University of Alberta

COTS Acquisition Process:
Incorporating Business Factors into COTS Vendor Evaluation Taxonomies

by

©

Hean Chin Yeoh

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of Master of Science.

Department of Electrical and Computer Engineering

Edmonton, Alberta

Fall 2004

Your file Votre référence
ISBN: 0-612-95886-8
Our file Notre référence
ISBN: 0-612-95886-8

Table of Contents

1. INTRODUCTION 1
1.1 OVERVIEW 1
1.2 DEFINITION 3
1.2.1 Federal Acquisition Regulations 3
1.2.2 Software Engineering Institute (SEI) 4
1.2.3 Basili and Boehm 5
1.2.4 Carney and Long 5
1.2.5 COTS Based System 6
1.3 DRIVING FORCES 7
1.4 POTENTIAL BENEFITS OF COTS 9
1.4.1 Shorter Development Time 9
1.4.2 Lower Development Effort and Cost 9
1.4.3 More Stable System 9
1.4.4 More Alternative Solutions 10
1.4.5 Increasing Level of System Interoperability 10
1.5 POTENTIAL RISKS OF COTS 10
1.5.1 COTS Risks and Challenges 11
1.5.2 COTS Risk Category 12
1.5.2.1 COTS Selection and Evaluation 13
1.5.2.2 System Integration 13
1.5.2.3 Development Process 13
1.5.2.4 System Quality 14
1.5.2.5 System Evolution 14
1.6 CONCLUSION 15

2. RELATED WORK 19
2.1 OVERVIEW 19
2.2 COTS DEVELOPMENT MODEL AND ACQUISITION PROCESS 19
2.2.1 Off-the-shelf Option (OTSO) 20
2.2.2 COTS-based Integrated Systems Development (CISD) 22

2.2.3 COTS Acquisition Process (CAP) 23
2.2.4 Evolutionary Process for Integrating COTS Based Systems (EPIC)...25
2.2.5 COTS Acquisition Evaluation Process (CAEP) 29
2.3 COTS Risk Management 30
2.3.1 FAA Risk Mitigation Guide 31
2.3.2 COTS Usage Risk Evaluation (CURE) 32
2.4 Summary of Literature Review 33
2.5 Conclusion 35

3. PROPOSED SOLUTION 41

3.1 Overview 41
3.2 VERPRO Decision Making Model 47
3.2.1 Measurement Theory 48
3.2.2 Alternative Decision Making Methods for COTS Selection 49
3.2.3 Analytic Hierarchy Process (AHP) 51
3.2.3.1 Advantages of AHP 52
3.2.3.2 AHP's Application in VERPRO 54
3.2.3.3 Conclusion 60
3.3 VERPRO Business Factors (VBF) 61
3.3.1 Definition and Justification 61
3.3.1.1 Product Leadership 62
3.3.1.2 Financial Strength 63
3.3.1.3 Business Risks 64
3.3.2 VERPRO Business Metrics (VBM) 64
3.3.2.1 Definition 64
3.3.2.2 Sources of Data 66
3.3.2.3 Interpretation of Financial Statements 70
3.3.2.4 Formulas and Protocols 74
3.4 VERPRO Implementation 87
3.4.1 VERPRO Process Flowchart 88
3.4.2 Symptoms of Business Distress 94
3.4.3 Financial Reference Profiles 96

3.4.3.1 R&D Budget 96
3.4.3.2 Revenue Growth 98
3.4.3.3 Current Ratio 98
3.4.3.4 Cash Margin 99
3.4.3.5 Cash Reserve 99
3.4.3.6 Debt to Equity Ratio 100

4. APPLICATIONS OF VERPRO 101

4.1 OVERVIEW 101

4.2 EXAMPLE 1 101

4.2.1 Background 101
4.2.2 A Brief Profile of the Alternatives 103
4.2.3 Quantifying VERPRO Business Metrics 105
4.2.4 Implementing VERPRO Decision Making Model 107

4.3 EXAMPLE 2 116

4.3.1 Background 116
4.3.2 A Brief Profile of the Alternatives 119
4.3.3 Quantifying VERPRO Business Metrics 120
4.3.4 Implementing VERPRO Decision Making Model 123

5. FUTURE WORK 133

6. CONCLUSION 136

BIBLIOGRAPHY: 139

List of Tables

Table 3.1.1: Decomposition of VERPRO Business Factors......46
Table 3.2.1: Pairwise Comparison Scale (Saaty 1980)......56
Table 3.3.2: A Summary of VERPRO Business Metrics (Yeoh 2004)......66
Table 3.3.3: A Summary of VERPRO Measures and Data Sources......70
Table 3.3.4(A): Worldwide ERP Software Application New License Revenue Market Share Estimates for 2002 (Graham 2004)......79
Table 3.4.1: Early Signs of Financial Distress and Mitigating Strategy......95
Table 3.4.2(A): R&D Budget Benchmark of Public Software Companies in US......97
Table 3.4.2(B): Revenue Growth of Public Software Companies in US......98
Table 3.4.2(C): Current Ratio of Public Software Companies in US......99
Table 3.4.2(D): Cash Margin of Public Software Companies in US......99
Table 3.4.2(E): Cash Reserve of Public Software Companies in US......100
Table 3.4.2(F): Debt to Equity Ratio of Public Software Companies in US......100
Table 4.2.1: Results of the Metrics......107
Table 4.2.2: Nine Point Pairwise Comparison Scale (Saaty 1980)......107
Table 4.2.3(A): Pairwise Comparison for Market Share......108
Table 4.2.3(B): Pairwise Comparison for Revenue Growth......109
Table 4.2.3(C): Relative Importance of Market Share and Revenue Growth......109
Table 4.2.3(D): Overall Score of the Alternatives with respect to Product Leadership......109
Table 4.2.3(E): Pairwise Comparison for Current Ratio......110
Table 4.2.3(F): Pairwise Comparison for Cash Margin......111
Table 4.2.3(G): Pairwise Comparison for Cash Margin......111
Table 4.2.3(H): Relative Importance of Current Ratio, Cash Margin and Cash Reserve with respect to Financial Strength......112
Table 4.2.3(I): Overall Score of the Alternatives with respect to Financial Strength 112
Table 4.2.3(J): Pairwise Comparison for Debt to Equity......113
Table 4.2.3(K): Pairwise Comparison for Revenue Risk......113
Table 4.2.3(L): Pairwise Comparison for Legal Liability......113

Table 4.2.3(M): Relative Importance of the Objectives 114
Table 4.2.3(N): Overall Score of the Alternatives with respect to VBF.................... 115
Table 4.2.4: Comparison of Mercury with Industry Profiles.................................... 116
Table 4.3.1: Results of the Metrics 123
Table 4.3.2(A): Pairwise Comparison for Market Share 124
Table 4.3.2(B): Pairwise Comparison for R&D Budget... 124
Table 4.3.2(C): Pairwise Comparison for Revenue Growth..................................... 125
Table 4.3.2(D): Relative Importance of Market Share, R&D Budget and Revenue Growth with respect to Product Leadership... 126
Table 4.3.2(E): Overall Score of the Alternatives with respect to Product Leadership.. 126
Table 4.3.2(F): Pairwise Comparison for Current Ratio... 127
Table 4.3.2(G): Pairwise Comparison for Current Ratio.. 127
Table 4.3.2(H): Pairwise Comparison for Cash Reserve.. 128
Table 4.3.2(I): Relative Importance of Current Ratio, Cash Margin and Cash Reserve with respect to Financial Strength.. 128
Table 4.3.2(J): Overall Score of the Alternatives with respect to Financial Strength129
Table 4.3.2(K): Pairwise Comparison for Debt to Equity 129
Table 4.3.2(L): Pairwise Comparison for Legal Liability 129
Table 4.3.2(M): Pairwise Comparison for Legal Liability 130
Table 4.3.2(N): Relative Importance of the Objectives 131
Table 4.3.2(O): Overall Score of the Alternatives with respect to VBF.................... 131
Table 4.3.4: Comparison of Ruby with Industry Profiles.. 132

List of Figures

Figure 2.2.1(A): Required Approach for COTS-Based Systems 25
Figure 2.2.1(B): An Iteration in EPIC (Albert 2002) 27
Figure 2.2.1(C): EPIC Phases (Albert 2002) 29
Figure 2.4.1: COTS Vendor Evaluation Taxonomy 35
Figure 3.1.1: VERPRO Vendor Evaluation Taxonomy (Yeoh 2004) 43
Figure 3.2.1: Decision Hierarchy (Saaty 1980) 48
Figure 3.3.1(A): Software Segmentation Structure (Graham 2004) 76
Figure 3.3.1(B): Enterprise Application Software Segmentation Structure (Graham 2004) 77
Figure 3.3.1(C): Infrastructure Software Segmentation Structure (Graham 2004) 78
Figure 3.3.2(B): R&D Budget 79
Figure 3.3.2(C): Revenue Growth 80
Figure 3.3.2(D): CMMI Hierarchy (SEI) 81
Figure 3.3.2(E): Current Ratio 82
Figure 3.3.2(F): Cash Margin 82
Figure 3.3.2(G1): Cash Reserve (when cash flow is positive) 83
Figure 3.3.2(G2): Cash Reserve (when cash flow is negative) 84
Figure 3.3.2(H): Debt to Equity Ratio 84
Figure 3.3.2(I): Debt Coverage 85
Figure 3.3.2(J): Cash to debt ratio 85
Figure 3.3.2(K): Revenue risk 86
Figure 3.3.2(L): Claim Coverage 86
Figure 3.4.1(A): Process Flowchart for VERPRO Inception Phase 91
Figure 3.4.1(B): Process Flowchart for VERPRO Elaboration Phase 92
Figure 3.4.1(C): Process Flowchart for VERPRO Transition Phase 93

1. INTRODUCTION

1.1 OVERVIEW

The use of commercial off-the-shelf (COTS) and other pre-existing components is gaining popularity, particularly in communities where the organization's needs match those of one or more commercial information technology (IT) marketplace segments. The market that delivers COTS software components ranges from software development environments to operating systems, database management systems, and increasingly, business and mission critical applications. At the same time, the use of COTS products as elements of larger systems also leads to faster development, reduced effort and higher quality (Albert 2002). According to studies performed by Gartner Group, at least 70% of all new software applications developed in 2003 involved COTS. Although the term COTS can represent hardware, software or a combination of both, this thesis will focus on COTS software only. Over the years, there have been numerous attempts to provide a comprehensive definition for COTS and the most widely adopted definitions in North America are given by:

- United States Federal Acquisition Regulations
- Software Engineering Institute
- Basili and Boehm
- Carney and Long

Despite the differences in the definitions, COTS software generally has the following characteristics (Albert 2002):

- The product is developed for sale, lease or license to the general public at a fair market value
- The buyer has no access to the source code
- The vendor controls its development and evolution
- The product has a nontrivial installed base meaning it has more than one customer and more than a few copies.

Commercial software is a significant component of the global economy, generating annual revenues of approximately $75 billion in 2004. The drive for greater efficiency and the increasing globalization of business markets spurs much of the spending, as companies must continue to invest in the hardware and software infrastructure that enables them to operate in increasingly competitive markets (Graham 2004). COTS software, which comprises a significant portion of the software market, contains a wide variety of products and technologies. As a result, there is a need to classify the COTS software industry into logical segments for better understanding and more in-depth and segment-specific studies. The classification of a product should take into account a variety of factors, which include the product's technical features, target audience, competitive positioning and perceived usage by the customers. There is more than one approach to classify the industry; one of the most comprehensive is provided by Gartner Dataquest. Gartner Dataquest classifies the products into two broad categories (Graham 2004):

- Infrastructure Software
- Enterprise Application Software

The focus of Infrastructure Software is to increase the performance of IT resources. Infrastructure Software is further categorized into (Graham 2004):

- Application development
- Application integration and middleware
- Database management systems
- Business intelligence
- Data warehousing tools
- Network and systems management
- Security systems

The focus of Enterprise Application Software is to increase the performance of business or personal resources. It enables users to leverage the power of computers toward achievement of their business, professional or personal objectives or goals. Enterprise Application Software is further categorized into (Graham 2004):

- Customer relationship management
- Project portfolio management
- Enterprise resource planning
- Supply chain management
- Collaborative and knowledge management

The rationale for using COTS products is that it will involve less development time by reusing existing, market proven, and vendor supported components, thereby reducing overall system development costs. Yet, the promise is too often not realized in practice (Carney 2000). Many organizations find that COTS-based systems are difficult and costly to build and support. One of the causes is that organizations building these systems tend either to assume that COTS components can be thrown together or fall back on force fitting COTS based system into traditional software engineering models which have been shown not to work. As a result, in the absence of a proper process, COTS based development faces many potential development, operational and maintenance risks. The consequence can range from slipped schedule, overrun cost, to unreliable and functionally invalid systems that are difficult or impossible to maintain (Morisio 2001).

1.2 DEFINITION

At the time of this writing, COTS is still an evolving concept with more than one definition. With over a decade of experience, there have been more than one attempt to define COTS, and the four that are presented in the following sub sections are by far the most comprehensive and widely accepted.

1.2.1 Federal Acquisition Regulations

The United States Federal Acquisition Regulations breaks down the acronym COTS into 'commercial' and 'off-the-shelf', and defines these terms separately. The term 'commercial' is defined as (Torchiano 2001):

- Property customarily used for non-governmental purposes and has been sold, leased, or licensed (or offered for sale, lease or license) to the general public.
- Any item evolved from an item in the above through advances in technology and is not yet available commercially but will be available in time to satisfy the requirement.
- Any item that would satisfy the above but for modifications customarily available in the commercial marketplace or minor modifications made to meet Federal Government requirements.
- Any combination of items meeting the above.
- Services for installation, maintenance, repair, training, etc. if such services are procured for support of an item in the above, as offered to the public or provided by the same work force as supports the general public; or other services sold competitively in the marketplace.
- A non-developmental item developed exclusively at private expense and sold competitively to multiple state and local governments.

The term "off-the-shelf" is defined as a piece of software that is not developed by the user but already exists such as (Torchiano 2001):

- Development tools (e.g., compilers)
- Integral parts of the new system (e.g., libraries)
- Both development tools and parts of the new system (e.g., DBMS, compilers with run-time libraries, OS with APIs).

1.2.2 Software Engineering Institute (SEI)

The SEI defines a COTS product as a piece of software that is (Torchiano 2001):

- Sold, leased, or licensed to the general public
- Offered by a vendor trying to profit from it
- Supported and evolved by the vendor, who retains the intellectual property rights
- Available in multiple, identical copies

- Used without source code modification

1.2.3 Basili and Boehm

Basili and Boehm define COTS product as a piece of software that (Torchiano 2001):

- The buyer has no access to the source code
- The vendor controls its development
- The product has a nontrivial installed base

This definition is more restrictive and does not take into account some types of software products like software products developed for special purposes and not widely deployed, special version of commercial software products and open source software.

1.2.4 Carney and Long

Carney and Long classify COTS products according to their origin and modifiability. The possible values for the attribute *origin* are (Torchiano 2001):

- Independent Commercial Item
- Special Version of Commercial Item (A product developed by a commercial vendor and slightly modified for a client, where the modification may or may not be included in the next commercial release of the product.)
- Component Produced by Contract or Sub-contractors
- Existing Components from External Sources (Components that are not developed internally, and usually not paid for either)
- Component Produced In-house

The possible values for the attribute *modifiability* are:

- Extensive reworking of code
- Internal code revision
- Necessary tailoring and customization

- Simple parameterization
- Minor modification
- No modification

1.2.5 COTS Based System

A COTS based system is a software-based application that integrates one or more COTS. Carney classifies COTS based systems into turnkey system, intermediate system, and integrated system (Carney 1997).

- Turnkey systems are built around a suite of commercial products, such as product data management, enterprise resource planning, workflow management, financial management, or manufacturing execution. Such systems arise when a well-bounded range of business processes creates a sufficiently large market segment to justify the cost of developing the COTS product. Only one COTS is used, and customization does not change the nature of the initial COTS.
- Intermediate systems are built around one COTS such as Oracle Database Management System but integrate other components, commercial or legacy. The central COTS software is the main part of the system, but integration of other components is the key process.
- Integrated systems are built by integrating several COTS, all on the same or varying level of importance. The final system is not dominated by any single COTS component; integration is the key to building the system.

Wallnau classifies COTS-based systems into COTS solution system and COTS intensive system (Wallnau 1998). In COTS solution systems one core COTS component is tailored to provide a turnkey solution. The main characteristics are:

- Generic solutions
- Very tightly coupled to business process
- Tailoring and parameterization focus
- Maintained by vendor
- Vendor infrastructure

- May change business process in order to adapt to the tool

The designation of the term "COTS intensive" moves along a spectrum of complexity. For instance, some systems might make use of a major commercial product but add significant functionality making the end product more like a custom-made solution that relies on a commercial foundation. It is difficult to accurately characterize the scale of measurement to determine "What percentage of the system is COTS?" But one can infer that the more COTS components are involved, the more "COTS intensive" is a system. COTS intensive system are often integrated, glued, and combined from disparate products from different and often competing vendors to provide functionality that is unavailable from any single vendor (Shaffer 2002). The main characteristics are:

- More flexible to business process change
- Integration, engineering focus
- Maintain by purchaser
- Own infrastructure
- More complex to maintain

1.3 DRIVING FORCES

The increasing directives and mandates in using COTS in government and business organizations is due to the fact that these organizations typically spend far too much effort on defining to the lowest level of detail the desired characteristics of systems and how the contractors are to build those systems to achieve those characteristics. Thus a lot of resources are wasted in developing systems and components that often already exist or exist in good enough form with nearly the same capabilities (Albert 2002). The prevailing approach was to develop the systems from the ground up, which resulted in unique systems that are:

- Very expensive, with only one customer to bear the development and maintenance costs over the life of the component or system
- Inflexible and unable to easily capitalize on advances in technology
- Historically fielding technology that is in excess of ten years old

Shifting to a paradigm in which systems are built primarily from components that are available commercially offers the opportunity to lower costs by sharing them with other users, thus amortizing them over a larger population, while taking advantage of the investments that industry is putting into the development of new technologies. The factors that lead to the widespread use of COTS software in government agencies and business organizations are (Tran 1997):

- The increasing market competition for delivery of more complex and reliable solutions in shorter time frame.
- The increasing demand for larger and more integrated software solutions that cannot be effectively built by a single software organization. These software solutions often span multiple domains that include networking, GUI, database, and workflow.
- The increasing availability of both generic and domain specific COTS products. These products are often built to provide architectural frameworks for families of applications belonging to specific domains. In addition to implementing many of the domain specific functionalities, these products provide generic abstractions to these functionalities that enable simpler integration with other domain applications.
- The increasing level of interoperability and compliance to industrial standards for better and faster integration. Software is no longer produced to function as a standalone product but to co-exist within a much larger framework. Therefore, industrial standards provide a strong foundation for the development of interoperable software products.
- The increasing research and development for better component packaging techniques and approaches that form the basis for production of complex COTS components.
- The increasing recognition that software reuse is one of the most important means to achieve better software solutions with lower overall development costs.

1.4 POTENTIAL BENEFITS OF COTS

The benefits of COTS have been documented in (Tran 1997), (Basili 2001), and (Shaffer 2002), they can be summarized into five main points:

- Shorter development time
- Lower development effort and cost
- More stable system
- More alternative solutions
- Increasing level of system interoperability

1.4.1 Shorter Development Time

The use of COTS allows the organization to focus its resources on the development of software that reflects its core competencies without having to commit extra resources in developing generic software that are available in the marketplace at considerably lower cost. As a result, the resources can be kept low; the range of skill sets required needs not be widened, and the organization can focus on its niche and stay competitive (Tran 1997).

1.4.2 Lower Development Effort and Cost

More than 99 percent of all executing computer instructions come from COTS products (Boehm 1999). Each instruction passed a market test for value. Economic necessity drives extensive COTS use because few organizations can afford to write a general-purpose operating system or database management system. As a result, every project ought to consider the COTS while carefully weighing its benefits, costs, and risks against other options.

1.4.3 More Stable System

Unlike internally developed software components that are often only used within a single organization, COTS components are used across multiple organizations and environments. Because of its larger stake and impact, COTS components often undergo more stringent and extensive testing. As a result, the final system that is built from COTS is often much more stable (Boehm 1999).

1.4.4 More Alternative Solutions

The increasingly prevalent use of COTS components has attracted a huge capital pool to the industry, resulting in an explosion of COTS software in the market. In many application domains such as database, operating systems, GUI, and network communications, there are a large number of choices with common functionalities. In order to compete, the market players have to constantly reinvent themselves, and the result is better quality products and lower price (Basili 2001).

1.4.5 Increasing Level of System Interoperability

The knowledge accumulated from the integration of widely used COTS products to the final system will help improve the level of interoperability in the next generation COTS products. For example, the adoption of HP OpenView network management framework created a valuable knowledge base for the development of future generation SNMP-based COTS products. While the level of interoperability between COTS products is still far from ideal, the continued accumulation of knowledge and data from the existing systems coupled with strong market forces will ensure that consistent progress will continue to be made (Basili 2001).

1.5 POTENTIAL RISKS OF COTS

Although COTS does deliver real advantages, it also brings about a new set of risks and challenges such as lack of software support skills, incompatibility with underlying hardware platform, complexity of interfaces with other applications, middleware, glue code, and legacy systems, accidental modification of a system functionality that exceeds the product tolerance, accidental addition of corrupted codes into the system, confusing licensing options, sole source dependency for critical software components, limited data rights and questionable information security (Tran 1997), (Basili 2001), (Shaffer 2002).

1.5.1 COTS Risks and Challenges

The risks and challenges of COTS based projects are depicted in the following anecdotes:

- The Standish Group's CHAOS report cited that COTS based system is a high-risk activity with effort and schedule overruns exceeding traditional systems. The median applications software cost overruns of traditional system is about 50 percent, with 4 percent of the projects overrunning beyond 400 percent, and schedule overruns is about 100 percent, with 1 percent of the projects having overruns beyond 400 percent. Surprisingly, comparable COTS based systems have cost and schedule overruns exceeding 400 percent (Basili 2001). The cause can be attributed to lack of visibility in COTS products, vendors' temptations to over-promise on vaporware, and the difficulty of estimating glue-code size in source code lines coupling with over-optimism and desire to produce unrealistic cost and schedule commitments.
- Basili and Boehm reported that COTS based development and post deployment efforts can scale as high as the square of the number of independently developed COTS products targeted for integration. Integrating n COTS products involves potentially $n*(n - 1)/2$ interfaces. The theoretical justification for this relationship stems from the architectural incompatibilities that pose difficulties in integrating any two COTS products. Although most empirical evidence is still lacking, various COTS based projects have experienced results consistent with this relationship. As a result, excessive use of COTS components in a system is counter-productive; at times four can be too many. Making the scaling law approximately linear involves using sound interface standards, modular domain architectures, wrappers around COTS products, and well-planned multi-COTS-product refresh cycles (Basili 2001).
- Ground System Architectures reported that the average COTS product undergoes a new release every eight to nine months, with active vendor support for only its latest three releases. Although the use of COTS products aims to solve many infrastructure change adaptation problems,

COTS vendors must evolve their products to sustain the market's competition, depending on their perceptions of fluctuating market demands. If the direction of your COTS based system is inconsistent with the market's evolution, your applications will have significant adaptive maintenance costs, even during production, and can introduce new risk.

- Torii laboratory reported in the 2000 International Software Engineering Research Network Workshop that more than half the features in large COTS software products are unused. Individuals working alone used 12 to 16 percent of Microsoft Word and PowerPoint measurement features, whereas a 10-person group used 26 to 29 percent of these features. Although adding features is an economic necessity for vendors, it introduces complexity for COTS adopters. This added complexity requires expanded computer resources, such as speed and memory, to provide functionality that is not needed (Basili 2001).

1.5.2 COTS Risk Category

The majority of the problems in COTS based development are due to the poor understanding and management of the risks involved. According to (Kotonya 2001), COTS risks can be categorized into five main categories on the basis of the development stages. The risks in one category often overlap with the risks in the other categories and also cut across several stages of the development cycle. Therefore, categorizing the risks makes it possible to better understand their overlapping nature and creates a more targeted risk mitigation strategy. The development stages include:

- COTS selection and evaluation
- System integration
- Development process
- System quality
- System evolution

1.5.2.1 COTS Selection and Evaluation

COTS software is often delivered as "blackbox" components with limited specification making it difficult to predict how the components behave under different environments. There is also a general lack of methods for mapping user requirements to COTS based architectures. Moreover, the evaluation process is made complicated by the different forms (function libraries, off-the-shelf applications) and features (granularity, tailorability, platform support, distributed system support, and interoperability) of COTS components.

1.5.2.2 System Integration

Most COTS integration processes are impacted by inflexibility and poor component evaluation. This problem is often compounded by a lack of interoperability standards between component frameworks and adequate vendor support. Most COTS software is generally not modifiable. Therefore, significant effort may be required to build wrappers around components in order to tailor the application to new situations. As the system evolves these wrappers must be maintained. There is also a general lack of interoperability standards to facilitate the integration of components implemented using different technologies. Components for well-understood domains are more likely to have readily available COTS equivalents than specialized domains such as safety-critical systems. The variability of specialized systems often makes it difficult to tailor components for different application without major modifications.

1.5.2.3 Development Process

The existing software development model practiced by the organization might not be suitable for COTS based development. For example, both waterfall model and evolutionary development are unsuitable for COTS because in the waterfall model requirements are identified at an earlier stage and the components that are selected later might not possess the required features. Likewise, evolutionary development assumes that additional features can be added when required but COTS components cannot be upgraded for one particular development team. In addition, the lack of code availability hinders the development team from tailoring them to their needs.

1.5.2.4 System Quality

The design assumptions of a COTS component are unknown to the system integrator. Coupled with poor component specification, the testing quality on the component is seriously impacted, and the potential for the component to fail to interact with other components also significantly increases. COTS component heterogeneity may result in complicated licensing arrangements in which no single vendor has complete control over the development artifacts for the purpose of testing. A new version of COTS software is likely to contain new features that are not used by the system but may still have some indirect impact on the system behavior. The perception of quality may vary across COTS software vendors and application domains. For many vendors the time-to-market may be more important than delivering a high-level of reliability. The use of COTS software introduces a vulnerability risk that may compromise system security, especially for distributed systems and safety-related systems.

1.5.2.5 System Evolution

The different customer-vendor evolution cycles may result in an uncertainty about how often COTS components in a system may have to be replaced and the extent of the impact of such a change on the rest of the system. This makes it difficult to plan and predict costs over the life cycle of a system. Upgrading to a new version of COTS software poses several risks. First, hidden incompatibilities may cause unforeseen side effects in the system necessitating a complete system update. Second, changes in the quality attributes of a new version of COTS software may be incompatible with the user requirements. Finally, a new version of COTS software may provide additional undocumented capabilities.

1.6 CONCLUSION

Despite the documented risks and challenges, COTS software continues to thrive and is expected to constitute over 90 percent of the software applications running in major corporations (Graham 2004). The driving forces can be attributed to three key factors:

- Economic
- Technological
- Psychological

Economic factor deals with the comparison between the true cost of using COTS with the true cost of custom development taking into consideration the opportunity cost as a result of schedule gain or delay. It also includes the integration and maintenance cost. The emergence of TQM and Six Sigma practices has imposed an enormous pressure on organizations to move toward industrial best practices, which in turn drive the consolidation and standardization of business processes. This phenomenon has created a huge market potential for software packages and suites such as operating system (OS), database management (DBM), enterprise resources planning (ERP), supply chain management (SCM) etc. and led to the proliferation of software powerhouses such as IBM, Microsoft, Oracle, Peoplesoft, SAP AG etc. These companies have enough installed base to achieve the kind of technological advantages and economy of scale that make it impractical for individual companies to custom develop software that can be purchased from them.

Technological factor deals with the ability of the company to cope with the technical complexities required to develop the software. Complex software packages such as DBM, ERP, and SCM are complicated, risky, costly, and take years of experience to accumulate the technical know how to master the process. Therefore, even with sufficient funds, individual companies may not be able to acquire the technology or hire the right resources to custom develop the software. This has left most companies with the only option, which is to buy when it comes to such complex software packages.

Psychological factor deals with the behavior of the person making the build or buy decision as a result of peer pressure and institutional imperatives. For example, an executive often feels pressured to mimic the behavior of his peers in order not to look foolish. Therefore, if SAP is adopted by most Fortune 500 companies as the de facto ERP suite, the Chief Information Officer (CIO) of a corporation is more likely to go for the same software suite than to risk his career with custom development software especially when a huge portion of the business is at stake. By doing what others are doing, the CIO gets a peace of mind that even if SAP does not turn up to be what it promises, the business should still run with its basic features. In extreme cases where SAP fails entirely, the CIO is less likely to be blamed for his decision because everybody else is also using SAP. If the CIO chooses custom developed software and it fails to run, his career is probably at stake. Therefore, there is a saying that nobody ever gets fired by choosing IBM.

For over a decade, numerous research works have been published to address the issues faced by COTS development team. The work has focused on two key areas, COTS Development Model and Acquisition Process, and COTS Risk Management.

- Development Model and Acquisition Process. This is one of the most critical areas in COTS based development and also by far the most actively researched area. In the absence of a COTS specific development model, traditional development models such as Waterfall and Spiral have been widely used for COTS based development and produced varying results ranging from acceptable to disastrous. To correct the deficiencies and inherent limitations of Waterfall and Spiral models, various works have been published since 1995 starting with the introduction of Off-the-Shelf Option (OTSO) by (Kontio 1995) when the term COTS was still relatively new. Although the method was highly applicable to COTS, Kontio emphasized his work on reusable components treating COTS only as a subset because COTS was still a relatively unknown term in academia at that time. There are at least seven major publications on COTS development model and acquisition process following OTSO. Although some of the proposed methods and approaches vary in scope and technique, they share the same goal. Some of their common characteristics are their attempt to:

- Address the deficiencies of the previous work.
- Expand the scope.
- Refine the model.
- Test the practicability of the proposed models.

These publications, listed in chronological order, include:

- COTS-based Integrated Systems Development (CISD) by (Tran 1997).
- Procurement Oriented Requirement Engineering (PORE) by (Maiden 1998).
- Scenario Based COTS Selection by (Feblowitz 1998).
- COTS Acquisition Process (CAP) by (Ochs 2000).
- Requirement-driven COTS Product Evaluation Process (RCPEP) by (Lawlis 2001).
- Evolutionary Process for Integrating COTS Based Systems (EPIC) by SEI (Albert 2002).
- COTS Acquisition Evaluation Process: Preacher's Practice by (Sai 2004).

- Risk Management. The majority of the work in this area consists of a compilation of lessons-learned derived from an extensive analysis of common government and industrial experiences that are documented in numerous technical documents. Each approach represents a specific technique or method that is aimed at addressing certain risk factors. The major publications on COTS Risk Management, arranged in chronological order, include:
  - Risk Mitigating Model for the Development of a Reliable and Maintainable Large Scale COTS Integrated Software System by (Tran 1997).

- A Strategy for Managing Risk in Component-based Software Development by (Rashid 2001)
- FAA Risk Mitigation Guide: Practical Methods for Effective COTS Acquisition and Lifecycle Support by (Shaffer 2002).
- Identify COTS Product Risk: A COTS Usage Risk Evaluation (CURE) by (Carney 2003).

In Section 2, the related literatures will be discussed and a detailed analysis of each approach and a discussion of its pros and cons will be presented. Section 2 will be concluded with a discussion of the general limitations and deficiencies of the existing approaches and a proposal of how these deficiencies can be addressed. In Section 3, the proposed solution will be elaborated, and two examples will be presented in Section 4 to reinforce the understanding and application of the proposed solution. Section 5 will provide a brief discussion of the future work and Section 6 will conclude the thesis.

2. RELATED WORK

2.1 OVERVIEW

Making the change to a COTS approach is like making the change from being a developer and producer of systems to being a consumer and integrator. Many of the changes involve evaluation of technologies and products, and the design and engineering of systems (Carney 2003). Becoming an effective consumer and integrator of COTS products necessitates the application of new strategies for licensing components, negotiating data rights, estimating system development and maintenance cost, predicting schedules, managing personnel, and identifying and reducing risks. An effective COTS consumer also requires knowledge of how to build sufficient flexibility into procurement and contract documents to foster a variety of creative solutions. Effective COTS consumers also must identify appropriate steps for determining when a system is a good candidate for COTS (Albert 2002). The strategy for identifying best acquisition and management practices requires continual contact with commercial and federal organizations experienced with aspects of COTS approaches.

In the subsequent sections, an elaborated discussion of the various COTS development models and acquisition processes, and risk management approaches will be presented. These topics are not mutually exclusive; in many cases they are complementary. After these approaches are examined, the pros and cons will be discussed, and a solution will be proposed to address the issues.

2.2 COTS DEVELOPMENT MODEL AND ACQUISITION PROCESS

The goal of a COTS development model and acquisition process is to provide a well-defined, efficient, reliable, and customizable process that effectively identifies the COTS components that correctly interface and integrate to meet the objectives of the new system. The most prominent publications in this area are:

- Off-the-shelf Option (OTSO)
- COTS-based Integrated Systems Development (CISD)

- COTS Acquisition Process (CAP)
- Evolutionary Process for Integrating COTS Based Systems (EPIC)
- COTS Acquisition Evaluation Process: Preacher's Practice

For each topic, the following subject matters will be covered:

- The background and motivation behind the work
- A brief description of the model and its strengths
- A brief discussion of its pros and cons

2.2.1 Off-the-shelf Option (OTSO)

Published in late 1995 by Kontio, OTSO represents one of the earliest efforts to address issues that stem from software reuse involving in-house libraries of reusable components and commercial off-the-shelf components. When OTSO was first published, the use of COTS was still relatively new, therefore, the discussion mainly focused on reusable components while treating COTS as a subset. The project was a joint effort between Hughes Information Technology Corporation and NASA in response to the mandate by the United States Department of Defense to maximize the utilization of reusable software components as well as the potential benefits that can be realized from reuse programs. Despite the increasing popularity and importance of software reuse, the selection process typically is not defined and each project team has its own approach. In addition, the project teams are often under schedule pressure and there are no mechanisms to learn from previous selection cases. The other problem faced by the project team was the lack of a reliable method to estimate the effort associating with software reuse. The cost estimation of reusable software is often problematic as traditional cost models are not applicable.

The objective of the project was to develop a well-defined solution that supports the search and selection of reusable components in software development. The resulting solution was named OTSO. OTSO supports the search, evaluation and selection of reusable software and provides specific techniques for defining evaluation criteria, comparing the costs and benefits of alternatives, and consolidating the evaluation results for decision making. The OTSO method consists of five phases (Kontio 1995):

1. Defining evaluation criteria
2. Searching for potential COTS alternatives
3. Shortlisting the alternatives
4. Evaluating the shortlisted alternatives
5. Analyzing and synthesizing the result

The OTSO method was subsequently tested in two case studies that were carried out with Hughes Corporation in the EOS program being developed for NASA. The case studies were documented in a conference paper by Kontio in March 1996. According to Kontio, the result of the case studies indicated that the OTSO method was feasible and had a low overhead. The method also resulted in more efficient and consistent evaluations and increased the confidence of the decision makers.

OTSO is not strictly a COTS development model as it does not address the issues of the simultaneous definition and trade off of requirements among the stakeholders, the marketplace, the architecture, and the programmatic risks that are central to any COTS development model. In addition, because of the inherence difficulties in defining a baseline for any software project, the cost and benefits claimed by the OTSO method can be described as anecdotal at best. Its cost estimation model also found limited application since the publication of COCOTS by Barry Boehm in 1999. Nonetheless, by documenting a comprehensive review of the state-of-the-art on software reuse and COTS, OTSO represents an important pioneering effort in recognizing the increasing prevalence of COTS-based development and lays the groundwork for further research. Furthermore, its advocate of the use of analytic hierarchy process in the component selection process is a revolutionary improvement over the prevailing weighted average approach.

2.2.2 COTS-based Integrated Systems Development (CISD)

Published in 1997 by Tran and Liu in response to an increasing migration from development-centric toward a more procurement-centric software development approach and an increasing evidence of the inadequacies of the development centric models such as Waterfall and Spiral in coping with the use of COTS, CISD can be regarded as one of the earliest effort to address the need for a unique COTS-based development model. Unlike the OTSO method, which treated COTS components as a subset, the CISD model was exclusively developed for large scale and complex integrated COTS-based development (Tran 1997).

The objective of the CISD model was to attempt to generalize the technical process of selecting, evaluating, and integrating COTS products to support COTS-based software development. The CISD model consists of three distinct phases (Tran 1997):

- Product Identification, which includes the process of collecting and understanding the overall system requirements, identifying and classifying COTS products in product sets, and prioritizing them for subsequent evaluation.
- Product Evaluation, which includes the process of integrating, evaluating, and comparing the product sets to select the most optimal combination for integration.
- Product Integration, which includes the building of all necessary software adapters and enhancements to the selected COTS product set to implement the required integrated system.

The model found some practical applications in Motorola's IRIDIUM system, MCI's TNMNVu system, and Mitsubishi's DiamondWeb system. The overall product selection, evaluation, and integration process is often iterative in nature and the number of iterations depends on the size of the system and the resources available. With the introduction of more advanced and comprehensive models such as CAP and EPIC, CISD is unlikely to play a pivotal role in any future COTS-based development. Nonetheless, CISD represents one of the pioneering models that advocates simultaneous definition and trade off of stakeholder requirements, availability of COTS components, and architectural requirements. It is fair to say that CISD laid the groundwork for the evolvement of COTS-centric models that led to the creation of CAP and EPIC.

2.2.3 COTS Acquisition Process (CAP)

First published in 2000 in an ESCOM Conference, CAP was created by a group of researchers in Siemens Health Services Group in recognition to the need for having a more practicable reference model for COTS-based development. Although OTSO and CISD did provide the basis for a more systematic and repeatable COTS-based development approach, a large part of the work remained theoretical. The details of how the individual criteria can be measured were not elaborated. As a result, the COTS project team often encounters difficulty in selecting the best alternative out of the increasing number of options. This has created a vacuum for the development of an efficient and systematic measurement-based COTS selection process that allows the COTS project team to conveniently screen, shortlist and rank the alternatives using a set of pre-determined criteria that can be objectively and unambiguously measured using pre-established quantitative or qualitative metrics.

The foundation of CAP lies with its focus on measurement-based decision-making that utilizes analytic hierarchy process. CAP consists of three process components (Ochs 2000):

- The CAP Initialization Component (CAP-IC)
- The CAP Execution Component (CAP-EC)
- The CAP Reuse Component (CAP-RC).

CAP Evaluation Taxonomy constitutes the core part of CAP-IC and CAP-EC. It breaks down the evaluation criteria into four main categories:

- Functional
- Non Functional
- Domain and Architecture

- Strategic

These categories are known as the 1st level categories. Each component of the 1st level categories is then broken down into the 2nd level categories and further into the 3rd level, and so on until an objective measure or metric can be assigned to evaluate the criteria. A measure is considered objective when there exists a well-defined method such as a formula or a pre-established protocol and a sufficiently accurate scale to interpret the result. To illustrate, CAP breaks down functional into suitability, accuracy, interoperability, and security. Suitability can be measured by Feature Implementation Coverage (FIC), a pre-defined quality metric.

The CAP Evaluation Taxonomy comprises a set of more than 100 pre-defined quality metrics, approximately 60 of them are derived from the ISO 9126 standard on product quality, and the rest are derived from expert interviews, literature reviews and applied research activities. CAP was evaluated in a case study at Siemens Health Services (SHS) over a 5 months period, from October 99 to February 00 to test the hypothesis that CAP will produce more systematic and reliable decisions for COTS components selection than any ad-hoc practice without significantly increasing the effort. The result of the case study demonstrated a reduction in total effort, which more than compensated for the additional cost incurred by the higher degree of formality with CAP.

Compared to CISD, CAP has a much narrower scope, which targets only the selection process. With its focus, CAP did an excellent job in putting theory into practice. Although COTS selection process was also covered by CISD, there was little detail of how the criteria can be measured. CAP, with its measurement-based Evaluation Taxonomy, provides the practitioner with a pragmatic solution that can be applied to formalize and improve the COTS selection process. The case studies did demonstrate a positive cost benefit with the use of CAP but the method that was used to quantify the return on investment remains subjective and debatable. Despite this shortcoming, CAP still provides one of the best methods in determining the best alternatives during the selection process.

2.2.4 Evolutionary Process for Integrating COTS Based Systems (EPIC)

EPIC is the synthesis of years of extensive research work that was undertaken by the SEI COTS-Based System Initiative in response to the escalating use of COTS in mission critical systems in government and private organizations. EPIC extends the Rational Unified Process (RUP) to provide a risk-based spiral development process that integrates COTS lessons learned, disciplined engineering practice, and end-user business processes to help organizations build, implement, and support solutions based on COTS and other pre-existing components. EPIC redefines acquisition, management, and engineering practices to more effectively leverage the COTS marketplace and other sources of pre-existing components through concurrent discovery and negotiation of diverse spheres of influence that includes user needs and business processes, applicable technology and components, the target architecture, and programmatic constraints (Albert 2002). The transformation from the traditional approach to the EPIC approach is captured in Figure 2.2.1(A).

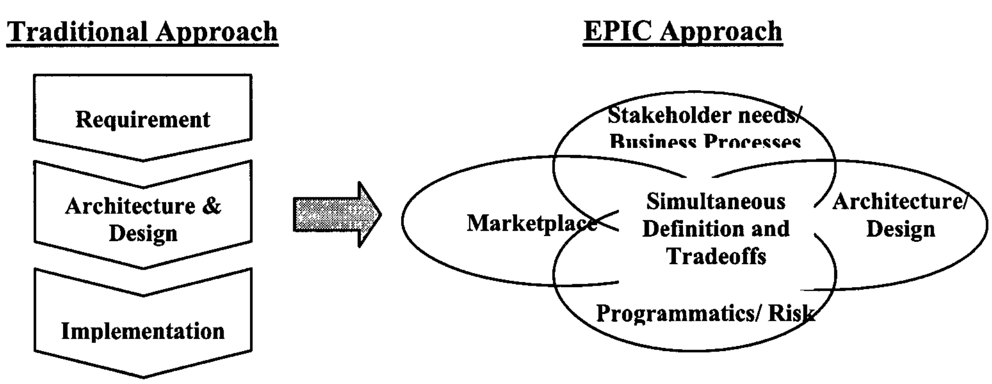

Figure 2.2.1(A): Required Approach for COTS-Based Systems

Each EPIC cycle consists of four phases: Inception, Elaboration, Construction, and Transition. The use of EPIC begins with the definition of a need for a new or changed capability and a commitment to provide the resources necessary to identify, acquire, build, implement, and support a solution that will deliver that capability, and ends only when the solution is retired or replaced with a new solution. Each EPIC phase consists of one or more iterations. The iteration begins with the development of a detailed plan and ends with assessing whether or not the objectives in that plan are met.

As demonstrated in Figure 2.2.1(A), the key to COTS approach requires the simultaneous acts of defining and trading off among the four spheres of influence. The tradeoffs are driven by the desire to leverage components from the marketplace. The four spheres represent competing interests that must be considered in forming a viable solution that effectively leverages COTS. These four spheres are simultaneously defined and traded through the life of the project because a decision in one sphere will likely constrain the decisions that can be made in another sphere.

Each EPIC phase consists of one or more iterations. As shown in Figure 2.2.1(B), each iteration begins with the development of a detailed plan and ends with assessing whether or not the objectives in that plan are met. Iteration planning uses the current understanding of risk to establish goals and objectives, and defines the specific tasks as well as the cost, schedule, and resources needed for the iteration.

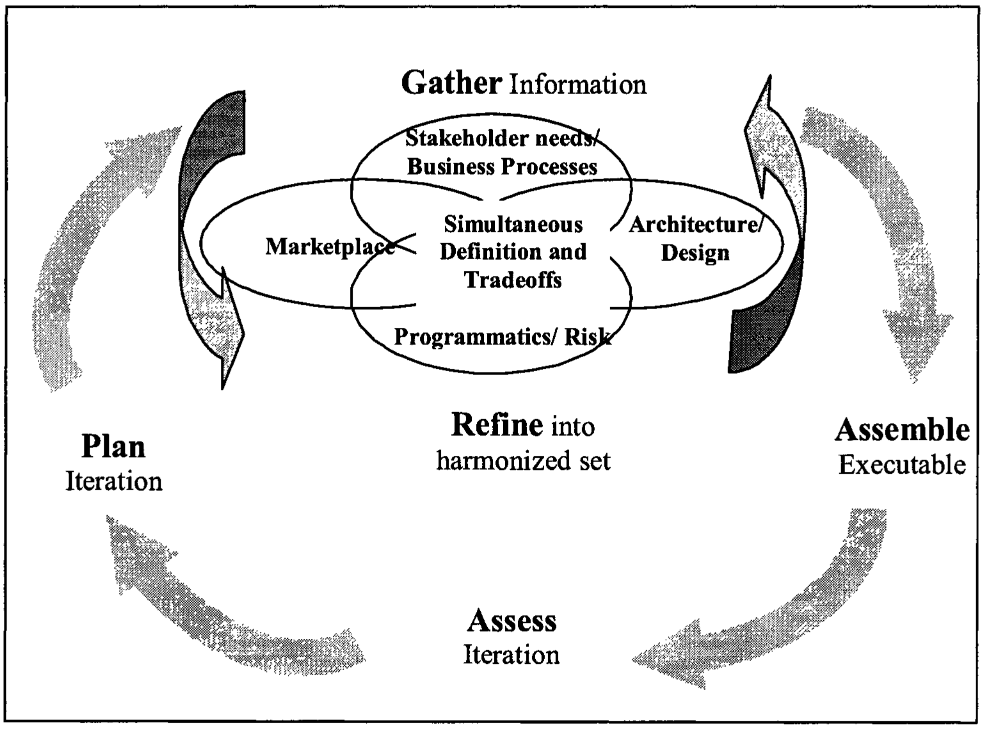

Figure 2.2.1(B): An Iteration in EPIC (Albert 2002)

Each iteration consists of the activities that continuously gather information from each of the four spheres and refine the newly gathered information through analysis and negotiation with affected stakeholders, to form the harmonized knowledge needed to assemble an Executable Representation of the solution. This knowledge is captured in artifacts that start at a very high level in early iterations and are expanded through cycles of gathering and refining across iterations as increasingly detailed information is harmonized. It may take many cycles of gathering and refining information within an iteration to produce knowledge sufficiently detailed and harmonized across the four spheres to meet the iteration objectives. Every iteration assembles an Executable Representation of the solution that exhibits the common understanding of the solution that has been achieved among affected stakeholders to that point and demonstrates the adequacy of the solution to meet the iteration objectives.

The four spheres continuously evolve through successive iterations. Each iteration is designed to meet specific objectives and will nominally take one to eight weeks to complete. As shown in Figure 2.2.1(C), EPIC iterations are managed by the four RUP phases (Albert 2002): Inception, Elaboration, Construction, Transition, and three associated anchor points:

- Life-cycle Objectives (LCO): When one or more candidate solutions that meet the high-level objectives for the solution are identified and concurred by the key stakeholders. The phase exit criteria and the fulfillment of the phase objectives are reviewed, stakeholder concurrence is validated, and approval to examine the most viable candidate solutions in greater depth is sought.
- Life-cycle Architecture (LCA): When all components have been selected and procured, any integration mechanisms to incorporate the components and any other components are validated, and the risk, cost and schedule for completion of the project have been predicted within an acceptable range.
- Initial Operational Capability (IOC): When a production-quality release of the solution is ready for fielding to at least a subset of the operational users as an initial fielding or beta test.

Each EPIC phase consists of one or more EPIC iterations. Iterations in each phase build on and strengthen stakeholder understanding of the available components and each component's impact on requirements and end-user business processes, architecture and design, and the cost, schedule, and risk of implementing the solution. The EPIC phases accommodate the continuous change induced by the COTS marketplace. Each phase has explicit objectives, activities, artifacts, and phase exit criteria. Each phase ends with an anchor point that provides an opportunity to review progress, ensure continued stakeholder commitment to the evolving solution, and to decide to proceed, change direction, or terminate the project.

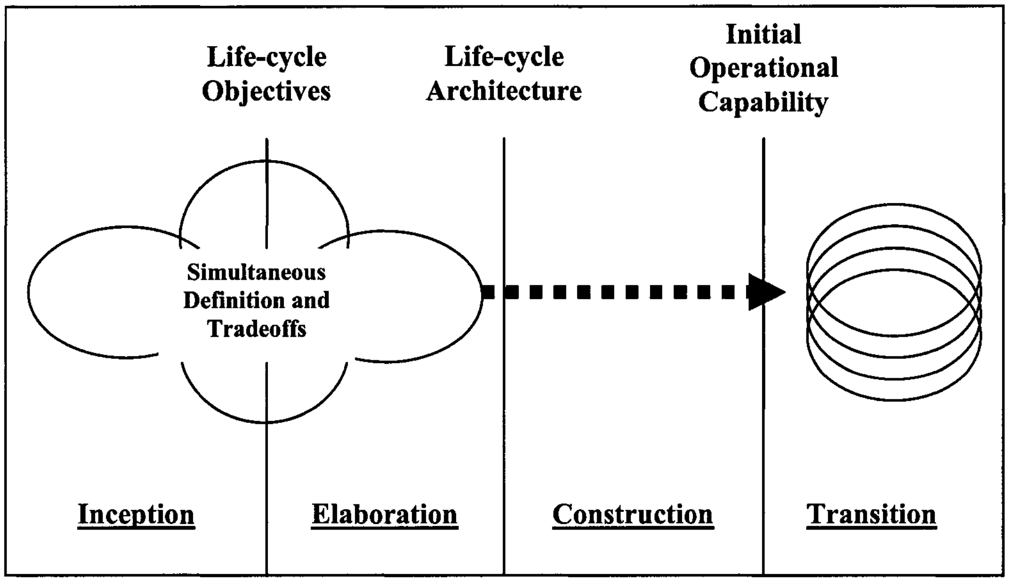

Figure 2.2.1(C): EPIC Phases (Albert 2002)

Unlike the other publications, EPIC is more than a revolutionary COTS framework. With over 250 pages of theories and artifacts, EPIC represents the gold standard for the study and practice of COTS-based development. It is by far the most comprehensive work available to date and has the widest scope. The conceptual framework presented in EPIC is general enough to be applicable in almost all instances in COTS-based development and the collection of artifacts is complete and comprehensive enough to be adoptable in most COTS-based projects. First released by the SEI as a technical report in November 2002, EPIC will continue to evolve in response to lessons learned and market environment to ensure its continued relevance to the dynamics and emerging demands of COTS-based systems.

2.2.5 COTS Acquisition Evaluation Process (CAEP)

CAEP is not a new COTS framework or acquisition process, it is a recapitulation of a COTS-based project carried out by the Financial and Business Services (FABS) and Information Technology (IT) departments at the SEI. The objectives of this technical paper are to provide a practical view of how the theoretical models are put into practice in a real life project, discuss the limitations of the existing models, and warn the readers about the critical challenges that were not anticipated in the theoretical models. Like CAP, this paper has a narrow scope covering only the selection process. Because each software system is unique, the project team had to modify the existing COTS selection process to adapt to the unique requirements of the systems. The team also suggested that there is no one model that fits all instances, therefore it is important for the project team to recognize the unique needs of the individual project and make the necessary modification to the existing framework to suit the situation.

CAEP breaks down the COTS selection process into three sub-processes; each sub-process is made up of a sequence of sub-tasks (Sai 2004). Running a COTS-based development project is a difficult and risky task. Despite the availability of references and reference models such as CISD, CAP, and EPIC, the contents can be intimidating to novice project managers. By transforming theories and frameworks into actionable processes and tasks, CAEP helps bridge the gap between theory and practice, and provides practitioners with a useful guide in coping with COTS-based development projects.

2.3 COTS Risk Management

The expansion in the use of COTS products has been accompanied by an increase in program failures. Many of these failures have been due to a lack of familiarity with the changed approach that COTS products demand. The research in COTS risk management is done in parallel with and independently from the research in COTS-based development models described in Section 2.1. Although both branches of research share a common goal, COTS risk management views COTS related issues from a distinct perspective and proposes a set of solutions that are different but complementary to COTS-based development models. The goal of COTS risk management research is to develop a tool to reduce the number of program failures attributable to COTS software. The two most widely known COTS-based risk management tools are:

- FAA Risk Mitigation Guide
- COTS Usage Risk Evaluation (CURE)

2.3.1 FAA Risk Mitigation Guide

Developed by the United States Federal Aviation Administration and documented in (Shaffer 2002), this guide is intended to capitalize on the lessons-learned from government and industry and document them in a practical manner within the context of an acquisition management process to more effectively acquire and provide life cycle support for COTS-based systems. The core contents include:

- The elicitation and elaboration of the top ten risk factors that are derived from past projects. These risk factors are:
    1. Rapid changes
    2. Different obsolescence impacts
    3. No access to source codes
    4. High life cycle costs
    5. Multiple configurations
    6. Different quality practices
    7. "As is" configurations
    8. Commercial standards
    9. Dependent on vendor for maintenance
    10. Information security susceptibility
- A detailed description of thirteen common risk-mitigating strategies that are developed in accordance to a "what, why, when, and how" approach.
- A systematic process that links the appropriate risk-mitigating strategies to the individual risk factors.

This guide has played and will continue to play a pivotal role in serving as a reference risk management model for COTS practitioners. Although some of the strategies have been presented in the earlier work, none was comprehensive and detailed enough to be considered practicable. Supported by the FAA, this guide will constantly be updated with new information and will continue to evolve to fit the demand of the market.

2.3.2 COTS Usage Risk Evaluation (CURE)

CURE was developed by the SEI in response to a call from the United States Department of Defense after experiencing several high profile failures in COTS based development. CURE is intended for use with any program office and contractor that are creating large-scale software systems that rely on COTS products. While the majority of its applications were done on government programs, CURE is not just limited to government projects but can also be used by any organization that is building COTS-based systems. CURE has been performed on the Department of Defense, other government agencies, and industrial organizations. The projects include weapons systems, information systems, and command, control, communication, and intelligence systems. Unlike EPIC, CURE is not aimed at bringing a philosophical change to the organization, but at diagnosing and fixing a problem before it gets out of control.

CURE is usually conducted at the early stage of a project. The CURE process consists of four core activities, which include:

1. Preliminary data gathering
2. On-site interview
3. Analysis of data
4. Presentation of results

Successful completion of these activities requires significant amount of skills and experience. Therefore, CURE is often carried out by specially trained individuals from organizations such as the SEI or its affiliates. Although the CURE process is not entirely proprietary, the information that is released to the public is far from sufficient for carrying out a successful CURE. Therefore, as of today, it is advisable to hire qualified individuals such as SEI certified CURE evaluators to carry out the evaluation. A detailed description of the CURE process is provided by (Carney 2003).

Based on the data collected on CURE conducted over the 1998-2002 period, COTS risks can be grouped into a collection of 21 risk areas that fall into four broad categories:

1. Programmatics and Management
2. Technical Areas such as Design, Evaluation, Testing
3. Mission and Stakeholders
4. Product-Specific Issues

CURE remains a tool that is only available to organizations that can afford the service. A systematic process to train and certify new CURE evaluators is still unavailable. Until such mechanism is put in place, CURE has yet to become a widely adoptable tool that is readily available to organizations and enterprises that practice COTS-based development. Despite this shortcoming, the CURE process proves to be an effective data collection mechanism on project risk, and has contributed to the development of a COTS risk database, statistical data, and lessons learned.

2.4 Summary of Literature Review

The use of COTS software continues to thrive and is expected to constitute at least 80% of all new software application developed in 2005, and the trend is likely to continue to rise (Albert 2002). This is due to (Shaffer 2002):

- Government acquisition policies that favor COTS, i.e. Federal Acquisition Regulation mandate in FAR 12.102.
- Increasing realization of COTS benefits, i.e. faster time to market, shared development cost, etc.
- Increasing maturity of COTS software marketplace. An article in CIO magazine stated that more than 2000 new software products hit the market every month.
- Emergence of COTS Development Model (EPIC), COTS Selection and Acquisition Framework (OTSO, CAP), and COTS Risk Management Tool

(CURE) to support and improve the formality of COTS-based development.

The search for probable COTS alternatives starts with the exploration of a set of COTS software alternatives that match the initial SRS. Researching and creating a list of probable alternatives is a difficult task. According to CAP and EPIC, a good starting point is to review industrial publications on software segmentation and search for the desired COTS components based on its segments. The information regarding software segmentation can be obtained from the annual software market research report published by Gartner Dataquest.

Once the alternatives are identified, a set of evaluation criteria, commonly known as COTS Vendor Evaluation Taxonomy (VET), is established. CAP contains a pre-defined VET. The VET typically consists of a set of quality metrics that focus on the product and cost aspects of the alternatives. The product aspect evaluates the functional, non-functional and domain/architecture requirements. The cost aspect evaluates the initial cost, integration cost, and cost benefits. Some organizations also include a third aspect, the service aspect, which evaluates the service and support capability of the alternatives. A diagrammatic representation of the commonly used VET is presented in Figure 2.4.1. The majority of the product metrics are derived from ISO 9126, the remaining metrics are derived from expert interviews, literature reviews, past experiences, and applied research activities (Ochs 2000). Despite the widespread use of VET in COTS selection process, there is no one set of VET that fits all organizations and all instances. The actual VET differs from one organization to another depending on the acquiring organization's internal practice, available resources and the criticality of the new system. According to (Kontio 1996), COTS selection process is rarely well defined in practice. Few organizations have detailed procedures to govern the selection and evaluation process, and the project personnel in charge often need to define their own approach.

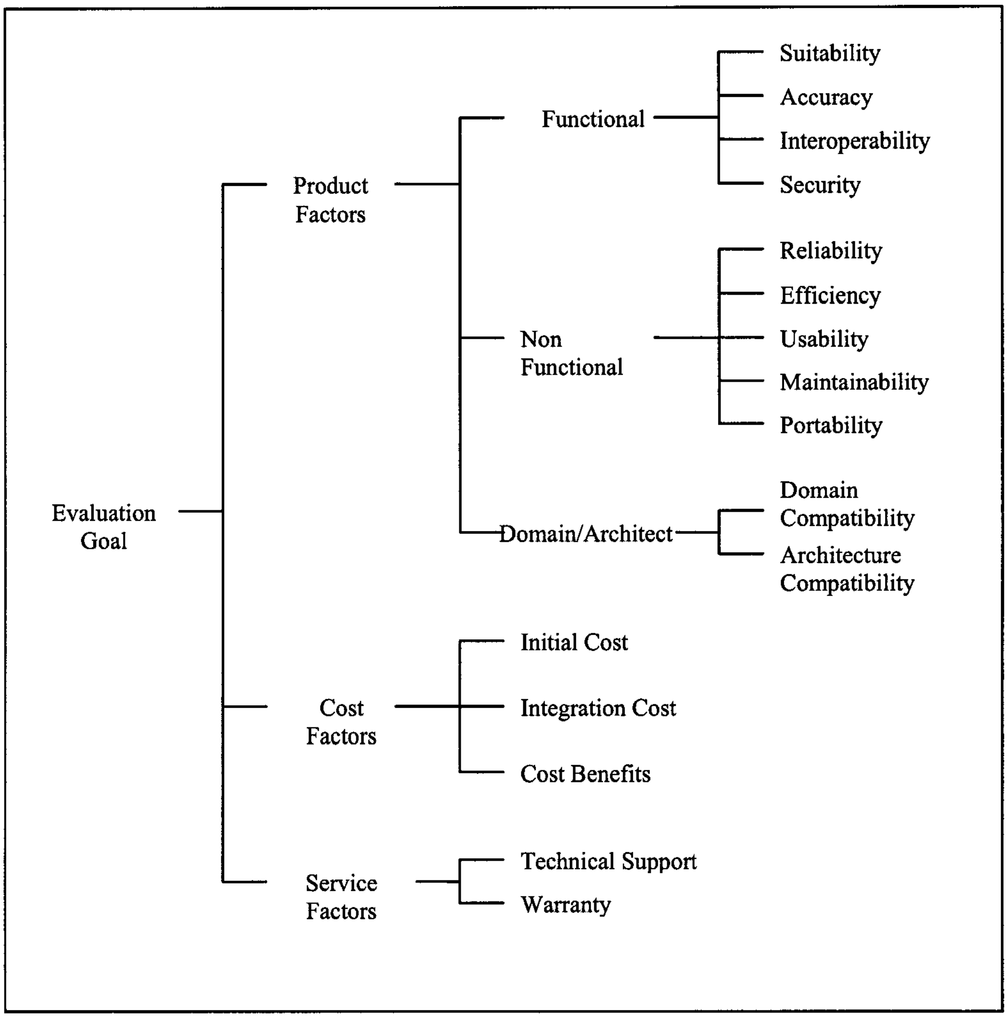

Figure 2.4.1: COTS Vendor Evaluation Taxonomy

2.5 CONCLUSION

Migrating to COTS based development is not merely a technical change. It has profound impact on business, acquisition, and management practices, and organizational structures. A general lack of experience and the absence of a dependable COTS process had led to some high profile failures in both government and non-government organizations. A well-publicized example is the 19 percent drop in quarterly income in 1999 at Hershey Food Inc. that is believed to be due to the complications of a newly installed COTS ERP system (Carney

2003). For every high profile disaster, there are hundreds or even thousands of unpublicized lower profile failures, which cost the industry over a billion dollar of economic loss. This has led to the creation of a mission to redefine acquisition, management, and engineering practices to more effectively leverage the COTS marketplace, a task that is led by the SEI, along with other key organizations associated with the government, academic, and civil agencies.

The state-of-the-art and state-of-the-practice of COTS related research have so far focused on improving the knowledge in the following area:

- Evaluating technologies and products: Developing evaluation criteria, conducting a satisfactory evaluation, and selecting viable technologies and products based on the criteria. Determining the amount of time to allocate for evaluation during the acquisition.
- Developing requirements: Developing techniques to optimize the trade-offs between COTS products, system architecture, and stakeholder requirements.
- Managing system and COTS product evolution: Learning how the development and maintenance of the systems will need to change as a result of the continual release of COTS product updates.

The research work is still in the early stage, and a large part of the effort is consolidated into a comprehensive COTS-based development model such as EPIC, and a few widely known selection models, namely OTSO and CAP, whose greatest contribution lies within a set of measurement-based Vendor Evaluation Taxonomy (VET). As summarized in Section 2.3, the existing VET is dominated by product centric factors such as functionality and cost. The systematic use of a VET, as documented in OTSO and CAP has brought in tremendous advantages over the previously undefined or even chaotic vendor selection process. Supported by a series of case studies, (Kontio 1995) and (Ochs 2000) claimed that the use of VET has turned in positive results in terms of net reduction in effort. In spite of these successes, there is an increasing concern over the scope and bias of the existing VET, which lacks the ability to evaluate the business viability of COTS vendors.

Business viability is not just limited to the ability of the vendor to stay in business; it includes all other non-product factors such as the vendor's strategic position, financial strengths, and business risks. For this thesis, these factors are collectively named Vendor Business Factors (VBF). A negative development in these factors is likely to have a significant impact on the vendor's existing product lines and customers, and future product development and releases. The potential damage to the acquiring organizations can be viewed from the following scenarios:

- The first scenario involves a disruption in the vendor's financial stability when they are still actively involved in the development or maintenance of the COTS based system. The impact to the acquiring organization can range from a minor hiccup in the schedule to a serious disruption in the business operation. The extent of the damage will be proportionate to the size and complexity of the system, and the criticality and ramification of the particular component. Although an alternative vendor can be considered, the process is complicated and risky. Despite the attempts of the COTS community to design COTS products to mimic the "plug and play" capability of the hardware world, the reality is COTS software seldom plug into anything easily, therefore it is usually difficult to obtain "spares" for COTS software components.
- The second scenario involves the vendor's failure or delay in introducing a next generation product that matches the market demands. A classical example was the rush to introduce a web-enabled version of virtually any software applications during the explosion of the Internet in the mid 1990s. The impact to the acquiring organization will depend on how important the desired features are to the organization's image and strategic position, and how big the switching cost is.

The need to evaluate VBF is hardly a new discovery; the SEI COTS Software Initiative Team had recognized the importance of VBF as early as 1998. (Wallnau 1998) proposed incorporating vendor financial well being into the COTS selection process. The authors further suggested using published financial reports such as Dun & Bradstreet Business Profile to assess the vendor's financial stability. This method has found limited applications because it raised more questions than answers. Although such reports represent a good starting point, they are insufficient by themselves. The method to interpret the financial measures remained unexplored, and what constitutes "good enough" remained unknown. Moreover, the COTS evaluation team is often dominated by software designers, who are usually ill equipped to perform the kind of business and financial analysis that is necessary to differentiate financially sound vendors from their financially troubled counterparts. Over the years, despite the surge in COTS use, there has not been much progress in the development of a more concrete method to evaluate VBF, resulting in a situation where vendor risk factors remain largely unknown. As indicated in FAA Risk Mitigating Guide and the lessons learned from CURE, most acquiring organizations are well aware of this risk, however, in the absence of a proper tool to assess VBF, these organizations are forced to resort to a strategy that either assumes the worst or neglects the risk altogether. By assuming the worst, the organization has to purchase redundant components or keep a team of spare developers, which can be a waste of resources if the likelihood of the vendor going out of business is nil or close to nil. On the other hand, by neglecting the risk, the organization may experience the full impact of the damage should the event materialize. Either extreme incurs cost that can be avoided or reduced if VBF were known.

The urgency to equip the existing COTS VET with the capability to assess VBF has become more compelling as a result of a series of events that prove the possibility of COTS vendor going out of business is far from remote.

- The Jul 2001 issue of the CIO Magazine indicated that the software industry is fundamentally unstable. Every year there is a hot new trend in IT. Whether it is ERP, e-business, CRM or wireless, vendors flood the market, hoping to make a killing off of a CIO's need to stay competitive. But the intense competition inevitably leads to a shakeout. The weaker companies' earnings start to slip, and in time they declare bankruptcy or are acquired by a larger competitor. Such was the case with Chicago-based System Software Associates (SSA), which left at least one ERP customer floundering when it declared bankruptcy in April 2000. Even industry leaders are not immune to financial trouble. After all, they are under even more pressure from Wall Street to meet earnings expectations, and this pressure sometimes leads companies to cook the books. Belgian speech recognition and translation technologies company Lernout & Hauspie, for example, made its revenues look better than they actually were, by recording sales before contracts were signed. This alleged financial fraud spurred an investigation by the Securities and Exchange Commission (SEC), which led to its filing for bankruptcy and its customer stranded with disrupted or even discontinued support (Levinson 2001).

- With the emergence of the Internet in the 90s, propriety information quickly turns into commodity, causing new technology to emerge and replace old technology at a pace so fast that it is unimaginable just a few years ago. This phenomenon has resulted in a higher rate of failure in software organizations. Based on the data published by BankruptcyData.com, an Internet based bankruptcy research firm, the number of publicly traded software companies that filed for bankruptcy increase from an average of one per year in the early 90s to an average of 5 per year in early 00s, a five fold increase over a decade.
- An article published by Gartner Dataquest indicated that dealing with dramatic drops in revenue, most software vendors are in major turmoil. Many vendors have been forced to reduce labor and cut research and development investments. The last six quarters have been tough and are showing little signs of relief, at least into the middle of 2003. Gartner Dataquest predicts that up to 50 percent of the software vendors in 2000 will be gone by the end of 2004 by merger, acquisition or demise. Acquisitions are slower than expected, as many vendors with cash are

being cautious. The number of vendors on the market declined by about 75 percent, primarily due to de-listing and bankruptcy (Correia 2003).

The objectives of this thesis are to:

1. Refine the state-of-the-art COTS selection process by extending the existing COTS Vendor Evaluation Taxonomy to include Vendor Business Factors.
2. Elaborate these factors to the level of details that the measures can be quantified and the results can be combined with the results of the existing measures to form the basis for screening and ranking COTS alternatives.
3. Establish a method to identify early signs of financially troubled vendors and develop a strategy to manage such vendors.
4. Establish a method to monitor the financial health of the existing COTS vendors and develop a strategy to protect the interest of the acquiring organizations.

3. PROPOSED SOLUTION

3.1 OVERVIEW

Vendor **E**conomics and **R**isk **Pro**filer (VERPRO) is developed to address the inadequacies of the existing COTS Vendor Selection Process and to fulfill the objectives of this thesis as stated in Section 2. VERPRO is applied in the following activities:

1. Identification of the top ranked candidate(s) from a list of COTS alternatives in the selection process.
2. Evaluation of the business risks of the top ranked candidate(s), and formulation of an appropriate risk management strategy. Because of the inherent instability in software industry, even the top ranked alternative is not immune to financial troubles; therefore, after singling out the best alternative, it is essential to assess its financial health before finalizing the risk management strategy. The assessment can be performed by comparing the financial profile of the best alternative to a reference financial profile that is derived from published financial data. The details about how the comparison is performed and how the reference data is derived are covered in Sections 3.3 and 3.4.
3. Monitoring of the business risks of the existing COTS vendors, and assessment of the adequacy of the existing risk management strategy. Because of rapid changes in technology, the financial health of a vendor may deteriorate over time. As a result, after qualifying a vendor, it is essential to periodically monitor its financial result to ensure its continual ability to support the system and the adequacy of the existing risk management strategy to cope with the vendor. Financially strapped vendors may result in rapidly deteriorating service levels and generate excessive levels of customer dissatisfaction. Therefore, it is essential for IT executives to conduct periodic performance review meetings with major COTS vendors to analyze their financial conditions, review their achieved versus guaranteed performance, and discuss any backup strategies if necessary. These meetings should be considered as important

activities to help protect the acquiring organizations from unpredictable situations that impact network performance and prevent reliable user access to business applications. The details about the monitoring process and the associating risk management strategy are covered in Section 3.4.

VERPRO expands the existing COTS Vendor Evaluation Taxonomy (VET) as depicted in Figure 2.3.2 to include Vendor Business Factors (VBF); the resulting VET is presented in Figure 3.1.1. VERPRO VET is made up of four main factors:

1. Product: Product factors, which are largely derived from ISO 9126, have been covered in considerable details in CAP and will not be the focus of this thesis.
2. Cost: Cost factors, which measure the total cost of ownership of the individual COTS alternatives, are evaluated using COCOTS (Boehm 1998) and will not be the focus of this thesis.
3. Service: Service factors, which measure the service level of the individual COTS vendors, have been briefly described in CAP and will not be the focus of this thesis.
4. Business: Business factors, which measure the individual COTS vendors' strategic position, financial strengths, and business risks, will be the main focus of this thesis.

VERPRO VBF consists of three key elements:

1. Product Leadership: Relates to the ability of the vendor to produce next generation products that meet future market demands.
2. Financial Strength: Relates to the vendor's short-term liquidity, long-term profitability, and access to capital.
3. Business Risk: Relates to the internal and external forces that influence the strategic position of the vendor.

These elements are collectively named VERPRO Business Factors and the acronym VBF is used as a plural term. The definition and derivation of these factors are discussed in detail in Section 3.3.

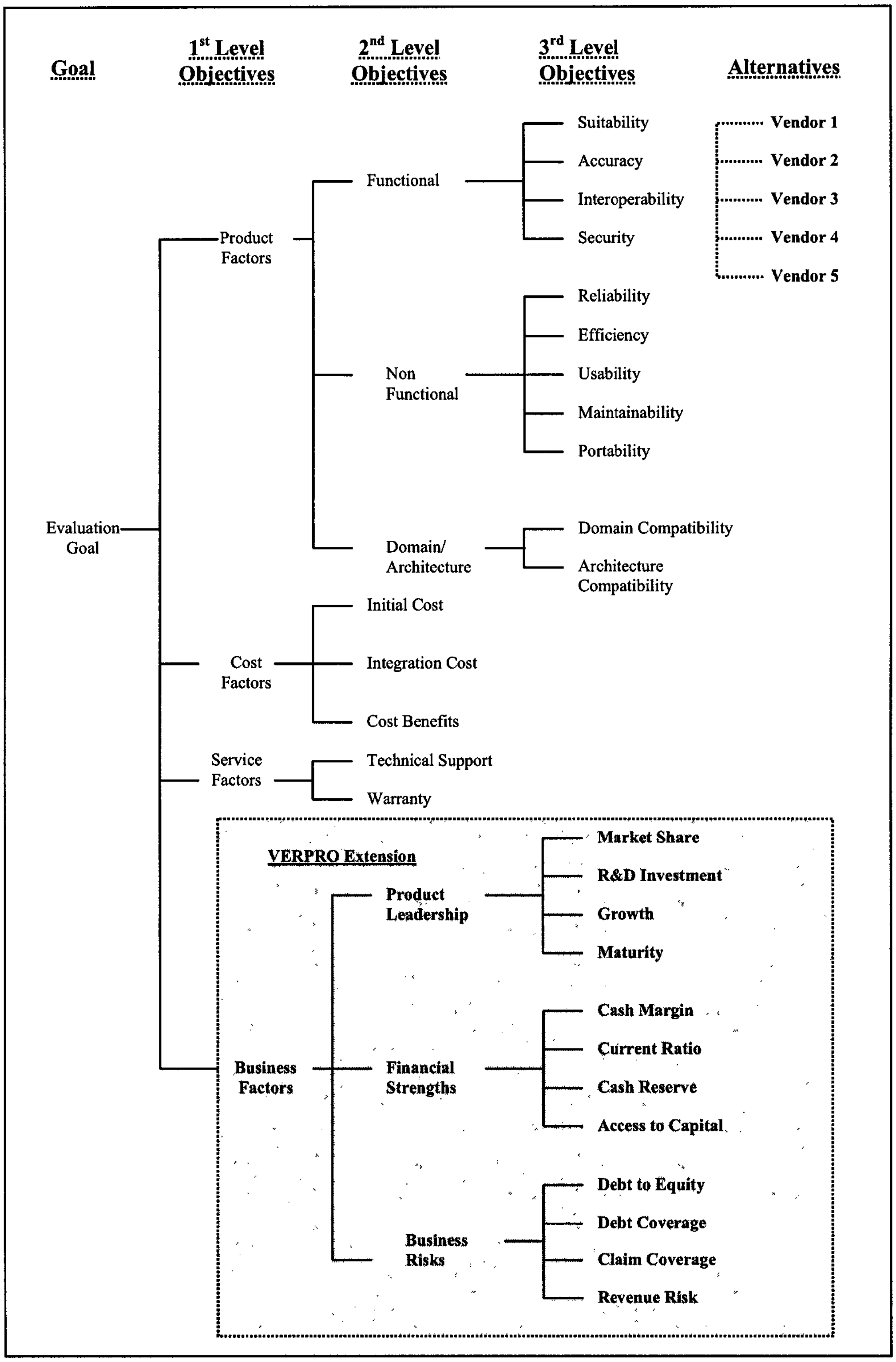

Figure 3.1.1: VERPRO Vendor Evaluation Taxonomy (Yeoh 2004)

VERPRO consists of three major elements:

1. Input Elements
    a. System Requirement Specification (SRS)
    b. Vendor Evaluation Taxonomy (VET)
    c. COTS Alternatives
    d. Supporting Data
2. Execution Elements
    a. COTS Acquisition Team
    b. Multi-criteria decision making tool
3. Output Elements
    a. Numerical score of the alternative
    b. Numerical rank of the alternative

A high level diagrammatic representation of VERPRO is presented in Figure 3.1.2.

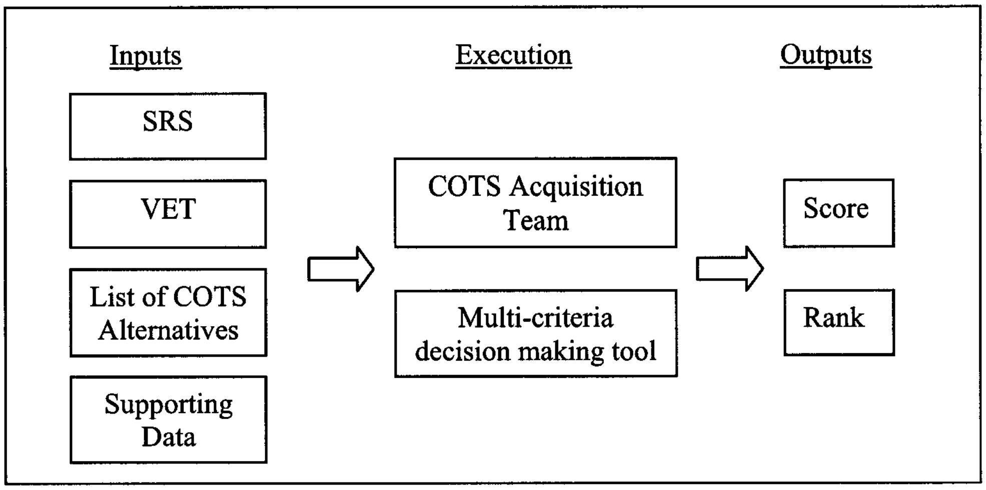

Figure 3.1.2: VERPRO I High Level Process Flow

VERPRO Multi-criteria decision making tool is built from a multiple-objective, multiple-hierarchy decision making model that is based on the Analytic Hierarchy

Process (AHP). AHP application for VERPRO is covered in detail in Section 3.2. As indicated in Figure 3.1.1, VERPRO Decision Making Model consists of:

1. Evaluation Goal
2. 1st Level Objectives
3. 2nd Level Objectives
4. 3rd Level Objectives
5. Alternatives

The evaluation goal is decomposed into the 1st Level Objectives, which consists of Product, Cost, Service and Business factors. These 1st Level Objectives are too high level for performing any objective assessment. Therefore, each 1st Level Objectives are decomposed into the 2nd Level Objectives, which are still too high level and are further decomposed into the 3rd Level Objectives. The decomposition will continue until an objective measure can be assigned to evaluate the alternatives. A measure is considered objective when there exist a well-defined method such as a formula or a pre-established protocol and a sufficiently accurate scale to interpret the result. For consistency in nomenclature, an objective measure will thereafter be referred to as a metric.

VERPRO Decision Making Model follows a bottom up approach and is modeled after the Decision Hierarchy developed by Thomas Saaty at Wharton School of Business. The alternatives are first compared in a pairwise fashion against the lowest level objectives. At each level in the hierarchy the relative importance or priority of the objectives is assessed by comparing them in pairs. Finally, the global priorities for each alternative are summed to yield the overall priorities. The most preferred alternative is the one with the highest priority. The decomposition of VBF is illustrated in Table 3.1.1.

1st Level Objective	2nd Level Objective (VBF)	3rd Level Objectives (VBM)	Alternatives
Business	Product Leadership	Market Share	Vendor 1
			Vendor 2

1st Level Objective	2nd Level Objective (VBF)	3rd Level Objectives (VBM)	Alternatives
			Vendor 3
			Vendor 4
		R&D Budget	Vendor 1
			Vendor 2
			Vendor 3
			Vendor 4
		Revenue Growth	Vendor 1
			Vendor 2
			Vendor 3
			Vendor 4

Table 3.1.1: Decomposition of VERPRO Business Factors

To derive the ranking, the alternatives are compared in pairs with respect to the 3rd Level Objectives. In VERPRO, the 3rd Level Objectives are named VERPRO Business Metrics (VBM). For each metric, the comparison is scored using a nine-point scale. The justification for using a nine-point scale is presented in Section 3.2. The scoring is to be substantiated by either hard or soft data that characterize the individual metric. For example, to compare how Vendor 1 performs relative to Vendor 2 with respect to Market Share, market share data for both vendors are obtained and analyzed before a score is assigned. Subsequently, the relative importance or priority of the metric is determined using the same nine-point scale. For example, how important Market Share is relative to R&D Budget with respect to Product Leadership is determined. The same exercise is repeated until the priorities of the objectives in all hierarchies are determined. All judgments are made by the COTS Acquisition Team in consultation with an experienced Financial Analyst. Finally, the overall rating for each alternative is synthesized using AHP. The computation is supported by a commercial tool known as Expert Choice.

Section 3.2 provides the background of some common decision making methods in COTS selection, justifies why AHP is selected as the de facto method for VERPRO and describes in detail how AHP is applied in VERPRO. Section 3.3 explains in detail how VBF are selected, quantified, and interpreted. These topics include the definition of the rating process, the search for the relevant data, the justification of the rating scale, and development of the rating guidelines. At the same time, a brief introduction to financial accounting is also provided as the preliminary education for the understanding of the quantification of VBF. Section 3.4 covers the development and interpretation of financial reference profiles, and the associating strategies for coping with a wide spectrum of financial conditions. In Section 4, a worked example is presented to illustrate the application of VERPRO, discuss its strengths and opportunities for improvement.

3.2 VERPRO DECISION MAKING MODEL

VERPRO Decision Making Model is based on the Analytic Hierarchy Process that was developed in the early 70s at the Wharton School of Business by Thomas Saaty (Saaty 1980). The purpose was to develop a theory and provide a methodology for modeling unstructured problems in the economic, social and management sciences. The resulting model allows decision makers to model a complex problem in a hierarchical structure showing the relationships of the goal, objectives, sub-objectives, and alternatives.

The basic model is analogous to a pie chart; the entire pie represents the goal of the decision problem. The pie is organized into wedges, where each wedge represents an objective contributing to the goal. AHP helps determine the relative importance of each wedge of the pie. Each wedge can then be further decomposed into smaller wedges representing sub-objectives. Finally, wedges corresponding to the lowest level sub-objectives are broken down into alternative wedges, where each alternative wedge represents how much the alternative contributes to that sub-objective. By adding up the priority for the wedges for the alternatives, we determine how much the alternatives contribute to the organization's objectives. A diagrammatic representation of the model is presented in Figure 3.2.1.

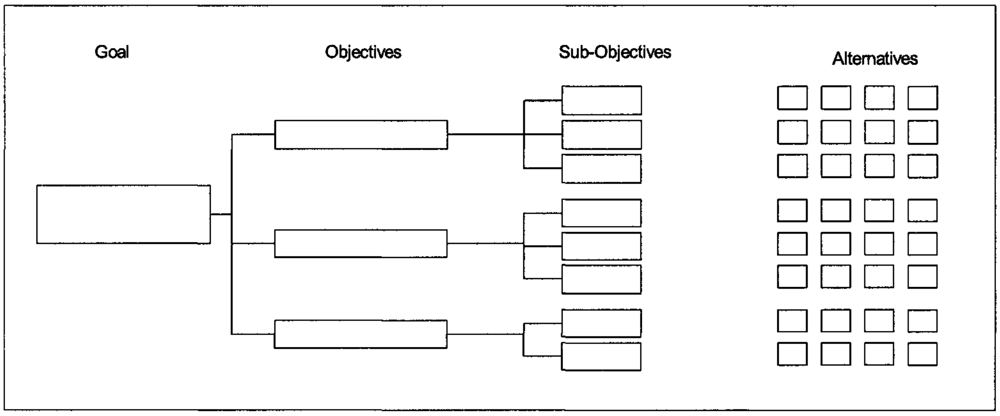

Figure 3.2.1: Decision Hierarchy (Saaty 1980)

In Section 3.2.1, a brief description of measurement theory is covered. The purpose of this section is to provide a basic understanding of AHP. In Section 3.2.2, alternative decision-making methods for COTS selection, and their pros and cons are discussed. In Section 3.2.3, a detailed description of AHP, as well as how it is applied in VERPRO is provided.

3.2.1 Measurement Theory

There are four levels of measurement:

1. Nominal
2. Ordinal
3. Interval
4. Ratio

Nominal numbers, the lowest level in terms of the meaning conveyed, are just numerical representations for names. Nominal numbers are used for identification purposes only and imply nothing about the ordering. Telephone numbers and social security numbers are nominal. Ordinal numbers implies an order or ranking among elements but nothing more. It does not imply anything about the differences or intervals between items. The order can be increasing or decreasing but ordinal data cannot be added or multiplied. Interval level data possesses the meaning of Nominal and Ordinal data, plus the meaning about the intervals between objects. Corresponding intervals on different parts of an interval scale have the same meaning. Interval data can be added or multiplied. However, after adding interval level data, one cannot infer that a total of 100 is twice as good as a total of 50. Ratio level data is the highest level, having Nominal, Ordinal, and Interval properties, as well as the property of ratios. Corresponding ratios on different parts of a ratio scale have the same meaning. A ratio scale is often defined as one having a true zero point (Forman 2002).

3.2.2 Alternative Decision Making Methods for COTS Selection

The fundamentals of good decision-making are a clear understanding of the decision itself and the availability of properly focused information to support the decision. Decision-making techniques help with both these problems. However, the techniques should be thought of as aids to decision-making and not the substitutes for it. Numerous decision-making techniques have been proposed as effective methods of ranking COTS alternative in the selection process. One of the most widely adopted techniques is the Weighted Scoring Method (WSM).

A typical WSM consists of a matrix with criteria listed in columns and alternatives listed in rows. The criteria are typically assigned weights using a 0 to 10 or 0 to 100 scale. The alternative is then scored against each criterion; the score is then multiplied by the criterion's weight and finally summed to give a total score for each alternative. This score represents the overall preference for the alternative. There are several practical difficulties with this method. First, when assigning weights, what do the numbers really mean? On a scale of 0 to 100, what is an 80, and what is a 40? Is the score 80 twice as important as 40? If the answer is yes, then the weights possess the ratio scale property. However, experiments have proven that the human brain is limited in short-term memory capacity and discrimination ability to about nine things. Therefore, we cannot be consistent in our judgment when dealing with more than nine criteria. Another problem with WSM involves dealing with orders of magnitude. For example, if one criterion is assigned a .02 on a scale of 0 to 10, and another is assigned a 9, do we really mean that one is 450 times more important than another? Our ability to accurately compare things that differ by such order of magnitude is not nearly as good as our ability to compare things that differ by an order of magnitude, which is about from 1 to 9 (Forman 2002), (Saaty 1980).

Another difficulty with WSM is the arbitrary assignment of weights and scores. For example, when we assign a score of 8 for customer satisfaction, what does the 8 really mean? Why not a 7, or a 9? This difficulty is due to two causes. The first is our attempt to assign weight and score in an absolute fashion rather than a relative fashion. The second is the implied precision of numbers in situations where the precision is not justifiable. Justification would be much easier if a less precise way is used for expressing judgments, such as words instead of numbers. Words are often easier to justify than numbers. For example, if we say alternative A is 3 times more preferable than alternative B, it is difficult to justify why it is exactly 3? Why not 2.9, or 3.1? But if we say A is moderately more preferable than B, this can be justified with a variety of arguments or even hard data. But how do we quantify verbal judgments such as moderate? Words have different meanings to different people (Forman 2002).

Despite the widespread use of WSM, it is flawed when applied in COTS selection process. Decision-making in COTS selection is a very complex process that combines probability judgments that may be affected by the evaluator's beliefs and underlying preferences. Some factors that give rise to problems in evaluating and assessing COTS software are:

- Large number of component attributes or features that have to be considered
- Various combinations of hardware platforms, operating systems and application software need to be considered
- Rapid technological changes in all aspects of computing, the business environment and the needs of the users

- Most users lack the technical expertise or time to develop criteria, measurements and testing procedures for performance assessments and to conduct the actual evaluations
- Great variations in performance between the attributes of each component and across the components for each attribute.

Furthermore, selecting the best COTS product from a number of potential alternatives is a complex decision making process that may include conflicting quantitative, qualitative criteria and multiple decision makers. The problems can be summarized in the following points:

- Selection decisions are invariably multileveled and multidimensional in nature.
- Decisions involve information that comes from different sources.
- Decisions are governed by multiple measures of merit and performance.
- All the information required to make a decision may not be available.
- Some of the information used in making a decision may be hard, that is, based on scientific principles and some information may be soft, that is, based on the selectors judgment and experience
- The problem for which a decision is being made is invariably loosely defined and open and is characterized by the lack of a singular, unique solution.
- The decisions are less than optimal and represent satisfying solutions, that is, not the 'best' but 'good enough'.

3.2.3 Analytic Hierarchy Process (AHP)

AHP allows for the application of data, experience, insight, and intuition in a logical and thorough way. AHP enables decision-makers to derive ratio scale priorities or weights as opposed to arbitrarily assigning them. In so doing, AHP not only supports decision-makers by enabling them to structure complexity and exercise judgment, but also allows them to incorporate both objective and subjective considerations in the decision process. AHP is a compensatory decision methodology because alternatives that are deficient with respect to one or more objectives can compensate by their performance with respect to other objectives.

3.2.3.1 Advantages of AHP

AHP addresses WSM's issues by structuring complexity as a hierarchy and by deriving ratio scale measures through pairwise relative comparisons. The pairwise comparison process can be performed using words, numbers, or graphical bars, and typically incorporates redundancy, which results in a reduction of measurement error and producing a measure of consistency of the comparison judgments. Human beings are much more capable of making relative rather than absolute judgments. The use of redundancy permits accurate priorities to be derived from verbal judgments even though the words themselves are not very accurate. Therefore, we can use words to compare qualitative factors and derive ratio scale priorities that can be combined with quantitative factors.

By using AHP pairwise comparison process, priorities are derived from a set of judgments. While it is difficult to justify weights that are arbitrarily assigned, it is relatively easy to justify judgments by hard data, knowledge or experience. Furthermore, the priorities that are derived from judgments automatically incorporate the necessary non-linearity in measuring utility. For example, when considering a vehicle for city driving, the preference for a vehicle with a top speed of 40 miles per hour is probably more than twice that of a vehicle with a top speed of 20 miles per hour. But the preference for a vehicle with a top speed of 100 miles per hour would be much less than twice as preferable as a vehicle with a top speed of 50 miles per hour (Forman 2002).

The conscious mind constantly attempts to understand what is perceived by relating it all together in some coherent way. For example, if we say that $A > B$, $B > C$ and $C > A$, we have been inconsistent. Consistency of judgment follows this simple transitive property. But we are very seldom perfectly consistent in making comparative judgments, particularly when we deal with intangibles that have no scales of measurement. And, we should not expect to be totally consistent. The real world often lacks consistency, and we must be able to reflect that in our models. For example, Team A can beat Team B, and Team B can beat Team C, yet Team C might then beat Team A (Forman 2002). Therefore, it is important to emphasize that the objective is to make good decisions, not to be perfectly consistent. It is easy to make perfectly consistent judgments that are nonsensical and result in terrible decisions. Good decisions are most often based on consistent judgments, but the reverse is not necessarily true.

AHP allows inconsistency but provides a measure for it, called the Consistency Ratio (CR), in each set of judgments. Contrary to what its name suggests, CR is a measure of inconsistency, the bigger the ratio, the more inconsistent the set of the judgment is. A CR value of 1 indicates a random judgment. CR is defined as the ratio of the Consistency Index (CI) for a particular set of judgments to the average Consistency Index for random comparisons for a matrix of the same size. CI is an important by-product of the process of deriving priorities based on pairwise comparisons. The mathematical expressions for CI are provided by (Saaty 1980). The computations for CI and CR are supported by a commercial tool for matrices up to 15x15. According to (Saaty 1994), if CR is 0.10 or less, the inconsistency is generally considered tolerable. If CR is considerably more than 0.10, then a re-examination of the judgments is recommended. The degree of inconsistency that indicates a significant problem depends on the specific situation where the model is applied.

The most common causes of inconsistency are:

- Lack of information
- Lack of concentration
- Inherent inconsistency in most real world situations
- Data entry error when a commercial tool is used

If one has little or no information about the factors being compared, then judgments will be random and a high inconsistency ratio will result. Lack of concentration occurs as a result of fatigue or a lack of interest in the decision. The real world is rarely perfectly consistent (Forman 2002). As a result, real world inconsistencies do exist and will appear in our judgments. It is important to note that consistency is not the only goal of a decision-making process. A low inconsistency is necessary but not sufficient for a good decision. It is possible to be perfectly consistent but consistently wrong. Thus, it is more important to be accurate than consistent (Saaty 1980).

3.2.3.2 AHP's Application in VERPRO

In general, AHP has two practical applications. The first application is to assign weights to a set of pre-determined objectives and sub-objectives, and then make a decision out of several scenarios or alternatives. For example, weights are assigned to several criteria in the personnel selection exercise. Then, the assessors can score each candidate with respect to the weighted criteria and choose the one with the highest total score. VERPRO falls into this category of application. The second application is to prioritize objectives in order to identify the key objectives. This application is useful for organizations in determining the allocation of resources. When an organization works on several projects simultaneously, ranking the relative importance level of individual tasks may help better allocate the resources in order to minimize the costs for storage, extra transportation, and risks of out of stock and stoppage.

The application of AHP involves seven major steps:

1. Define the goal.
2. Decompose the goal into a hierarchy of objectives and sub-objectives until no further decomposition is needed.
3. Perform pairwise comparison for the alternatives with respect to the lowest level objectives.
4. Perform consistency measure to evaluate the consistency of the judgments.
5. Using pairwise comparison, derive the priorities (relative weights) of the objectives for each level of the hierarchy.
6. Multiply the score of each alternative by the relative weights of the objectives and sum up to obtain the total score.

7. Perform sensitivity analysis.

Step 1: Define the goal

In order to obtain an accurate decision, we have to unambiguously define our goal. In VERPRO, the goal and objectives are defined by the stakeholders and documented in the SRS via a series of intensive elicitation process. For example, the evaluation goal of a COTS selection process is to identify the alternative that best satisfies the SRS with the lowest possible risk.

Step 2: Decompose the goal

In VERPRO, the goal is decomposed into four 1st level objectives, and each objective is further decomposed until an objective measure can be assigned for judging the alternatives. The decomposition for VBF is represented in Table 3.1.1. In VEPRO, the 2nd Level Objectives are known as the VBF and the 3rd Level Objectives are known as the VBM. Section 3.3 discusses why these metrics are selected and what other secondary metrics can be used if these core metrics are unable to provide conclusive results. Section 3.3 also explains how these metrics are measured and provides details of where the relevant data and facts are obtained to support the measuring. Finally, it is important to take note that although these metrics can be quantified with hard data, the final judgments should not be derived directly from the data but made by the COTS Acquisition Team in consultation with an experienced financial analyst. The purpose of the data is for justifying the judgments and not vice versa. This is due to the non-linearity of the metrics. For example, an R&D budget of 10% is probably twice as desirable as a budget of 5% but a budget of 20% is probably less than twice as desirable as a budget of 10% because historical data has indicated that a 10% budget is sufficient and the incremental return diminishes as the budget goes beyond 10%.

Step 3: Perform pairwise comparison

In this step, the alternatives are compared in a pairwise fashion with respect to VBM. Before performing the comparison, we have to select an appropriate scale and ensure that the number of alternatives does not exceed nine. This is due to the limitation in our ability to process more than 7 plus minus 2 items in our short term memory (Saaty 1980) and thus the consistency of our judgments diminishes as the number of alternatives exceeds our natural limit. If we have more than nine alternatives, we should attempt to establish some preliminary criteria to screen out the alternatives that do not meet the "must" requirements. A nine-point scale is used to rate the alternatives. The use of a nine-point scale is justified by experiments performed by Saaty and the advantages of a nine-point scale over other alternative scales are presented by (Saaty 1980). A nine-point pairwise comparison scale is presented in Table 3.2.1.

Numerical Value	Verbal Definition	Explanation
1	Equal importance	Two elements contribute equally
3	Moderate importance of one over another	Experience and judgment favor one element over another
5	Strong importance of one over another	An element is strongly favored
7	Very strong importance of one over another	An element is strongly dominant
9	Extreme importance of one over another	An element is favored by at least an order of magnitude
2,4,6,8	Intermediate values	Used to compromise between two judgments

Table 3.2.1: Pairwise Comparison Scale (Saaty 1980)

Pairwise comparisons can be made in three contexts:

1. Importance: when comparing objectives or players with respect to their relative importance.
2. Preference: when comparing the preference for alternatives with respect to an objective.

3. Likelihood: when comparing uncertain events or scenarios with respect to the probability of their occurrence.

For example, when we compare four alternative vendors (A, B, C, D) with respect to product leadership, a 4x4 matrix will be formed. Let us assume that after thorough investigation, the team concluded that A is 3 times more preferable than B and C, and 5 times more preferable than D, B is equal to C and 5 times more preferable than D, and C is equal to D, the scores are entered into the matrix as follows:

	A	**B**	**C**	**D**
A		3	3	5
B			1	5
C				1
D				

The next step is to fill out the diagonals with 1 as A is equal to A, B is equal to B, and so on. Finally, we fill out the remaining numbers with the reciprocals. The resulting matrix is as follows:

	A	**B**	**C**	**D**
A	1	3	3	5
B	1/3	1	1	5
C	1/3	[illegible]	1	1
D	1/5	1/5	[illegible]	1

The following step is to compute a vector of priorities from the above matrix. In mathematical terms, the vector of priorities is obtained by normalizing the principal eigenvector (Saaty 1980). In actual practice, a commercial tool (Expert Choice) is used to perform the computations. By using Expert Choice, the priorities of the alternatives with respect to product leadership are given below:

	Priority
A	0.509

	Priority
B	0.243
C	0.155
D	0.094

In the absence of a commercial tool, a rough estimate of the priorities can be obtained by dividing the elements of each column by the sum of that column and then add the elements in each resulting row and divide this sum by the number of elements in the row. Three other estimation techniques are provided by (Saaty 1980).

Step 4: Perform consistency measure

This step is used to evaluate the consistency of our judgments, which is denoted by the consistency index (CI) and the consistency ratio (CR). For example; if A is 3 times more dominant that B, and 6 times more dominant than C, then A = 3B, and A = 6C, therefore, 3B should equal 6C, B should equal 2C, and if the numerical judgment in the (B,C) position is different from 2, then the judgment would be inconsistent. CR is defined as the ratio of the Consistency Index (CI) for a particular set of judgments to the average Consistency Index for random comparisons for a matrix of the same size. The mathematical expressions for computing CI are provided by (Saaty 1980). Like step 3, the computation is supported by Expert Choice. Certain CR thresholds need to be achieved in order for the judgments to be considered consistent. If the CR exceeds the threshold value, Step 3 will be re-examined to identify the sources of inconsistency and mitigation actions will be taken if deemed necessary. The guidelines below on CR thresholds are provided by (Saaty 1994):

1. CR threshold of 0.05 for a 3x3 matrix
2. CR threshold of 0.08 for a 4x4 matrix
3. CR threshold of 0.10 for matrices of higher order

Step 5: Estimate relative weights of the objectives

In this step, the priorities of the objectives for each level of the hierarchy are derived from pairwise comparison. The judgments are to be made based on the relative importance of each objective with respect to the higher-level objective. For example, to derive the priorities of the 2^{nd} level objectives: product leadership, financial strengths, and business risk, we need to consider the relative impact of these 2^{nd} level objectives to the overall system in the event of a negative development. The relative impact differs greatly from one organization to another. The difference is depicted by the following scenarios:

- If you are a big organization with significantly stronger financial resources than the vendor, vendor financial strengths might be lower in relative importance to you because you have the option to provide financial support to the vendor if the negative impact to you is big enough to justify the action.
- If you are a small to medium size organization that outsources the critical part of the development and maintenance process to the vendor, vendor financial strengths might be higher in relative importance to you because you do not have the resources to influence the financial position even if the impact to you is big.

Once these considerations are thoroughly assessed, pairwise comparisons are performed in a similar fashion as in Step 3, and CI is computed to check the consistency of the judgments. Like Steps 3 and 4, the computations are supported by Expert Choice.

Step 6: Compute the overall score

This step is also known as synthesis. Synthesis is the process of weighting and combining priorities throughout the model after judgments are made to yield the final result. Global priorities are obtained for hierarchies throughout the model by applying each hierarchy's local priority and its parent's global priority. The global priorities for each alternative are then summed to yield overall priorities. The most preferred alternative is the one with the highest priority. Like the other steps, the computations are supported by Expert Choice.

Step 7: Perform sensitivity analysis

Sensitivity analysis is performed to see how well the alternatives performed with respect to each of the objectives as well as how sensitive the overall rating are to slight changes in the judgments. For example, we can study the changes of the overall rating with respect to changes in the relative importance of the objectives.

3.2.3.3 Conclusion

All theories are based on axioms and guiding principles. The theory will be more general and applicable if the axioms are simpler and fewer. AHP is based on four key axioms (Forman 2002), which include:

1. Reciprocity
2. Homogeneity
3. Bottom Up
4. Justifiable Outcome

The reciprocity axiom states that if A is 5 times larger than B, then B is one fifth as large as A. The homogeneity axiom states that the elements being compared should not differ by too much, else there will tend to be larger errors in judgment. When constructing a hierarchy of objectives, one should attempt to arrange elements in a cluster so that they do not differ by more than an order of magnitude. Judgments beyond an order of magnitude generally result in a decrease in accuracy and increase in inconsistency. The bottom-up axiom states that judgments about the elements in a hierarchy may depend on lower level elements. For example, in choosing a laptop computer, the relative importance of speed vs. weight might depend on the specific alternatives being considered, if the alternatives were about the same weight but differed greatly in speed, we might then infer that speed is more important than weight. The justifiable outcome axiom states that individuals who have reasons for their beliefs should make sure that their ideas are adequately represented for the outcome to match these expectations.

3.3 VERPRO Business Factors (VBF)

The factors that influence the success of a business are characterized by its strategic resource base, which consists of its:

- Tangible and intangible assets
- Financial assets
- Human assets
- Organizational assets

Tangible assets include state of the art property, plant and equipment; intangible assets include patents, trademarks, trade secrets, and customer base. Financial assets include a strong balance sheet, excellent cash flow, and a strong financial track record. Financial assets reflect a company's competitive position, market success, and ability to invest in future. Human assets include strong leadership, experienced managers, and well trained, highly qualified, and motivated employees. Organizational assets include specific skills, knowledge and competencies such as manufacturing experience, brand equity, innovativeness, and ability to adapt to change (De Kluyver 2003).

3.3.1 Definition and Justification

Section 3.3.1 will provide a detailed definition of VBF and its corresponding metrics, and describe how VBM are quantified and characterized. Section 3.3.2 will provide details of how the individual VBF are measured. The measuring process involves:

- Defining the metrics
- Obtaining the supporting data (the majority of which are financial data)
- Understanding financial statements
- Defining the equations or protocols for the metrics

Section 3.4 will recapitulate the implementation of VERPRO in a COTS vendor selection process and define guidelines for interpreting the results generated from the equations as defined in Section 3.3. The guidelines are derived from a reference profile that portrays how the software industry performs with respect to the individual metrics. The profile includes statistical information such as range, average, median and standard deviation. These guidelines provide the basis for the evaluators to judge the relative performance of the alternatives to the performance of the industry as a whole. As a result of this comparison, additional action can be instigated if the alternatives are found to be significantly below or above the industry average or median.

3.3.1.1 Product Leadership

Product leadership relates to the ability of the vendor to produce the next generation products that meet future market demands. For example, if we purchased an Enterprise Resource Planning (ERP) suite from SAP in the early 90s, we would expect SAP to come up with a web-enabled version of the suite as soon as the explosion of the Internet became evident in the mid 90s. Since ERP intercepts with almost all facets of our business process; from planning to procurement, and inbound logistics to manufacturing, packaging, and outbound logistics, the switching cost would be enormous. If SAP was unable to deliver a web-enabled version of the suite before our competitive position is threatened, we would have to make a painful tradeoff; to switch and endure an enormous cost or to stay and watch our competitive position losing ground to the other players. (SAP was able to deliver a web-enable version of the suite with minimal impact to its client through a few acquisitions with its huge war chest of cash). Regardless of the ending, the lesson learned from this anecdote is that prior to acquiring SAP's product, we should have anticipated such a scenario and made a choice that would minimize the risk of us having to make the painful tradeoff. This is the main reason why product leadership comes into the picture, especially when the products we intend to acquire are mission critical or incur high switching cost. Product leadership can only be measured indirectly. The quantitative metrics that influence product leadership include:

- Market share
- R&D budget as a percent of revenue
- Revenue growth

The qualitative metrics include:

- Maturity
- Technology prowess

3.3.1.2 Financial Strength

Compared to product leadership, financial strength is less abstract and thus a more definitive judgment can be drawn by analyzing the financial statements. For our purpose, we will look at financial strength from three perspectives:

- Short term liquidity
- Long term profitability
- Access to capital

Short term liquidity relates to the ability of the vendor to fulfill his promise during the development phase. For example, a vendor facing short term liquidity issues may have to impose cost cutting measures that could lead to an exodus of key employees, under investment in key technology areas, and lack of focus to the customers. All these implications could significantly impact the schedule and quality of the product delivered. Long term profitability relates to the vendor's ability or desire to continue to stay in business, and the vendor's ability or interest to continue to invest in product development. For example, if a business continues to lose money, the shareholders may eventually decide to exit the business and divest the capital to other ventures or liquidate the business and return the capital to the shareholders. The negative implications to the customers range from interrupted service to early retirement of the service level agreement. The customer may have to decide to switch or hire a team to take over the maintenance. Either choice is risky and incurs substantial cost. Access to capital relates to the vendor's ability to stay in business and make adequate investment for product development. For example, it is common for a startup company, especially in software industry, to lose money in the initial phase. Therefore, cash reserve serves as a good measure for its financial strength. Short term liquidity can be measured by adequacy of working capital, long term profitability can be measured by cash margin, and access to capital can be measured by cash and cash equivalent, and the ability to raise capital.

3.3.1.3 Business Risks

Business risks consist of all internal and external forces that influence the strategic position of a business. For our purpose, we will focus on the risk factors that are present as a result of the strategic choices that a company makes or some specific conditions that a company faces. The strategic choices include capital structure and market segment pursued. The specific condition includes legal liability. These risk factors are measured by the following metrics:

- Debt to equity ratio
- Debt coverage
- Concentration of revenue sources
- Presence of any legal liability

Unlike product leadership or financial strengths, business risks can be considered incidental because some of the risk may not exist in many of the cases. For example, capital structure is a financing decision; there is nothing wrong with incurring a certain amount of debt if the company is able to service the debt. However, a company with a high debt load is more prone to negative macroeconomic changes and interest rate fluctuations. Likewise, legal liability does not exist in most companies, but it can be very taxing or even financially damaging to the company if it does. Therefore, the existence of legal liability should be examined.

3.3.2 VERPRO Business Metrics (VBM)

Section 3.1 has introduced VBF and the corresponding VBM. Section 3.3.1 has provided an elaborated definition of VBF and a brief account of VBM. Section 3.3.2 will provide a detailed account of VBM, which include the definition of the metric, the evaluation method, the source of data in the event hard data is required, and the interpretation guidelines.

3.3.2.1 Definition

Table 3.3.2 presents a summary of the metrics. Column (1) lists the VBF. Column (2) lists the corresponding VBM. Column (3) lists the measurement type for the individual metrics. A measure is categorized as quantitative if the definition is given by a mathematical expression and the variables can be substituted with hard data, otherwise, the measure is categorized as qualitative. Column (4) lists the level of measurement. For the quantitative measures, ratio level data is preferred. Column (5) lists the precedence of the measure. The precedence is classified into primary, secondary and conditional. We recommend looking at all primary measures before a judgment is made. If the primary measures cannot provide conclusive judgment, secondary measures will be considered. Conditional measure is performed only after a certain condition is met. For example, claim coverage will only be assessed when legal liability exists. Likewise, debt coverage will only be assessed after exceeding a certain amount of debt.

(1) Sub Objectives	(2) Measures	(3) Measurement Type	(4) Level of Measurement	(5) Precedence
Product Leadership	Market share (%)	Quantitative	Ratio	Primary
	R&D budget (%)	Quantitative	Ratio	Primary
	Revenue growth (%)	Quantitative	Ratio	Primary
	Capability maturity model	Qualitative	Ordinal	Secondary
	Technology prowess	Qualitative	Ordinal	Secondary
Financial Strength	Current ratio	Quantitative	Ratio	Primary
	Cash Margin (%)	Quantitative	Ratio	Primary

(1) Sub Objectives	(2) Measures	(3) Measurement Type	(4) Level of Measurement	(5) Precedence
	Cash reserve (year)	Quantitative	Ratio	Primary
Business Risk	Debt to equity ratio	Quantitative	Ratio	Primary
	Debt coverage	Quantitative	Ratio	Conditional
	Cash to Debt ratio	Quantitative	Ratio	Conditional
	Revenue risk (%)	Qualitative	Ratio	Primary
	Claim coverage	Qualitative	Ratio	Conditional

Table 3.3.2: A Summary of VERPRO Business Metrics (Yeoh 2004)

3.3.2.2 Sources of Data

The data required to support the computations of the measures can be primarily obtained from:

1. The company's financial statements for free from the Internet.
2. Independent market research reports for a nominal fee from independent market research firms.
3. Business financial database for a nominal fee from independent financial data services firms.

If the vendor is a public company, its financial statements can be obtained from its annual report to the shareholders. The financial statements typically consist of three sections:

1. Income statement
2. Balance sheet
3. Cash flow statement

The information in the annual report is regulated by a securities commission of the country or region in which the company is traded. The securities commission is a regulatory agency that administers federal securities laws to protect stockholders. In US, it is known as the Securities and Exchange Commission (SEC), and in Canada, one of the most prominent is the Ontario Securities Commission. The most recent annual report usually contains financial data from the most recent three fiscal years, which is sufficient for our purpose.

If the vendor is a private company, its financial statement can be purchased from independent financial data services firms such as Moody Corporation, Standard and Poor, or Dun and Bradstreet Corporation at a nominal fee. We recommend using the Comprehensive Report published by D&B Small Business Solution, a subsidiary of Dun and Bradstreet Corporation because it contains all the financial data necessary to perform our analysis, and D&B database, covering more than 1.6 Million businesses in the US and Canada alone, is one of the most complete financial database in the world. A sample of the Comprehensive Report can be downloaded from http://sbs.dnb.com/credserv.asp. It is important to note that private companies are not required to make their accounting and financial information available to the public. As a result, the information is harder to obtain and not subject to the same scrutiny and regulation that public companies face. Market share data can be found in the company's annual report but a more comprehensive coverage can be purchased from independent market research firms such as IDC or Gartner Dataquest.

Table 3.3.3 presents a summary of the data sources for the individual metrics. Column (1) lists the individual VBM. Column (2) lists the company type, which has two possible values; public and private. The difference between a public and private company is that public companies sell stock to the public and private companies do not. Because public companies sell stock to the public, they must make their accounting and financial information available to the public. However, the financial information of a private company is harder to obtain and not subject to the same scrutiny that public companies face. Column (3) lists where the information can be obtained. The possible values include Annual Reports, Market Research Reports, Comprehensive Financial Reports, Vendor Survey, etc. Column (4) points to where the piece of information can be found in the report. The subsequent sub section, Section 3.3.2.3(A), provides some preliminary education for interpreting financial statements.

	(1) VBM	(2) Company Type	(3) Data Source (Recommended)	(4) Location
1	Market share (%)	Public	Annual Report	US: EDGAR http://www.sec.gov/edgar/searchedgar/webusers.htm Canada: SEDAR http://www.sedar.com/homepage_en.htm
		Private	Market Research Reports	Gartner Dataquest http://www3.gartner.com/Init
2	R&D budget (%)	Public	Annual Report	Income statement Management discussion
		Private	Comprehensive Report	D&B Small Business Solution http://sbs.dnb.com/
3	Revenue growth (%)	Public	Annual Report	Income statement Management discussion
		Private	Comprehensive Report	Income statement
4	Capability maturity model integration (CMMI)	Public	Independent Survey	Request from vendor
		Private	Independent Survey	Request from vendor

	(1) VBM	(2) Company Type	(3) Data Source (Recommended)	(4) Location
5	Technology prowess	Public	Independent Survey	Assessed by acquiring organization
		Private	Independent Survey	Assessed by acquiring organization
6	Current ratio	Public	Annual Report	Balance sheet
		Private	Comprehensive Report	Balance sheet
7	Cash margin (%)	Public	Annual Report	Income statement Cash flow statement
		Private	Comprehensive Report	Income statement Cash flow statement
8	Cash reserve in years of capital needs (year)	Public	Annual Report	Balance sheet Cash flow statement Management discussion
		Private	Comprehensive Report	Balance sheet Cash flow statement
9	Debt to equity ratio	Public	Annual Report	Balance sheet
		Private	Comprehensive Report	Balance sheet
10	Debt coverage	Public	Annual Report	Income statement
		Private	Comprehensive Report	Income statement
11	Cash to Debt ratio	Public	Annual Report	Balance sheet
		Private	Comprehensive Report	Balance sheet
12	Revenue	Public	Annual Report	Management discussion

	(1) VBM	(2) Company Type	(3) Data Source (Recommended)	(4) Location
	risk (%)	Private	Vendor Survey	Classified information Request from vendor
13	Claim coverage	Public	Annual Report	Legal proceedings Balance sheet
		Private	Vendor Survey	Request from vendor

Table 3.3.3: A Summary of VERPRO Measures and Data Sources

3.3.2.3 Interpretation of Financial Statements

As described in Section 3.3.2.2, the typical financial statements consist of three separate statements:

1. Income Statement, a.k.a. Statement of Earning
2. Balance Sheet, a.k.a. Statement of Financial Position
3. Cash Flow Statement

Financial accounting is not an exact science; therefore the result may vary quite significantly depending on who prepares the statements. In order to counter this flaw, federal securities laws require publicly-owned companies to follow a set of rules and financial reporting guidelines, which are jointly developed by the Financial Accounting Standards Board (FASB), a private organization of accounting professionals, and the SEC, a U.S. government agency. These generally accepted accounting principles (GAAP) help ensure that the financial information reported is reliable and consistent. GAAP also helps safeguard against investor fraud, and makes it possible to compare financial data with other companies. In addition, financial statements are also fairly standard within an industry, making it possible to compare the performance of the companies in the same industry.

Although all companies are required to follow GAAP, the management of a company still has some leeway to report a set of numbers that do not fairly reflect the true financial position of the company by taking advantage of the loopholes in GAAP. This practice ranges from creative accounting such as extending the useful life of an asset to inflate earnings to outright fraud such as creating bogus revenues (Enron) or capitalizing operating expenses (WorldCom). Creative accounting is not easy to spot therefore it is important for the reader to be aware that such practices are common, and be skeptical about the numbers that appear too good to be true. A reader of financial statements can make more reliable assessment of the company's financial health by taking the basic information in the financial statements and extending it to identify:

- A company's internal strengths and weaknesses
- Company and industry trends
- Performance in the larger business environment

The income statement shows how much revenue a company brings into the business by providing goods or services, or both, to its customers for a set period, usually one quarter or one year. It also shows the costs and expenses associated with earning that revenue during that time. The key elements in the statement of earnings include: Revenue, Gross profit, Operating income, Net earnings, and Earning per share. In an annual report, the Income Statement often includes sales revenue, and expenses for at least the last three years. The net earnings or loss shows how much the company earned or lost.

A thorough analysis of the income statement include looking at trends in revenue, operating income, and gross profit rates or gross margins, and calculating the return on assets, and the return on equity. It is crucial to look at the changes from year to year, both in the raw numbers and in the percentage. It is difficult to generalize about a good rate of change; it depends on the line item being analyzed and the rates of change in prior years. If a company's sales rose 15 percent in each of the past three years, and rose only 10 percent this year, it would trigger further examination, such as adding the industry growth rate into the analysis before determining if the lower growth rate is a negative signal. Another useful measure is the ratio of total operating income to total revenue, also known as operating margin. Growth in operating margin usually implies that a company is growing its operating income and becoming more efficient over time in managing its costs and operating expenses. There is no ideal percentage as the target percentage varies from industry to industry.

The balance sheet reports a company's financial status at a set date noted on the statement. The statement is like a snapshot because it shows what the company is worth at that set date. The statement shows:

- What the company owns
- What the company owes
- What belongs to the owners

The balance sheet is primarily used to examine a company's liquidity and to gain insight into the state of the company's debt and inventory. One of the most important and commonly examined measures is current ratio, ratio of current assets to current liabilities. The relationship of the balance sheet and the income statement should also be examined. Disproportionate increases in account receivable with sales can be an early sign of problem. The company may be too aggressively pursuing sales by offering long payment terms and extending excessive credit to high credit risk customers. The collection of accounts receivable from financially weak customers can be difficult, and may lead to bad debts. The company may be forced to borrow money to finance these receivables, and at the same time, incurs extra debt. Another useful measure is the ratio of total liabilities to total stockholders' equity. In general, a lower ratio of liabilities to equity means a lower risk for a company's creditors and lower costs when the company borrows money. Yet, how much debt a company carries compared with stockholders' equity varies widely according to the norms for the industry and the company's financial strategy. Just because a company has a high debt ratio is not a signal of weakness, if the ratio is within an acceptable range for the industry. In addition, not every asset and liability can be measured in accounting terms. Statements of financial position often omit assets that are difficult to measure. For example, the Coca-Cola Company does not report the company trademark, which is estimated to be worth more than $50 billion on its balance sheet. Industrial company like Bethlehem Steel may overstate the actual economic value of its fixed assets because the replacement value of its fixed assets is often much higher than the carrying value of its existing assets.

The cash flow statement reports the flow of cash into and out of a company in a given year. Cash is a company's lifeblood. Cash includes currency, checks on hand, and deposits in banks. Cash equivalents are short-term investments such as treasury bills, certificates of deposit, or commercial paper that can be quickly and easily converted to cash. A company uses cash to pay bills, repay loans, and make investment, allowing it to provide goods and services to customers. If all goes well, a company uses the cash to generate even more cash as a result of higher profits.

The cash flow statement is analyzed to determine how effectively a company generates and manages cash. The cash from operating activities is to be examined closely in evaluating a company's potential for long-term success because this figure shows how efficiently the company can produce and sell its primary product or service. These activities represent the basic business of the company. During some periods in its life, particularly the high growth phase, a business may not generate enough cash to finance its growth. In these periods, the business will spend more than it generates from its operating activities, resulting in a situation known as negative free cash flow. The additional cash need will come from the capital market or borrowing from banks. Nevertheless, if a company consistently fails to make money at its basic business, it will have a hard time surviving. In healthy mature companies, operating activities normally result in positive cash flows. Cash flows should be evaluated in relation to net earnings from the income statement. For example, in some cases, a company can report positive earnings on the income statement and still report a negative net cash flow on the cash flow statement. This situation may occur when a company is unable to meet the current demand for its products and consequently invests its profits, or even borrows additional money, to expand its manufacturing capability. The implication of such a situation should be examined to determine if this is a positive or negative signal. It is important to determine if the prospective demand for the company's product is great enough to justify the expenditures and new debt. The investing and financial activities of a company are often more difficult to analyze. For example, negative investing cash flows may indicate only that the company is growing and buying assets that enable it to manufacture more products. Financing cash flows are affected by a company's borrowing and the amount paid in dividends during the year. These numbers should be interpreted together with the company's strategies, and compared with the numbers of other companies in the same industry.

The scope of the analysis and financial information comparisons should be extended beyond one year's figures in a single set of financial statements. A more reliable way is to look at the same ratios and performance measures over three year periods. In addition, these figures should also be compared with several other numbers from other sources of business and economic information (Graham 1988). For example, it is important to:

- Compare many growth figures, such as growth in sales, with the rate of inflation, and look for rates that exceed that of inflation
- Look at figures for competing companies to determine a company's strength in its industry
- Examine industry averages to gain a historical perspective in order to evaluate a company's chances for long-term success

3.3.2.4 Formulas and Protocols

A) Market Share:

Market share is a measure of a company's competitive strength in its segment. The relative importance of market share as a strategic goal for businesses has been a subject of considerable controversy. However, although no business should pursue market share at any cost, there is evidence that market share is an important determinant of long term profitability. Large market share is both a reward for providing better value and a means of realizing lower costs. Under most circumstances, enterprises that have achieved a large market share are considerably more profitable. As a result, market share has been recognized by corporate executives and consultant as a pseudo measure for business dominance and product leadership (De Kluyver 2003). While some research encompasses the entire software industry, the majority of research is done at the segment level. Figure 3.3.1(A), 3.3.1(B) and 3.3.1(C) present an example of how the entire software industry is categorized into logical segments, which allows for in-depth and segment-specific research. Before interpreting the data, it is important to first identify which segment the alternatives belong to and understand how the data is derived. If the report is published by a research firm such as Gartner Dataquest, the research methodology and data collection principles are well defined. Since the data may vary depending on the methodology, the alternatives should be assessed based on the same report. In addition to the current market share, the trend for the most recent three years should also be studied to observe if there is any deterioration in any player's market position. If such a signal is observed, further information should be sought to understand the changes. In general, the segmentation structure, definitions of terminology, and research methodology are revised, altered or expanded each year to reflect changes in software technologies and the software marketplace. If data is not available for a particular product category, the data from its parent category can be used. An example of market share report published by Gartner Dataquest is presented in Table 3.3.4(A).

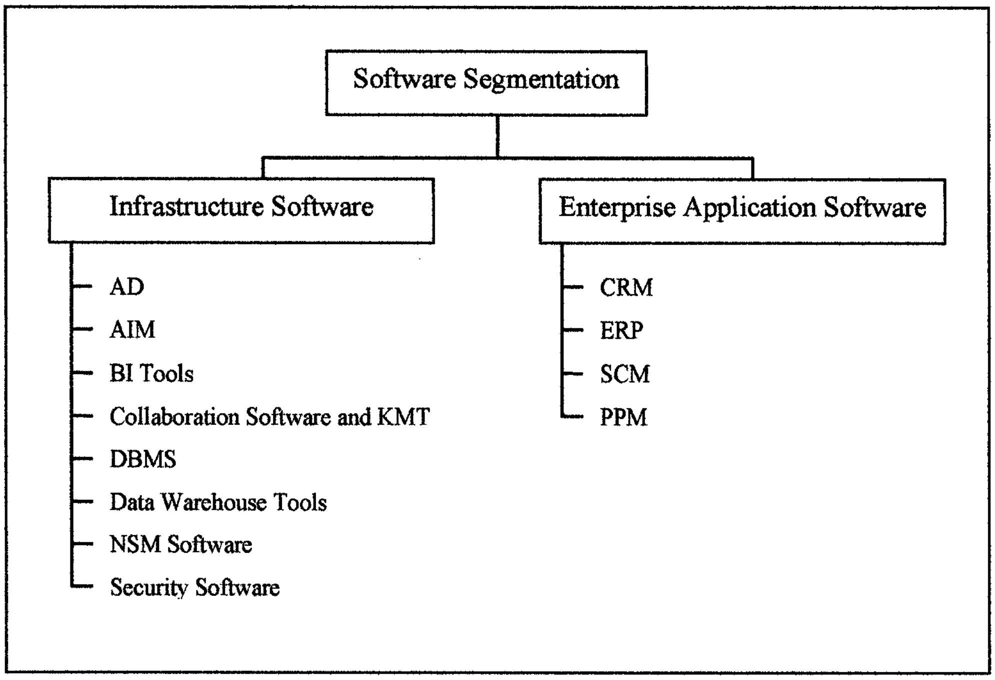

Figure 3.3.1(A): Software Segmentation Structure (Graham 2004)

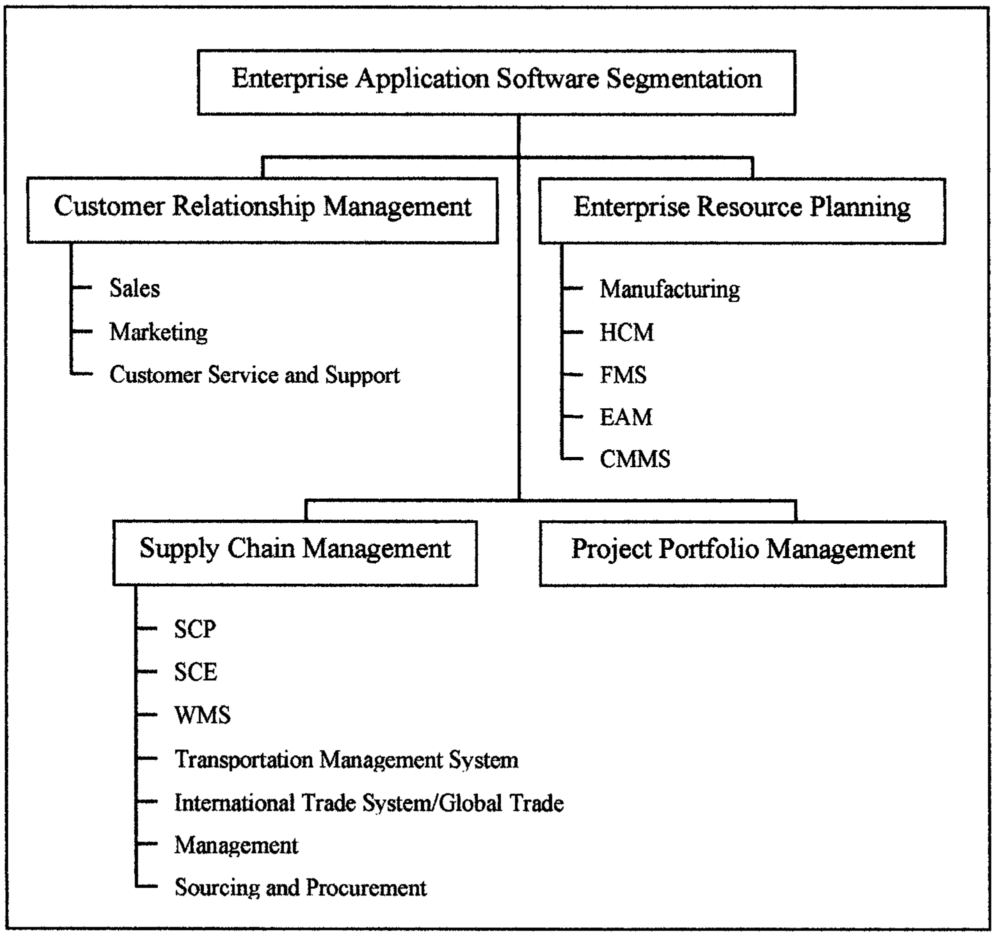

Figure 3.3.1(B): Enterprise Application Software Segmentation Structure
(Graham 2004)

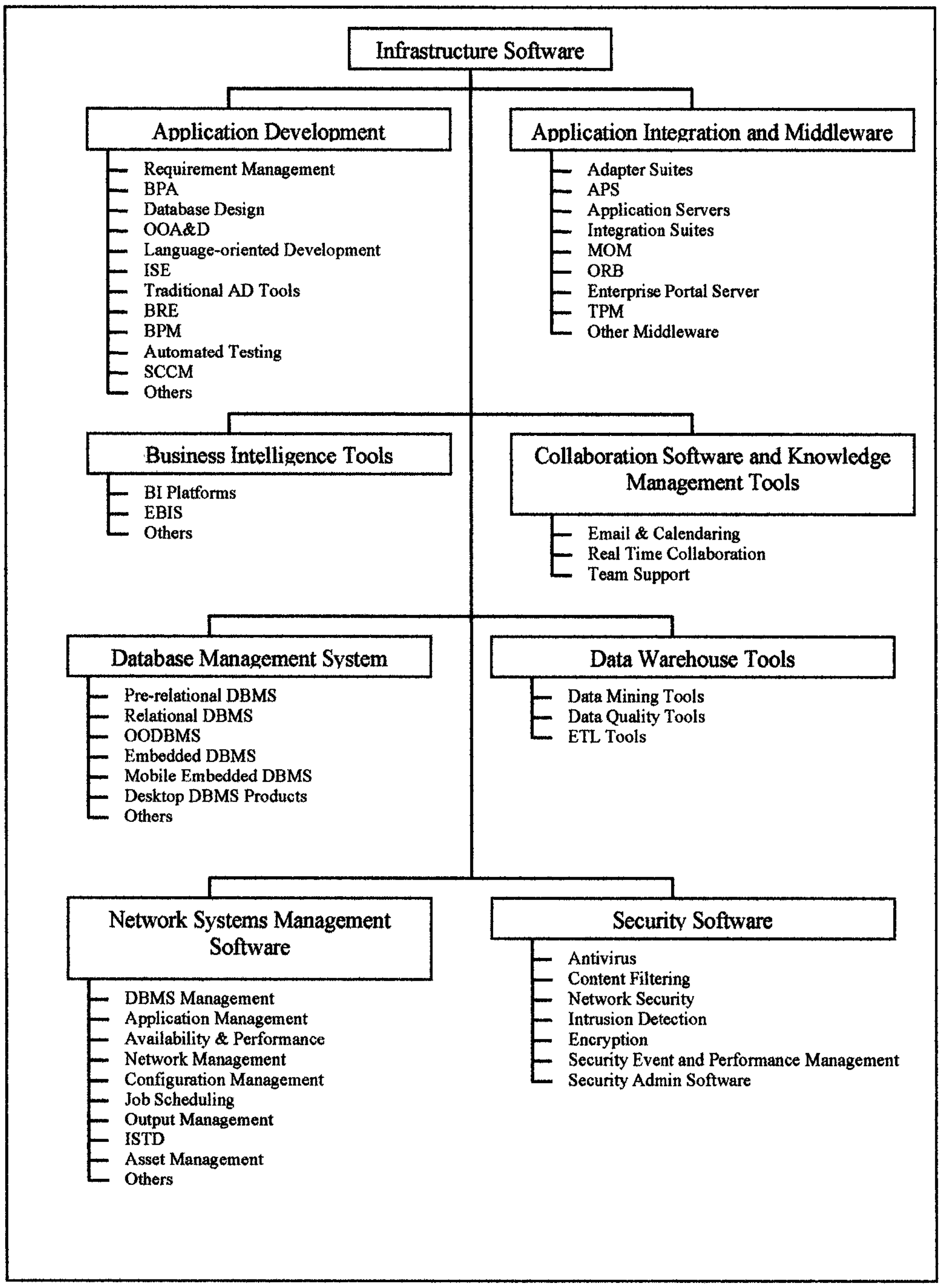

Figure 3.3.1(C): Infrastructure Software Segmentation Structure (Graham 2004)

Company	2002 Market Share (%)	2001 Market Share (%)
SAP AG	25.1	24.7
Oracle	7.0	7.9
PeopleSoft	6.5	7.6
SAGE	5.4	4.6
Microsoft Business Solutions	4.9	4.6
Others	51.1	50.3
Total Market Share	100.0	100.0

Table 3.3.4(A): Worldwide ERP Software Application New License Revenue Market Share Estimates for 2002 (Graham 2004)

B) R&D Budget:

R&D budget is an investment decision that reflects the company's commitment to current and future product development. There is no absolute scale defining how much R&D spending is desired. Therefore, judgment should be made by observing how the alternatives perform relative to each other, and how they perform relative to the industry benchmarks. The industry benchmarks are the industry leaders for a particular segment. For example, Microsoft for Operating System, Oracle for Database Management Solution, Symantec for Security Software. The result of this metric should be based on the average of the most recent three years data. The trend for the most recent three years should also be studied to observe if there is any radical development that requires further investigation. R&D budget is given by:

$$\frac{\textit{Investment in research \& development}}{\textit{Total revenue}} \times 100\%$$

Figure 3.3.2(B): R&D Budget

C) Revenue Growth

Revenue growth is an indication of a company's products or services gaining market acceptance. Like R&D budget, there is no specific number to dictate how much growth is desirable. Therefore, judgment should be made from peer comparison with reference to the industrial average. The result of this metric should be based on the average of the most recent three years data. The trend for the most recent three years should also be studied to observe if there is any radical development that requires further investigation. Companies operating in the same industry segment usually have a small variation in revenue growth. If any alternative is found to have significantly higher or lower growth, further investigation should be undertaken. Revenue growth is given by:

$$\frac{Current\ Year\ Sales - \mathrm{Pr}evious\ Year\ Sales}{\mathrm{Pr}evious\ Year\ Sales} \times 100\%$$

Figure 3.3.2(C): Revenue Growth

Previous year and current year sales are usually tabulated side by side in a single statement. If the company breaks down its revenue into product and services, the total is to be used.

D) Maturity

One of the most widely accepted measure for maturity is Capability Maturity Model Integration (CMMI). CMMI for software is a model for judging the maturity of the software processes of an organization. CMMI helps organizations identify the key practices required to increase the maturity of these processes. CMMI was developed by the software community with stewardship by the Software Engineering Institute (SEI). CMMI is a de facto standard for assessing and improving software processes. Through CMMI, an effective means of modeling, defining, and measuring the maturity of the processes used by organizations developing and maintaining software-intensive systems can be established. The CMMI hierarchy, which consists of five levels, is presented in Figure 3.3.1(D)

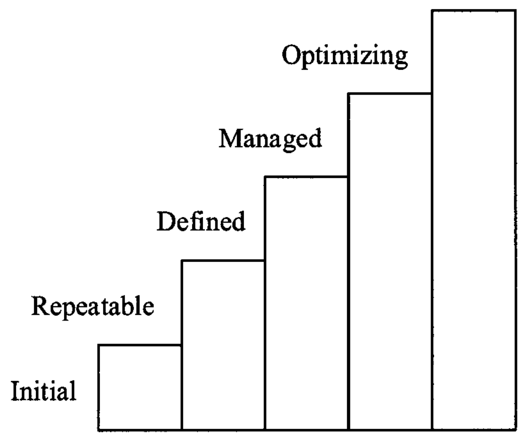

Figure 3.3.2(D): CMMI Hierarchy (SEI)

Enhanced with feedback from the industry, the CMMI categorizes software organizations based on their current practices from level 1 (least mature) to level 5 (most mature). Organizations will then attempt to climb the ladder from their current level to level 5. Each maturity level is associated with a set of process goals and improvement priorities. Categorization helps the firm understand where it stands, and the immediate areas it should look into to move up to the next plateau. This way, organizations can do a little at a time to reach maturity. Organizations in each maturity level show similar characteristics. For example, level 1 organizations are generally not stable. Planned procedures are not in place, and ad-hoc policies are made up by the team leader along the way. Most small start-ups are in this category. As the team grows and projects get larger, haphazard management which may work for smaller teams will crumble. Level 1 organizations should strive to make processes repeatable, a characteristic of level 2 organizations. Successful projects should be studied to produce a set of process guidelines so that future projects can repeat the previous success by following what has been proven to work. On the far end of the spectrum, level 5 organizations are already efficient and productive. Level 5 organizations should be concentrating on how to continually improve their processes. The details about CMMI can be found in this website: http://www.sei.cmu.edu/cmmi/. Although a higher level is desirable, the acquiring organizations need to weigh the vendor's

CMMI level against their expectations. For example, if level 3 is sufficient, the incremental advantage diminishes beyond level 3. If the alternatives being assessed do not currently have a CMMI rating, the CMMI audit checklist can be used to conduct an informal assessment to determine the approximate CMMI level of the alternatives.

E) Current ratio:

Current ratio is a measure of a company's ability to carry on its normal business comfortably and without financial stringency, to expand its operations without the need of new financing, and to meet emergencies and losses without disaster (Graham 1975). A ratio of less than one is an early indication of a shortage in working capital, which may result in slow payment of bills, poor credit rating, interruptions in operations, and inability to turn around and make progress. The result of this metric should be based on the most recent year data. The trend for the most recent two years should also be studied to observe if there is any radical development that requires further investigation. Current ratio is given by:

$$\frac{\textit{Total current assets}}{\textit{Total current liabilities}}$$

Figure 3.3.2(E): Current Ratio

F) Cash Margin:

Cash margin is a measure of cash profitability of a company. It also reflects the ability of a company to generate cash to pay for all its operating expenses and capital expenditure. Cash margin is given by:

$$\frac{\textit{Net cash provided by operating activities} - \textit{Capital Expenditure}}{\textit{Total revenue}}$$

Figure 3.3.2(F): Cash Margin

Capital expenditure consists of investing activities that are related to maintaining and growing the business. These activities include purchases of property, plant and equipment, investment in research and development, and acquisition of businesses. Cash margin can be positive or negative. A positive value reflects profitability and a negative value reflects losses. Although all companies ultimately need to achieve positive cash margin to survive, it is common for startup companies to have negative cash margin because a significant amount of cash is spent on building and growing the business. Therefore, a negative cash margin does not necessarily mean the company is having financial problems; we need to look further into the trend and cash reserve before any conclusive judgment can be made. When calculating cash margin, we recommend taking the average of the most recent three years, this is to reduce variations from capital expenditure. In addition, the three-year-trend should also be scrutinized for any abnormality. All abnormalities should be investigated and explained.

G) Cash Reserve:

Cash reserve is a measure of a company's staying power. The unit of measure is given by the number of years of capital need. Capital need is the estimate of the amount of cash a company needs to spend each year to stay in business. Cash reserve also reflects a company's ability to make strategic or opportunistic investment. If net cash provided by operating activities (cash flow) is positive, cash reserve is given by:

$$\frac{\textit{Cash} + \textit{Cash Equivalents}}{\textit{Capital Expenditure}}$$

Figure 3.3.2(G1): Cash Reserve (when cash flow is positive)

Otherwise, it is given by:

$$\frac{\textit{Cash} + \textit{Cash Equivalents}}{\textit{Net cash used by operating activities} + \textit{Capital Expenditure}}$$

Figure 3.3.2(G2): Cash Reserve (when cash flow is negative)

When cash flow is negative, the term "provided" is substituted by "used". There is no hard rule to determine how much cash reserve is sufficient. Therefore, judgment should be made by observing how the alternatives perform relative to each other, and how they perform relative to the industry. When calculating cash reserve, the numerator should be based on the most recent year data. The denominator should be based on the average of the most recent three years data. This is to reduce the year over year variation of components that form the denominator.

H) Debt to Equity Ratio

Debt to equity ratio is a measure of how much financial risk a company is taking. A vendor having higher debt to equity ratio than the industry indicates that it is taking more risk than the industry. This does not necessarily indicate an unhealthy sign if the vendor is able to generate a higher rate of return than the cost of debt. Debt to equity ratio is given by:

$$\frac{Total\ short\ term\ debt + Total\ long\ term\ debt}{Total\ assets - Total\ liabilities}$$

Figure 3.3.2(H): Debt to Equity Ratio

The numerator represents all interest bearing liabilities. Total short-term debt consists of loans, notes payable, short-term borrowing and current maturities of long-term debts. Long-term debt is usually a one line item. The denominator is also known as Shareholder's equity or book value. The result of this metric should be based on the most recent year data. If the ratio is greater than zero, two additional measures will be evaluated before a judgment can be made. These two additional measures are debt coverage as represented by Figure 3.3.1(I) and cash to debt ratio as represented by Figure 3.3.1(J).

I) Debt Coverage:

Debt coverage is a measure of a company's ability to service its debt. The numerator should be based on the average of the most recent three years. This is to reduce year over year variations in operating income as a result of fluctuations in revenue and depreciation. The denominator should be based on the most recent year data. Debt coverage is given by:

$$\frac{\textit{Operating Income}}{\textit{Total Interest Expenses}}$$

Figure 3.3.2(I): Debt Coverage

If debt coverage is greater than 7 (Graham 1975), we can conclude that the concern from debt is negligible.

J) Cash to Debt Ratio:

Cash to debt ratio is a measure of a company's ability to pay back the principal of its debt without incurring additional debt or equity. The numerator and denominator should be based on the most recent year data. Cash to debt ratio is given by:

$$\frac{\textit{Cash} + \textit{Cash Equivalents}}{\textit{Total Debt}}$$

Figure 3.3.2(J): Cash to debt ratio

If cash to debt ratio is greater than 1, we can conclude that the concern from debt is negligible.

K) Revenue Risk:

Revenue risk is a measure of the extent of a company's dependence on its top or top few customers. Revenue risk is given by:

$$\frac{\textit{revenue from top x customer}}{\textit{Total revenue}}$$

Figure 3.3.2(K): Revenue risk

The variable x represents the number of customers. When making comparison, we can substitute x with 1, 5, 10 and observe the trend. In general, the risk increases as the percent goes higher. There is no absolute scale defining how many percent is tolerable. Therefore, judgment should be made by observing how the alternatives perform relative to each other. The overall industry data is not available because this is not a required disclosure. Although some public companies do disclose this information, it is often embedded in lengthy management discussion, which makes large-scale analysis tedious. Other factors to examine include the existence of any long term contracts, or repeatable licensing deals that provide a layer of protection to the company's revenue.

L) Claim Coverage:

Regardless of the outcome, any litigation may require the company to incur significant litigation expense and may result in significant diversion of management attention. Claim coverage is a measure of the company's ability to cover the claims (confirmed or pending) and cost of litigation from its deployable financial resources such as cash & cash equivalents and cash flow. The numerator and denominator should be based on the most recent year data. Claim coverage is given by:

$$\frac{1.5 \times \textit{Combined Exposure}}{\textit{Cash} + \textit{Cash Equivalents}}$$

Figure 3.3.2(L): Claim Coverage

Because of the complexity of litigations and lack of relevant data, there is no one universal equation to quantify it; therefore, a more conservative approach is preferred. A safety constant of 1.5 is used because litigations often overrun budget. Again, there is no one universal constant that suits all situations. In the software industry, most of the claims involve patent and licensing issues and the awards seldom exceed the claims filed by the plaintiff. In addition, the justice system is unlikely to endorse a ruling that would bankrupt a company. Therefore we feel that a safety constant of 1.5 is sufficient to cover any surprises. In most cases, this risk does not exist, if it does, the risk is considered low if the ratio is less than one, which means the company has enough cash to pay for the claims without affecting the operation.

3.4 VERPRO IMPLEMENTATION

Section 3.3 has described how VBF are derived and detailed how the VBM are measured. Section 3.4 integrates the information presented in Sections 3.1 to 3.3 into a process flowchart detailing the implementation of VERPRO in a COTS selection and monitoring process. Section 3.4 also presents the guidelines for interpreting the results generated from the formulas and protocols defined in Section 3.3. The guidelines consist of two parts:

1. Symptoms of business distress, which consists of a list of warning signs that prompt the acquiring organization to take further actions to re-evaluate the financial conditions of the vendor. Section 3.4.2 will detail the symptoms and the mitigating actions that can be taken to minimize the risks.
2. Financial reference profiles for certain metrics, which portray how software industry in general performs in any given period. These profiles consist of statistical information such as range, average, median and standard deviation, which help the evaluators to judge the relative performance of the alternatives to the performance of the industry as a whole. As a result of this comparison, additional action may be triggered if the alternatives are found to be significantly below or above the industry average or median.

3.4.1 VERPRO Process Flowchart

VERPRO implementation is loosely based on the Rational Unified Process which consists of three phases:

1. Inception: This phase consists of eleven major process steps. The detailed process together with the input and output artifacts are presented in Figure 3.4.1(A). Activity 1 (A need is identified) is triggered when the need for a new software application is recognized. The stakeholders in Activity 2 refer to the representatives of the organization who have a vested interest in the outcome of the project. The representatives can be an end user, a purchaser, a contractor, a developer, or a project manager. The COTS Acquisition Team usually consists of the stakeholders and individuals who are assigned to the team for their expertise. These individuals can be a programmer, a process expert, a financial analyst or a consultant. Activities 4 to 7 are the process steps that lead to the creation of a preliminary SRS. These activities often involve a vigorous discussion of the high-level understanding of the end-user needs, expectations, and constraints. The stakeholder requests are challenged to ensure that each need and the implication of not meeting that need are fully understood. The essential features in the preliminary SRS are used as the basis to search for potential COTS alternatives. Potential alternatives are identified from various sources such as industrial databases, market reports, conferences, depository etc. (Albert 2002). In Activity 9, the potential alternatives are examined to ensure they are capable of delivering the essential features in the preliminary SRS. This can be done through end user evaluation or limited experiment. At the end of Activity 9, the list of alternatives is finalized. Then, a request for quotation (RFQ) is sent to these alternatives. Once the RFQ is received, the business profiles of the alternatives are obtained. The business profiles can be obtained from various sources such as the vendor, industrial database, internet search, market reports etc.
2. Elaboration: This phase consists of nine major process steps. The main objective of this phase is to identify the best candidate out of the list of

alternatives. This phase also encompasses VERPRO implementation (Activity 1 to 7). The detailed process together with the input and output artifacts are presented in Figure 3.4.1(B). The VET in Activity 1 is established with reference to VERPRO VET (Figure 3.1.1). The VET is established based on the context of the acquisition. For example, if all of the alternatives offer comparable Product, Cost and Service factors, only Business factors are evaluated. In Activity 2 (Collect Measures), the metrics of the individual alternatives are quantified and tabulated for comparison. An example of this activity is presented in Table 4.2.1. In Activity 3 (Perform Pairwise Comparison), the alternatives are compared in a pairwise manner and the pairwise ranking is recommended based on the scale defined in Table 4.2.2. Because Activity 3 is rather tedious, it is usually conducted by highly specialized individuals such as data and financial analysts. In Activity 4 (Review and Finalize Results), the recommended ranking is reviewed and finalized by the COTS Acquisition Team. In Activities 5-7, the ranking is input to a software application (Expert Choice), the software application will then output the scores and the associating consistency ratios. If the consistency ratio exceeds the pre-defined threshold, the input will be revised and re-input to the software application. This step is repeated until the consistency ratio is within the threshold. An example of these activities is presented in Section 4.2.4. At the end of Activity 7, each alternative will have an overall score and the alternative with the highest overall score is selected. At this point, we have identified the best alternative out of the list of potential alternatives. The next step is to ensure that the best alternative is free of possible business distress. This is accomplished by Activity 8 (Review Top-Ranked Alternatives for Possible Business Distress) where the metrics of the selected alternative is compared with the financial reference profile of the software industry (Section 3.4.3). An example of this process is presented in Table 4.2.4. At the same time, the selected alternative is evaluated to ensure it is free of symptoms of business distress. Some common symptoms of business distress in presented in Table 3.4.1. In

Activity 9 (Create Risk Management Plan), a monitoring plan is established. The recommended monitoring frequency is once a year.

3. Transition: The key activities in this phase include creating a monitoring plan, reviewing the contractual breaches and performance issues, evaluating the vendor's business for any symptoms of distress, reviewing contracts from non-performing vendors, and negotiating new contracts. These activities are repeated until the software application is retired. The detailed process is presented in Figure 3.4.1(C).

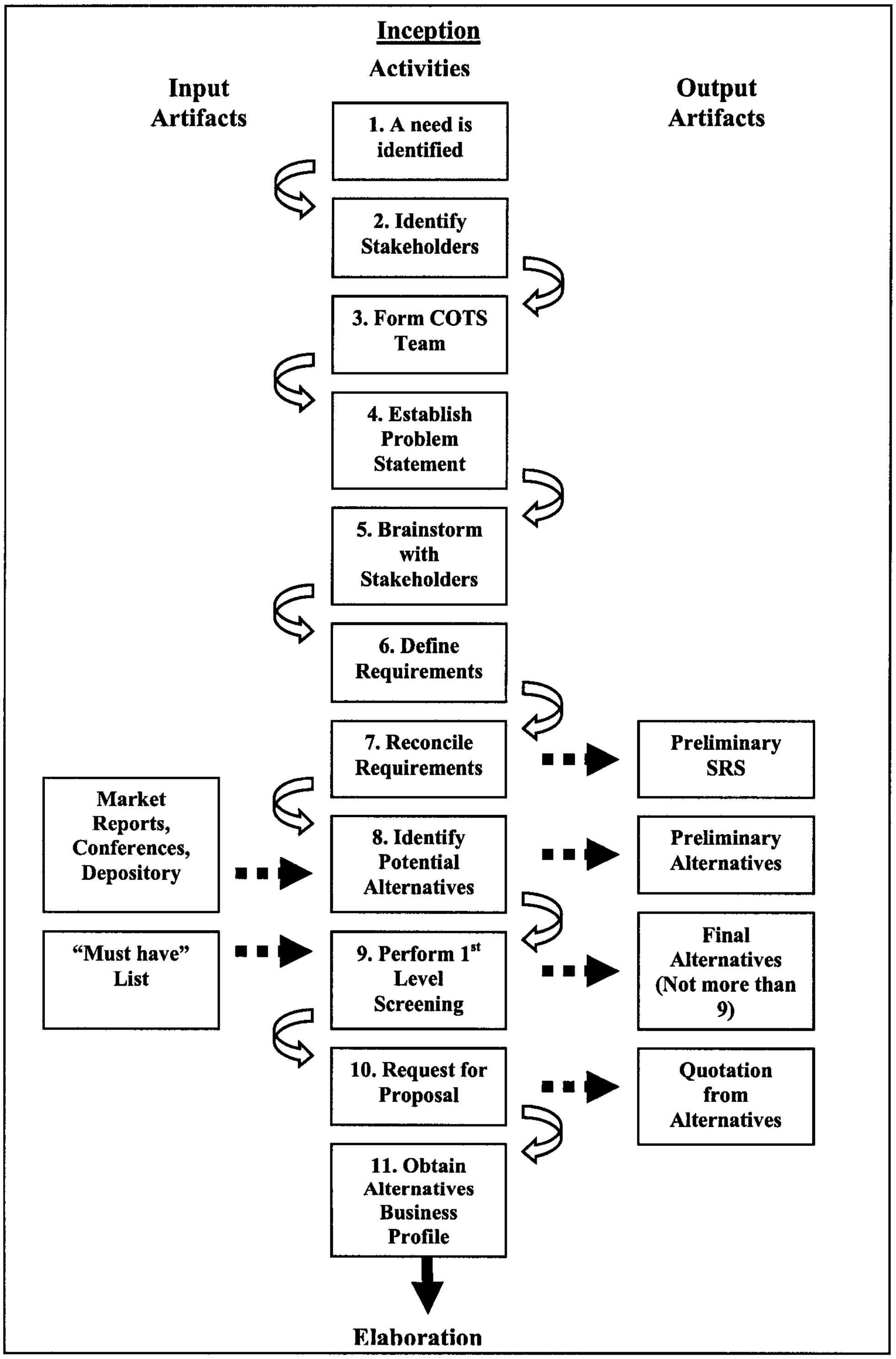

Figure 3.4.1(A): Process Flowchart for VERPRO Inception Phase

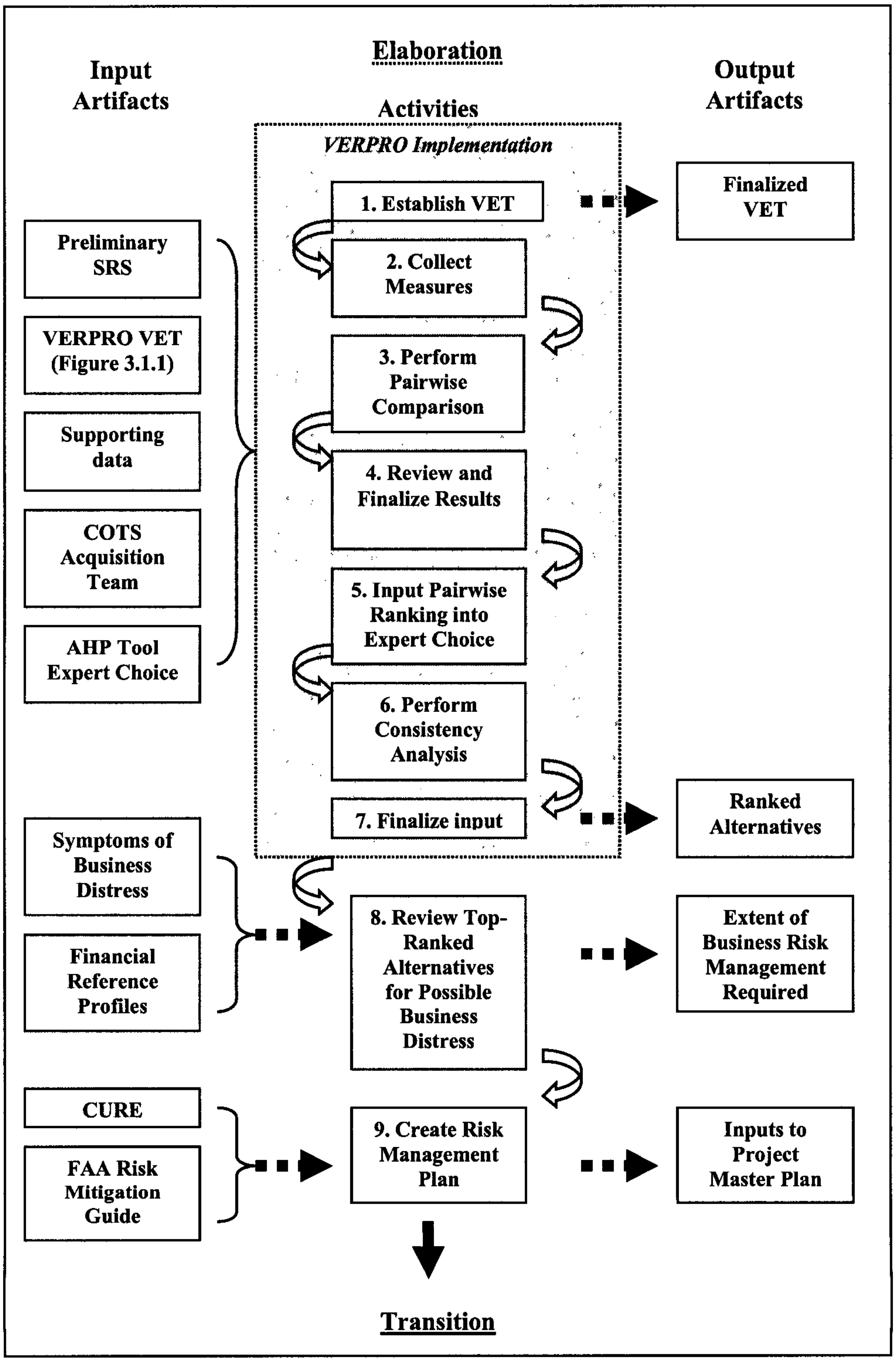

Figure 3.4.1(B): Process Flowchart for VERPRO Elaboration Phase

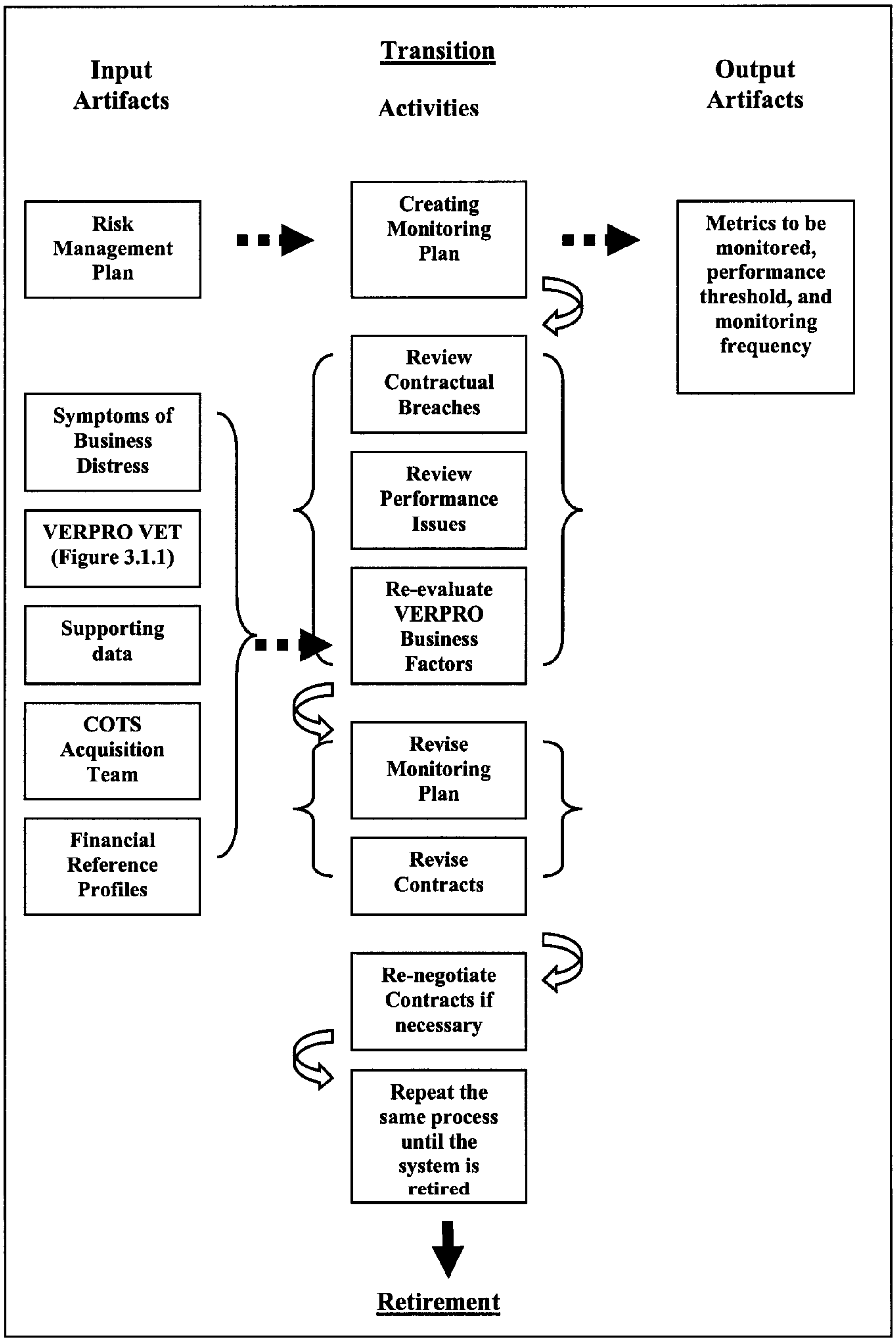

Figure 3.4.1(C): Process Flowchart for VERPRO Transition Phase

3.4.2 Symptoms of Business Distress

Even during the current state of vendor unpredictability and financial instability, there are warning signs before a business goes into financial distress. We can track the signs by:

- Scrutinizing the company's news release i.e. restructuring, downsizing, and exodus of key employees.
- Reviewing the company's quarterly financial statements i.e. negative changes to the top and bottom lines.
- Observing the company's employees movement i.e. exodus of a large group of employees.
- Reviewing the company's new products and product pipelines.

The most common symptoms of financial distress are presented in Table 3.4.1.

Symptom	Description
Restructuring	Restructuring happens as a result of external change forces (prolonged economic downturn, shift in technological trend) that are beyond the control of the company. It usually involves radical changes (abandonment of the existing product line in favor of a hot new product, divestment of capital to other venture). Although not all restructurings are bad, they are often risky, and all stakeholders should be aware of the consequence.
Layoff	Layoff is usually part of a restructuring plan that is undertaken to improve profitability. Although some layoffs are legitimate (save the company from bankruptcy), there are still negative consequences (low morale, reduced service, sabotage, employee lawsuits) and the repercussions to the stakeholders should not be underestimated.
Sudden Resignation of Key	This can be a sign of illegal business practices by the management team that can potentially break up the company. Other similar warning sign is the investigation by federal

Symptom	Description
Employees	agencies (SEC, FBI, Justice Department)
Exodus of a large group of Employees	Employees are often the first to learn about the problems inside a company. The leaving of a large group of employees can also cause reduced or interrupted service.
Class Action Lawsuits	Class Action Lawsuits are usually costly and can potentially breakup a company. It will strain management resources and divert the company's attention away from productive activities (developing new products, serving customers)
Reduction in R&D Investment	This is an indication of financial difficulty or divestment of capital to other ventures that may not be beneficial to the existing stakeholders.
Sudden Reduction in Sale Price	This is also an indication of financial problem. A reduction is sale price will lead to a drop in margin and inability to make future investment.

Table 3.4.1: Early Signs of Financial Distress and Mitigating Strategy

Once such symptoms are detected, the acquiring organizations can take the following steps to contain the risk:

1. Define: Analyze the problem by applying the 5W (who, what, why, where, when) and 1H (how) method. For example, when a restructuring is announced, we need to investigate who (employees, shareholders, suppliers, customers) are impacted and to what extent; what product lines are affected, i.e. any product discontinued, divested, taken out of focus; why is the restructuring taking place, i.e. economic, technical, political, strategic, legal; where is the restructuring taking place, i.e. corporate, regional; when does it start and end, if the customers are impacted, how soon will the repercussions reach the customer; how do we protect ourselves from any negative repercussion, i.e. immediately explore alternative sources for critical systems, if the cause is financial, explore if we can provide financial assistance (most large companies such as

Microsoft, Intel, Dell etc. have a venture capital arm that can provide financial assistance to their strategic partners in exchange for equity ownership).

2. Contain: After analyzing the situation, our conclusion will range from no impact to immediate impact. Once the criticality and the urgency of the problem are determined, we need to contain the problem before the long term corrective action is undertaken. For example, as a result of a restructuring and massive layoff, certain clauses in our service contract are voided; we can consider hiring the service team from the vendor to temporarily take over the service function. This action, although costly, will ensure continuity of the operation and minimize service interruption.
3. Corrective Action: Once containment is in place, we can review the options we have to restore the system back to normal operating conditions. Depending on the cause of the problem and our ability, the options include replacing the entire or a portion of the system, hiring the employees from the vendor, rewriting part of the codes, offering financial assistance to the vendor.

3.4.3 Financial Reference Profiles

Except for R&D Budget, the data used to derive the reference profile are obtained from sixty publicly traded software and programming companies in the United States. The company's size is represented by the number of employees. To ensure consistency, all data are for fiscal year 2003 and the raw data are downloaded from http://www.sec.gov/edgar/searchedgar/webusers.htm.

3.4.3.1 R&D Budget

Table 3.4.2(A) presents a summary of how software industry in general performs. Unlike the other metrics, the reference profile for R&D Budget is not derived from the industry average but from the industry benchmark. Because R&D Budget is an investment decision, it is more meaningful to compare the results with companies that consistently make the right investment decision. The benchmark list is selected from the top software and programming companies within its segment. To qualify as industry benchmark, the company must fulfill the following requirements:

1. Top three players in its market segment.
2. Consistently profitable or cash flow positive in the past consecutive three years.

The industry benchmark is not sensitive to macroeconomic climate such as economic boom or recession. Therefore, comparison for other periods can be made based on the same benchmark data.

Company	Segment	Revenue (Million)	2001	2002	2003	Average
Microsoft	OS	$32,187	17%	15%	14%	16%
Oracle	Database	$9,475	10%	11%	12%	11%
SAP AG	ERP	$8,849	10%	11%	12%	11%
Adobe	Digital Imaging	$1,295	18%	21%	21%	20%
Intuit	Personal Finance	$1,651	15%	15%	18%	16%
Peoplesoft	Enterprise Application	$2,267	14%	18%	19%	17%
Symantec	Security	$1,407	15%	15%	14%	15%
BEA System	Application Infrastructure	$1,012	12%	14%	14%	13%
Average		$7,268	14%	15%	16%	15%
Maximum		$32,187	18%	21%	21%	20%
Minimum		$1,012	10%	11%	12%	11%
Median		$1,959	15%	15%	14%	15%
Std Dev		$10,651	3%	3%	3%	3%
Sample		8	8	8	8	8
Std Dev = Standard Deviation						

Table 3.4.2(A): R&D Budget Benchmark of Public Software Companies in US

3.4.3.2 Revenue Growth

Table 3.4.2(B) presents a summary of how software industry in general performs. The size is represented by the number of employees. The overall industry revenue growth is sensitive to macroeconomic climate such as economic boom or recession. Therefore, comparison should only be made for the same period.

Size	<100	101 – 500	501 – 1000	1001-5000	>5000	Overall
Average	-16.61%	6.63%	-10.19%	2.38%	10.15%	0.11%
Maximum	41.10%	92.80%	76.00%	50.90%	28.90%	92.80%
Minimum	-56.60%	-41.50%	-86.10%	-29.00%	-2.00%	-86.10%
Median	-25.95%	-1.15%	-17.00%	-2.15%	6.85%	-3.80%
Std Dev	33.21%	31.63%	39.85%	21.31%	13.57%	0.31
Sample	8	26	10	12	4	60

Table 3.4.2(B): Revenue Growth of Public Software Companies in US

3.4.3.3 Current Ratio

Table 3.4.2(C) presents a summary of how software industry in general performs. The overall industry current ratio is not very sensitive to macroeconomic climate, if data from the same period is not available, the data from other period can be used as a close proximate for comparison.

Size	<100	101 – 500	501 – 1000	1001-5000	>5000	Overall
Average	4.96	3.46	3.32	2.00	2.86	3.31
Maximum	28.92	17.29	8.32	3.47	4.90	28.92
Minimum	0.64	0.45	0.90	1.06	1.28	0.45
Median	1.75	2.43	2.83	2.03	2.64	2.25
Std Dev	9.71	3.70	2.23	0.73	1.54	4.31

Size	<100	101 – 500	501 – 1000	1001-5000	>5000	Overall
Sample	8	26	10	12	4	60

Table 3.4.2(C): Current Ratio of Public Software Companies in US

3.4.3.4 Cash Margin

Table 3.4.2(D) presents a summary of how software industry in general performs. The overall industry cash margin is not very sensitive to macroeconomic climate, if data from the same period is not available, the data from other period can be used as a close proximate for comparison.

Size	<100	101 – 500	501 – 1000	1001-5000	>5000	Overall
Average	-21%	-2%	-6%	13%	12%	-2%
Maximum	5%	50%	31%	46%	36%	50%
Minimum	-116%	-69%	-87%	-38%	0%	-116%
Median	0%	0%	1%	13%	6%	1%
Std Dev	41%	22%	33%	21%	17%	28%
Sample	8	26	10	12	4	60

Table 3.4.2(D): Cash Margin of Public Software Companies in US

3.4.3.5 Cash Reserve

Table 3.4.2(E) presents a summary of how software industry in general performs. The overall industry cash reserve is not very sensitive to macroeconomic climate, if data from the same period is not available, the data from other period can be used as a close proximate for comparison.

Size	<100	101 – 500	501 – 1000	1001-5000	>5000	Overall
Average	13.3	16.4	19.4	26.3	16.1	18.6
Maximum	78.0	103.8	76.5	79.5	22.4	103.8
Minimum	0.1	0.3	1.5	0.3	4.6	0.1
Median	0.9	8.2	15.3	13.6	21.2	9.5
Std Dev	28.7	22.4	22.0	27.3	10.0	23.4
Sample	8	26	10	12	4	60

Table 3.4.2(E): Cash Reserve of Public Software Companies in US

3.4.3.6 Debt to Equity Ratio

Table 3.4.2(F) presents a summary of how software industry in general performs. The overall industry debt to equity ratio is not sensitive to macroeconomic climate, if data from the same period is not available, the data from other period can be used as a close proximate for comparison.

Size	<100	101 – 500	501 – 1000	1001-5000	>5000	Overall
Average	0.08	0.07	0.07	0.34	0.01	0.12
Maximum	0.35	0.85	0.27	1.24	0.05	1.24
Minimum	0.00	0.00	0.00	0.00	0.00	0.00
Median	0.02	0.00	0.00	0.22	0.00	0.00
Std Dev	0.13	0.18	0.11	0.40	0.02	0.25
Sample	8	26	10	12	4	60

Table 3.4.2(F): Debt to Equity Ratio of Public Software Companies in US

4. APPLICATIONS OF VERPRO

4.1 OVERVIEW

This section consists of two examples aimed at providing the details that are fundamental to the understanding of the implementation of VERPRO in a COTS acquisition process. The 1st example describes how a fictional organization, Solar, Inc., evaluates four alternative COTS products from Mercury Tech, Venus Tech, Mars Tech, and Jupiter Tech, with respect to its preliminary SRS for an internal corrective action tracking tool, which is categorized as a workflow management product. The 2nd example describes how another fictional organization, Monster, Inc., evaluates four alternative COTS products from Diamond Tech, Emerald Tech, Ruby Tech, and Jade Tech, with respect to its preliminary SRS for an enterprise security software solution for managing user identities and access. For simplicity, it is assumed that all four alternatives for both cases are able to deliver equivalent products at the same price range and service level. Therefore, the final selection decision will be based on the business factors.

4.2 EXAMPLE 1

The company profiles and financial data for Mercury Tech, Venus Tech, Mars Tech, and Jupiter Tech, are derived from Hyperion Solution Corp., Microstrategy Inc., Parametric Technology Corp., and Agile Software Corp. respectively. These companies belong to the Software and Programming segment in NASDAQ (A stock exchange in the US). They are selected because:

- Their financials resemble those of the COTS vendors in workflow software segment, where no individual players have a dominant position.
- Their financial data are regulated and readily accessible via the Internet.

Except for market share, all data are derived from the companies' 2003 annual report.

4.2.1 Background

Solar, Inc., with annual revenue of US40 Billion and worldwide employees of 35,000, designs, develops, manufactures, markets, services and supports a range of computer systems, including enterprise systems (servers, storage and networking products and workstations), notebook computer systems, desktop computer systems and software and peripherals. The company is headquartered in the US and has operations throughout the world. It is organized into four geographic business units (Americas, EMEA, APAC, Japan) and a few centralized functional units (Finance, IT, Design, HR, Legal) that support the geographic business units.

The IT department has staff of over 3000 supporting all facets of the operation. Despite the size of the IT department, over 90% of the company's mission critical systems (ERP, Supply Chain, CRM, and Manufacturing Tracking) are based on COTS. The IT staffs have extensive experience in integration, maintenance and support but not development.

The company also has a venture capital arm, which allows it to make strategic acquisitions or alliances when the needs or opportunity arise. Through this venture capital arm, the company currently has equity or direct investments in more than 80 technology firms worldwide.

In response to an incident that costs the company US5 Million of sales because of a failed follow up to a corrective action request from a key customer, the CEO has commissioned a team to look into the company's corrective action tracking system. The results of the investigation revealed that the existing corrective action tracking system is highly fragmented; each business/functional unit has its own tracking system on separate platforms that are not linked to a central database, as a result, there is no systematic way to track a corrective action that involves more than one business or functional unit. In addition, the business process is also not standardized, some has stricter closure criteria than the other, and because of invisibility, a lot of the corrective actions are still open after a few months. In the absence of a centralized tracking system, it is impossible for the executive to monitor the status and react to the urgency of the issues. Upon reviewing the report, the CEO has decided to adopt a company wide internal corrective system that is supported by a workflow tool. The project will involve standardizing the individual business/functional unit's existing corrective action process, and building or acquiring a software tool to support the process. Because of the company's historical preference of using COTS products, the decision will lean toward buy unless a suitable product is not available in the market.

The team that is commissioned to work on this project consists of the process owners from the business/functional units (to re-engineering the current process, act as end user), an IT consultant (to provide technical assistance), a financial analyst (to provide cost analysis), and a purchasing consultant (to deal with vendor and negotiate contract). After a few intensive meetings, a preliminary SRS is created. The top five requirements are:

1. The system must be web-based and compatible with any widely used web-browser. No installation on the end user system shall be required.
2. The system must be expandable to support up to 10,000 users without any upgrade.
3. The system shall have a Windows based Administrator tool that enable an internal staff to maintain the day to day operation of the system (create and modify workflow, create and modify user, etc) without the need to involve the vendor.
4. The system shall be compatible with the company's existing database management system without the need to develop additional middleware.
5. The system must be able to communicate with the company's other systems such as email, ERP, Supply Chain etc.

Based on the SRS, the team screened through 24 potential products and ultimately singled out 4 after eliminating 20 that did not meet the "must" list.

4.2.2 A Brief Profile of the Alternatives

Mercury Tech, with annual revenue of 510 Millions and total employees of 2500, is a provider of business performance and workflow management software that enables companies to translate strategies into plans, monitor execution and provide insight to manage and improve financial and operational performance. The Company's applications, along with its development and deployment platform, enable business performance management across a wide variety of functional and operational areas beyond the finance areas of the business. The Company also offers support and services from offices in 20 countries, and works with over 330 partners to provide solutions to more than 6,000 customer organizations worldwide.

Venus Tech, with annual revenue of 175 Millions and total employees of 800, is a provider of business intelligence and workflow management software that enables companies to analyze the raw data stored across their enterprise to reveal the trends and insights needed to develop solutions to manage their business effectively. The Company's software delivers this critical information to workgroups, the enterprise and extranet communities via e-mail, Web, fax, wireless and voice communication channels. The Company offers an integrated business intelligence platform, which is designed to enable businesses to turn information into strategic insight and make more effective business decisions.

Mars Tech, with annual revenue of 670 Millions and total employees of 3500, develops, markets, and supports product lifecycle and workflow management software solutions and related services that help manufacturers improve the competitiveness of their products and product development processes. The Company's solutions enable manufacturing companies to create virtual computer-based products, collaborate on designs within the enterprise and throughout the extended supply chain and control the digital product information throughout the product lifecycle. Its software solutions are complemented by its services and technical support organizations, as well as resellers, systems integrators and other strategic partners, who provide training, consulting, ancillary product offerings, implementation and support to customers worldwide.

Jupiter Tech, with annual revenue of 70 Millions and total employees of 400, develops and sells a broad suite of product lifecycle and workflow management software applications that enable enables distributed organizations to cost effectively leverage the Web to quickly and easily capture and route paper and electronic documents into a workflow process.

4.2.3 Quantifying VERPRO Business Metrics

The markets for workflow management software are intensely competitive and subject to rapidly changing technology. The industry segment belongs to a domain that can be characterized by strong change forces and weak resistance (De Kluyver 2003). When the forces of change far exceed the threshold of resistance and change occurs continuously, no individual player or even a small group of players can block the forces of change. As a result, many companies in these markets are offering, or may soon offer, products and services that may compete with each other.

Before performing pairwise comparison for the alternatives, we need to first compute the metrics (VERPRO Business Metrics) for each alternative to substantiate the ranking. The results for the primary metrics are presented in Table 4.2.1. In addition to the metrics, the trends, and the relative performance of the alternatives to the industry are also included. The trend is studied to determine if the condition is improving or deteriorating. All metrics are computed based on the formulae and protocols presented in Section 3.3.2.4.

Metrics	Mercury	Venus	Mars	Jupiter
Product Leadership:				
Market Share	8%	4%	6%	5%
Market Share Trend	Stable	Down	Up	Stable
R&D Budget	15%	17%	18%	37%
R&D Budget Trend	Down	Down	Up	Up
Relative to the Average of Industry Benchmark	Same	Higher	Higher	Higher
Relative to the Median of Industry Benchmark	Same	Higher	Higher	Higher

Metrics	Mercury	Venus	Mars	Jupiter
Revenue Growth	4%	19%	-9%	-9%
Relative to Industry Average	Higher	Higher	Lower	Lower
Relative to Industry Median	Higher	Higher	Lower	Lower
Financial Strength:				
Current Ratio	2.79	1.45	1.40	8.57
Current Ratio Trend	Up	Up	Down	Down
Relative to Industry Average	Higher	Lower	Lower	Higher
Relative to Industry Median	Higher	Lower	Lower	Higher
Average Cash Margin	8%	1%	-3%	-29%
Cash Margin Trend	Up	Up	Down	Down
Relative to Industry Average	Lower	Higher	Lower	Lower
Relative to Industry Median	Lower	Higher	Lower	Lower
Cash Reserve	15	10	8	8
Cash Reserve Trend	Down	Up	Up	Down
Relative to Industry Average	Lower	Lower	Lower	Lower
Relative to Industry Median	Higher	Lower	Lower	Same
Business Risk:				
Debt to Equity	0.13	No debt	No debt	No debt
Debt coverage	17	N/a	N/a	N/a
Cash to debt	8	N/a	N/a	N/a

Metrics	Mercury	Venus	Mars	Jupiter
Number of Customers	6000	2500	3500	850
Revenue risk, top 1	<1%	<1%	<1%	<5%
Revenue risk, top 10	<5%	<5%	<10%	<10%
Lawsuits	Yes	Yes	Yes	Yes
Claim Coverage	0.1	0.6	0.4	1.2

Table 4.2.1: Results of the Metrics

4.2.4 Implementing VERPRO Decision Making Model

The relative preference between the alternatives is compared using a nine point scale commonly used in Analytic Hierarchy Process (Saaty 1980). The definition of the scale is presented in Table 4.2.2. The entire exercise is done using Expert Choice. A positive value denotes more favorable and a negative value denotes less favorable.

Numerical Value	Verbal Definition
1	Equally dominant
3	Moderately more dominant
5	Strongly more dominant
7	Very strongly more dominant
9	Extremely more dominant
2,4,6,8	Intermediate values

Table 4.2.2: Nine Point Pairwise Comparison Scale (Saaty 1980)

Market share data range from 4% to 8% indicating none of the alternatives command a dominant position. However, Mercury with 8% shares is strongly more dominant than Venus with 4% shares, and moderately more dominant than

Mars and Jupiter. Mars and Jupiter are equally dominant, and moderately more dominant than Venus. The pairwise comparison is summarized in Table 4.2.3(A).

Market Share	**Mercury**	**Mars**	**Jupiter**	**Venus**
Mercury		3	3	5
Mars			1	3
Jupiter				3
Venus				
Consistency ratio: 0.02				

Table 4.2.3(A): Pairwise Comparison for Market Share

R&D Budget for Mercury, Venue and Mars closely trace the industry benchmark (Table 3.4.2), the R&D Budget for Jupiter is more than twice the industry benchmark because it is still in its early stage of development and a huge portion of its R&D investment has not turned into final product yet. In terms of R&D Budget, no one has a distinct advantage over the other.

Revenue growth data range from -9% to 19%, with Mercury and Venus performing better than the industry average, while Mars and Jupiter trailing the industry average. In terms of Revenue growth, Venus is moderately more dominant than Mercury (because Venus has a much smaller revenue base), and very strongly more dominant than Mars and Jupiter. Mercury is strongly more dominant than Mars and Jupiter. Mars and Jupiter are equally dominant. The pairwise comparison is summarized in Table 4.2.3(B).

Rev Growth	**Mercury**	**Mars**	**Jupiter**	**Venus**
Mercury		5	5	-3
Mars			1	-7
Jupiter				-7
Venus				

Rev Growth	Mercury	Mars	Jupiter	Venus
Consistency ratio: 0.03				

Table 4.2.3(B): Pairwise Comparison for Revenue Growth

Market Share is moderately more important than Revenue Growth with respect to Product Leadership. The pairwise comparison between Market Share and Revenue Growth is summarized in Table 4.2.3(C).

	Market Share	Rev Growth
Market Share		3
Rev Growth		
Consistency ratio: 0.00		

Table 4.2.3(C): Relative Importance of Market Share and Revenue Growth

The output from Expert Choice for Product Leadership is summarized in Table 4.2.3(D).

	Market Share (0.750)	Rev Growth (0.250)	Overall Score
Mercury	.522	.282	.467
Mars	.200	.067	.169
Jupiter	.200	.067	.169
Venus	.078	.583	.194

Table 4.2.3(D): Overall Score of the Alternatives with respect to Product Leadership

Current ratio data range from 1.40 to 8.57, with Mercury and Jupiter performing better than the industry average, while Venus and Mars trailing the industry average. In terms of Current ratio, Mercury and Jupiter are equally dominant, and both are moderately more dominant than Venus and Mars. Although the current ratio for Jupiter is about four times that of Mercury, they are considered equally dominant because according to (Graham 1975), any number above two indicates sufficient margin of safety. The pairwise comparison is summarized in Table 4.2.3(E).

Current Ratio	**Mercury**	**Mars**	**Jupiter**	**Venus**
Mercury		3	1	3
Mars			-3	1
Jupiter				3
Venus				
Consistency ratio: 0.00				

Table 4.2.3(E): Pairwise Comparison for Current Ratio

Cash margin data range from -29% to 8%, with Venus performing better than the industry average and the rest trailing the industry average. In terms of Cash margin, Mercury is moderately more dominant than Venus, and very strongly more dominant than Mars and Jupiter. Venus is strongly more dominant than Mars and Jupiter, and Mars and Jupiter are equally dominant. Although Mercury trails the industry average, 8% is an impressive value, which indicates 8% of its revenue is converted into cash to strengthen its financial position. Although Jupiter has a much lower number compared to that of Mars, they are considered equally dominant because Jupiter has a much lower revenue base, meaning the absolute cash burn is lower, and the possibility of its cash margin turning into the positive territory is greater with new product release. The pairwise comparison is summarized in Table 4.2.3(F).

Cash Margin	**Mercury**	**Mars**	**Jupiter**	**Venus**
Mercury		7	7	3
Mars			1	-5
Jupiter				-5

Cash Margin	Mercury	Mars	Jupiter	Venus
Venus				
Consistency ratio: 0.03				

Table 4.2.3(F): Pairwise Comparison for Cash Margin

Cash reserve data range from 8 to 15 years, with all trailing the industry average. Mercury is moderately more dominant than Venus and strongly more dominant than Mars and Jupiter. Venus is moderately more dominant than Mars and Jupiter, and Mars and Jupiter are equally dominant. Although none of the alternatives have a near term insolvency risk, the alternatives with stronger cash reserve have more flexibility and advantage to make strategic investment or acquisition to strengthen its product lines and market position, and longer staying power in the event of a prolonged downturn. The pairwise comparison is summarized in Table 4.2.3(G).

Cash Reserve	Mercury	Mars	Jupiter	Venus
Mercury		5	5	3
Mars			1	-3
Jupiter				-3
Venus				
Consistency ratio: 0.02				

Table 4.2.3(G): Pairwise Comparison for Cash Margin

Cash Margin and Cash Reserve are moderately more important than Current Ratio. The pairwise comparison is summarized in Table 4.2.3(H).

	Current Ratio	Cash Margin	Cash Reserve
Current Ratio		-3	-3
Cash Margin			1
Cash Reserve			

	Current Ratio	Cash Margin	Cash Reserve
Consistency ratio: 0.00			

Table 4.2.3(H): Relative Importance of Current Ratio, Cash Margin and Cash Reserve with respect to Financial Strength

The output from Expert Choice for Financial Strength is summarized in Table 4.2.3(I).

	Current Ratio (0.143)	Cash Margin (0.429)	Cash Reserve (0.429)	Overall Score
Mercury	.375	.583	.560	.531
Mars	.125	.067	.095	.090
Jupiter	.375	.067	.095	.141
Venus	.125	.282	.249	.237

Table 4.2.3(I): Overall Score of the Alternatives with respect to Financial Strength

In terms of Debt to equity ratio, no one has a distinct advantage over the other. Although Mercury does carry some long term debt, its debt coverage and cash to debt ratio indicate its ability to service its debt load. The pairwise comparison is summarized in Table 4.2.3(J)

Debt to Equity	**Mercury**	**Mars**	**Jupiter**	**Venus**
Mercury		1	1	1
Mars			1	1
Jupiter				1
Venus				
Consistency ratio: 0.00				

Table 4.2.3(J): Pairwise Comparison for Debt to Equity

In terms of revenue risk, no one has a distinct advantage either. None have over 10% of its revenue coming from its top customer. The pairwise comparison is summarized in Table 4.2.3(K).

Rev Risk	Mercury	Mars	Jupiter	Venus
Mercury		1	1	1
Mars			1	1
Jupiter				1
Venus				
Consistency ratio: 0.00				

Table 4.2.3(K): Pairwise Comparison for Revenue Risk

In terms of legal liability, Mercury is moderately more dominant than Venus and Mars, and strongly more dominant than Jupiter. Venus and Mars are equally dominant and both are moderately more dominant than Jupiter. Jupiter has a claim coverage of more than one which indicates a potential default risk in the event of unfavorable rulings. The pairwise comparison is summarized in Table 4.2.3(L).

Legal	Mercury	Mars	Jupiter	Venus
Mercury		3	5	3
Mars			3	1
Jupiter				-3
Venus				
Consistency ratio: 0.02				

Table 4.2.3(L): Pairwise Comparison for Legal Liability

For Solar, Inc., Product Leadership is deemed moderately more important than Financial Strength and Business Risk. Product Leadership has the top priority because Solar, Inc., despite its large IT department, has little experience in software development, and therefore is more dependent on its vendor in this area. In addition, none of the alternatives have a dominant market position. Financial Strength has a lower priority because Solar, Inc., through its venture capital arm, has the ability to provide financial assistance to its vendor should the need arise. Business Risk also has a lower priority because none of the alternatives have a distinct advantage in terms of debt and revenue risk, and except for Jupiter, none is likely to be financial damaged by lawsuits. The relative importance of Product Leadership, Financial Strength, and Business Risk is summarized in Table 4.2.3(M).

	Product Leadership	**Financial Strength**	**Business Risk**
Product Leadership		3	5
Financial Strength			3
Business Risk			
Consistency ratio: 0.00			

Table 4.2.3(M): Relative Importance of the Objectives

The overall output from Expert Choice is summarized in Table 4.2.3(N).

	Product Leadership (0.637)	**Financial Strength (0.258)**	**Business Risk (0.105)**	**Overall Score**
Mercury	.467	.531	.522	.490
Mars	.169	.090	.200	.152
Jupiter	.169	.141	.078	.152
Venus	.194	.237	.200	.206

Table 4.2.3(N): Overall Score of the Alternatives with respect to VBF

Based on the results, Mercury has the highest overall score, followed by Mars. Venus and Jupiter have the lowest score. At the end of this exercise, Mercury is selected as the most preferred alternative. The subsequent step is to assess Mercury's financial position and determine if there is any impending risk. A summary of the comparison with the industry profiles is presented in Table 4.2.4. The comparison shows that Mercury has an overall low risk in terms of product leadership, financial strength, and business risk. We recommend a monitoring frequency of once per year because this is when new financial data are released.

Metrics	Mercury	Average	Median	Risk
Product Leadership:				
Market Share	8%	N/a	N/a	N/a
Market Share Trend	Stable			
R&D Budget	15%	15%	15%	No
R&D Budget Trend	Down			
Revenue Growth	4%	2.38%	-2.15%	Low
Financial Strength:				
Current Ratio	2.79	2.00	2.03	No
Current Ratio Trend	Up			
Average Cash Margin	8%	13%	13%	Low
Cash Margin Trend	Up			
Cash Reserve	15	26	13	No
Cash Reserve Trend	Down			
Business Risk:				
Debt to Equity	0.13	0.34	0.22	No
Debt coverage	17			
Cash to debt	8			
Number of Customers	6000	N/a	N/a	No
Revenue risk, top 1	<1%			

Metrics	Mercury	Average	Median	Risk
Lawsuits	Yes	N/a	N/a	Low
Claim Coverage	0.1			

Table 4.2.4: Comparison of Mercury with Industry Profiles

4.3 EXAMPLE 2

The company profiles and financial data for Diamond Tech, Emerald Tech, Ruby Tech, and Jade Tech, are derived from Verisign Inc., Netegrity Inc., RSA Security Inc., and Symantec Corp. respectively. These companies belong to the Software and Programming segment in NASDAQ (A stock exchange in the US). They are selected because:

- They are dominant players in the user identity and access management solution segment and they offer products that match the SRS of Monster, Inc.
- Their financial data are regulated and readily accessible via the Internet.

Except for market share, all data are derived from the companies' 2003 annual report.

4.3.1 Background

Monster, Inc., (thereafter referred to as the company) with annual revenue of US 300 Million and worldwide employees of 330, is an online closeout retailer offering discount, brand-name merchandise for sale primarily over the Internet. Its merchandise offerings include bed-and-bath goods, kitchenware, watches, jewelry, electronics, sporting goods and designer accessories. The Company also sells books, magazines, CDs, DVDs, videocassettes and video games (BMV). It offers its customers an opportunity to shop for bargains conveniently, while offering its suppliers an alternative inventory liquidation distribution channel. It typically offers approximately 12,000 non-BMV products and approximately 500,000 BMV products in up to 12 departments on its Websites. The Company also offers travel services, including airline tickets, hotel reservations and car rentals. It has a direct business in which it buys and takes possession of excess inventory for resale.

The company has a relatively small IT department with a full time staff of only 50 supporting all facets of the operation. Because of the size of the IT department, over 90% of the company's mission critical systems (ERP, Supply Chain, CRM, etc.) are based on COTS and outsourced. The IT staffs have some experience in integration, maintenance and support but not development. The company has a relatively small IT budget (<5% of revenue), which limits its ability to make strategic acquisitions or alliances with its technology partners.

With more and more business processes moving online, the company is faced with the challenge of conducting business securely both within and beyond traditional corporate boundaries. Upon recognizing this challenge, the CEO has commissioned a team to explore the need to address this challenge and expand the need into a preliminary SRS. After a few rounds of intensive meetings with the stakeholders, the team decided to focus the SRS on the following areas:

1. **Securely let business in:** to generate more revenue from existing customers, reach new markets, create new selling channels, deliver online customer service, integrate partners and suppliers into the supply chain, decrease time-to-market, and increase the company's competitive position. The need is further expanded into the following points:
    a. Support the needs of a highly heterogeneous user population.
    b. Create a virtual enterprise by sharing product information with a range of suppliers and distributors.
    c. Allow easy sales force access from remote locations.
    d. Provide online customer purchasing through the creation of a portal for brokers, distributors, business partners, as well as customers.
    e. Very high scalability to support potentially very large suppliers, distributors and customer base.

2. **Keep risk out:** to protect corporate assets, comply with privacy and disclosure regulations, and enhance corporate-wide governance. The need is further expanded into the following point:
    a. Flexible access management policies so that user access can be controlled based on a number of factors.
3. **Reduce costs:** by leveraging resources and information across the organization, eliminating redundancy, automating processes, and reducing cycle times. The need is further expanded into the following points:
    a. Delegated administration so that supply chain partners can manage the identities of their own users.
    b. Support for corporate-level mandates for standardization across all divisions.
    c. Self-service to allow business partners, distributors and customers to control their own profile information (with appropriate rights).

After some extensive research work, the team has decided to limit their search for a COTS solution within the Identity and Access Management solutions segment. The key requirements of the preliminary SRS are summarized into the following points:

- Equipped with an integrated and dynamic workflow processes.
- Supported by heterogeneous directories and databases in e-commerce and B2B environment.
- Fortified with strong security features so that inappropriate access can be easily detected and controlled.
- Supplemented by a comprehensive reporting and auditing module for monitoring access and ensuring regulatory compliance.
- Supported by Single Sign-On and federated identities within the company or beyond it to include the company's suppliers, distributors, and customers.
- Enhanced with a distributed administration module so that users can be managed by business units or partners.

- Designed to be platform independent and multiple-infrastructure compatible.
- Designed to have an open architecture compatible with existing IT infrastructure and emerging standards to enable integration of existing and future applications.
- Scalable, manageable and responsive to cope with rapid growth in number of users into the millions.

Based on the preliminary SRS, the team screened through 20 potential products and finally singled out 4 after eliminating 16 that did not meet the "must" list.

4.3.2 A Brief Profile of the Alternatives

Diamond Tech, with annual revenue of 1 Billion and total employee of 2500, is a provider of critical infrastructure services. The Company is organized into two service-based lines of business: the Internet Services Group and the Communications Services Group. The Internet Services Group consists of the Security Services business and the Naming and Directory Services business. The Security Services business provides products and services that enable enterprises and organizations to establish and deliver secure Internet-based services to customers and business partners, and the Naming and Directory Services business acts as the exclusive registry of domain names in the .com and .net generic top-level domains and certain country code top-level domains. The Communications Services Group provides Signaling System 7 network services, intelligent database and directory services, application services, and billing and payment services to wireline and wireless telecommunications carriers.

Emerald Tech, with annual revenue of 85 Million and total employee of 400, is a provider of enterprise security software solutions specifically for managing user identities and access. The Company's identity and access management product line gives companies a secure way to make corporate information assets and resources available online. With the Company's identity and access management products, companies are able to securely use the Web (Internet, intranet or extranet) to meet the information access needs of partners, suppliers, customers and employees. The company's solutions enable businesses to ensure that the right people have the right access to the right information across a variety of applications, business systems and computing architectures. The Company's core products offer a single source solution for integrated, centralized identity management, user access and administration and account provisioning/de-provisioning.

Ruby Tech, with annual revenue of 270 Million and total employee of 1050, helps organizations protect private information and manage the identities of the people and applications accessing and exchanging that information. The Company's portfolio of solutions, including identity and access management, secure mobile and remote access, secure enterprise access and secure transactions, are all designed to provide a seamless e-security experience. In addition to its portfolio of solutions, the Company operates a research center that provides cryptography and security technology services. The company has more than 14,000 customers worldwide.

Jade Tech, with annual revenue of 1.9 Billion and total employees of 4300, provides content and network security software and appliance solutions to enterprises, individuals and service providers. The Company provides client, gateway and server security solutions for virus protection, firewall and virtual private network (VPN), security management, intrusion detection, Internet content and e-mail filtering, remote management technologies and security services to enterprises and service providers worldwide. The company has offices in 36 countries worldwide. The Company views its business in five operating segments: enterprise security, enterprise administration, consumer products, services and other activities.

4.3.3 Quantifying VERPRO Business Metrics

The identity and access management market is characterized by rapid technological change, changes in customer requirements, new product introductions and enhancements and emerging industry standards. The players need to devote significant time and resources to analyze and respond to industry changes, such as those in operating systems, application software, security standards, networking software and evolving customer requirements, and expect to continue to make a substantial investment in research and development. The market is relatively new, rapidly evolving and highly competitive. Competition is expected to continue to increase both from existing competitors and new market entrants. The ability of the players to compete depends on factors within and beyond their control, including: the performance, reliability, features, price and ease of use of their products as compared to those of their competitors; their ability to secure and maintain key strategic relationships; their ability to expand domestic and international sales operations; their ability to support their customers; and the timing and market acceptance of new products and enhancements to existing products by their competitors.

Before performing pairwise comparison for the alternatives, we need to first compute the metrics (VERPRO Business Metrics) for each alternative to substantiate the ranking. The results for the primary metrics are presented in Table 4.3.1(A). In addition to the metrics, the trends, and the relative performance of the alternatives to the industry are also included. The trend is studied to determine if the condition is improving or deteriorating. All metrics are computed based on the formulae and protocols presented in Section 3.3.2.4.

Metrics	Diamond	Emerald	Ruby	Jade
Product Leadership:				
Market Share	12%	14%	20%	18%
Market Share Trend	Stable	Up	Stable	Up
R&D Budget	6%	26%	22%	15%
R&D Budget Trend	Up	Down	Down	Down

Metrics	Diamond	Emerald	Ruby	Jade
Relative to the Average of Industry Benchmark	Lower	Higher	Higher	Same
Relative to the Median of Industry Benchmark	Lower	Higher	Higher	Same
Revenue Growth	5%	-4%	-3%	28%
Relative to Industry Average	Higher	Lower	Lower	Higher
Relative to Industry Median	Higher	Lower	Lower	Higher
Financial Strength:				
Current Ratio	1.59	2.11	1.36	2.22
Current Ratio Trend	Up	Down	Down	Down
Relative to Industry Average	Lower	Lower	Lower	Higher
Relative to Industry Median	Lower	Lower	Lower	Higher
Average Cash Margin	14%	-11%	23%	31%
Cash Margin Trend	Up	Up	Up	Down
Relative to Industry Average	Higher	Lower	Higher	Higher
Relative to Industry Median	Higher	Lower	Higher	Higher
Cash Reserve	4	17	43	9
Cash Reserve Trend	Up	Up	Up	Down
Relative to Industry Average	Lower	Higher	Higher	Lower
Relative to Industry	Lower	Higher	Higher	Lower

Metrics	Diamond	Emerald	Ruby	Jade
Median				
Business Risk:				
Debt to Equity	No debt	No debt	No debt	No debt
Debt coverage	N/a	N/a	N/a	N/a
Cash to debt	N/a	N/a	N/a	N/a
Number of Customers	>5000	800	>5000	>5000
Revenue risk, top 1	<1%	<1%	<1%	<1%
Revenue risk, top 10	<5%	<10%	<5%	<5%
Lawsuits	Yes	No	Yes	Yes
Claim Coverage	0.04	N/a	0.03	0.003

Table 4.3.1: Results of the Metrics

4.3.4 Implementing VERPRO Decision Making Model

Like Example 1, the relative preference between the alternatives is compared using a nine point scale commonly used in Analytic Hierarchy Process (Saaty 1980). The definition of the scale is presented in Table 4.2.2. The entire exercise is done using Expert Choice. A positive value denotes more favorable and a negative value denotes less favorable.

Market share data range from 12% to 20% indicating these four alternatives command 64% of the total market. Ruby with 20% shares is in fact the biggest player in this segment, followed by Jade with 18% shares. Diamond and Emerald are one of the top ten players in this segment. In comparison, Ruby is moderately more dominant than Jade, strongly more dominant than Emerald, and very strongly more dominant than Diamond. Jade is moderately more dominant than Emerald and strongly more dominant than Diamond, and Emerald is moderately more dominant than Diamond. The pairwise comparison is summarized in Table 4.3.2(A).

Market Share	Diamond	Emerald	Ruby	Jade
Diamond		-3	-7	-5
Emerald			-5	-3
Ruby				3
Jade				
Consistency ratio: 0.03				

Table 4.3.2(A): Pairwise Comparison for Market Share

R&D Budget data range from 6% to 26%, with Emerald, Ruby and Jade trace or exceed the industry benchmark (Table 3.4.2), while Diamond significantly lower than the industry benchmark. Emerald and Ruby have significantly higher R&D Budget because they are still in the early stage of development and a large portion of its investment has not turned into revenue yet. Diamond has a significantly lower R&D Budget because the company acquires most of its products and technology through mergers and acquisitions. In terms of R&D Budget, Emerald, Ruby and Jade are equally dominant; however, they are strongly more dominant than Diamond. Acquiring products through merger and acquisition is usually more expensive and difficult to integrate. The pairwise comparison is summarized in Table 4.3.2(B).

R&D Budget	Diamond	Emerald	Ruby	Jade
Diamond		-5	-5	-5
Emerald			1	1
Ruby				1
Jade				
Consistency ratio: 0.00				

Table 4.3.2(B): Pairwise Comparison for R&D Budget

Revenue growth data range from -4% to 28%, with Diamond and Jade performing better than the industry average, while Emerald and Ruby trailing the industry average. In terms of Revenue growth, Jade is strongly more dominant than Diamond, and very strongly more dominant than Emerald and Ruby. Diamond is moderately more dominant than Emerald and Ruby, and Emerald and Ruby are equally dominant. Diamond is considered only moderately more dominant than Emerald and Ruby despite its considerably better result because much of its growth is purchased through merger and acquisitions. This type of growth is considered less desirable than organic growth. The pairwise comparison is summarized in Table 4.3.2(C).

Rev Growth	**Diamond**	**Emerald**	**Ruby**	**Jade**
Diamond		3	3	-5
Emerald			1	-7
Ruby				-7
Jade				
Consistency ratio: 0.03				

Table 4.3.2(C): Pairwise Comparison for Revenue Growth

Market Share is strongly more important than R&D Budget and moderately more important than Revenue Growth. Revenue Growth is moderately more important than R&D Budget. The pairwise comparison with respect to Product Leadership is summarized in Table 4.3.2(D).

	Market Share	**R&D Budget**	**Rev Growth**
Market Share		5	3
R&D Budget			-3
Rev Growth			
Consistency ratio: 0.00			

Table 4.3.2(D): Relative Importance of Market Share, R&D Budget and Revenue Growth with respect to Product Leadership

The output from Expert Choice for Product Leadership is summarized in Table 4.3.2(E).

	Market Share (0.637)	R&D Budget (0.105)	Rev Growth (0.258)	Overall Score
Diamond	.055	.063	.191	.085
Emerald	.118	.313	.076	.144
Ruby	.565	.313	.076	.416
Jade	.262	.313	.657	.355

Table 4.3.2(E): Overall Score of the Alternatives with respect to Product Leadership

Current ratio data range from 1.36 to 2.22, with Jade performing better than the industry average, and the rest trailing the industry average. In terms of Current ratio, Jade and Emerald are equally dominant and both are moderately more dominant than Diamond and Ruby. Diamond and Ruby are equally dominant. The pairwise comparison is summarized in Table 4.3.2(F).

Current Ratio	**Diamond**	**Emerald**	**Ruby**	**Jade**
Diamond		-3	1	-3
Emerald			3	1
Ruby				-3
Jade				
Consistency ratio: 0.00				

Table 4.3.2(F): Pairwise Comparison for Current Ratio

Cash margin data range from -11% to 31%, with Diamond, Ruby and Jade performing better than the industry average and Emerald trailing the industry average. In terms of Cash margin, Ruby and Jade are equally dominant, both are moderately more dominant than Diamond, and very strongly more dominant than Emerald. Diamond is strongly more dominant than Emerald. Although Jade has a higher number than Ruby, they are considered equal because a cash margin of over 20% indicates a very strong financial position and number higher than that may not be sustainable. The pairwise comparison is summarized in Table 4.3.2(G).

Cash Margin	**Diamond**	**Emerald**	**Ruby**	**Jade**
Diamond		5	-3	-3
Emerald			-3	-7
Ruby				1
Jade				
Consistency ratio: 0.04				

Table 4.3.2(G): Pairwise Comparison for Current Ratio

Cash reserve data range from 4 to 43 years, with Emerald and Ruby performing better than the industry average, and Diamond and Jade trailing the industry average. Ruby is moderately more dominant than Emerald and Jade, and very strongly more dominant than Diamond. Emerald and Jade are equally dominant, and both are strongly more dominant than Diamond. Despite its higher number, Emerald is considered equally dominant with Jade because Jade has a much stronger cash margin. The pairwise comparison is summarized in Table 4.3.2(H).

Cash Reserve	**Diamond**	**Emerald**	**Ruby**	**Jade**
Diamond		-5	-7	-5
Emerald			-3	1

Cash Reserve	Diamond	Emerald	Ruby	Jade
Ruby				3
Jade				
Consistency ratio: 0.03				

Table 4.3.2(H): Pairwise Comparison for Cash Reserve

Cash Margin and Cash Reserve are moderately more important than Current Ratio. The pairwise comparison with respect to Financial Strength is summarized in Table 4.3.2(I).

Financial Strength	Current Ratio	Cash Margin	Cash Reserve
Current Ratio		-3	-3
Cash Margin			1
Cash Reserve			
Consistency ratio: 0.00			

Table 4.3.2(I): Relative Importance of Current Ratio, Cash Margin and Cash Reserve with respect to Financial Strength

The output from Expert Choice for Financial Strength is summarized in Table 4.3.2(J).

	Current Ratio (0.143)	Cash Margin (0.429)	Cash Reserve (0.429)	Overall Score
Diamond	.125	.163	.052	.117
Emerald	.375	.047	.210	.160
Ruby	.125	.395	.528	.397
Jade	.375	.395	.210	.326

Table 4.3.2(J): Overall Score of the Alternatives with respect to Financial Strength

In terms of Debt to equity ratio and revenue risk, no one has a distinct advantage over the other. The pairwise comparison is summarized in Table 4.3.2(K) and (L).

Debt to Equity	Diamond	Emerald	Ruby	Jade
Diamond		1	1	1
Emerald			1	1
Ruby				1
Jade				
Consistency ratio: 0.00				

Table 4.3.2(K): Pairwise Comparison for Debt to Equity

Revenue Risk	Diamond	Emerald	Ruby	Jade
Diamond		1	1	1
Emerald			1	1
Ruby				1
Jade				
Consistency ratio: 0.00				

Table 4.3.2(L): Pairwise Comparison for Legal Liability

In terms of legal liability, Emerald and Jade are equally dominant, and both are moderately more dominant than Diamond and Ruby. Diamond and Ruby are equally dominant. Although none of the alternatives have an impending financial risk from lawsuits (all have claim coverage significantly lower than 1), the alternatives with lawsuits are disadvantaged because regardless of the outcome, any litigation may require the company to incur significant litigation expense and a significant diversion of management attention. The pairwise comparison is summarized in Table 4.3.2(M).

Legal	Diamond	Emerald	Ruby	Jade
Diamond		-3	1	-3
Emerald			3	1
Ruby				-3
Jade				
Consistency ratio: 0.00				

Table 4.3.2(M): Pairwise Comparison for Legal Liability

For Monster, Inc., Financial Strength is moderately more important than Product Leadership and very strongly more important than Business Risk, and Product Leadership is strongly more important than Business Risk. Financial Strength is deemed the most important because Monster, Inc., being a relatively small online retailer does not have the ability to support its vendor in the event of insolvency. Therefore, its top priority is to work with vendor that is least likely to run into financial distress. Product Leadership is also deemed important because Monster, Inc., with a relatively small IT budget, has very limited IT capability. Therefore, it has very high dependence on its vendor. Product Leadership is deemed less important than Financial Strength because all four alternatives, with a combined 64% market share, are considered dominant players in the user identity and access management segment. Business Risk is deemed the least important because none of the alternatives has a distinct advantage in terms of debt and revenue risk, and none is exposed to any lawsuits that would damage their financial position. The relative importance of Product Leadership, Financial Strength, and Business Risk is summarized in Table 4.3.2(N).

	Product Leadership	Financial Strength	Business Risk
Product Leadership		-3	5
Financial Strength			7

	Product Leadership	Financial Strength	Business Risk
Business Risk			
Consistency ratio: 0.06			

Table 4.3.2(N): Relative Importance of the Objectives

The overall output from Expert Choice is summarized in Table 4.2.3(O).

	Product Leadership (0.279)	Financial Strength (0.649)	Business Risk (0.072)	Overall Score
Diamond	.085	.117	.125	.110
Emerald	.144	.160	.375	.175
Ruby	.416	.397	.125	.378
Jade	.355	.326	.375	.337

Table 4.3.2(O): Overall Score of the Alternatives with respect to VBF

Based on the results, Ruby has the highest overall score, closely followed by Jade, Emerald, and Diamond has the lowest score. At the end of this exercise, Ruby is selected as the most preferred alternative. Since Jade closely trails Ruby, we can keep Jade as a backup vendor. The subsequent step is to assess Ruby's financial position and determine if there is any impending risk. A summary of the comparison with the industry profiles is presented in Table 4.3.4. The comparison shows that Ruby has an overall very low risk in terms of product leadership, financial strength, and business risk. We recommend a monitoring frequency of once a year.

Metrics	Ruby	Average	Median	Risk
Product Leadership:				

Metrics	Ruby	Average	Median	Risk
Market Share	20%	N/a	N/a	Low
Market Share Trend	Stable			
R&D Budget	22%	15%	15%	No
R&D Budget Trend	Down			
Revenue Growth	-3%	2.38%	-2.15%	No
		Financial Strength:		
Current Ratio	1.36	2.00	2.03	No
Current Ratio Trend	Down			
Average Cash Margin	23%	13%	13%	No
Cash Margin Trend	Up			
Cash Reserve	43	26	13	No
Cash Reserve Trend	Up			
		Business Risk:		
Debt to Equity	N/a	0.34	0.22	No
Debt coverage	N/a			
Cash to debt	N/a			
Number of Customers	>5000	N/a	N/a	No
Revenue risk, top 1	<1%			
Lawsuits	No	N/a	N/a	Low
Claim Coverage	N/a			

Table 4.3.4: Comparison of Ruby with Industry Profiles

5. FUTURE WORK

A viable method to validate the effectiveness of VERPRO is using case study. Case study is an ideal methodology when a holistic, in-depth investigation is needed. Case studies have been used in varied investigations, particularly in sociological studies, but increasingly, in the field of Information Technology and Software Engineering. Case study research is not sampling research. Case studies tend to be selective, focusing on one or two issues that are fundamental to understanding the theory being examined. However, selecting cases must be done so as to maximize what can be learned in the period of time available for the study.

The case study will consist of three stages

1. Design the case study protocol
    a. Determine the selection criteria for the subject
    b. Develop and review the protocol
2. Conduct the case study
    a. Prepare for data collection
    b. Develop and review data collection protocol
    c. Collect data
3. Analyze case study evidence
    a. Develop analytic strategy
    b. Establish conclusions and recommendations
    c. Elaborate implications

The subjects of the case study must fulfill the following criteria:

1. An organization that adopts COTS-based development model and uses COTS in at least 50% of its mission critical software systems. This is to ensure the organization has sufficient knowledge base in COTS and is willing to allocate sufficient resources in COTS-based systems.

2. An organization that averages at least one new COTS-based project in a year. This is to ensure the organization has sufficient data from COTS-based system for analysis.
3. An organization that has a well documented COTS selection process that includes a sufficiently defined vendor evaluation taxonomy (at least the equivalent of the vendor evaluation taxonomy depicted in Figure 2.4.2 but without considering vendor business factors). This is for constructing a well defined base line for tracking the improvement made by adding VBF.
4. An organization that has a well defined issue tracking system and maintains a complete log of issues generated from the existing COTS-based system. Preferably the issues are pre-classified into technical and non-technical, and non-technical issues are further classified into different bins and one of them must be vendor business related. The classification criteria must be defined and documented to ensure consistency. Once the issue is classified into the individual bins, the criticality of the issue must be assignment and the cost associating with the issue is estimated. This issue log will form the base line i.e. the number of vendor business related issues per system per year and the cost of vendor business related issues per system per year before the implementation of VERPRO.

Once the subjects of the case study are identified, a cut-off date will be issued and the subjects will start implementing VERPRO for their new COTS-based system while all other processes remain unchanged. The subject is given a year to collect and record issues generated from the new systems using the exact same method as before. The after data is then compared with the before data, and if the number of COTS vendor business related issues and the cost related to COTS vendor business related issues have gone down, we can conclude that VERPRO has helped the organization reduce COTS vendor business related issues.

Although the study provides a viable way to empirically validate the effectiveness of VERPRO, there are a few limitations:

- It may be a challenge to find a suitable subject for this study because few organizations have well-documented COTS acquisition process, even fewer formally employs a well-defined vendor evaluation taxonomy.
- No two software projects are identical. Therefore, the baseline that is constructed from historical data may not be suitable for future projects. In addition, two software projects can be very different in nature even though they may be equivalent in size (Line of Codes, Function Points), the most widely accepted way of measuring a software project.
- It is difficult to classify software issues because the root cause of the issues is sometimes unknown. As a result, the validity of the data is affected.

6. CONCLUSION

The migration to COTS based development from traditional development is having a profound impact on business, acquisition, and management practices, and organizational structures. A general lack of experience and the absence of a dependable COTS process have led to some high profile failures in both government and non-government organizations. This has led to the creation of a mission to redefine acquisition, management, and engineering practices to more effectively leverage the COTS marketplace, a task that is led by the SEI, along with other key organizations associated with the government, academic, and civil agencies.

The state-of-the-art of COTS related researches have so far focused on improving the knowledge in the evaluating technologies and products, developing requirements, and managing system and COTS product evolution. The majority of the research work is still in the early stages, and a large part of the effort is consolidated into a few COTS-based development models such as EPIC, and a few widely used COTS selection models such as OTSO and CAP, which are heavily focused on product centric factors such as functionality and cost. However, there is an increasing concern over the scope and bias of the existing models which lacks the ability to evaluate the business factors of COTS vendors. Business factors consist of the vendor's strategic position, financial strengths, and business risks.

The need to evaluate vendor business factors has been recognized by the SEI COTS Software Initiative Team as early as 1998. Despite some preliminary efforts, a method to interpret financial measures remain unexplored, and what constitutes "good enough" remains unknown. Over the years, despite the surge in COTS use, there has not been much progress in the development of a more concrete method to evaluate vendor business factors, resulting in a situation where vendor risk factors remain largely unknown. The urgency to extend the existing COTS vendor evaluation taxonomy to evaluate vendor business factors was triggered by a series of events in the early 00s that show the number of COTS vendors running into business distress is increasing in alarming rate.

VERPRO addresses the deficiencies of the state-of-the art COTS vendor selection process by extending the vendor evaluation taxonomy to include business factors. The foundation of VERPRO lies within a measurement based vendor evaluation taxonomy that categorizes the evaluation criteria into four main factors: product, cost, service and business, with business unique to VERPRO. VERPRO decision making tool is a multiple-objective, multiple-hierarchy decision making model that is based on the Analytic Hierarchy Process. VERPRO business factors consist of three key elements: product leadership, financial strengths and business risks. The thesis has explained how these elements are linked to the overall COTS selection process and presented a systematic process to measure these factors.

Product can never be separated from economic realities, by applying VERPRO, the acquisition community is able to refine the selection process by analyzing the economic conditions of the alternatives. The strengths of VERPRO can be summarized into the following points:

- VERPRO is addressing a valid and important issue in the acquisition community. The need to evaluate COTS vendor financial conditions has been highlighted as early as the late 90s by SEI COTS Initiative Team. The validity of this issue is further reinforced by a number of articles in the CIO magazine (Levinson 2001).
- VERPRO is an extension of the existing vendor selection process, therefore no major relearning is required to apply VERPRO.
- VERPRO decision making tool is based on the Analytic Hierarchy Process which is widely regarded as the best decision making mechanism available, and AHP has already been applied in Requirement Engineering, OTSO and CAP.
- VERPRO Business Factors (VBF) are not arbitrarily determined but systematically derived from historical data of failed software and

programming companies. This derivation method greatly improves the validity of VBF to other software companies.

- VERPRO Business Metrics are measurable and the supporting data are readily available.
- There are no complex economic models involved, therefore VERPRO can be applied without the need to employ high cost consulting services.
- VERPRO vendor evaluation taxonomy is open-ended, therefore, it can be extended when new information becomes available.

VERPRO transforms a risk mitigation strategy from one that is subjective and reactive, to one that is objective and proactive. It enables the acquisition community to incorporate business factors into vendor selection process and provides an objective, measurement based technique to identify financially weak vendors. VERPRO is a revolutionary approach that merges the knowledge of financial analysis into software engineering and elevates COTS acquisition management to another level.

BIBLIOGRAPHY:

Albert C, Brownsword L. 2002. Evolutionary Process for Integrating COTS based Systems: An Overview, SEI, Carnegie Mellon University.

Altman E. 2002. *Bankruptcy, Credit Risk, and High Yield Junk Bonds*, Blackwell Business.

Altman E. 1968. Financial Ratios, Discriminant Analysis and the Prediction of Corporate Bankruptcy. *Journal of Finance.*

Altman E, Haldeman R, Narayaman P. 1977. ZETA analysis: A New Model to Identify Bankruptcy Risk of Corporations. *Journal of Banking and Finance.*

Altman E. 1984. The Success of Business Failure Prediction Models. *Journal of Banking and Finance.*

Basili V, Boehm B. 2001 COTS-Based Systems Top 10 List. *IEEE Computer*, Vol. 34, No. 5, May 2001.

Bernhardsen E. 2001. *A Model of Bankruptcy Prediction*, University of Oslo.

Boehm B.1989. Tutorial: Software Risk Management. *IEEE Computer Society.*

Boehm B, Abts C.1999. COTS Integration: Plug and Pray. *IEEE Computer*, January 1999.

Boehm B. 1999. Managing Software Productivity and Reuse. *IEEE Computer*, September 1999.

Carney D, Oberndorf P. 1997. The Commandments of COTS: Still in Search of the Promised Land. *Crosstalk*, May, 1997.

Carney D. 1997. Assembling Large Systems from COTS Components: Opportunities, Cautions, and Complexities. SEI Monographs on Use of Commercial Software in Government Systems, Software Engineering Institute.

Carney D. 2000. COTS and Risks: Some Thought on How They Connect. SEI, Carnegie Mellon University.

Carney D, Morris E, Place P. 2003. Identifying Commercial Off-the-Shelf Product Risks: The COTS Usage Risk Evaluation. *SEI Technical Report,* CMU/SEI-2003-TR-023. 2003.

Carney D, Oberndorf P, Place P. 2003. A Basis for an Assembly Process for COTS-Based Systems (APCS). *SEI Technical Report,* CMU/SEI-2003-TR-010. Software Engineering Institute, Carnegie Mellon University. 2003.

Correia J. 2003. Market stalled in Global Economy's Slow Engine. *Gartner Dataquest Guide.* LE-19-4552, February 2003.

Crnkovic I. 2002. Component-based Software Engineering: Building Systems from Software Components. Malardalen University, Department of Computer Engineering.

Davidson E. 1999. Joint Application Design in Practice. *Journal of Systems and Software* 45(3): 215-223.

Down A, Coleman M, Absolon P. 1994. *Risk Management for Software Projects.* London, McGraw-Hill Book Company

Feblowitz M, Greenspan S. 1998. Scenario-Based Analysis of COTS Acquisition Impacts. *In Requirements Engineering*, 3(3/4), 1998, pp 182-201.

Forman E, Selly MA. 2002. *Decision by Objectives*, Expert Choice Inc.

Friedlob G, Schleifer L. 2003. *Essentials of Financial Analysis*, John Wiley & Son Inc.

Gilb T. 1988. *Principles of Software Engineering Management,* Wokingham, England: Addison-Wesley.

Goulao M, Abreu F. 2002. The Quest for Software Component Quality. *Information System Group*, Portugal.

Graham B, Dodd D, Cottle S, Murray R, Block F. 1988. *Security Analysis 5th Edition.* McGraw Hill.

Graham B, Meredith S. 1975. *The Interpretation of Financial Statement 3rd Edition*, Harper Business.

Graham C, Latimer N, Biscotti F, Correia J, Eschinger C, Pang C, Topolinski T. 2004. Software Market Research: Methodologies and Definitions. *Gartner Dataquest Guide*, SOFT-WW-GU-0011, January, 2004.

Humphrey W. 1989. *Managing the Software Process*. Addison-Wesley.

Keil M, Cule P, Lyytinen K, Schmidt R. 1998. A Framework for Identifying Software Project Risks. *Communication of the ACM.*

Kontio J. 1995. OTSO: A Systematic Process for Reusable Software Component Selection. University of Maryland, Technical report, December 1995.

Kontio J. 1996. A Case Study in Applying a Systematic Method for COTS Selection. Proceeding 18th ICSE, *IEEE Computer.*

Kontio J. 2001. *Software Engineering Risk Management: A Method, Improvement Framework, and Empirical Evaluation*, Helsinki University of Technology.

Kruchten P. 2000. *The Rational Unified Process: An Introduction, 2nd Edition.* New York, NY: Addison-Wesley Object Technology Series, March 2000.

Lawlis P, Mark K, Thomas D, Courtheyn T. 2001. A Formal Process for Evaluating COTS Software Products. *IEEE Computer*, Vol. 34, No. 5, May 2001.

Levinson M. 2001. Do Diligence. *CIO Magazine*, July 2001.

Maiden N, Ncube C. 1998. Acquiring COTS Software Selection Requirements. *IEEE Software* March/April 1998, pp 46-56.

MacCrimmon R. 1972. Proceedings of Multiple Criteria Decision Making. University of South Carolina, October 26-27, 1972.

McDermid J, Talbert N. 1998. The Cost of COTS. *IEEE Computer Society.*

Mizuno O, Kikuno T, Takagi Y, Sakamoto K. 1999. Characterization of Risky Projects based on Project Managers' Evaluation. Osaka University.

Morisio M, Seaman C, Basili V, Parra A, Kraft S, Condon S. 2001. COTS-based software development: Processes and open issues. *The Journal of Systems and Software.*

Ochs MA, Pfahl D, Chrobok-Diening G, Nothhelfer-Kolb B. 2000. A COTS Acquisition Process: Definition and Application Experience. *Proceedings of the 11th ESCOM Conference*, Shaker, Maastricht, 2000. pp.335-343

Ochs MA, Pfahl D, Chrobok-Diening G, Nothhelfer-Kolb B. 2001. A Method for Efficient Measurement-based COTS Assessment and Selection – Method Description and Evaluation Results. *Proceedings IEEE 7th International Software Metrics Symposium*, London, England, 4-6 April 2001. pp. 285-296.

Ohlson J. 1980. Financial ratios and the probabilistic prediction of bankruptcy. *Journal of Accounting Research.*

Pressman R. 2001. *Software Engineering: A Practitioner's Approach.* McGraw-Hill, New York.

Reifer D, Basili V, Boehm B, Clark B. 2002. Eight Lessons Learned during COTS-Based Systems Maintenance. *IEEE Computer Society.*

Saaty T. 1980. *The Analytic Hierarchy Process*. McGraw Hill, New York.

Sai V. 2004. COTS Acquisition Evaluation Process: Preacher's Practice. Technical Note CMU/SEI-2004-TN-001.

Shaffer G, McPherson G. 2002. FAA COTS Risk Management Guide: Practical Methods for Effective COTS Acquisition and Life Cycle Support" FAA.

Torchiano M. 2001. Selected Literature on COTS Products. *INCO Project Internal Report.*

Tran V, Liu DB 1997. A Procurement Centric Model for Engineering Component-based Software System. *IEEE Computer*, 1997.

Turner J. 1993. The Handbook of Project-based Management. *McGraw-Hill,* London.

Vitharana P. 2003. Risks and Challenges of Component-Based Software Development. *Communications of the ACM,* August 2003.

Voas J. 1998. The Challenges of Using COTS Software in Component-Based Development. *IEEE Computer*, June 1998.

Voas J. 1998. Certifying Off-The-Shelf Software Components. *IEEE Computer.*

Wallnau K, Carney D, Pollak B. 1998. How COTS Software Affects the Design of COTS-Intensive Systems. *SEI Interactive.*

Wallnau K, Hissam S, Seacord R. 2002. *Building Systems from Commercial Components*. New York, NY: Addison-Wesley.

Westfall L. 2000. Software Risk Management. *ASQC.*

Wilcox J. 1976. The Gambler's ruin approach to business risk. *Sloan Management Review.*

Yin R. 1994. *Case study research: Design and Methods*. Thousand Oaks, CA, Sage Publishing.

Yeoh HC, Miller J. 2004. COTS Acquisition Process: Incorporating Business Factors in COTS Vendor Evaluation Taxonomy. *10th IEEE International Software Metrics Symposium*, September 2004.