

Technical Report

CMU/SEI-95-TR-014

ESC-TR-95-014

October 1995

An Experiment in Software Development Risk Information

Ira Monarch

David P. Gluch

Risk Program

Technical Report

CMU/SEI-95-TR-014

ESC-TR-95-014

October 1995

An Experiment in Software Development Risk Information



Ira Monarch
David P. Gluch

Risk Program

Unlimited distribution subject to the copyright.

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

This report was prepared for the

SEI Joint Program Office
HQ ESC/AXS
5 Eglin Street
Hanscom AFB, MA 01731-2116

The ideas and findings in this report should not be construed as an official DoD position. It is published in the interest of scientific and technical information exchange.

FOR THE COMMANDER

(signature on file)

Thomas R. Miller, Lt Col, USAF
SEI Joint Program Office

This work is sponsored by the U.S. Department of Defense.

Copyright © 1995 by Carnegie Mellon University.

Permission to reproduce this document and to prepare derivative works from this document for internal use is granted, provided the copyright and "No Warranty" statements are included with all reproductions and derivative works.

Requests for permission to reproduce this document or to prepare derivative works of this document for external and commercial use should be addressed to the SEI Licensing Agent.

NO WARRANTY

THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

This work was created in the performance of Federal Government Contract Number F19628-95-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 52.227-7013.

This document is available through Research Access, Inc., 800 Vinial Street, Pittsburgh, PA 15212. Phone: 1-800-685-6510. FAX: (412) 321-2994. RAI also maintains a World Wide Web home page. The URL is <http://www.rai.com>

Copies of this document are available through the National Technical Information Service (NTIS). For information on ordering, please contact NTIS directly: National Technical Information Service, U.S. Department of Commerce, Springfield, VA 22161. Phone: (703) 487-4600.

This document is also available through the Defense Technical Information Center (DTIC). DTIC provides access to and transfer of scientific and technical information for DoD personnel, DoD contractors and potential contractors, and other U.S. Government agency personnel and their contractors. To obtain a copy, please contact DTIC directly: Defense Technical Information Center / 8725 John J. Kingman Road / Suite 0944 / Ft. Belvoir, VA 22060-6218. Phone: (703) 767-8222 or 1-800 225-3842.]

Use of any trademarks in this report is not intended in any way to infringe on the rights of the trademark holder.

Table of Contents

1	Introduction	1
2	Overview of the Approach	3
2.1	Principal Process Steps	4
2.2	Identifying Concepts	4
2.3	Establishing Relationships	6
2.4	Generating Leximappes: The Algorithm	7
3	Description of the Data	9
4	Objectives of the Experiment	11
5	Detailed Observations	13
5.1	Risk and Leximappe Comparisons	14
5.1.1	Risk 1 and Risk 5	14
5.1.2	Risk 2, Risk 13, and Risk 15	16
5.1.3	Risk 3, Risk 4, and Risk 16	18
5.1.4	Risk 9, Risk 10, Risk 12, and Risk 14	19
5.1.5	Risk 6	20
5.1.6	Risk 7 and Risk 11	21
5.1.7	Risk 12	23
5.1.8	Risk 13 and Risk 15	24
5.1.9	Risks Not Covered by any Leximappe	25
5.2	Leximappes Not Covered by any Risks	25
6	Discussion	29
6.1	Comparison of Leximappes and Top N	29
6.2	Knowledge and Reading Leximappes	30
6.3	OnGoing Investigations	30
6.4	Extensions	30
6.5	K-SAV Technology as a Supplement to TRM Identification	31
6.6	General Conclusion	31
	Acknowledgments	33
	References	35

List of Figures

Figure 1	Producing Leximappes	3
Figure 2	Example Leximappe 'decision'	3
Figure 3	The Leximappe 'acceptance'	14
Figure 4	The Leximappe 'documentation'	15
Figure 5	The Leximappe 'performance'	16
Figure 6	The Leximappe 'card'	17
Figure 7	The Leximappe 'translation/lab'	18
Figure 8	The Leximappe 'memory'	18
Figure 9	The Leximappe 'date/translation schedule'	19
Figure 10	The Leximappe 'time'	20
Figure 11	The Leximappe 'vendor'	20
Figure 12	The Leximappe 'testing/hardware'	21
Figure 13	The Leximappe 'month/budget/customer'	22
Figure 14	The Leximappe 'time'	23
Figure 15	The Leximappe 'translation/lab'	23
Figure 16	The Leximappe 'card'	24
Figure 17	The Leximappe 'interface'	25
Figure 18	The Leximappe 'understanding/support'	26
Figure 19	The Leximappe 'qualification'	26
Figure 20	The Leximappe 'personnel'	27
Figure 21	The Leximappe 'countryx'	27

List of Tables

Table 1	Process Steps for Identifying Concepts	4
Table 2	Example Output of Extracting Shared Word Clusters	5
Table 3	Process Steps for Establishing Relationships	6
Table 4	The Top N Risks	13
Table 5	Leximappes Generated in the Experiment	14

An Experiment in Software Development Risk Information Analysis

Abstract: The following report summarizes the results of an experiment that uses terminological structures derived from the application of knowledge summarization, analysis, and visualization (K-SAV) technology to textual data from the Software Engineering Risk Repository (SERR) resident at the Software Engineering Institute. This study evaluates the use of several tools including shared word clustering [Monarch 94] and a co-word analysis software program, leximappe [Teil 92]. The experiment seeks to determine whether an application of co-word analysis to baseline risk assessment data would enable a reduction of the information load while simultaneously providing a succinct but encompassing picture of the risk information within the program. This study is based upon a somewhat limited data set. Nevertheless, the results of this investigation are encouraging and suggest that there may be value and potential for the effective use of co-word analysis and K-SAV technology more generally in risk management. Additional investigations are underway to confirm, alter, or challenge the results.

1 Introduction

The Software Engineering Institute (SEI), a federally funded research and development center sponsored by the U.S. Department of Defense, began to formally investigate and develop risk management in January 1990 and has developed a suite of processes, methods, and tools for managing risks within large software-intensive development and maintenance programs [SEI 92], [Higuera 93], [Higuera 94a], [Higuera 94b]. Through SEI risk management practices, all identified risks are managed somewhere within the organization. One of the major challenges, at any level of the organization, is focusing on the risks and the aspects of risk for which the application of program resources will provide the most cost-effective leverage toward successful risk mitigation and risk management generally.

This work attempts to describe and evaluate how knowledge summarization, analysis, and visualization (K-SAV) technology can be used to focus risk management activities and comprehensively represent risk information. K-SAV uses natural language analysis (NLA) tools such as taggers and parsers to extract syntactically well-formed phrases that are further processed in order to build clusters of phrases based on shared words and association networks of co-occurring terms. These methods are viewed as possible adjuncts to the selection methods currently used within SEI risk management processes and as potentially providing a broader picture of the full set of risks, perhaps through the representation of the broad base of issues and concerns.

This paper describes an initial evaluation of the co-word analysis approach. The evaluation compares representations of the risk data produced by this analysis with risks identified as

most important in a baseline risk assessment conducted using the SEI team risk management approach. The team risk management (TRM) approach [Higuera 93], [Higuera 94a], [Higuera 94b] employs the taxonomy-based questionnaire [Carr 93] interview method to elicit risks and a selection process that identifies as most important a finite number (N) of risks from all of the risks identified. These most important risks are termed the top N risks and are used as a focus for senior management in the overall risk management process [Higuera 93], [Higuera 94a]. The TRM methods have been successfully tested in a number of applications.

By addressing the top N risks, this experiment explored whether co-word analysis applied to a risk data set derived from a single baseline risk assessment, can provide a succinct representation that puts into sharp focus the important aspects and relationships of **risk** in the program while at the same time suggesting other important and relevant concerns, not included explicitly in the risks identified as most important to the program.¹

¹. The data has been carefully reviewed in order to remove any terms that would in any way indicate the organization or program whose data is being analyzed.

2 Overview of the Approach

The K-SAV approach evaluated in this work generates terminological networks using natural language processing (NLP) and co-word analysis applied to textual data, as depicted in Figure 1. The terminological networks are represented graphically as network maps termed leximappes [Callon 86], [Courtial 89], [Callon 91], [Teil 92].

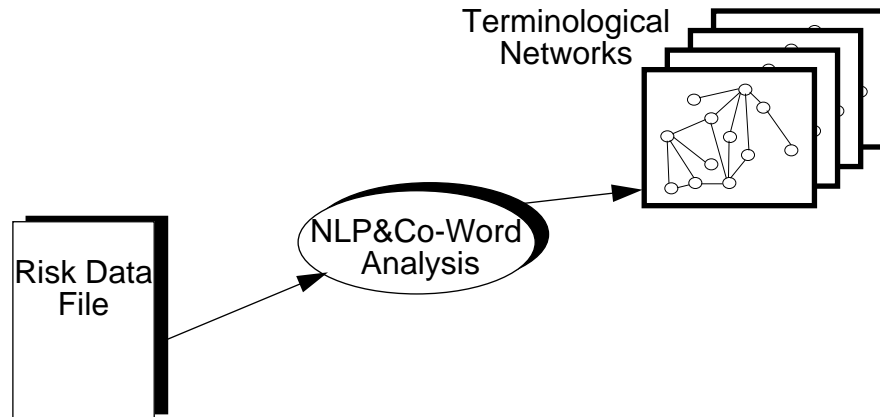


Figure 1: Producing Leximappes

Leximappes depict terminological networks consisting of nodes and links which represent terms (concepts) and inter-relationships, respectively (See Figure 2). It is a premise of this work that these networks, appropriately interpreted, show relationships and patterns among concepts that are both explicit and implicit within the text being analyzed.

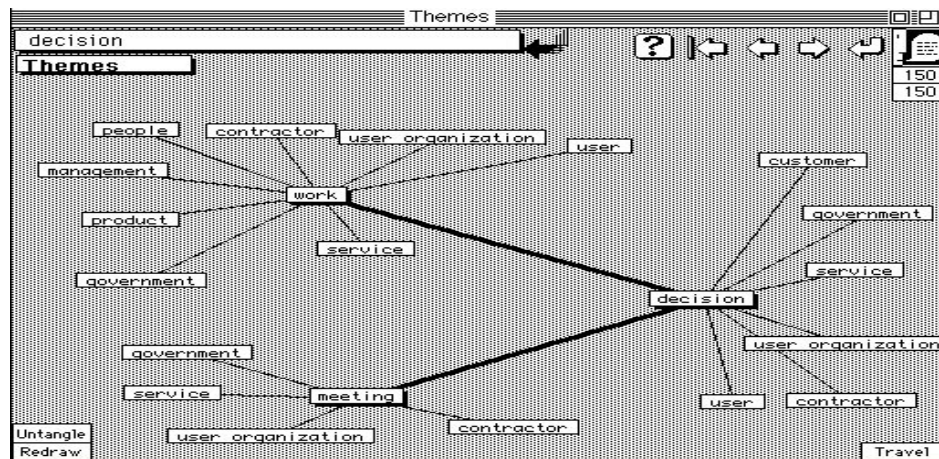


Figure 2: Example Leximappe 'decision'

Interpretation of the leximappes is based upon the following assumptions:

- Concepts are determined as the shared word in each of the clusters of noun phrases. They are extracted from text using natural language processing and shared word clustering.
- Relationships are based on the relative number of times a concept occurs together with another concept over the entire data set.

2.1 Principal Process Steps

The natural language-based analysis consisted of two principal steps:

1. **Identifying Concepts:** Concepts are identified by a shared word (shared phrase) cluster analysis process [Monarch 94] which produces clusters of phrases that all share the same syntactic unit — usually a noun or noun phrase.

In some cases, certain phrases in a cluster are themselves shared elements of clusters. These are called child clusters. Concepts tend to correspond to terms that are shared elements of clusters with children. See Table 2.

2. **Establishing Relationships:** Terms that are identified as corresponding to concepts are put through a co-word analysis process. A co-word analysis tool called “leximappe” [Teil 92] is used to create structured graphic representations of terminological networks. In these networks concepts are represented as nodes and the strength of the connection between two nodes represents the strength of their co-occurrence, that is, how often they occur together in relation to how often they occur separately.

2.2 Identifying Concepts


Concepts are identified as occurring in the data through the generation of shared word clusters. Shared word clusters are groupings of terms that share a common word or phrase. This common word is interpreted as a significant concept implied by the data. The process steps for identifying concepts are shown in Table 1.

Process Step	Description
Preprocess the data	This involves partitioning the contiguous text into individual “chunks” of information. For the experiment, a chunk was defined as an individual risk statement and its associated context.
Tag all words	Tag all words in the text as either a noun, verb, adjective, etc.
Parse the text	Parse the entire file into a list of noun phrases, grouped by partition (chunk).
Extract shared word clusters	Using a shared word clustering algorithm, extract groups of phrases that share a single word or contiguous words - a common phrase or concept. These shared words can be single or multi-word phrases.

Table 1: Process Steps for Identifying Concepts

The result of the steps above is a set of shared word clusters that characterize the data set. These clusters are used to define the concepts within the data set and can be used to form an index into the data set. An example of the output from step 4 is shown in Table 2.

shared term



```

*** (customer) *** Unique Terms: 54 Total Terms: 244
* Terms *
93 (customer)
26 (product for use by customer)
18 (customer as user organization)
18 (many different customer for individual organization within program structure)
18 (many different customer)
18 (user organization as customer)
3 (customer expectation)
2 (customer faction)
2 (appropriate program member in meeting with customer)
2 (development system to customer)

...etc.

* Child Cluster(s)*
2 (customer faction) Unique Terms: 5 Total Terms: 6
1 (customer intent) Unique Terms: 4 Total Terms: 4
1 (various customer) Unique Terms: 4 Total Terms: 4
3 (customer expectation) Unique Terms: 3 Total Terms: 5
1 (customer involvement) Unique Terms: 3 Total Terms: 3
1 (customer agreement) Unique Terms: 3 Total Terms: 3

```

Table 2: Example Output of Extracting Shared Word Clusters

The example cluster with the shared word “customer” is extracted from the data set consisting of the software risk taxonomy questionnaire [Carr 93], including questions, prompts, definitions, etc. These data were partitioned into chunks according to the attributes of the taxonomy.

In the first line of the output, the term in parentheses and bracketed with ***s is the shared term of the phrases collected in the cluster. This shared term defines the cluster. The number after the entry **Unique Terms:**, represents the number of different terms collected in the cluster. The number after the entry **Total Terms:**, represents the combined total of the number of times all of the different terms in the cluster occur in the data set analyzed.

In the example shown in Table 2, the cluster “customer” has 54 different phrases, each of which has the term customer within it; these occur collectively a total of 244 times.² Often a phrase occurs more than once, and each occurrence is counted in the total.

In subsequent lines of output, the number on the far left of each line of the output represents the number of times the phrase or term in parentheses just to the right of the number occurs

² Note that plurals are eliminated in the phrases collected in different clusters. Thus, “various customers” becomes “various customer.”

in the data set. This applies to all the lines whether listed under the heading * **Terms** * or ***Child Cluster(s)***

The terms under the heading * Child Cluster(s) * are terms that are collected in the cluster that themselves define another shared word cluster. In the example shown in Table 3, the phrase ‘customer faction’ occurs twice in the data set, and defines a shared word cluster that has 5 different phrases. In the entire data set, all of these phrases occur collectively a total of 6 times.

2.3 Establishing Relationships

Relationships between the concepts identified in the text are generated using a software package for co-word analysis called “leximappe” [Teil 92]. This processing results in a structured network of concepts, graphically presented with concepts displayed as rectangular nodes and their relationships as straight lines (links) between nodes. The process steps are shown in Table 3.

Process Step	Description
Index each chunk with the identified concepts	<p>Each chunk of the original text file is indexed using a subset of the concepts identified using term clustering. The index concepts are selected from the complete set by excluding very abstract and vague concepts like: aspect, level, point, way, rest, example, one, difference, etc. and including only those concepts which appear in at least a minimum number of chunks.</p> <p>Currently the selection process is done manually, at the discretion of the co-word analysis investigator. This process can proceed more automatically by basing selection just on a cutoff criterion, e.g. a concept must appear in a minimum number of chunks.</p>
Execute the leximappe program	<p>The leximappe program calculates a “strength” of co-occurrence for all pairs of index terms (noun phrases) in the index file. Terms co-occur if they occur in the same chunk.</p> <p>The co-occurrence “strength,” used to characterize the relationship between two terms in a leximappe, involves an inverse weighting by the number of times these terms occur in the data set. Consequently, two terms that occur frequently in a data set will need to co-occur more frequently than two terms that occur only rarely in order to have the same co-occurrence strength.</p>
Generate the graphical representations	Use the graphical display capability of the leximappe program to display the networks.

Table 3: Process Steps for Establishing Relationships

The final output of the leximappe program is a series of graphical presentations or maps of concept term networks. Each map is constructed such that

- each node represents a concept (index term) in the file
 - internal nodes, shown as shaded boxes, are primary (typically more strongly co-occurring) to a given network
 - external nodes are shown as thin-lined boxes. These nodes are not themselves internal nodes but are related to one or more internal nodes within the network and are internal nodes in another network
- the connections between pairs of nodes represent the co-occurrence of those concepts
- there are two levels of connection (association) between nodes:
 - internal links (nodes included in the internal links are connected by heavy connection lines in a network)
 - external links (This is an association between the internal nodes in a network and internal nodes from other networks. These links are shown as the thinner lines in the map)

Figure 2 is an example of a leximappe. The name of a leximappe is based on the internal node or nodes with the most links, both internal and external. The nodes of the leximappe contain the concepts derived from the shared phrase (shared word) analysis of each data set.

In a leximappe, concepts are associated with other concepts according to their co-occurrence strength. The co-occurrence strength is a relative numerical representation of how often two concepts (terms) co-occur in a document, inversely weighted by their frequency of occurrence. Links between pairs of nodes represent relationships (co-occurrences) between concepts.

2.4 Generating Leximappes: The Algorithm

The first pass of the leximappe program through the data generates a series of maps of networks consisting only of internal nodes (concepts) and internal links. A second pass through the data adds the external nodes and external links.

The initial leximappe generated starts with the two concepts that are linked at the highest co-occurrence strength found in the data set. This is represented by an internal link between two internal nodes. A new link is added to one of the original two nodes if it has the highest co-occurrence strength of all the remaining links to those two concepts. Once the new link and concept are added, the network consists of three nodes and two links. The network is further extended by adding links to any internal node in the same manner, one link at a time.³ Expanding the network continues until some cutoff point specified by the user with respect to nodes and/or links. Usually the maximum number of nodes is set at 10, chosen for coverage

³ From this point on, a link can occur between two internal nodes already in the network.

and readability. Depending on the nature of the data set, it may or may not be necessary to specify a maximum number of links per map.

Once the addition of links and nodes reaches the cutoff, a new map is begun with a linked pair of nodes. The link between the initial pair of nodes of any of these subsequent maps represents the highest co-occurrence strength between two concepts that have not appeared in any previous networks. Subsequent maps are built in the same way as the initial map until there are no more concepts that co-occur a minimum number of times. The minimum co-occurrence number depends on the size of the data set. In the current study it was set at three.

The lowest value of co-occurrence strength included in the internal links defines the lower (internal association) bound for the map. Thus, for a given map, the two internal concepts linked with the lowest co-occurrence strength determine the lower bound of the strength of co-occurrence for that map. The internal association lower bound is shown as the top number located at in the upper right-hand corner of the leximappe.

At the conclusion of the first pass of the leximappe program through the data set, a set of maps depicting only internal links between internal nodes has been generated. At this point, the leximappe program goes through the data set a second time, adding external links and external nodes (represented as thin lines and thin boxes) to each of the maps already generated.

The external links are determined by identifying, for every internal node in a given map, all the co-occurrences with concepts represented in other maps. If the number of co-occurrences associated with these links exceeds a minimum cut-off, then the concepts are considered to be externally related in a given map. These external links (relationships) are labeled with the thinner connection lines and the nodes external to the map are presented in thin line boxes.

The lowest value of co-occurrence strength included in the external links defines the lower external association bound for a map. This lower bound is shown as the second (bottom) number in the upper right-hand corner of the map.

3 Description of the Data

The data used for this experiment was gathered during a single baseline risk assessment of a project, was purely textual, and consisted of risk statements and associated context [Gluch 94]. A risk statement captures the essential elements of a risk in a single phrase or brief sentence. The context is a textual description of the events, circumstances, and interrelationships that may affect the risk. The context generally consists of (1-10) brief phrases or sentences extracted from the interview discussions that led to the capture of the risk statement. In the case of a baseline there are independent interview groups wherein discussions are initiated by distinct questions. These are further partitioned into discussions which provide a context for each identified risk.

Baseline risk assessments of a large software development project conducted using the Taxonomy Based Questionnaire [Carr 93] typically produce approximately 100 risk statements. For the case studied here, there were 82 risks identified in the risk baseline assessment. Using the team risk management processes, a subset of 16 risks, the top N most important risks, were identified. While all risks are managed within the Software Engineering Institute team risk management approach [Higuera 93], [Higuera 94a], the top N risks - the risks considered most important to the program - are generally the focus of management. The top N risks were the focus of comparison for this investigation.

It should be noted here that the data collected during the baseline risk assessment had to be altered slightly. The SEI has established confidentiality agreements with clients; consequently, the name of the organization involved in the baseline assessment is not divulged in this report and other sensitive (confidential) information is edited. In this study, the analysis tools were applied to the unedited data and edits were only made in the final forms of the maps. The exclusions and changes made in the maps and this report do not affect the results of the analysis. Substitute terms are identified with an underscore in this document. Most of these terms are formed by adding an 'x' and sometimes a letter (a, b, c, etc. when there is more than one of the same kind) to the end of a descriptive but more general term replacing the sensitive term (usually a proper name). The second letter indicates that there were a number of items named in the data by different proper nouns that were all replaced by the same new term (common noun). For example, systemxb means that there were two things named by proper nouns that can both be called a systemx, i.e., systemxa and systemxb.

4 Objectives of the Experiment

This experiment explores the effectiveness of natural language processing and co-word analysis to succinctly represent risk information, to capture important relationships among the top risks, and to aid in highlighting potentially important risk information not included explicitly in the top N risks.

The effort investigates the extent to which maps of networks generated from the complete risk data set cover and elaborate the information represented in the top 16 most important risks identified in a baseline assessment (e.g., identify additional relations or concerns); whether or not such maps can be used by a project manager or software developer to identify and analyze important risk information and provide a basis for making decisions and taking action.

The specific objectives of this experiment are

- to determine the extent to which the derived maps of networks correspond to (cover) the top 16 risks identified in the baseline assessment
- to determine whether the maps exhibit relations among the top 16 that might be important in managing them
- to determine whether the maps represent important considerations not represented in the top 16 risks

5 Detailed Observations

The 16 most important (top N) risks from the subject baseline assessment are summarized in Table 4.

Risk	Statement
Risk 1	It takes too long to resolve issues with the customer.
Risk 2	Processing power (<u>systemxb</u>) to handle the data throughput requirements.
Risk 3	Translation has potential sizing problem fitting into memory.
Risk 4	Size of Ada executable and slow execution may exceed hardware and timing limitations.
Risk 5	Customer approval of deliverable documentation content (CDRL).
Risk 6	The Ada compiler is not reliable (<u>vendorxa</u>) and not easy to fix. Lacks CM; two compilers (<u>vendorxb</u> , Target); lack coordinated configurations.
Risk 7	Inadequate budget and schedule for software development; software engineering budgets are based upon optimistic estimates of performance (SLOC counts, productivity).
Risk 8	Casualty recovery philosophy is not specified at this time.
Risk 9	May not meet translation schedule due to late GFI/lab delivery.
Risk 10	Access to integration lab facility at <u>facilityxb</u> may not be adequate.
Risk 11	Inadequate budget and schedule for testing.
Risk 12	Translation testing on target machine is a choke point (limited resource for debug).
Risk 13	Risk that engineering development model will not be available for start of formal qualification test at customer site.
Risk 14	Potential slip in availability of software and test procedures may not allow for full use of integration resources and time.
Risk 15	<u>graphics processorx</u> redesign may not meet the performance goals and it may have impact on software, i.e., which I/O interfaces to simulate.
Risk 16	The Ada language does not permit as much control of timing issues as provided by other languages.

Table 4: The Top N Risks

Each leximappe is identified by a primary node(s) within the map. There were 16 distinct leximappes (LMs) generated as a result of the analysis. These are summarized in Table 5.

• acceptance	• vendor	• date/translation schedule	• countryx
• card	• translation/lab	• documentation	• interface
• qualification	• understanding/support	• month/budget/customer	• memory
• performance	• testing/hardware	• time	• personnel

Table 5: Leximappes Generated in the Experiment

5.1 Risk and Leximappe Comparisons

In this section each of the top N risks is compared with leximappes.

5.1.1 Risk 1 and Risk 5

Risk 1	It takes too long to resolve issues with the customer.
Risk 5	Customer approval of deliverable documentation content (CDRL).

The ‘**acceptance**’ leximappe shown in Figure 3 captures a specific instance of Risk 1. The important relations are between ‘**customer approval**’ ‘**expectation**,’ ‘**acceptance**,’ and ‘**late acceptance**.’ Relations to other nodes such as ‘**srs**’ (software requirement spec) ‘**schedule**,’ ‘**format**,’ and ‘2167a’⁴ constitute a *specific* issue about format that takes too long to resolve.

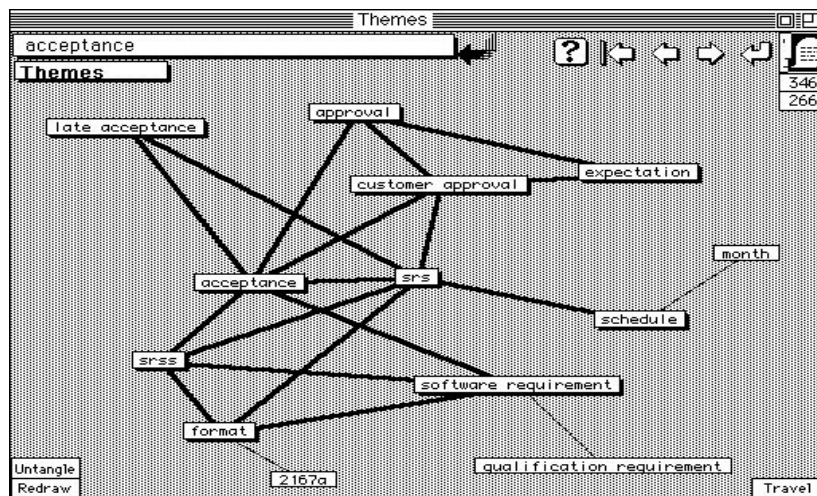


Figure 3: The Leximappe ‘acceptance’

4. Note that internal nodes are in bold and external nodes are in plain text unless reference is being made to more than one leximappe.

The **'documentation'** leximappe (Figure 4) corresponds to Risk 5. The important relations are between **'documentation,'** **'data,'** **'2167a,'** and **'customer approval.'** This LM also puts Risk 5 in a richer context, most importantly, perhaps, by emphasizing **'2167a'** in relation to **'pdr.'**

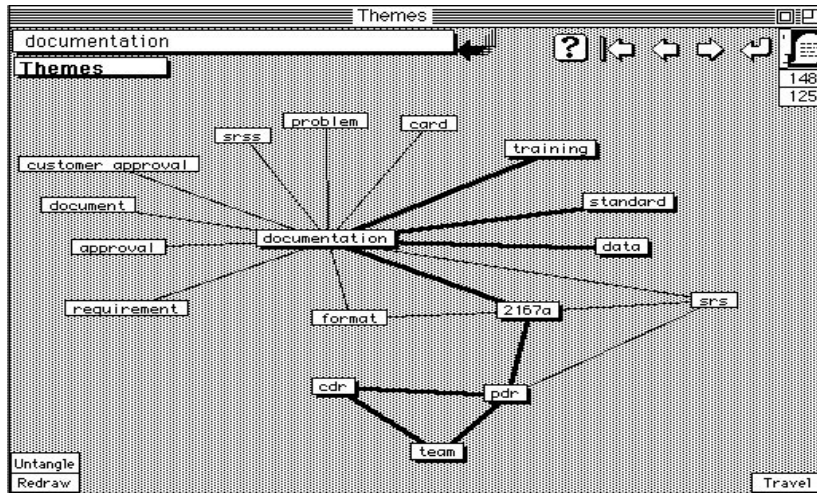


Figure 4: The Leximappe 'documentation'

The two LMs shown here importantly overlap in the concepts **'format,'** **'2167a,'** **'customer approval,'** and **'srs'**—thus, perhaps indicating a relationship between Risk 1 and Risk 5.

5.1.2 Risk 2, Risk 13, and Risk 15

Risk 2	Processing power (<u>systemxb</u>) to handle the data throughput requirements.
Risk 13	Risk that engineering development model will not be available for start of formal qualification test at customer site.
Risk 15	<u>graphics processorx</u> redesign may not meet the performance goals and it may have impact on software, i.e., which I/O interfaces to simulate.

Risk 2 is addressed by the ‘**performance**’ leximappe (LM), shown in Figure 5. This map points to a ‘**timing**’ ‘problem’ with respect to ‘**throughput**’ and ‘**power**’ of the ‘**processor**’ as well as the ‘**performance**’ of the ‘**systemxb**,’ thus covering the main thrust of Risk 2.

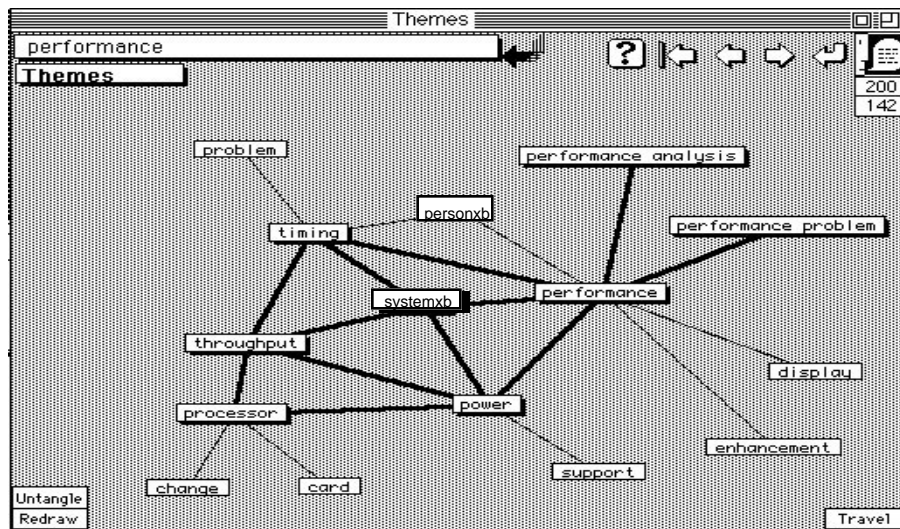


Figure 5: The Leximappe ‘performance’

The ‘**performance**’ LM also associates ‘**personxb**’ with ‘**timing**’ and ‘**performance**.’ Since the two latter terms are fairly vague and may or may not have anything to do with ‘**throughput**,’ ‘**systemxb**,’ or ‘**power**’ in regard to ‘**personxb**,’ it is unclear from the structure of the leximappe whether ‘**personxb**’ has some role to play in managing Risk 2.

As can be seen in the ‘**card**’ LM, shown in Figure 6, ‘**personxb**’ is more intensely associated with ‘**edm**’ on the one hand and especially ‘**display**’ ‘**card**’ on the other. Personxb may therefore have a more important role to play in the availability of the engineering development model (edm) for Risk 13 or, perhaps, in graphics processorx display card redesign for Risk 15.

To re-emphasize, our hypothesis is that when faced with such LMs or other types of conceptual graph, people involved in a project will be able to make the appropriate distinctions and

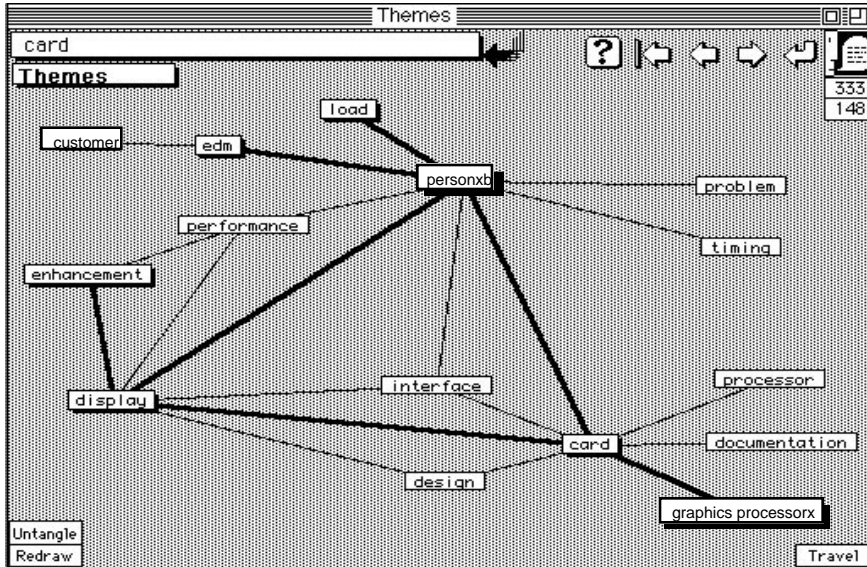


Figure 6: The Leximappe 'card'

inferences for interpreting them. In fact, personxb did play a significant role in the in both Risk 2 and in the edm.

5.1.3 Risk 3, Risk 4, and Risk 16

Risk 3	Translation has potential sizing problem fitting into memory.
Risk 4	Size of Ada executable and slow execution may exceed hardware and timing limitations.
Risk 16	The Ada language does not permit as much control of timing issues as provided by other languages.

The leximappe **'translation/lab'** (see Figure 7) shows there is a concern about **'ada'** with respect to **'timing,' 'time,'** and **'translation.'** The leximappe **'memory'** (see Figure 8) shows a concern with the **'size'** of **'memory.'** The two together cover fairly well Risks 3, 4, and 16.

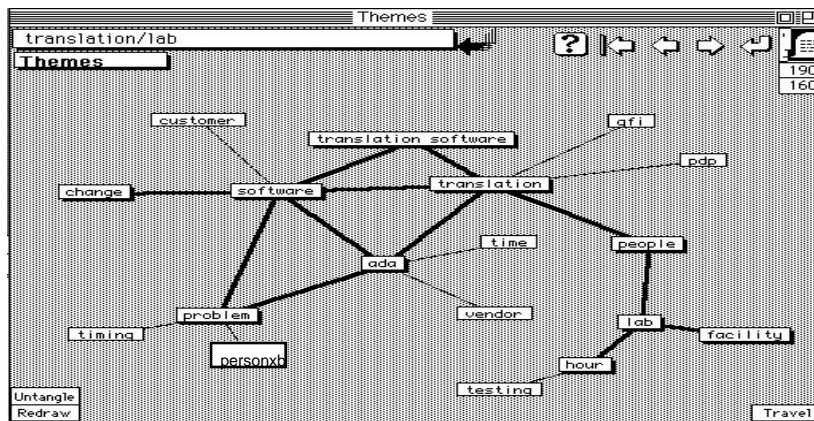


Figure 7: The Leximappe 'translation/lab'

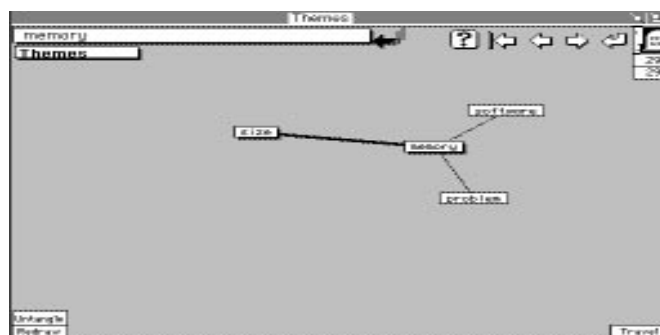


Figure 8: The Leximappe 'memory'

However, the mapping is incomplete. While translation is clearly related to the size and memory problem in Risk 3, it is not explicitly related in any of the leximappes. Moreover, while **'ada'** and **'timing'** issues are clearly related in Risk 16, they are only indirectly related in **'transla-**

tion/lab.’ In fact, one might argue that insofar as ‘**timing**’ is a problem, it is more related to the ‘**throughput**’ of the ‘**systemxb**’ as is shown in the ‘**performance**’ leximappe in the discussion of Risk 2. There is another way of looking at this, more favorable to the pertinence of the leximappes. The ‘**timing**’ issues may be related. This is perhaps suggested in Risk 4.

5.1.4 Risk 9, Risk 10, Risk 12, and Risk 14

Risk 9	May not meet translation schedule due to late GFI/lab delivery.
Risk 10	Access to integration lab facility at <u>facilityxb</u> may not be adequate.
Risk 12	Translation testing on target machine is a choke point (limited resource for debug).
Risk 14	Potential slip in availability of software and test procedures may not allow for full use of integration resources and time.

The ‘**translation/lab**’ LM does show the interrelations of almost all of the risks concerning issues of ‘**translation**’ and ‘**translation software**’ including both Risk 9 with its concern about late GFI/lab delivery and Risk 12 with its concern for testing, though this concern is better represented in the ‘**data/translation schedule**’ LM (Figure 9). The concern about late GFI/lab delivery is also represented more directly by the LM ‘**date/translation schedule**’ which overlaps with ‘**translation/lab**’ at ‘**translation**’ and ‘**gfi**.’

Risk 10 also has some correspondence with the LM ‘**translation/lab**’ in its concern about people’s access to the integration lab facility. The concern about people’s access to the ‘**integration**’ lab ‘**facility**’ could also be inferred from the leximappe ‘**time**’ (Figure 10). Risk 14 is also covered by ‘**date/translation schedule**.’ Especially important here are ‘**slip**,’ ‘**resource**,’ ‘**availability**,’ and ‘**test**.’

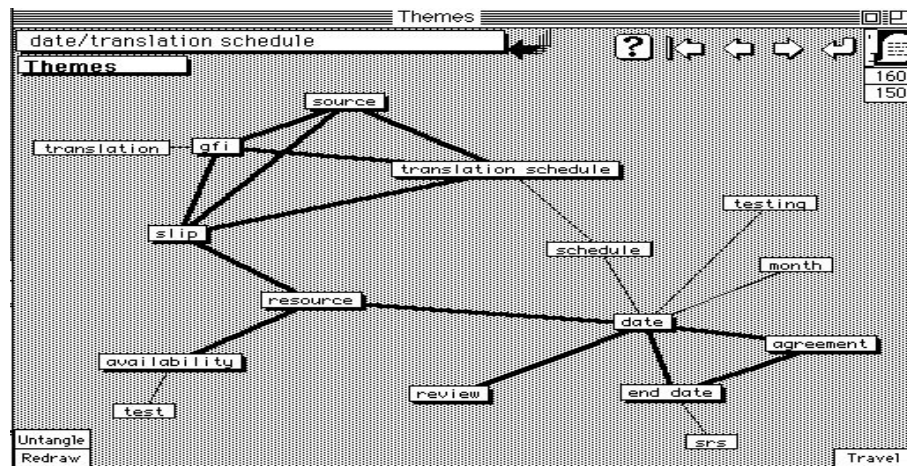


Figure 9: The Leximappe ‘date/translation schedule’

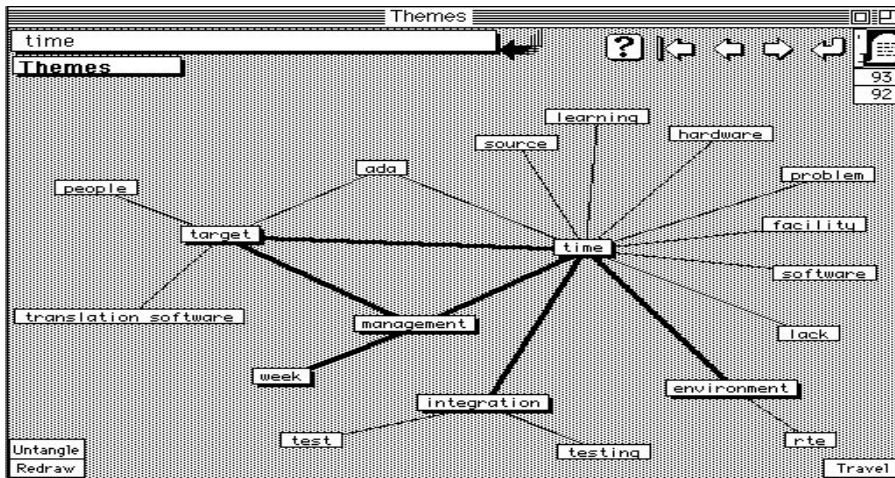


Figure 10: The Leximappe 'time'

5.1.5 Risk 6

Risk 6	The Ada compiler is not reliable and not easy to fix. Lacks CM; two compilers; (<u>vendor</u> , Target) lack coordinated configurations.
--------	---

The leximappe 'vendor' (Figure 11) covers the concern to 'fix' the 'ada compiler' in Risk 6 and adds that this concern applies to the 'vendor.' However, nothing is shown about the lack of coordinated configurations.

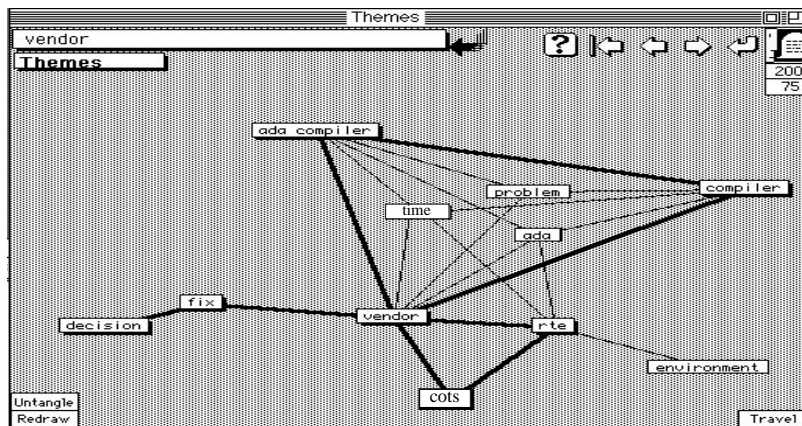


Figure 11: The Leximappe 'vendor'

5.1.6 Risk 7 and Risk 11

Risk 7	Inadequate budget and schedule for software development; software engineering budgets are based upon optimistic estimates of performance (SLOC counts, productivity).
Risk 11	Inadequate budget and schedule for testing.

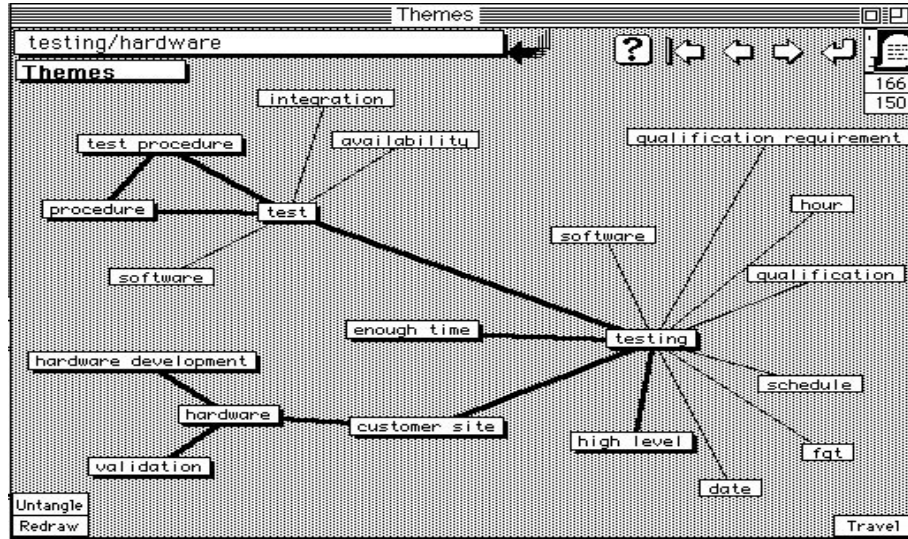


Figure 12: The Leximappe 'testing/hardware'

Both Risk 7 and Risk 11 concern inadequate budget and schedule -- the former for software development in general and the latter for testing. The LM 'testing/hardware' (Figure 12) shows a strong relation between 'testing,' 'enough time,' 'date' and 'schedule,' and the LM 'month/budget/customer' (Figure 13) shows a relation between 'budget' and 'schedule,' especially in the guise of 'completion,' though a somewhat indirect and less strong connection to 'testing.' The two LMs are connected through 'schedule' and 'testing.' The 'testing/hardware' LM may be a basis for relating Risks 7 and 11 to 'hardware development' and 'validation' at the 'customer site' as well as to 'test' and 'testing' procedures for 'integration,' thus relating them to Risks 12 and 14.

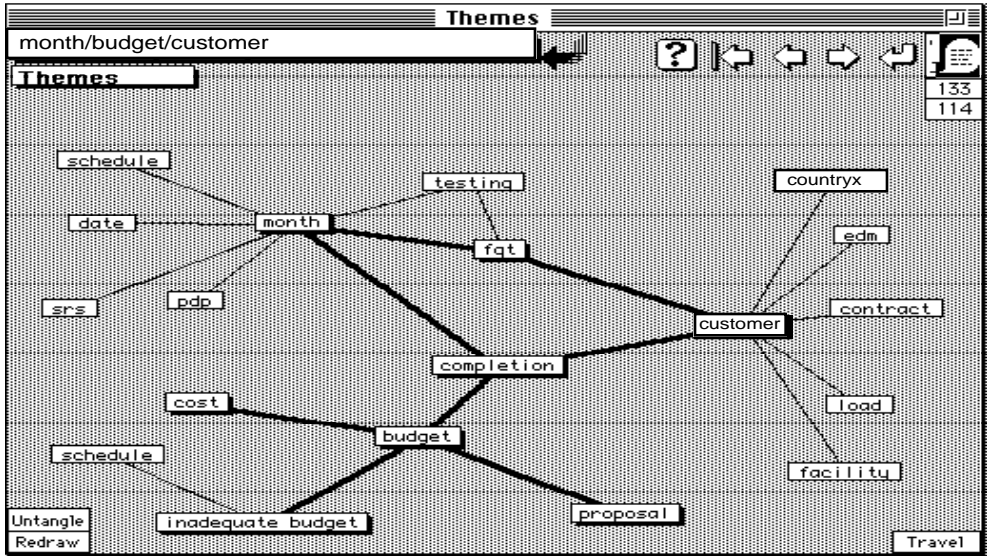


Figure 13: The Leximappe 'month/budget/customer'

5.1.7 Risk 12

Risk 12	Translation testing on target machine is a choke point (limited resource for debug).
---------	--

The LM 'time' (Figure 14) shows a relationship between the 'target' of 'translation software' and 'time' on the one hand and 'integration,' 'testing,' and 'test' on the other. This covers "translation testing on target machine is a choke point" in Risk 12, provided the meaning of "choke point" is that a lot of time will be taken due to limited resources for debugging. This is reinforced if the 'target' in the LM 'time' is part of the 'lab' 'facility' in 'translation/lab' (Figure 15). This should be determinable by the project members (of this sub-language community) at this site.

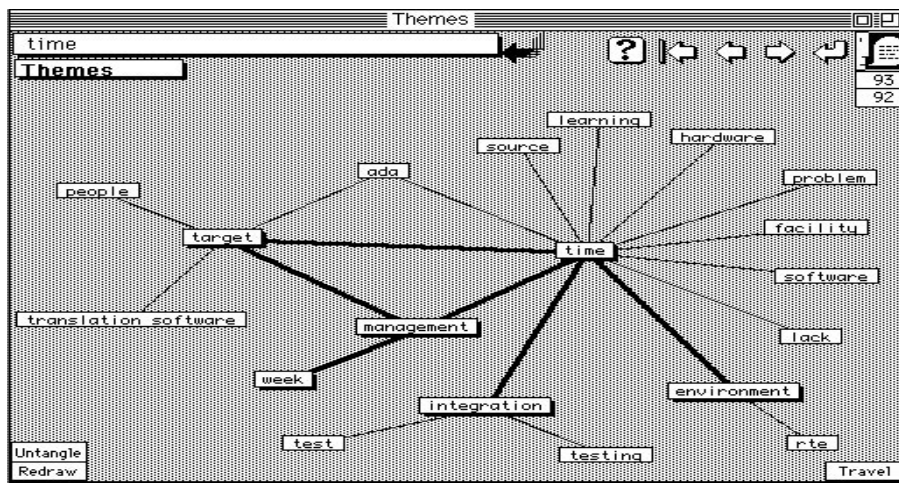


Figure 14: The Leximappe 'time'

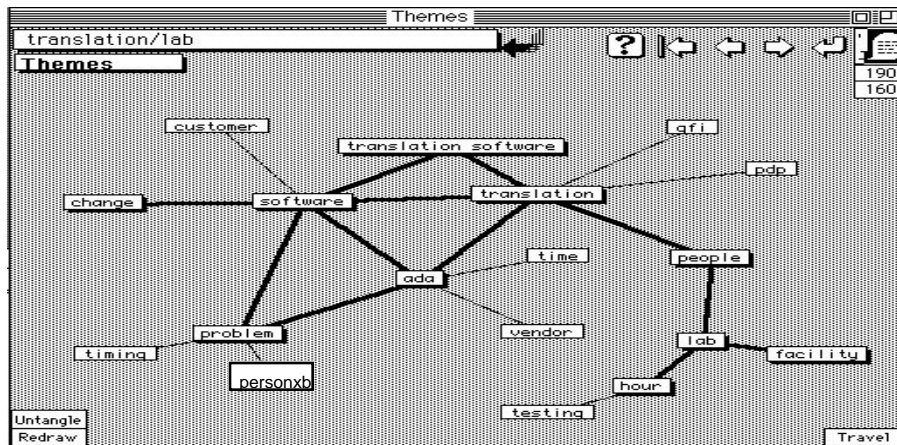


Figure 15: The Leximappe 'translation/lab'

6 Discussion

We have explored several questions concerning the potential of Leximappes to represent and communicate risk information based on text data gathered in a baseline risk assessment. The questions all have the same basic form: “Could someone involved in a development project ‘read’ these Leximappes and derive conclusions that both agree with and extend the insight provided by the top 16 risk statements?”

The following summarizes these results and offers some general observations:

- There is significant overlap between the Leximappe representations of the entire data set and the top 16 risks identified in the program. In the case investigated here, the co-word analysis technique accurately captures most of the important risks.
- Leximappes, in addition to showing the relationships that are explicit in the risk statements themselves, show relationships among risks through the concepts they share, even though these relationships are not evident explicitly in the risk statements. This suggests that Leximappes may be useful as a basis for considering all risk information while partitioning it into chunks that can be readily acted upon.
- Because Leximappes show relationships among concepts, they may be useful in identifying reasons (sources or causes) for risk and/or provide specific examples of risks not explicit in the risk statements. For example, the Leximappes suggest that risk 5 and risk 1 are concerns because of lack of understanding of the process and the document for SRSs (see the LMs ‘**acceptance**’, ‘**documentation**’, and ‘**understanding/support**’) or that the reason they are concerns is because of qualification requirements on the part of the customer (see the LMs ‘**acceptance**’ and ‘**qualification**’). In fact, the interrelationships among these Leximappes suggests that qualifying requirements is a problem because of lack of understanding on the part of the customer. These results suggest that Leximappes can be useful in providing guidance on identifying sources or causes for specific risks.
- By providing both relationships and potential reasons (sources or causes), Leximappes may be useful in probing for and identifying common sources or causes and forming a basis for defining mitigation actions that would address multiple risks.
- Leximappes appear to offer a broad perspective on the entire risk data set that suggests causal and inter-risk associations and may be an effective tool for guiding the continuing analysis and planning activities of risk management.

6.1 Comparison of Leximappes and Top N

Leximappes and identifying the top N risks are alternative ways of filtering the large amount of risk information that is normally collected in a risk assessment. Both attempt to sift what is important in the risk data collected from what is not. The premise underlying the Leximappe approach is that relationships common to different interviews of different groups of people with

different perspectives are indicative of concerns that are common across a number of program personnel, and not merely idiosyncratic. The Leximappe approach takes account of individual voices, but only to the extent that these voices share something with other individual voices.

The top N approach also requires a certain amount of consensus but admits concerns into the top N that *prima facie* have nothing in common with other concerns. What is interesting is that, at least in this case, the two approaches converge to a significant degree, despite this difference.

There are also some other important differences. The Leximappe approach addresses significantly more of the risk information than the top N approach, whereas the top N approach is more focused in what it does cover. Moreover, the Leximappe approach finds what appear to be important relationships among the risks that the top N does not show explicitly.

6.2 Knowledge and Reading Leximappes

One of the hypotheses of the present investigation is that real world knowledge both in terms of background and current experience will enable a reader of a Leximappe to make many of the appropriate distinctions and inferences regarding the context of risks. Such knowledge can be used to pin down whether an indirect relationship applies in a specific way to a certain situation even when all the relevant factors are not presented. This hypothesis is being investigated as part of ongoing work.

6.3 Ongoing Investigations

Although the results of this investigation are encouraging and suggest that there may be value and potential for the effective use of the Leximappe approach in risk management, it should be noted that this study is based upon a somewhat limited data set. Additional investigations are being conducted to confirm, alter, or challenge the results. These investigations have involved further experimentation with the present NLA tools, including Leximappe, as well as extending the system by adding additional modules.

Improvements are currently being added for viewing the relationships between multiple leximappes to help identify sources or causes of risk and aid in grouping risks to support risk mitigation and management. These extensions will need to be evaluated as the approach is applied to more and more risk data.

6.4 Extensions

The NLA tools can also be modified and extended. The present NLA tools use only nouns and noun phrases as potential shared elements of clusters. They could be extended to use verbs and verb phrases as well. This would enable the latter to function as a manifestation of a concept. For example, if 'sizing' would have counted as an instance of the concept size, the relation between translation and size exhibited in Risk 3 may have been captured in a Leximappe.

This would also require new morphological capabilities, but inexpensive COTS software exists to provide them.

In addition, the present lexicons could be updated to handle better the abbreviations in the software engineering domain. This would enable the current pre-parsing NLA facilities to expand a phrase like “lacks cm” to “lacks cm (configuration management)”—thus increasing the co-occurrence potential for the term “configuration management.”

6.5 K-SAV Technology as a Supplement to TRM Identification

Future investigations could explore the view that using K-SAV is supplementary to the use of team risk management’s methods. Along with team risk management risk identification, NLA and Leximappes could be used to capture relationships among the top risks and track risk information not identified in the top N. In fact, K-SAV may be helpful in predicting which risks not in the top N have the best chance of becoming more highly ranked in the future.

6.6 General Conclusion

It is important to view the natural language analysis and Leximappe tools as producing new modes of communication which increase opportunities for knowledge sharing rather than as simply a tool for automating data analysis. One implication of this is that these tools do not simply derive risks from data so much as provide intermediary representations that facilitate the construction of practical knowledge and form the basis for informed decisions. In this sense, they become media for the representation, communication, and integration of knowledge into management processes. One of the aims of this work is to discover and evaluate those intermediaries most conducive to the effective integration of knowledge into general decision making processes.

Acknowledgments

The authors would like to thank Audrey Dorofee, William Hayes, Dick Murphy, Julie Walker, and Ray Williams for reviewing and commenting on the evolving versions of this report. Their input helped substantially to improve the over all quality of the document. Appreciation is also extended to Bob Lang for his editorial review and helpful suggestions and to Jodi Horgan, Tanya Jones, and Dana Siler for final preparation of the manuscript. Also, the authors would like to thank Genevie Teil for providing the Leximappe software used in these analyses.

References

- [Callon 86] Callon, M; Law, J.; & Rip, A. "Quantitative Scientometrics." *Mapping of the Dynamics of Science and Technology*, London: McMillian, 1986.
- [Callon 91] Callon, M; Courtial, J. P.; & Laville, F. "Co-Word Analysis as a Tool for Describing the Network of Interactions between Basic and Technological Research: The Case of Polymer Chemistry." *Scientometrics* 22, 1 (January 1991): 153-203.
- [Carr 93] Carr, Marvin; Konda, Suresh; Monarch, Ira; Ulrich, Carol; & Walker, Clay. *Taxonomy Based Risk Identification* (CMU/SEI-93-TR-6 ADA 266992). Pittsburgh, Pa.: Software Engineering Institute, Carnegie Mellon University, 1993.
- [Courtial 89] Courtial, J. P. & Law, J. "A Co-Word Study of Artificial Intelligence." *Social Studies in Science*, London, SAGE, (1989): 301-311.
- [Gluch 94] Gluch, David P. *A Construct for Describing Software Development Risks* (CMU/SEI-94-TR-14). Pittsburgh, Pa.: Software Engineering Institute, Carnegie Mellon University, 1994.
- [Higuera 93] Higuera, Ronald P.; & Gluch, David P. "Risk Management and Quality in Software Development." *Proceedings of the Eleventh Annual Pacific Northwest Software Quality Conference*, Portland, Oregon, October 18-20, 1993:58-73.
- [Higuera 94a] Higuera, Ronald P.; Gluch, David P.; Dorofee, Audrey J.; Murphy, Richard L.; Walker, Julie A.; & Williams, Ray C. *An Introduction to Team Risk Management Version 1.0* (CMU/SEI-94-SR-01). Pittsburgh, Pa.: Software Engineering Institute, Carnegie Mellon University, 1994.
- [Higuera 94b] Higuera, Ronald P. *Team Risk Management: A New Model for Customer-Supplier Relationships* (CMU/SEI-94-SR-05). Pittsburgh, Pa.: Software Engineering Institute, Carnegie Mellon University, 1994.
- [Monarch 94] Monarch, Ira. "An Interactive Computational Approach for Building a Software Risk Taxonomy," *Third Annual Conference on Software Risk*, Pittsburgh, PA, April 1994.
- [SEI 92] Software Engineering Institute. "The SEI Approach to Managing Software Technical Risks." *Bridge*, October 1992:19-21.
- [Teil 92] Teil, G. *Guide Pratique Pour L'Utilisateur de Leximac*. Genevive, 1992.

