The Use of the Virtual Company for the Development of Mission Critical Software

By

Paul A. Soukup
B. Engineering (Mechanical)
McGill University, 1986

SUBMITTED TO THE SYSTEM DESIGN AND MANAGEMENT PROGRAM IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE IN ENGINEERING AND MANAGEMENT

AT THE

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

FEBRUARY 2000

© 2000 Paul A. Soukup. All rights reserved.

The author hereby grants to MIT permission to reproduce and to distribute publicly paper and electronic copies of this thesis document in whole or in part.

Signature of Author: ____________________________

Design and Management Program
December 15, 1999

Certified by: ____________________________

John Rockart
Thesis Supervisor
Senior Lecturer and Director, Center for Information Systems Research

Accepted by: ____________________________

Thomas A. Kochan
LFM/SDM Co-Director
George M. Birken Professor of Management

Accepted by: ____________________________

Paul A. Lagace
LFM/SDM Co-Director
Professor of Aeronautics & Astronautics and Engineering Systems
The Use of the Virtual Company for the Development of Mission Critical Software

By

Paul A. Soukup

Submitted to the System Design and Management Program on December 15, 1999 in Partial Fulfillment of the Requirements for the Degree of

Master of Science in Engineering and Management

ABSTRACT

With increasing competition, companies must shorten their product development cycle times, reduce costs, and increase flexibility, all the while maintaining world-class capabilities. An effective way of handling these issues is to focus on an outsourcing strategy and develop relationships with companies that have the required expertise. This thesis provides a conceptual framework for formulating and implementing a product development outsourcing strategy for mission critical software.

The research is based on field data and literature reviews, which were combined and used in a strategic context. Decision drivers and a logical sequence of steps were developed that a company can use to strengthen its chances of success in its outsourcing ventures. The thesis then takes the framework beyond traditional outsourcing and considers issues and methods for implementing a networked or virtual company. By networking to various software development companies, either domestically or abroad, a company can produce products at world-class levels. Various business models and examples of current virtual companies were assessed to determine critical attributes of a successful organization. Also, by decomposing the software product development process, various organizational architectures were generated and evaluated for flexibility, complexity, and interactivity.

The results of this research were applied to AlliedSignal's Defense and Space Systems division in order to develop an outsourcing strategy that will allow them to become more competitive. A set of decision drivers and trade-offs were used, which resulted in a plan that retains systems engineering, and outsources all other processes to partner companies on a risk/reward sharing basis.

Thesis Supervisor: John Rockart
Title: Senior Lecturer and Director, Center for Information Systems Research
Table of Contents

1 Introduction ..............................................................................................................................11
  1.1 Background ......................................................................................................................11
  1.2 Motivations ......................................................................................................................11
    1.2.1 Flexibility and time to market .....................................................................................13
    1.2.2 Demand Variability ......................................................................................................13
    1.2.3 Cost ..............................................................................................................................13
  1.3 Thesis Approach ...............................................................................................................14
  1.4 Thesis Organization .........................................................................................................15

2 Software Outsourcing .............................................................................................................16
  2.1 Background ......................................................................................................................16
  2.2 Issues ................................................................................................................................16
    2.2.1 Outsourcing Risks ........................................................................................................17
    2.2.2 Insourcing Risks ...........................................................................................................17
  2.3 To Outsource or Not to Outsource? ..................................................................................17
    2.3.1 Core Competencies ......................................................................................................18
    2.3.2 Proficiency ...................................................................................................................19
    2.3.3 Considerations and Risks ............................................................................................19
    2.3.4 Flexibility and Control ..................................................................................................20
  2.4 Types of Relationships ......................................................................................................21
    2.4.1 Contract Basis ..............................................................................................................21
    2.4.2 Risk/reward Sharing .....................................................................................................22
    2.4.3 Long term Strategic Partnership ..................................................................................22
    2.4.4 Selective Sourcing ......................................................................................................23
    2.4.5 Offshore .......................................................................................................................24
  2.5 Outsourcing Examples ......................................................................................................26
    2.5.1 Xerox ..........................................................................................................................26
    2.5.2 Wipro ........................................................................................................................26
    2.5.3 AlliedSignal Aerospace Canada (ASACa) ....................................................................27
    2.5.4 MCI .............................................................................................................................27

3 Outsourcing Case Studies ......................................................................................................29
  3.1 AlliedSignal Inc. ..............................................................................................................29
  3.2 Ford Motor Company ........................................................................................................30
3.3 Keane Inc.................................................................31
3.4 IT Solutions Inc ................................................................................................................33
3.5 Outsourcing Critical Factors .................................................................37
  3.5.1 Relationship Factors ..................................................................................37
  3.5.2 Type of Software to be Outsourced ........................................................................37
  3.5.3 Value and Cost Factors ..................................................................................38
  3.5.4 Success and Risk Factors ..................................................................................39
3.6 Outsourcing Conclusions ..................................................................................41

4 The Virtual Company ...........................................................................42
  4.1 The Virtual Company Defined ........................................................................42
  4.2 Virtual Vertical Integration ...........................................................................43
  4.3 Virtual Business Processes ...........................................................................44
  4.4 Architecture of the Company ........................................................................45
  4.5 Virtual Companies ........................................................................................47
    4.5.1 Nike ........................................................................................................47
    4.5.2 Verifone Inc. ........................................................................................48
    4.5.3 First Virtual Corporation ........................................................................48
    4.5.4 Dell Computer ........................................................................................48
    4.5.5 Nokia ........................................................................................................49
    4.5.6 IBM .........................................................................................................49
  4.6 Strategic Attributes ......................................................................................49
    4.6.1 Relationship Type ....................................................................................49
    4.6.2 Information Flow ......................................................................................50
    4.6.3 Components ...........................................................................................51
    4.6.4 Relationship Management ..........................................................................51
    4.6.5 Factors that Favor the Virtual Company Arrangement ...............................52
    4.6.6 Factors that Inhibit the Virtual Company Arrangement .............................52
  4.7 Virtual Interactions ........................................................................................52
  4.8 Business Models ..........................................................................................54
    4.8.1 Centralized Strategic Alliances ..................................................................56
    4.8.2 Network Strategic Alliances .......................................................................57
    4.8.3 Offshore Partners .....................................................................................58
    4.8.4 Online Communities ..................................................................................58
    4.8.5 Online Auctions .......................................................................................59
List of Figures

Figure 1: Synopsis of SEI CMM Levels ................................................................. 12
Figure 2: Outsourcing Decision Matrix ................................................................. 18
Figure 3: Needs versus Proficiencies Envelopes.................................................... 19
Figure 4: ITS's Communication Infrastructure ....................................................... 36
Figure 5: Vertical Production Stream of PC Manufacturers ................................... 43
Figure 5: System Decomposition Techniques ....................................................... 46
Figure 6: Disaggregation of the Value Chain and Production Chain .................... 55
Figure 7: Internet Communities of Interest (COIN) ............................................. 59
Figure 8: Business Models - Value Creation vs. Control ...................................... 60
Figure 9: Software Development Life Cycle .......................................................... 61
Figure 10: Value Creation Potential for a Life Cycle Process .................................. 62
Figure 11: Decision Matrices for D&SS ................................................................. 63
Figure 12: Decomposition Using Trees ............................................................... 65
Figure 13: Decomposition Using Layers .............................................................. 65
Figure 14: Strategic Partnership of Companies in the Virtual Company Supersystem .... 66
Figure 15: Community of Companies in the Virtual Company Supersystem ............ 66
Figure 16: Internet based IT Infrastructure ............................................................ 74
Figure 17: Strategic Forces in Tension in the Virtual Company ............................... 75

List of Tables

Table 1 : Outsourcing Decision Considerations ................................................... 20
Table 2: Global Software Professional's Salary .................................................... 24
Table 3: ITS Software Specifications ................................................................. 34
Table 4: Outsourcing Relationships ................................................................. 37
Table 5: Most Suitable Types of Software to be Outsourced............................... 38
Table 6: Value and Costs of Outsourcing ........................................................... 39
Table 7: Success and Risk Factors ................................................................. 40
Table 8: Summary of Critical Factors .............................................................. 41
Table 10: Critical Success Factors for D&SS ...................................................... 64
Table 11: Business Model Decision Drivers ..................................................... 68
Table 12: Partner Selection Criteria ............................................................... 70
1 Introduction

1.1 Background
In recent years, aerospace companies have been transforming themselves from large, slow moving organizations into agile, commercially oriented companies competing for a share of the private market. One such company is AlliedSignal. AlliedSignal's Defense and Space Systems division (D&SS) is moving from producing military and government aerospace products to mission critical commercial products. In order to be competitive with commercial companies, however, AlliedSignal's product development cycle time must be shortened, costs must be reduced, and the company must become more flexible.

An effective way of handling these issues is to focus on the core competencies of the company and outsource activities that can be performed by other specialized companies at a lower cost. One strength that differentiates AlliedSignal is its process-oriented systems engineering. Process-oriented software engineering is also a strength, but software development can be done equally effectively at substantially lower costs by other dedicated software houses. In order to remain competitive, the Defense and Space Systems division must develop a strategy that will give it access to the latest software techniques but still maintain the very high level of quality required by mission critical applications. The strategy must also increase the flexibility, cost effectiveness, and speed of development, if its products are to be commercially viable and reach the market in time.

This thesis explores AlliedSignal's options and develops a software development strategy by first examining software outsourcing and then secondly, examining the attributes and benefits of taking outsourcing to the next level by establishing a virtual company. This study attempts to determine what critical parameters and trade-offs must be considered, and if a virtual company is to be utilized, what structure and architecture should be implemented.

1.2 Motivations
Electronics used in aircraft and space vehicles, known as avionics, have been utilizing a rapidly growing component of software in recent decades. The growth in size has occurred in parallel with a growth in complexity, resulting in large cost overruns and software that has failed to meet all user requirements. The software is also very difficult to modify and maintain. Two initiatives have been used in order to deal with the increasing size and complexity of avionics software.
First, systems engineering has been given more presence in the development process. System engineering analyzes the user requirements and develops a high level architecture for the software and hardware. The systems engineer then decomposes the requirements, either functionally or via objects, with consideration given to interfaces and system testing, and passes on the tasks to the software developers. Complexity, and therefore, cost, is reduced at the software level. The resulting software is also easier to test, modify, and reuse since these issues have been considered at a high level.

Secondly, the increasing importance of process in system and software engineering has allowed developers to produce more robust code at lower costs. This is due to the enforcement of key practices including requirement traceability, peer reviews, and defect prevention. The maturity of the software process in an organization is measured by a capability maturity model (CMM) created by the Carnegie-Melon Software Engineering Institute. The CMM allows an organization to evaluate and improve the maturity of their software processes in terms of an evolutionary path from ad hoc, chaotic processes to mature, disciplined software processes. The CMM uses five levels to describe the maturity of the software processes, each level is described in terms of key practices that describe the activities needed to satisfy each level:

![Figure 1: Synopsis of SEI CMM Levels](image)

**Level 1 Initial:** The software process is ad hoc, and occasionally even chaotic.

**Level 2 Repeatable:** Basic project management processes are used.

**Level 3 Refined:** The software process for both management and engineering activities is documented, standardized, and integrated into a standard software process.
Level 4 Managed: Detailed metrics of the process and product quality are collected.
Level 5 Optimizing: Continuous process improvement is driven by quantitative feedback.

The software processes at the D&SS division are considered to be operating at a level 3, and the system processes are operating at possibly a level 2. This is a significant accomplishment for this division since most organizations operate at level 1.

These two initiatives are effective with dealing with the complexity and cost issues that are critical in the aerospace industry, however, in order to compete in the commercial marketplace, other issues become prominent and must be dealt with.

1.2.1 Flexibility and time to market
AlliedSignal must have access to the latest software techniques at all times. In addition to having access, the organization must quickly become proficient in its use if it is to be implemented in a new product so that it will be first to market. By developing relationships with vendors that have world-class capabilities, AlliedSignal can learn new techniques and processes.

1.2.2 Demand Variability
As AlliedSignal is gaining a foothold in the commercial market, the demand for software development may not be constant over time. Product lines will hire many software developers as the project ramps up, and when the product ramps down, if another project is not coming on line, developers will have to be let off. Vendors can be used to absorb the peak demands, while the client maintains a steady level of developers.

1.2.3 Cost
Costs have been reduced substantially with the emphasis on systems engineering and software processes. However, AlliedSignal carries an expensive aerospace infrastructure to support manufacturing, shipping, human resources, etc. There is a large overhead, therefore, that must be built into the cost of each product. By sending development outside of the company, the overhead costs can be reduced substantially.

This thesis develops a strategy that addresses these issues by exploring and evaluating the use of a virtual company to allow AlliedSignal’s D&SS division to successfully compete in the commercial market place.
1.3 Thesis Approach

A four-step approach is used to develop the strategy to evaluate the multitude of options available to D&SS:

1. Study the software outsourcing industry to understand what should be outsourced and understand the characteristics and forces that effect the industry.
2. Develop several case studies from surveys and interviews to determine key factors that effect the outcome of outsourcing events.
3. Study current examples of virtual companies in various industries to determine their positive and negative attributes and produce decision drivers for implementation.
4. Apply the results of the outsourcing and virtual company studies to D&SS to generate trade-offs and formulate an overall strategy for commercial software development.

To study the different aspects of outsourcing, this thesis uses a case-study method combined with conclusions from literature. The cases are developed from surveys given to experts in the outsourcing field and supplemented by literature reviews. The survey is included in Appendix A and the summarized results of the surveys are included in Appendix B. Examples are used throughout the thesis where insufficient material was gathered to complete a case, but the material would bring additional insight into aspects of outsourcing that are useful to this study. The intent was to cover a wide spectrum of outsourcing experiences. The results of the interviews are compiled to determine keys of success of an outsourcing relationship, and critical factors to determine what type of relationship should be initiated.

Unfortunately, people experienced with virtual companies were not available to interview, so material on virtual companies is exclusively generated from literature reviews. Current examples of virtual companies are examined and attributes for success are collected. Various business models are also studied and trade-offs created. System architecture techniques are applied to determine the best structure for the virtual organization. The research data is used to develop a set of decision drivers to evaluate what to outsource and what kind of business model to establish.
1.4 Thesis Organization

Chapter 2 documents the outsourcing industry by studying the criteria to outsource, the risks, and the types of outsourcing relationships. Various examples are used to illustrate the material.

Chapter 3 uses the results of surveys and interviews with outsourcing experts to develop four case studies. This chapter uses the case material to develop overall critical factors for an outsourcing relationship.

Chapter 4 explores the characteristics of virtual companies in a wide variety of industries. Positive and negative attributes are generated and then applied to several potential business models and system architecture principles are applied to generate a structure for a virtual company.

Chapter 5 applies the learnings of the previous two chapters to AlliedSignal's D&SS division to evaluate the best business models and sourcing relationships to make the division commercially viable in today's competitive software market.

Chapter 6 offers conclusions and reflections on the material presented in this thesis.
2 Software Outsourcing

AlliedSignal's strategic options range from using a limited amount of outsourcing to creating a virtual company. Both extremes entail the use of outsourcing relationships since they are the building blocks of a virtual company. Therefore, an understanding of the characteristics and risks of outsourcing is essential.

2.1 Background

In industry today, software outsourcing is the largest component of a rapidly growing trend of outsourcing information technology (IT). The outsourcing IT movement is considered to have been initiated by the landmark Kodak Eastman deal, which in 1989 outsourced most of its IT services. In the U.S., spending on IT outsourcing has exploded from $7.2 billion in 1990 to $35 billion in 1995, with application development making up 28% of the overall expenditure. According to the results of a survey performed by the Outsourcing Institute, the top five strategic reasons that companies outsource are to:

1. Improve business focus
2. Gain access to world-class capabilities
3. Accelerate reengineering benefits
4. Share risks
5. Redirect IS resources to more strategic activities

And the top five tactical reasons that companies outsource:

1. Reduce or control operating costs
2. Make capital funds available
3. Generate a cash infusion
4. Secure resources not available internally
5. Gain control of difficult or out of control functions

The factors driving the movement towards outsourcing are the pressures from global competition, skill shortage, the need for timely information, and the necessity to reengineer business processes.

2.2 Issues

Many of the issues of outsourcing application software for IT departments are applicable to outsourcing real time software for mission critical applications. In both cases there must be a clear understanding of the company's core competencies and business goals.
2.2.1 Outsourcing Risks
Some of the main issues organizations have with outsourcing are loss of control, hidden costs, declining service levels, and the effect on internal staff. Michael Earl has studied outsourcing and the risks involved, the following are the eight most significant:

1. **Possibility of weak management** – IT managers may not be able to manage an external vendor.
2. **Inexperienced staffs** – experienced IT professional may join the vendor company or leave altogether.
3. **Business uncertainty** – if the company's future needs are not clear, outsourcing arrangements may be restraining and flexibility may have to be paid for.
4. **Outdated technology skills** – it is difficult to ensure that the vendor's skill stay current.
5. **Endemic uncertainty within projects** - problem identification and conflict resolution can be difficult and expensive to build into a contract.
6. **Hidden costs** – companies tend to underestimate setup and management costs.
7. **Lack of organizational learning** – there is a risk that tasks, which are outsourced, may become strategic in the future.
8. **Technological indivisibility** – problems can occur at the interface between the vendor and the client.

2.2.2 Insourcing Risks
However, there are also risks and costs to keeping work in house. Outsourcing can be beneficial by pushing development or capital risks onto the suppliers. Also, outsourcing can extend a company's technology range so that the company is not limited to its own technical and innovative capabilities. World-class vendors can provide new product and process ideas that the client could not have developed on its own. According to Von Hippel, two-thirds of all innovation occurs at the customer-supplier interface, therefore, with good management, and through the use of multiple vendors, the potential for innovation is great.

2.3 To Outsource or Not to Outsource?
Two primary parameters are used to determine whether to outsource or retain the service or task in house: (1) the business or strategic value of the technology, and (2) the operational performance, proficiency, or competency of the company in performing the task.
Where the strategic value and performance of the task are low, the task should be outsourced, since this provides an opportunity to increase quality and cost effectiveness, with minimal risk (quadrant III in figure 2). Conversely, if the strategic value and performance are high, then the task should be kept in house (quadrant II) since outsourcing would provide minimal benefits but would increase risk. For the other two situations (quadrants I and IV), other parameters must be considered, such as flexibility, risks, resources, technology, or culture. For example, if value is low and performance is high (quadrant IV), and if the technology or the product will not change in the foreseeable future, then the service can be kept in house. In the opposite situation, where business value is high and performance is low (quadrant I), if the resources exist and an investment can be made, then the service should be kept in house and the performance upgraded, thereby moving the task into quadrant II.

![Outsourcing Decision Matrix](image)

Figure 2: Outsourcing Decision Matrix

The firm must determine where a particular task is positioned. First, the business value of the task must be determined - is this task a core competency? Second, the performance or proficiency of the company must be objectively determined.

### 2.3.1 Core Competencies

According to Quinn, core competencies are skills, not products or functions, which are uniquely important to the customer in the long term. Often a core competency will take advantage of a knowledge gap and the company will dominate in that area. Sometimes a core competency will be a skill that must be kept in-house in order to block competitors or to avoid giving sensitive knowledge to suppliers. The organization will only have two or three skills and they will be embedded in the company. To attempt to have many skills will result in
not being world-class in any of the skills and thereby lose competitive advantage. It is essential that the company be superior to all companies for the selected core competencies if it is to maintain competitive advantage.\(^5\)

### 2.3.2 Proficiency

The company can determine its capabilities or proficiencies by benchmarking themselves against the competition or other companies performing similar tasks. The company may find that their capabilities are significantly below other companies, but the level of proficiency may be sufficient in the current marketplace. If not, this activity may be worth the investment in time, money, and resources to be brought up to higher standards, otherwise, the activity should be outsourced to a world-class vendor.

Figure 3 illustrates how core competencies fall in the overlap between business needs and the proficiencies of the company.

![Figure 3: Needs versus Proficiencies Envelopes](Source: Forrester Research Inc.\(^9\))

### 2.3.3 Considerations and Risks

However, the decision to outsource is not as simple as studying the business needs versus the proficiencies of the company. In order to determine whether to outsource, the risks mentioned in the previous section must be weighed against the return. For example, most services should be kept in house if the executives are risk-averse, if major risks cannot be mitigated, or if strategic issues are more important than cost cutting. In those cases, only the most commodity-like services would be outsourced. The task should also be kept in house if the task is small, since it may not get the appropriate attention from a vendor. Conversely, if
the product is modular, so that the task can be broken out cleanly, or if it is not highly integrated in the organization, then the task should be outsourced. Considerations and risks, from the client’s perspective, are summarized in Table 1.

<table>
<thead>
<tr>
<th>Consideration</th>
<th>Keep In House</th>
<th>Outsource</th>
</tr>
</thead>
<tbody>
<tr>
<td>Management</td>
<td>• If strategic issues are critical</td>
<td>• Free to focus on core tasks</td>
</tr>
<tr>
<td></td>
<td>• If they are risk averse</td>
<td>• May lose cross functional benefits</td>
</tr>
<tr>
<td></td>
<td>• If the software is poorly defined</td>
<td>• May lose control</td>
</tr>
<tr>
<td></td>
<td>• If the task is small</td>
<td></td>
</tr>
<tr>
<td>Organization</td>
<td>• If software development is highly integrated in</td>
<td>• If software development is stand-alone wrt cost</td>
</tr>
<tr>
<td></td>
<td>the organization</td>
<td>and control</td>
</tr>
<tr>
<td></td>
<td>• If cultural barriers exist</td>
<td>• If experienced with restructuring</td>
</tr>
<tr>
<td>Vendors</td>
<td>• Are technically weak</td>
<td>• Are world-class</td>
</tr>
<tr>
<td></td>
<td>• Are monopolistic</td>
<td>• Possess unique capabilities or infrastructure</td>
</tr>
<tr>
<td></td>
<td>• Are potential competitors</td>
<td></td>
</tr>
<tr>
<td>Product</td>
<td>• Complex/customized products</td>
<td>• Simple/commodity products</td>
</tr>
<tr>
<td></td>
<td>• Integral products</td>
<td>• Modular products</td>
</tr>
<tr>
<td></td>
<td>• Short life cycle</td>
<td>• Long life cycle</td>
</tr>
<tr>
<td>Costs</td>
<td>• R&amp;D investment may be needed</td>
<td>• Lower costs due to less overhead and economies</td>
</tr>
<tr>
<td></td>
<td>• Personnel development costs</td>
<td>of scale</td>
</tr>
<tr>
<td></td>
<td>• Infrastructure investments</td>
<td>• Management costs may climb</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Hidden costs may appear</td>
</tr>
<tr>
<td>Expertise</td>
<td>• If possess world-class expertise</td>
<td>• If skills are obsolete</td>
</tr>
<tr>
<td></td>
<td>• May fall behind in technology</td>
<td>• Outside technology range</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• May lose critical skills</td>
</tr>
<tr>
<td>Competitive Edge</td>
<td>• Core competencies</td>
<td>• Support &amp; peripheral tasks</td>
</tr>
</tbody>
</table>

Table 1: Outsourcing Decision Considerations

2.3.4 Flexibility and Control
Once an outsourcing relationship has been identified, a balance must be established between flexibility and incentives for the vendor versus control and security for the client.

2.3.4.1 Flexibility
Flexibility is needed because technology, economic conditions, and services can change over the duration of a contract. The volatility of the environment and the amount of knowledge and innovation that flows between the two parties dictate the degree of flexibility needed in the relationship. This in turn is driven by factors such as complexity, volatility, and the amount of customizations involved with the product.

One difficulty customers have in optimizing flexibility is that vendors make it expensive to
sign short-term contracts. Therefore, managers often end up signing five or ten year contracts without being able to predict the changes in business and technology. One proposal is to develop the outsourcing strategy primarily around flexibility and control versus based purely on competencies and proficiencies. To maximize flexibility and control, multiple vendors, including the customer's internal departments, should compete for pieces of the project. By having multiple vendors, using short-term contracts, working on various pieces, the customer is able to easily switch vendors as business and technology conditions change.

2.3.4.2 Control
Giving control away is always a concern for a client, but may be overemphasized. Most companies already contract out services, such as telecommunications (i.e. to AT&T), that could potentially cause major disruptions if the vendor has a failure. The degree of control should be driven by the potential competitive advantage that could be derived from this task and the long-term vulnerability that could arise if there is a failure. For example, a loosely controlled vendor would be appropriate for a support task with little potential for competitive advantage and could temporarily leave the company vulnerable if there were an interruption in service. For a situation where the task could provide a competitive advantage and failure would be critical to the client, tight control is needed, if even outsourced at all.

2.3.4.3 Degree of Outsourcing
If indications show that outsourcing is a probable path, then the degree of outsourcing and type of relationship can be used to mitigate some outsourcing risks, allowing management to focus on the specific benefits. The degree of sourcing is driven by the level of vendor expertise and strategic issues such as the potential for a competitive edge. If too much is outsourced, the vendor may take control. Also, the vendor is now entering the client's vertical market, so they could become competitors.

2.4 Types of Relationships
The type of relationship is driven by the amount of flexibility and control required by the client and the vendor. There are many types of software outsourcing relationships that have been used, the major ones are summarized in the following sections.

2.4.1 Contract Basis
Outsourcing on a contract basis is primarily focused on cost savings. The relationship is usually based on a loose, hands-off approach. The simplest version of this relationship is the purchase of shrink wrapped or commercial off-the-shelf software from a software
manufacturer such as Microsoft or Lotus. A more sophisticated example of this relationship would be the outsourcing of desktop PC maintenance and help desk. This could be managed loosely with cost evaluations every six months to ensure that the vendor is performing at an acceptable level. These outsourcing relationships are usually based on one of the following two types of contracts:

- **time and material contracts**: the client pays the vendor on a monthly basis based on size and composition of the vendor's team. This option gives the client flexibility to balance the amount of time with projected workloads.

- **fixed fee contracts**: is a good way to shift the development risk to the vendor, but is only appropriate when the requirements are well defined.

### 2.4.2 Risk/reward Sharing

A variation of the contract relationship is one where the client and vendor share the risks and rewards. The contract is set up so that the vendor will take more responsibility for the software development, but will also take a share of the subsequent revenues. Risk/reward sharing has grown in popularity since the extra incentives for the vendor have improved their performance and successfully aligned their objectives with the client's. Also, the client enjoys a reduction in technology risk exposure. The vendor's incentive is the opportunity to participate in a potentially lucrative relationship.

### 2.4.3 Long term Strategic Partnership

Recently, risk/reward sharing relationships have developed into long term strategic partnerships. This has occurred in cases where the outsourcing is motivated by a higher strategic need such as redefining a portion of the business or where a high degree of adaptability is required. Other reasons include situations where one party is going to make a substantial investment as part of the contract, or if the nature of the work extends beyond a normal contract. The very tight customer/vendor relationships of Just-In-Time (JIT), increased competition, and higher demands by customers for outsourcing services have driven this trend. Partners offer a wider range of services than the narrowly defined tasks performed on a contract basis, and since there is more trust between the two parties, the vendors can carry out tasks that are much closer to core business functions.

The terms of the partnerships are usually long, but only as long as the partnership brings value, since eventually changes in technology and organizations will demand new alliances over time. Therefore, in order to survive as long as possible, flexibility must be built into the relationship so that it can evolve with business conditions. The vendor is aligned by
developing appropriate incentives and penalties, which are measured by tools that can objectively measure a vendor’s performance and facilitate communication between the two parties. Initially, the specific objectives of the relationship should be identified and set. The relationship is conducted on a win/win basis so that the customer and vendor operate as an integrated supply chain. In one case, a grade of less than 80% results in a warning that must be corrected in the next quarter, otherwise a disqualification process is initiated. 10 According to Dr. Robert Kepper, contracts are important in partnerships, but play a different role. The contract dictates the objectives, direction, and termination strategy of the partnership. It should also have the means to allow the relationship and requirements to change over time. 8

Ideally, partnerships are developed to handle strategic issues, such as integrating e-commerce technology, rather than tactical issues, such as cost savings. Often the customer will absorb new skills from the vendor along with new technology. Therefore, performance is typically measured based on new skills and revenue increases. 9 In some cases, clients tie in gain-sharing incentives to performance standards. 10

2.4.4 Selective Sourcing
Customers over time have come to the realization that they have become dependent on one outsource vendor and have lost a substantial degree of leverage needed to maintain control. Also, customers have concerns over how a supplier will perform over the length of the contract (possibly up to 10 years). Often vendors build in high switching costs in the contract in order to dissuade a firm from dropping the contract. An alternative is to revert back to insourcing the task, but it can be very difficult to insource again, since much of the technical competence has been absorbed by the vendor or disappeared. One solution that companies have implemented is outsourcing to a group of vendors. This solution eases the transition from one vendor to another if one under-performs and it also allows a customer to select the most suitable vendor for each specific task.

However, as a company should not have too many core competencies, they should also try and minimize the number of close vendors. For example, Japanese companies build on only a few, but close, interdependent relationships. 5 Managing many relationships would defeat the purpose of outsourcing since administrating the vendors would overwhelm the client’s management who are supposed to be free to focus on value adding activities.

British Petroleum (BP) experimented with selective outsourcing in order to escape escalating
fees and inflexible services that resulted from a single vendor. BP also believed that a single supplier could not excel in all areas, and therefore BP would not be able to take advantage of high quality technologies and services offered by other vendors. However, BP found that the separate contracts required far more management than it was worth. The primary weakness was that the vendors did not cooperate with each other, and BP was forced to manage the interfaces between them. This drove BP to develop an innovative way to motivate a new set of suppliers to cooperate. BP selected three vendors and made each one of them responsible for a specific domain and then closely measured their performance against industry standards. This arrangement enforced a mutual reliance on each other that ensured cooperation and coordination.

2.4.5 Offshore
Any of the above mentioned relationship types can be used with offshore vendors. The primary advantage of going offshore is the substantial cost reduction, possibly up to 50%, that can be achieved. Typical hourly onshore costs are about $70 to $125. In comparison, typical offshore costs are about $18 to $25 per hour. A second factor is the ability to speed up development by having 16 to 24 hours of programming per day without paying overtime. When U.S. workers finish for the day, they hand off the code to overseas developers who are just starting their day. A third factor is the availability of skilled labor. It is becoming increasingly difficult to find and keep professional talent in the US while there is a proliferation of well-educated computer specialists in Eastern Europe and India. While India has the most vendors, Eastern Europe, Ireland, Israel, Mexico, South America, China, Russia, and the Philippines also offer software vendors.

<table>
<thead>
<tr>
<th>Country</th>
<th>Number of Software Professionals</th>
<th>Average Annual Salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>China</td>
<td>1.3 Million</td>
<td>$7,150</td>
</tr>
<tr>
<td>India</td>
<td>947,100</td>
<td>$8,200</td>
</tr>
<tr>
<td>Russia</td>
<td>1.1 Million</td>
<td>$12,000</td>
</tr>
<tr>
<td>Brazil</td>
<td>665,663</td>
<td>$48,000</td>
</tr>
<tr>
<td>U.S.A.</td>
<td>2.46 Million</td>
<td>$55,000</td>
</tr>
</tbody>
</table>

Table 2: Global Software Professional’s Salary
(Source: Software Productivity Research Inc., 1996)
One of the biggest disadvantages of going offshore is the fear of losing intellectual property. Many foreign companies will sign non-disclosure agreements and negotiate copyright arrangements, however, clients still feel insecure without the US legal system to fall back on. Also, the large distances can make offshore development difficult, especially for projects that involve a substantial amount of day to day interaction or have complex specifications.

When selecting an offshore country, the country must have access to the following:

- Significantly lower cost software professionals
- A high level of expertise in the right types of software
- Use of English among software developers
- High education standards and availability of skilled software professionals
- Availability of advanced technology and technical training and support
- Good, high speed data transmissions and easy access to the Internet
- Good infrastructure facilities
- Stable political and economic environments
- Strong legal structure

A summary of recommendations from IS managers who have outsourced overseas:\(^2\)

- A methodological approach to project and change management
- A vendor supplied project manager at the customer site to overcome cultural and communication barriers with offshore developers
- A network manager at the vendor site to coordinate the communication systems
- An agreement on documentation on how to maintain the software

There are two models that can be used for offshore outsourcing:

- **Direct:** US companies can work directly with offshore companies. The offshore company will have a minimal presence on the customer site and most software development will occur offshore. The direct model leads to the lowest software development costs, but it is more difficult and costly to manage.

- **Indirect:** US companies can work through a US vendor that has offshore development centers. In this case, software development can take place either on the customers site, the vendor’s US development center, or offshore depending on the cost constraints and the complexity of the project. The advantage of this model is that the customer does not have to manage the issues of distance, language, and culture.

The two models have blurred together in cases where the offshore company has set up offices and software development facilities in the US, thereby reducing the complications due to distance. Issues of culture, language, and legalities still exist, however.
Chapter 2

2.5 Outsourcing Examples

In order to illustrate the various forms of outsourcing, four examples have been developed based on interviews and literature reviews.

2.5.1 Xerox

Xerox uses software outsourcing extensively, even running in house workshops on how to manage vendors. Jeff Carter of Xerox is currently managing the software development of a sales force automation tool. In his case, a three tier approach to outsourcing is used with onshore vendors to develop the software:

1. **Partners**: vendors that are co-located and have day to day interaction to develop highly customized products (time and material contract).
2. **Professional Services**: are vendors that are used on a one time implementation basis, at a distance with a small amount of customization (fixed price contract).
3. **Commercial off the shelf**: for “shrink wrapped” products that are mass produced by a vendor with no customization (fixed price).

Xerox also employs offshore outsourcing, with software development centers in the UK, Delhi, Singapore, and Brazil. The offshore centers are only staffed by Xerox employees who do three month assignments in the US to become educated on Xerox processes and problem solving techniques. Xerox employees are used offshore in order to control intellectual property and control the migration of knowledge. Other lessons learned from Carter are:

- The product’s end owner should be involved in the project management and system engineering.
- A minimum of one full time manager is needed to manage each partner.
- The systems engineering should always be kept in house.
- The number of vendors should be kept to a minimum
- The product architecture should be used to partition software tasks.

2.5.2 Wipro

In 1991, Wipro, a large Indian software development company, started developing software for GE out of a dedicated development center in Bangalore, the Silicon Valley of India. The center is a virtual extension of GE’s information management support operations and provided GE with software development, maintenance, and re-engineering. The long term
relationship model established with GE was then successfully applied to dozens of other US companies for a wide range of applications. In 1992, Wipro took the aggressive step of establishing offices in the US which, by 1998, grew to 14 offices and 700 software professionals, with 2,300 professionals in India supporting them.

Wipro has become known for its software development expertise, quality, and accelerated delivery. The quality reputation is based on its ISO 9001 and SEI level 4 certification, which are essential for mission critical software projects. As Mr. Vivek Chopra, Chief Executive for US operations stated, “We are now one of a handful of vendors that top tier companies will trust due to the mission-critical nature of the jobs we do for them and the fact that they must turn over key intellectual property. As a result, they rely on us for everything from high-end custom software applications to the smallest application-specific integrated circuit.” Wipro has also become know for its ability to offer a large pool of skilled resources in many different technology platforms.

2.5.3  AlliedSignal Aerospace Canada (ASACa)
Since December 1996, ASACa has been outsourcing aircraft embedded systems testing to an offshore company, Wipro Systems Inc. of India. Wipro Systems employed approximately 10 people both in Canada and India. However, for the first project, all personnel were working in Canada as the Wipro team developed experience and a communication infrastructure was established. Subsequent projects were partially performed in India that utilized a 64Kb dedicated satellite link to access the computer network at ASACa. By sharing the same network, duplicate development environments were not needed and work products were available to ASACa engineers at all times.

Wipro was subcontracted on a fixed price basis with a statement of work and a technical statement of work initially submitted to Wipro. Wipro then responded with a Software Development Plan (SDP). Performance was monitored with biweekly meetings. Management allocated a maximum of 20 hours per week to interface with Wipro, and both ASACa and Wipro established single point of contact. Email was the primary method of communication. ASACa is considering using Wipro Systems for future projects, including embedded maintenance programs and support environment tools/applications.

2.5.4  MCI
Richard Leibhaber, chief strategy and technology officer at MCI, stated that MCI has extensive relationships with small companies, which do about 40% of the software
MCI internally performs all the design, specification, process control, process rating, operational procedures, and system testing. The remainder of the software development tasks are outsourced to a combination of vendors.
3 Outsourcing Case Studies

Two clients and two vendors will be studied as cases. The data for the cases was drawn from interviews with managers involved with outsourcing, and the results of surveys they completed. The survey consisted of two sections, the first part asked specific questions about outsourcing projects that the subject has worked on. The second section of the survey pertained to general questions on managing outsourcing relationships.

3.1 AlliedSignal Inc.

AlliedSignal Defense and Space Systems (D&SS) division focuses on the development and manufacturing of airborne electronics, primarily on launch vehicles and military aircraft. The information technology (IT) group as been experimenting with outsourcing a few select functions. The director of the IT group, Craig Halterman, has had experience with outsourcing in other companies, albeit, with mixed results. In one case, an alliance agreement was formed between Dow Chemicals and a major IT outsourcing firm, which resulted in a failure. And very recently, a Y2K appraisal was outsourced to an Indian company, which resulted in dissatisfaction on both sides. The problem was one of expectations, the Indian company thought they were conducting an initial scoping for a larger job, while D&SS just wanted an appraisal of the company’s capabilities.

Halterman is skeptical about the cost savings that outsourcing can provide. He cites the extra costs that are often underestimated or overlooked. One cost in particular that has hurt him in the past are hidden costs or costs that are outside the contract, such as overtime, installation and maintenance of non-standard software, and other unexpected services. A second unexpected cost is the costs involved with testing and acceptance of the software. Also, two negative side effects occur when some IT functions have been outsourced. The first is retention of the best IT personnel is difficult since they perceive their jobs and job security as eroding. The second is the creation of “shadow organizations.” For example, an employee will write his or her own software instead of waiting for the vendor to write the software, due to time or hassle.

Halterman is also concerned with the cultural impact of bringing in an IT vendor. Barriers can be created between the IT staff and the vendors due to attitudes about outsourcing and the impact on job security. Declining job security also reduces employee satisfaction and
employee moral.

Halterman advocates ensuring that the vendor uses a dedicated program manager to interface with the client or the use of an Integrated Product Team (IPT) to communicate functionality and issues. Halterman concludes: "outsourcing is difficult to be cost effective, the contract must be written with extreme care in order to avoid hidden costs."

3.2 Ford Motor Company
To develop automobiles, Ford Motor Company uses an extensive array of design and manufacturing software tools, such as computer aided design (CAD) and computer aided manufacturing (CAM). One system that ties the data of many of the engineering tools together is the Project Information Management System (PIM). PIM manages data for CAD, assembly, and manufacturing. In 1995 Ford initiated the development of PIM using off-the-shelf products and outsourcing the software that was not available or needed to be customized. One off-the-shelf product was Metaphase, a core component of the PIM system. Customizations to the Metaphase PIM environment included WEB interface, Orbix interconnect, C method code, MoDEL data model, and scripts (MetaScript). The final product would be highly complex, "bleeding edge" technology, based on an Oracle client/server architecture. The final software would have approximately one million lines of custom code written for all Metaphase customizations for a wide variety of customer applications.

Ford could have trained their employees to develop and use Metaphase, however, management realized that employees would probably leave for more lucrative jobs after their training. Ford thus decided to outsource the Metaphase implementation to a US company specializing in Metaphase. In the end, Ford outsourced the overall development to three different companies based in Minneapolis, Cincinnati, and India. The two US based companies were chosen because of their specialization. The Indian company was chosen because it was talented, low cost, and had established a good reputation with another large US company. The Indian team was initially brought over to the US because of poor infrastructure and very detailed requirements and, eventually, one person remained in the US and six worked in India.
Functional Specifications were developed by the planning organization using a common format used by all project teams. Code reviews were performed by internal Ford personnel, and a separate Quality Assurance group (also internal) tested the functionality. Bugs were tracked in a database but no formal tracking mechanism was used for code reviews. If an error was found, the planner involved raised any issues back to the customer contact to discuss work-arounds and other issues with the specified functionality as development progressed.

Ford has a long term outsourcing strategy that includes plans to leverage internal resources on planning and overall project management, and utilize an outsource partner for development. For the PIMS project, Ford had a contractual relationship with agreed metrics and long term (multi-year) commitments. Program managers face off from both companies, with project leads from the vendor reporting to Ford supervision. For Mark Anders, supervisor of development and implementation, it is essential that the client's process mesh with the outsource activity, but the outsource vendor does not have to utilize the same client process in their own internal shop. Direct communication links and access to system resources will often play an important role in effective outsourcing (vendor accessing client equipment). The best forms of direct communication are: an on-site presence by the vendor, video-conferencing, or teleconferencing. For Anders, the keys to reducing risk in an outsourcing relationship is ensuring on site presence of the vendor and closely monitoring quality and costs.

3.3 Keane Inc.

Keane Inc. is a software services outsourcing company that uses information technology to help large companies achieve their business objectives. Keane focuses on application outsourcing, software development, project management, year 2000 compliance, and help desk services. Over the last 30 years, Keane has been a vendor of outsourcing services to hundreds of companies, using an internally developed management process to ensure customer satisfaction.\(^2\) In fact, Keane has achieved SEI Level 3 in software development process. This process is flexible depending on the customer and the potential for change that may occur during the project.

Keane has a highly structured approach to supplying outsourcing services, which uses
detailed information on the software, users, and service levels to define the requirements of
the software. Software projects range from one single application area to an entire
business’s applications portfolio. In addition, expectations are established with specific
metrics and monthly reports and meetings to measure performance against these metrics.
Quantitative metrics include response, repair and resolution times, number and type of
activities, and number and type of delivered products. Qualitative metrics include client
satisfaction. Over the duration of the project quarterly reviews are used to evaluate trends,
raise and resolve issues, and reset standards as needed. Other program management tools
include a communications strategy and a balanced scorecard.

Quality of the software is measured by establishing quality standards up front and then
testing to them. Generally software must pass several levels and types of testing including
user, unit, and systems tests as well as meeting established quality requirements (i.e.
documentation). When the software falls short of requirements, the product is reworked till
the product passes. Depending on the agreement, the work may be done at no cost to the
client.

According to Keane’s model the key to a successful venture is ensuring that there is mutual
reliance between the two parties. As Barbara Wentzel of Keane says, “It is critical that both
organizations have skin in the game”. Keane feels that having multiple suppliers leads to
poor relationships. The contract between the two companies is long and complex and it must
outline expectations and dissolution strategies. In terms of the organizational structure
between the customer and the vendor, the client should have a vendor liaison with enough
power to set priorities, evaluate performance, and clear roadblocks. The client’s processes
must be in sync. As Wentzel states, “vendor/client processes should be seamlessly
integrated and you also need a cultural match”. On the other hand, the vendor’s process
should ensure that the vendor integrates their best practices with the clients. The nature of
the communication between the client and the vendor is critical. Regardless of the size and
scope of the relationship, each organization should have an individual who is responsible to
oversee the engagement. Communications networks grow as the size and complexity of the
engagement grows. Keane’s methods of communication between the client and the vendor
are; periodic weekly/monthly status reporting, periodic monthly/quarterly metrics reporting,
periodic weekly/monthly and quarterly meetings, and quarterly/annual engagement reviews.
Wentzel concludes, “Pick the right vendor and clearly define expectations.”
3.4 IT Solutions Inc.

IT Solutions, Inc. (ITS) is a global software company engaged in providing high quality software development and project management services with revenues approximately $45-50 million each year. INC magazine ranked ITS as the 31st fastest growing company in the U.S. in 1998. With offices across the US and in the UK, and software centers in Bangalore and Chennai, India, ITS is positioned to deliver quality and cost effective software services to its clients. ITS currently employs about 600 people (250 in US, 300 in India, and 50 in UK). The client base includes AlliedSignal, Bank of America, Ingersoll-Rand, Lexis-Nexis, Netscape, Oracle, Pacific Bell, People Soft, Providian Bancorp, Silicon Graphics, Synopsys, and Wells Fargo.

ITS uses the indirect model for offshore outsourcing. In this way, ITS can offer the best of both worlds to its customers. On one hand, it has an extensive group of personnel in North America who can interface with the customer directly. And on the other hand, ITS can offer its customer very cost effective software by developing it in India. ITS takes care of all the global, legal, multi cultural issues that occur with foreign development teams and reduces the "emotional risk". In this way, US companies can further reduce costs by minimizing the management overhead needed to coordinate the offshore outsourcing tasks. The main benefit, as Jim Caruso of ITS states, is "communication by having a strong, accountable local person". ITS feels that the direct model is only appropriate for commodity software.

ITS is a Project Delivery organization. The type of software outsourced to ITS are in the areas including Web-based Solutions, Commercial Applications, System Software, ERP Solutions, Telecommunications, Tools and Utilities. Clientele include Information Technology Departments, Hardware Manufacturers, Product Suppliers, System Integrators and Publishing and Information Management companies. The size of the software handled depends on the project being executed and the platform. The size varies from a few thousand lines to over a million lines of code.

Depending on the type or nature of the project, the requirements could be specified by the customer in any of the following ways:
Table 3: ITS Software Specifications

In a number of cases, ITS has assisted the customer in documenting the requirements. These requirements are signed-off by the customer before proceeding to the next stage of development.

Program management is considered to be one of ITS's primary strengths. ITS conducts quarterly/half-yearly reviews with the customer and metrics reports at the program level. Metrics that are used to track the outsourcing performance include: on-time index, where projects are measured against schedule; review efficiency, number of defects found by ITS/(Number of defects found by ITS + customer); and effort overrun. The quality of the software is measured based on the number of defects reported by the customer (See Review efficiency) through testing. Testing is carried out at various levels including unit testing, done by the programmer, system/regression testing, done by the test team, field testing, done by ITS at customer location, and acceptance testing, done by the customer. When software is determined to fall short of established requirements, ITS performs a corrective action, to resolve the problem, and a preventive action, to analyze the error so that it does not repeat in the future. The corrective action could be at the customer site or an offshore location.

ITS has plans to measure the software quality based on an Error Prediction Model. The Model will predict the number of errors that can be introduced at the beginning of the project.
From the number of defects detected during the various phases of project and the number which was predicted, the quality of the software can be computed.

The relationship between ITS and its customers is that of a long-term partnership. ITS development centers are a virtual extension of the customer offices. The advantages for the customer with this kind of relationship include:

- **Flexibility Development Capacity.** The ITS resources could be ramped-up or down based on the customer needs.
- **Software Development Expertise.** As software development is ITS's main line of business, the customer has access to matured processes and practices.
- **Accountability:** ITS is accountable for Project Delivery and Project Management.
- **Cost Effectiveness:** The same quality of software would be delivered at a lower price.

The type of contract that ITS enters into with customers vary depending on the customer needs. Typically the contracts could be one or more of the following types:

- **Project Fixed Price Contract.** The customer is insured against cost overruns.
- **Time and material Contract.** Based on the resources used onsite or offshore and the stability of the requirements.

The client's processes are important and do affect the outsourcing activities. Apart from the need for the vendor to fully understand the client's business processes, practices, standards and terminology, the client may need to formalize its processes for requirement specification and change management. The vendor's process is important and does affect the outsourcing activities. However, the vendor would typically have a process that caters to the outsourcing requirements. ITS has a quality process based on a structured methodology. ITS feels that it is important that vendor processes are customizable for meeting each client's unique requirements.

The IT infrastructure is used for the develop tools for status tracking of the work outsourced metrics reporting and performance evaluation, and change/configuration management. ITS's IT communications bring together the US, UK, and Indian offices as shown in figure 4. The most effective methods of communication for ITS between the client and the vendor are face
to face, video/teleconference, and e-mail.

A typical engagement would start with a scope out of the clients needs, then requirements would be ironed out, then the type of contract would be determined (i.e. fixed price or time and materials), then a program manager would be assigned to coordinate all activities. ITS may position 1 to 10 people on site with the customer, depending on the complexity of the project, and about 35 people in India.

ITS believes that the two main ingredients driving offshore success is their approach of “accountability for the end result” and the process of building trust prior to any business transaction.
3.5 Outsourcing Critical Factors

Critical factors can be determined by studying the answers to the second section of the survey, pertaining to the general experiences of the two customers and two vendors. By isolating the factors that improve or are crucial to an outsourcing relationship, the factors can then be applied to AlliedSignal in order to determine the feasibility of outsourcing, and if so, how to successfully advance an organization to a “virtual” level.

3.5.1 Relationship Factors

<table>
<thead>
<tr>
<th>Question</th>
<th>AlliedSignal</th>
<th>Ford</th>
<th>Keane</th>
<th>IT Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimal level of collaboration?</td>
<td>Comb. of risk/reward with partnership</td>
<td>Contract basis – one vendor</td>
<td>Risk/reward sharing or Long term strategic partnership</td>
<td>Long term strategic partnership</td>
</tr>
<tr>
<td>What factors effect the level of collaboration?</td>
<td>Definition of requirements</td>
<td>In our environment a long-term strategic partnership is already being addressed at the corporate level for IT. Our internal initiative is better addressed by a contract that is periodically renewed.</td>
<td>Level of risk &amp; dependence on the success or failure of the vendor</td>
<td>Enables continuous learning and improvement in productivity and quality of services provided by the vendor.</td>
</tr>
</tbody>
</table>

Table 4: Outsourcing Relationships

Both customers and vendors recognize that the best relationship is one based on risk/reward sharing within a long-term strategic partnership. Only Ford acknowledges that a single vendor contract suits their purposes, but even they realize that at a higher level, a strategic partnership may be more appropriate. There were a variety of factors that led each organization to select the risk/reward sharing relationship as optimal.

3.5.2 Type of Software to be Outsourced

<table>
<thead>
<tr>
<th>Question</th>
<th>AlliedSignal</th>
<th>Ford</th>
<th>Keane</th>
<th>IT Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>What type of s/w is most suitable for outsourcing?</td>
<td>Business apps. are best, but all s/w, depending on nature of product and definition of requirements.</td>
<td>Integration projects (interconnects between existing systems), projects with multiple 3rd party software involved are both NOT good candidates for outsourcing. Basic coding, Oracle</td>
<td>Outsourcing decisions should be made around a firm’s internal competency and their level of performance. The most likely targets to outsource are those which are not related to the corecompetency of the business and</td>
<td>Maintenance and development projects with well defined requirements are best. The following type of software may be unsuitable for outsourcing: Off-the-shelf Product software</td>
</tr>
</tbody>
</table>


Chapter 3

| work, other commodity software are easier to outsource. | the where the firm is performing below industry standards. These offer the greatest opportunity for improved quality/cost. | Software requiring Rapid development Approach Evolving requirements using a spiral model for development |

Table 5: Most Suitable Types of Software to be Outsourced

Both customers and vendors agree that the best software for outsourcing are ones that can be sufficiently defined. Projects that involve a lot of integration or evolving requirements are not appropriate for outsourcing.

3.5.3 Value and Cost Factors

<table>
<thead>
<tr>
<th>Question</th>
<th>AlliedSignal</th>
<th>Ford</th>
<th>Keane</th>
<th>IT Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>How is outsourcing value added?</td>
<td>1. Reduction of costs-higher margins 2. Focus on core competency 3. Improved quality by higher skilled workers</td>
<td>Ultimately we expect to save money by aligning with a single vendor, using economies of scale to reduce the total costs to Ford. We can also expand &amp; contract the development resources needed at any point in time by simply incrementing or decrementing the funding to the selected vendor, providing Ford with more flexibility and reduced internal overhead.</td>
<td>1. Improved processes resulting in higher productivity and/or quality. 2. Improve management focuses on mission critical projects 3. Reduced cost</td>
<td>1. Flexible software development capacity enabling quick roll-out of products. 2. Same or higher quality software developed at cost-effective rates as the software is developed by organizations whose primary business is software development</td>
</tr>
<tr>
<td>Methods to evaluate cost?</td>
<td>Direct+indirect+ soft/transfer costs (i.e. telecom, travel, legal, time, etc.)</td>
<td>There are travel expenses and hardware/software acquisitions that vendor companies must make to support Ford work that are charged back to Ford.</td>
<td>You need to be able to assess the value that each organization brings to the table against the project's requirements. Examples include: • Subject matter &amp; industry applications knowledge • Technical expertise • Architecture • Processes, tools and methods • Experience etc.</td>
<td>The cost elements to be considered for in-house development are: • Salary costs • Hiring/Firing costs • Hardware and software costs • Office space and facilities costs • Management overheads The cost elements to be considered for out-sourcing are: • Fixed price as per contract or work-</td>
</tr>
</tbody>
</table>

38
Table 6: Value and Costs of Outsourcing

There were a variety of answers from the customers and vendors to the question “Where is the value in outsourcing?” The answers were a lot more consistent to the reciprocal question, “Where is the cost in outsourcing?” The primary costs are direct, indirect, and transfer costs.

3.5.4 Success and Risk Factors
Chapter 3

<table>
<thead>
<tr>
<th>Largest risks of outsourcing</th>
<th>How can risks be mitigated?</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Knowledge loss at end of project</td>
<td>Maintain an on-site presence of the vendor, internal QA, and regular monitoring of expenses to ensure alignment with budget.</td>
</tr>
<tr>
<td>2. Cost overruns</td>
<td>Pick the right vendor and clearly define expectations</td>
</tr>
<tr>
<td>3. Accountability of results</td>
<td>The Outsourcing relationship must be designed with the long term in mind – ideally 3-5 years. The fronted may be onsite to allow for knowledge building. As key managers and executives change the original intent of the relationship must be revisited and refined as necessary.</td>
</tr>
<tr>
<td>1. Loosing control of projects</td>
<td></td>
</tr>
<tr>
<td>2. Quality becoming diminished</td>
<td></td>
</tr>
<tr>
<td>3. Cost rising out of control</td>
<td></td>
</tr>
<tr>
<td>1. Failure of the vendor to deliver as promised. Responsiveness.</td>
<td></td>
</tr>
<tr>
<td>2. Instability of the vendor organization – disruption of the business.</td>
<td></td>
</tr>
<tr>
<td>3. Misunderstandings due to poor communications</td>
<td></td>
</tr>
<tr>
<td>1. Short term view of &quot;relationship&quot; will be eradicate the efficiencies</td>
<td></td>
</tr>
<tr>
<td>2. A rushed front-end process may only uncover tactical gains</td>
<td></td>
</tr>
<tr>
<td>3. Client/supplier personnel transitions could degrade the &quot;basis for the relationship&quot; established early on.</td>
<td></td>
</tr>
</tbody>
</table>

Table 7: Success and Risk Factors

By studying the success, failure, and risk factors in Table 7, one can see a common pattern emerge as keys to success for an outsourcing relationship. The following factors seem to be overwhelmingly important to both customers and vendors:

**Delivering as per requirements:** A failure to deliver is by far the most important factor for success or failure of the relationship. There are two elements to this: first, requirements and expectations need to be clearly defined and mutually understood. As Keane puts it – “managing expectations is critical”. And second, the delivered final product must match requirements and expectations.

**A flexible, communicative relationship:** An open, honest, flexible relationship that does not hinder communication is the second critical factor. Although this factor was more important to the vendors than the customers. The vendors were concerned with a good relationship that entails a good cultural fit. To the customers, cost control is also a critical parameter, however, this factor does not seem to be shared by the vendors.
3.6 Outsourcing Conclusions

In order to develop a strategy for AlliedSignal, it is imperative to study the experiences of others. Unfortunately, it is impossible to develop a general philosophy to outsourcing that will work for all customers, at all times. However, based on the evidence generated by the four cases, important considerations emerge which are summarized in Table 8. Using the table, the most suitable type of relationship can be determined based on the type of software, the level of economic or technological risk, the length of the relationship, the level of definition of the requirements, the amount of innovation required by the product, and the characteristics of the technology embedded in the product.

<table>
<thead>
<tr>
<th>Type of Relationship</th>
<th>Off the shelf or Contract</th>
<th>Risk/reward sharing</th>
<th>Keep in-house</th>
</tr>
</thead>
<tbody>
<tr>
<td>Software</td>
<td>Simple Software</td>
<td>Complex or custom software</td>
<td>Evolving software</td>
</tr>
<tr>
<td>Risks</td>
<td>Low risk</td>
<td>Medium/high risk</td>
<td>Very high risk</td>
</tr>
<tr>
<td>Time Frame</td>
<td>Short</td>
<td>Long</td>
<td>Short</td>
</tr>
<tr>
<td>Requirements/ Expectations</td>
<td>Well defined</td>
<td>Well defined</td>
<td>Not well defined</td>
</tr>
<tr>
<td>Product</td>
<td>Not innovative</td>
<td>Lots of innovation</td>
<td>Not innovative</td>
</tr>
<tr>
<td>Technology</td>
<td>Basic and non-dynamic</td>
<td>Very dynamic, not well understood</td>
<td>Very dynamic, well understood</td>
</tr>
</tbody>
</table>

Table 8: Summary of Critical Factors
Chapter 4

4 The Virtual Company

By studying the software outsourcing relationships of a variety of companies in the last section, an appreciation of the benefits and risks of virtual software development was achieved. To further the understanding, virtual companies in various industries will be studied. In this section, current virtual companies will be studied, and attributes will be extracted, which will then be applied to various business models that would support a virtual software organization.

4.1 The Virtual Company Defined

A virtual company can be defined as a group of people or companies that form temporary partnerships to contribute core competencies or skills to a common objective using information technology, thus sharing skills and costs and achieving economies of scale. The virtual company can gain a competitive edge due to its agility, the array of world-class skills, and the lack of geographic limitations. By quickly amassing world class skills, a virtual company can bring products to market substantially faster than conventional companies. The virtual company can be focused on creating value for the customer and can be closer to the customer since geography is not a limitation. It therefore can be more responsive and create products with more variety, customization and rapidity of delivery in different market segments. However, the virtual company is dependant on good management, trust and collaboration between members, and a robust IT infrastructure. The creation of virtual companies has recently been enabled by the ability to leverage the Internet and new business technologies. And the need for virtual companies has been driven by the increasing competitiveness of the business environment.

The virtual company is an excellent mechanism to implement what Forrester Research calls "dynamic trade", which is the ability to satisfy current demand with a customized response. Dynamic trading is a model that internet-based companies will need to adopt as global competition increases and customers become more knowledgeable. The virtual company will enable dynamic trading since it will increase the variety of software offerings across the whole spectrum from "stripped down to service laden" allowing more innovative and flexible forms of packaging of software, hardware and services. Also, the virtual company will minimize risk by ensuring an abundance of information to the customer and knowledge that the best players are developing the product.
4.2 Virtual Vertical Integration

The degree that a company is vertically integrated is a fundamental question for any organization. Companies are highly vertically integrated when upstream and downstream processes are internalized. Conversely, a company is not vertically integrated if it buys inputs from vendors on the open market and sells outputs on the open market. The virtual company gives an enterprise another option. The firm can be virtually vertically integrated whereby upstream and downstream entities are external but related with close partnerships. As an example, the three most dominant PC manufacturers are contrasted in their approach to integration (see figure 5). Compaq has chosen the highest degree of traditional vertical integration. Compaq produces many of the components of the PC in-house, assembles the PC, and then hands the PC over to its distribution channels. Alternatively, Gateway is less integrated since it buys all the components from vendors on a contract basis, assembles the PC in-house, and then distributes the PC directly to the customer. Dell appears to be the most vertically integrated, however, it achieves this integration not with internal capabilities, but with close partnerships with external vendors. Dell itself only assembles the PCs and then distributes the PC directly to the customer. As Michael Dell puts it, his company is “virtually integrated”.23

![Diagram](image)

Figure 5: Vertical Production Stream of PC Manufacturers
Many of the issues involved with determining the level of vertical integration can be applied to determining the scope of a virtual company. Primarily taxes and regulation, transaction costs, and access to information drive a traditional downstream company's choice of the level of integration. The upstream company's (vendor's) primary motivation to integrate is to establish a dependable market. Another important consideration for both the upstream and downstream companies is bargaining power. For example, if the vendor makes a significant investment for a specific customer and the customer renounces the deal, the vendor may not be able to recoup the cost if there is not a strong contract in place. Conversely, price and quality control over a vendor can be lost if a vendor makes a specific component that a customer needs and cannot readily purchase elsewhere. Therefore, having some form of vertical integration reduces the vulnerability for both vendors and customers.

Companies generally resist integration in order to increase flexibility and reduce overhead. By integrating, costs and partners become fixed, thereby losing flexibility. As long as the products being received from upstream vendors are not very specific, a vendor can be selected from the market place that has the right price and quality. A short-term contract is issued and if there are problems the vendor can be replaced. In a vertically integrated company, if there are problems with a department, turning that department around could be very difficult and costly.

Virtually integrated companies have the best of both worlds, since the company has the advantages of being vertically integrated and also the ability to be flexible. Since a virtual company can change its partners in the coalition, it has operational flexibility and due to the fact that the company increases its ratio of variable to fixed costs, it has financial flexibility. The virtual company does not have the overhead and inefficiencies of a vertically integrated company. "It's a way to gain scale without mass," says David Nadler, founder of New York-based Delta Consulting Group Inc. The virtual company, however, will have to balance the innovation and cost advantages of being able to easily shift vendors against the efficiencies of a tight vendor relationship.

4.3 Virtual Business Processes

Process reengineering has been implemented in many companies over the last two decades.
Recently, however, the trend of process outsourcing has been gaining momentum. During the past several years, specialist firms have emerged in areas such as accounting, inventory control, customer service, and telemarketing. This allows companies to outsource their business processes to a vendor that will manage the processes based on measurable metrics. An extension to this trend is the incorporation of the process specialists as partners in a virtual corporation. This is a necessary trend since each company in the virtual corporation will be focusing on their core competencies and will want to alleviate themselves of these tasks. Since many of these processes require lots of operational information, they will have to be supported by a secure and robust information system.

4.4 Architecture of the Company

The arrangement of components and interfaces of the virtual company needs to be analyzed in order to develop a suitable organization. Many of the considerations and criteria that are used to determine what to outsource (discussed in Section 2.0) are applicable to the formation of a virtual company. The company must consider aspects such as its core competencies, the internal skill set, flexibility and control, and the technology. However, the decomposition of the project is more complex for the virtual company than an outsourcing relationship. In an outsourcing relationship the interactions are primarily between only two or three companies. With a virtual company, interactions between many companies have to be coordinated and information controlled.

Using principles of system architecture, the composition of a generalized virtual company can be derived. By decomposing the required functions of the company, paying attention to upstream factors such as the product, the competition, the market, and technology, one can use the virtual company concept to develop the form of the company. Three methods to decompose functions are via trees, layers, or unstructured networks. When developing a rapidly changing system, architects consider layers to be more effective than trees. Since changes can be made in one layer without effecting other layers, while with trees, changes can propagate in unexpected ways. Also trees have difficulty communicating horizontally. Unstructured networks are the most flexible method, but could potentially lead to increased complexity. This may be a good method for simple virtual organizations. The other consideration when decomposing is to minimize the interfaces between elements. This will ensure the final system can integrate with minimal effort. In the case of the virtual company, it is preferable to minimize the amount of communication between companies because of
Examples of the different types of decompositions can be seen in existing organizations. The tree decomposition is a typical structure used in traditional hierarchical companies. The layered decomposition is used in virtual companies such as Nike and Iridium. For the production of satellites, Iridium has system integration (Motorola) in the top layer. Design, build, and test companies are in the second layer. And build to print companies are in the third layer.\textsuperscript{19}

By arranging the components of the architecture appropriately, sensitive information can be isolated to only the entities that need it for the product or task at hand. However, for a product that needs innovative knowledge across the whole system, it may not be possible to isolate sensitive information. Therefore, using the virtual company organization to develop a complex product that needs a high degree of innovation across the entire system is difficult and possibly dangerous since knowledge will have to be shared and coordinated between all partners.\textsuperscript{25} Partners have different objectives and standards making it difficult to resolve disputes and controlling competitive information. However, having one dominant partner can resolve disputes and provide leadership for the organization in order to achieve a common goal. Conversely, the virtual company is well suited to developing a product with limited innovation that can be kept within one partner and utilizing standards across partner interfaces. In this case, disputes can be minimized and information flow can be well defined.
Therefore, it is important to decompose the product or task between partners so that core competencies do not have to be shared and interfaces are simplified and kept to a minimum.

For example, during the development of the PC, IBM was managing new technologies across the whole system. When assessing the virtual company model, IBM had to trade off the need for fast development with the dangers of outsourcing critical components of the PC architecture (the microprocessor and the operating system). In contrast, Apple did not pursue the virtual company organization to the same extent as IBM and only outsourced many of its none core components, such as casing and assembly. Apple kept the operating system in-house and used a standard, off the shelf microprocessor.

4.5 Virtual Companies

There are three types of virtual companies:20 (1) A group of people working together across geographic or organizational boundaries toward achieving a common goal. This is more commonly known as a "virtual team". (2) A group of companies with specialized skills that partner up for a particular objective employing a high level of trust and collaboration between members. (3) A large company that outsources many of its functions to specialized companies, retaining just a few tasks in house, and focusing on what it does best. Currently, this is the most common form of the virtual company.

4.5.1 Nike

Nike produces athletic footwear almost entirely from outside producers. Nike uses a three-tier strategy to implement a virtual footwear company using its own R&D facilities, some corporately owned component factories, and its corporate offices.

The first tier is composed of "developed partners" and "volume producers."5 Developed partners produce the latest and most expensive products, sharing product development and investment costs. Conversely, volume producers are not dedicated solely to Nike, but can produce shoes for other companies. This allows them to handle larger volumes with greater volatility, but means that Nike does no development work with them.

The second tier supplies materials, supplies and sub-assemblies, while the third tier makes specialized components. The third tier is tightly controlled by Nike since they handle patented components and other components that differentiate Nike from its competitors. Nike
monitors and controls its vendors by having personnel at each site to liaison with corporate.

4.5.2 Verifone Inc.
Verifone Inc. makes credit card approval boxes and the network software that supports the boxes. Verifone was built around its information system and has grown from 5 employees to more than 2,500 over 13 years\(^2\)\(^1\). The company's philosophy is to have the employees physically close to the customers rather than physically close to each other. Verifone learned to make the virtual company work by ensuring employees were introduced to the necessary tools and given the motivation and guidance to make the best possible use of them.

4.5.3 First Virtual Corporation
The First Virtual Corporation (FVC), founded by Ralph Ungermann, produces equipment to allow people to conduct high-quality virtual meetings on their desktop computers\(^2\)\(^2\). The primary motivation to use the virtual company model was to keep up with the rapid changes in networking technology. Ungermann believes that speed and leverage are critical factors in the current high-technology environment. The virtual company is the organizational structure that can achieve both speed and leverage, through the use of partners that are specialized in the specific skills that FVC needs. Ungermann established a core group in Santa Clara to develop and market products, and he began teaming with outside firms to handle other basic business functions such as distribution, finance, human resources, manufacturing. As Ungermann states: "We only want to do a few things, and only if we can do them better than anybody else. Our goal is to find really strong partners and let them be part of our team." If the technology changes and a partner does not adapt, FVC finds a new partner that is on the leading edge of that technology. FVC maintains the marketing direction and technology strategy and outsources the rest to partners.

4.5.4 Dell Computer
Dell Computer is a $12 billion company that sells personal computing equipment primarily to corporate users.\(^2\)\(^3\) Dell is unique from other computer manufacturers both on the front-end to the customer and also on the back-end with suppliers. Dell's uniqueness is that the company relates directly to both customers and suppliers. Dell is also unique internally, since most computer companies produce everything themselves. Dell has chosen to focus on tasks that specifically produce value for the customer, such as delivering solutions and systems. It benefits from this philosophy because instead of unsuccessfully competing with other component manufacturers it can select the best manufacturer and use their components. Founder, Michael Dell, describes his company as being "virtually integrated" versus being vertically integrated like many of his competitors. Being virtually integrated has allowed Dell
Chapter 4

to grow into a $12 billion company in 13 years by not having the burden of a large infrastructure. Also, Dell can be responsive to change by not being dependent on any one manufacturing facility. If one supplier does not keep up in technology, quality, or capacity, Dell can switch to another supplier. The number of suppliers are kept to a minimum and kept as long as possible in a partnership relationship. The supplier’s engineers work in design teams as though they were part of the company. By having a partnership with suppliers, Dell can freely exchange information and plans allowing the supplier to respond quickly and accurately like an internal entity.

4.5.5 Nokia
Nokia was a failing 130 year old Finnish company in the early 1990’s when the new CEO realized that they must focus on an entirely new market with a high potential for growth.24 The CEO decided that cellular phones were an attractive opportunity and rapidly built a virtual corporation to manufacture and sell cellular phones. Based on purchased technology, Nokia outsourced most of the design, component production, and selling of the phones. With aggressive marketing in Europe, USA, and China and maintaining a lean structure, Nokia was able to grow from a small company to having revenues of $2.25 billion in two years. Nokia demonstrated how fast the virtual company can be by getting phones to market faster than their much larger competitors.

4.5.6 IBM
In 1981, IBM was one of the first technology companies to try the virtual model. In order to get the personal computer (PC) to market quickly IBM outsourced the microprocessor design to Intel and some of the operating system to Microsoft based on an “open” architecture philosophy. It was extremely successful, by 1984 the PC had surpassed the Apple computer, and eventually captured 41% of the market by 1985.25 However, due to the lack of proprietary measures intrinsic to the open architecture, IBM lost control when PC clones entered the market at significantly lower costs, using Intel’s chips and Microsoft’s operating system. IBM now only has a very modest portion of the PC market.

4.6 Strategic Attributes

4.6.1 Relationship Type
Close, flexible partnerships are essential for a virtual company since there is a substantial amount of interdependence and trust between them. As can be seen with Nike, Dell and other virtual companies, who use close partnerships with their suppliers, having such
proximity is essential for complete, quick information exchange. Being close makes it feel that the other company is practically an internal department. The interdependence and trust allow for less control by the customer on the vendor. Flexibility must be built into the relationship to compensate for changes in requirements, the market, or technology. If changes occur that imply that a vendor is not supplying world-class quality skills, it is understood that the vendor must bring the skills up to standard or a replacement will be considered.

4.6.2 Information Flow
Communication is closely related to the relationship type, but just because you have a close relationship does not mean you have good information flow. Michael Dell emphasizes information must flow freely between all elements of the virtual company. For Dell, information represents orders and goods. If suppliers have indications for what the customers are demanding, they can anticipate orders and avoid backlogs or lack of goods. Having this information reduces inventory buildups throughout the supply chain since suppliers do not have to compensate for unexpected conditions.

For software development, information primarily represents the specifications of the software and the final product. Though inventory is not an issue in software development, for complex or mission critical software, timely and accurate information reduces rework, thereby, shortening the product development cycle time and allowing for higher quality software. Using virtual concepts, companies such as Apple Computer, MCI and Xerox are already bringing new products to market faster. 26

Specification management is essential for any complex product development. One method of communicating and managing specifications is with Computer Aided Software Engineering tools (CASE). Lockheed Martin and Ford all use CASE tools extensively to communicate designs with their partners. Airbus, which is a consortium of European aerospace companies in four countries, has implemented a common CAD/CAM system so that the consortium can work virtually with common processes and procedures. 27

On site presence is common to all the virtual companies that have been studied in this thesis. Dell has supplier engineers in house and Nike has experienced personnel in all of their partner facilities. Nagel of Leigh University advocates the use of a dedicated high level manager to oversee the multiple linkups involved with a virtual company. This leader would
manage all external interactions including the outflow of technology to make sure that the company does not inadvertently lose the capability to compete.\textsuperscript{17}

The information flow must be based on a reliable, IT infrastructure, without which the company would collapse. The architecture of the infrastructure determines the organizational structure of the company and possibly the upstream and downstream partners. Thought must be given to the technology and standards that are used by customers and vendors so that limitations are not imposed. The increasing capabilities and reduction of costs of IT equipment have recently enabled the development of complex infrastructures and communication techniques. And the Internet provides a common platform from which all companies can communicate.

\subsection*{4.6.3 Components}
Analyzing which components to use in the virtual company leads to the following questions:

- What skill sets are needed?
- What tasks to give to the partners?
- How to select which partners to use?

By applying the architecture principles of section 4.4, skill sets can be identified which are crucial to the objectives of the organization. The tasks that will be delegated to each partner depend on many factors including core competencies, vulnerability due to knowledge loss, and cost efficiencies. The selection of partners follows from the architecture. By identifying the required skills of the partners, the virtual organization can start to converge on a final selection. Also, since trust is an important component of the virtual company a good track record will be crucial. Other criteria that must be used to determine suitable partners include experience, relationship flexibility, cost, legal issues, financial stability, and language.

\subsection*{4.6.4 Relationship Management}
Managing the virtual company can be an overwhelming experience if it is not set up correctly. If each relationship were purely a contractual arrangement, managing all the contracts would require a prohibitive overhead. Also, management must adjust to losing control of functions that are ceded to partners that could fail or take sensitive information. However, by learning from the experiences in the outsourcing cases (in section 3.0) and the existing virtual companies such as Dell and Nike, partnerships with mutual reliance is the most efficient basis for organizing the virtual company. Properly structured, the virtual corporation should be a win-win situation. The management is still not easy, however, since it is now based
more on trust than on control. New methods of measuring progress, cost, and value must be implemented. Each party must have a clear definition of what is expected and how to measure success. A virtual corporation requires different management skills, such as less control and direction and more negotiation and coordination.

Quality control and metrics are difficult in the virtual company, since the final product may go directly to the customer. Therefore, new ways must be devised to ensure high quality products from the partners. In the case of Nike, where shoes get shipped directly to the distribution channel, Nike corporate staff reside at the partner site (i.e. the manufacturing facilities in Asia) and monitor output to ensure high quality and conformance with Nike's requirements. Another aspect that is difficult when working virtually is conflict resolution. To handle this and other project management issues, the SEI has developed a CMM for software acquisition in order to embed outsourcing processes that will allow a company to acquire consistent software from outside vendors.

4.6.5 Factors that Favor the Virtual Company Arrangement

- Flexibility (customization, technology changes)
- Competitiveness (reduced costs, more innovation)
- Complexity (a product that a company can't be world-class in all aspects)
- Innovation (only when capabilities exist outside)
- Market (geographically dispersed market, diverse market segments)

4.6.6 Factors that Inhibit the Virtual Company Arrangement

- Simple, commodity-like products
- Task too complex for today's IT (information flow would be inadequate)
- Lack of viable partners (very new technology, competition, lack of cooperation)
- Proprietary issues (do not want to give away extensive internal knowledge, i.e. IBM)

4.7 Virtual Interactions

In order to manage virtual companies effectively, an understanding of virtual interactions, not just between the companies, but between the company's personnel, is essential. Without good virtual teamwork as a foundation, the virtual company will not succeed. An important consideration is the psychological aspect of communication. "[Virtual corporations] are supported by technology, but the challenge is that they are 90% people and 10% technology," says Jessica Lipnack, co-author of the book Virtual Teams. "That's where virtual
organizations are caught. Lipnack advocates three strong components in order to have a successful virtual team—people, purpose, and link. The people must be self-reliant and must be able to temporarily detach themselves from their company and integrate into the virtual company and then re-integrate themselves in the company once the engagement is finished.

The objectives of the team are much more important to virtual teams than conventional teams. Since they do not have a corporate structure to rely on, they must use the purpose as a guide. Lipnack has found that virtual teams often fail when they do not have a clear mission. One method to infuse enthusiasm for the mission is to get everyone together at the beginning of the project for a “kickoff” meeting. The meeting also allows team members to get to know one another and to ensure that everyone understands his or her place on the team. Lipnack advises, “If there is only one investment to make, make it an investment in social capital.”

Links are the methods that enable virtual, boundary crossing communications. The virtual company has recently become viable because of the explosion of cost effective means of communication. Every worker can now have powerful, easy-to-use means to communicate and process information. Laptop and handheld computers, electronic and voice mail, web sites and collaborative software, cellular and other wireless telephones are some technological innovations that have allowed different individuals to effectively work together even though they may be separated geographically and organizationally. However, William Pape found that when he founded Verifone Inc. he had to provide the guidance and motivation to use the tools. Methods he used included using e-mail for most communication, abolishing written memos, creating a company-wide electronic filing cabinet, and making the tools very easy to use. More recently, Pape has found that establishing a company intranet as quickly as possible is essential in order to create an electronic employee handbook to capture the policies and offer guidance to employees.

Through the links, people must develop trusting relationships. With the lack of face-to-face interactions and the lack of hierarchy, control, and structure from a traditional company, trust becomes an important success factor for virtuality. The trust is based on the knowledge that the relationships are dependent on a partner’s ability to maintain top tier capabilities for the required skills and perform at a level that generates revenue for the virtual company.
4.8 Business Models

Business models can be categorized across a control spectrum from hierarchical to self-coordinating. Traditional business models are based on a hierarchical structure of suppliers, purchasers, and customers where the flow of goods is one-dimensional. The rapid advances in information technology has enabled less hierarchical business models since information can be communicated quickly and cheaply at a global level allowing organizations to align themselves differently with respect to other companies and customers. At an extreme level these business models can be considered to be self-coordinating. Self-coordinating business models allow companies and the customers to come together in flexible, innovative ways that allow the customer to attain value more cost effectively. This thesis will consider business models across the entire control spectrum that are appropriate for developing software in a virtual organization, paying attention to how the business model affects the various attributes of the virtual company.

Business models can also be categorized across a customer value spectrum. Value means the balance between benefit and cost that is important to the customer. This could include the extent to which a product is customized to a customer’s precise needs, personal attention, timeliness and speed, low cost, reliability and consistency, security and accuracy. On one end of the spectrum, some businesses just repackage a product or offer a market place for the product but do not add value for the customer. On the other end, producers and manufacturers employ models that bring a lot of value to the customer.

In order to develop a business model for a software product, the value chain from the customer’s perspective can be disaggregated and each element evaluated for new value offerings. The same can be performed for the processes used to develop the software product. In one scenario, as illustrated in figure 6, the lead company (Company A) performs most of the critical customer interface, except for maintenance, which can be performed by the customer or another company (Company D). The software is developed by the virtual company which is composed of Companies A, B, and C. In this scenario, it is assumed that Company A’s core competencies lie in systems engineering so it holds on to the concept development, product architecture, and system implementation.
By breaking up software development both with respect to the value chain (customer's perspective) and the production chain (industry perspective), opportunities to add value, to add flexibility, and to reduce cost can be sought. For example, at the implementation stage the customer may perceive extra value by bundling extra services or features with the final product. Alternatively, different processes or group of processes can be outsourced to partners. Also, the linkages between processes can be evaluated for a preferable amount of control, flexibility, and knowledge transfer.
With the recent advances in information technology, customers are able to interact with companies at stages in product development that were traditionally inaccessible. By being involved with concept development or testing, customers can improve the product and bring further value to themselves.

4.8.1 Centralized Strategic Alliances
The traditional form of the virtual company is a group of companies that are linked via strategic alliances, usually with one dominant company controlling the products and interfacing with the customer. Nokia, Nike, and Dell are typical examples of this form of business model. Each company coordinates the long-term strategy, marketing and product design and allows partners to handle other tasks such as developing components. Information flow is controlled primarily either by company personnel residing at the partner's site (Nike) or partner's personnel residing at the company's site (Dell). But leaders will change - leaders of one virtual company may become just a partner in another virtual company as changes occur in the market place demand different core competencies. For example, IBM is dominant in mainframe development but follows established standards in the PC arena.

4.8.1.1 Sole Sourcing vs. Selective Sourcing
The virtual company requires world-class, cost-effective partners which leads to three decisions the lead company must answer: How much should each partner do? What if the partner does not maintain its standards? How do you still instill and maintain trust? On the first issue, there is a trade-off between trying to minimize the number of partners on one hand, but on the other, ensuring that each partner is not "spread too thin" and working beyond its scope of expertise. The second issue requires that the central companies have the ability to measure performance and change partners if necessary. If a partner does not maintain world-class standards it must be easy for the central company to replace it. Also, having suppliers compete against each other can create more innovation and less dependency on one vendor. Dell keeps suppliers performing at a high level by threatening to change. BP keeps its IT vendors at a high level by continually benchmarking them against industry standards. This leads to the third issue – trust. If the central company is too prone to change partners, it will be difficult to instill trust between partners. In conclusion a balance must be achieved between flexibility and trust. Both attributes are directly impacted by the number of partners and the type of relationship that is established. And both attributes are needed to develop a high quality product in a timely manner.
4.8.1.2 Stealth Company
One interesting form of the centralized strategic alliance business model is called the stealth company. The stealth company develops partnerships with dominant companies in specific areas of the value chain that interface with outside vendors or customers. Each partner is contractually sworn to secrecy, so the outside world only sees the partners but does not know who the actual company is. The advantage of the stealth company is that the competition does not know whom they are dealing with, it can easily undercut the competition with a low overhead, and it can flexibly change with market demands.

This model was used in the life insurance industry, where a 50-person company partnered, on the back-end, with world-class specialists in financial asset management, policy administration, actuarial analysis, and reinsurance. On the front-end, the virtual company used independent agents to sell policies to the public. Policyholders know only the bank that offered the insurance. In the meantime, the stealth company contained experts in insurance, negotiation, and information technology and controlled the network, the software, and stored the data. The incentive for the agents is a higher commission and the incentives for the partners are that they get business that they could not get alone.

4.8.2 Network Strategic Alliances
The virtual company can be set up as a joint venture between several, equal partners, each one focused on their specific tasks. This model resembles a dynamic network of pre-qualified companies that assemble and disassemble with market demands. It is preferable to apply this method where there is little innovation in the product or the innovation is limited and internalized within the partners since resolving disputes would be difficult without a strong leader. Also, as the IBM PC example shows, in a joint venture, companies can expose themselves to knowledge leaks. However, this model embodies a greater potential for agility and flexibility than the centralized alliance model.

4.8.2.1 DaVinci
DaVinci is a model virtual company developed by a partnership of high-tech companies,
including Anderson Consulting and Intel, to demonstrate the capabilities of a model based on
a network of strategic alliances using the latest in information technology. The model is
developed as general as possible so that it can be applicable to a wide range of industries
but this particular example was created to develop a gold and copper mine on the remote
island of Irian Jaya in Indonesia. The model can demonstrate various scenarios in the
development including the building of the hotel, maintenance of pumps, or booking a flight to
Indonesia. The DaVinci model illustrates how strategy and technology can come together to
allow partners of a virtual company to share product knowledge and market expertise.

4.8.3 Offshore Partners
Another variation for virtual companies is whether to utilize offshore partners. As discussed in
section 2.4.5, there are significant cost benefits but also notable risks. If intellectual property
is a very sensitive issue and costs are not so important, then offshore partners should be
avoided. Otherwise, if the capabilities exist overseas, and a good relationship can be
established with onsite presence, then going offshore can be very beneficial. Going direct or
indirect is another decision depending on the cost and complexity of the task.

4.8.4 Online Communities
The growth of the internet has enabled new and powerful business models that were not
possible before. The internet allows an exchange of information between business partners,
customers, and external channels at a global level without concerns for cost or computer
compatibility. One business model that has emerged is based on online communities or
COINs (communities of interest) of buyers and sellers. These communities are not only
information gathering and information disseminating conduits, they also provide an
environment for transactions. The COIN allows buyers to search for products and then
provides a means to conduct a transaction, charging a small transaction fee. A COIN can
focus on an industry, a market segment, or a slice of the value chain. An industry COIN can
provide customers visibility to all companies in the supply chain and allow them to interact
directly. This provides a powerful opportunity for suppliers to create additional value for the
customer by having more knowledge of the customer’s needs.

Currently, one of the largest industry sites is VerticalNet, which contains over 22 industry
specific COINs focused on vertical trade communities. Each specific COIN provides a
mechanism for members of that industry’s supply chain to conduct business with each other.
Other industry COINs are ChemConnect.com for the chemical industry and Loopnet.com for
the commercial real estate market. An example of a COIN focused on a piece of the value
chain is ChiatDay.com, which supports advertising.

![Figure 7: Internet Communities of Interest (COIN)](image)

4.8.5 Online Auctions
A COIN has not been found for software development. However, one online company, called Flashline.com, hosts auctions for software development. The process is: a company that needs a piece of code written posts the requirements on the website, then software companies (ranging from one-person outfits to large companies like IBM) bid on the project. The purchaser then selects the best bid and starts working directly with the winning vendor. As there is little control, the auction model works well with reused software and new, well-defined software that can be developed over a short time frame.

4.8.6 Self-Coordinating
The latest business model that has emerged with the growth of the internet uses a “self-coordinating” control structure. The only overlying control is a common goal, which results in a large amount of flexibility. However, with the lack of control, the results may not match exactly the requirements of the end user. Therefore, the end user usually has to take responsibility for customization and integration of the final product. Members usually work for free or a nominal fee, but are rewarded by receiving new knowledge or by just being a member of a club that creates a new standard. An example of this business model is the development of the Linux operating system software where thousands of individuals have developed a new, complex operating system with no central leadership, just a common purpose. Each person refines a small piece of the software, so that it continually improves. A person downloads the software from the Internet, works on it, such as adding new features or making it more robust to certain conditions, and then posts the modified operating system back to the Internet. To ensure quality, there exists an organization that tracks and controls the Linux operating system as it evolves.
4.8.7 Conclusions
As discussed earlier, business models can be described across two spectrums, control and value creation. These two important parameters can be used to determine which model is suitable and depending on factors such as the nature of the partners and the product, the most appropriate business model can be selected. The various models that have been discussed are plotted in figure 8. For example, developing a new product (high value) with uncertain partners (need control) would imply selecting a model from quadrant II. A third important parameter is flexibility or agility. Generally, flexibility comes with less control, so more flexible and agile business models can be found in quadrants III and IV. Other factors to consider when choosing a business model, is the amount of innovation and sensitive information that exists within the product or an individual process. Also, the amount of trust that is intrinsic to the relationship.

![Figure 8: Business Models - Value Creation vs. Control](image-url)
5 Strategic Implementation

5.1 Introduction

A roadmap needs to be developed to transform AlliedSignal’s D&SS division from a defense contractor into a flexible, agile, competitive company. It is too difficult to achieve this with an entirely internal solution. The organization and culture of the company has tremendous inertia and the skill set of the employees in critical domains is not up to the latest standards, especially in software development. Therefore, an external strategy needs to be created that will give the division long-term success in a rapidly changing marketplace.

In this chapter, an outsourcing strategy will first be developed that will be tailored to the development of mission critical software products. Second, an organizational architecture will be created that supports the strategy and captures the right types of partners and relationships. Third, guidelines will be laid out to assist the process of partner selection. Fourth, methods of monitoring performance of the company as a whole, and the individual partners, will be considered.

5.2 Outsourcing Strategy

The lack of an outsourcing strategy is one of the most significant shortcomings of companies that partake in such activities. Companies need a holistic, long-term approach to outsourcing. In order to develop an external strategy for software product development, the whole development life cycle has to be broken into basic processes. Figure 9 shows the software development life cycle and the processes that make it up. The development life cycle can be linear, following a “waterfall” model, or it can be iterative, using a “spiral” or a Rapid Application Development (RAD) model. In the waterfall model each phase is completed before moving on to the next phase. Alternatively, the RAD model iterates through the architecture-design-code phases to develop prototypes that best meet the customer requirements.

Figure 9: Software Development Life Cycle
Each process should be analyzed to determine how value can be created for the customer. For example, each process is evaluated as to whether it should be kept in house or outsourced. Also, each process should be studied to see what effect more customer interaction would have. For example, traditionally, the customer was not involved with the design phase, however, with the recent advances in technology, the customer can be given access to many of the design sub-processes allowing them to capture issues or opportunities at a relatively early stage. Another value adding opportunity is each process should be examined to see how new information could be brought to bear on the process. For example, in the design process, the latest development tools can be purchased, or even leased from the Internet, and used to develop a better product more efficiently.

![LIFE CYCLE PROCESS Diagram]

Figure 10: Value Creation Potential for a Life Cycle Process

Therefore, the analysis of outsourcing potential for each process can be based on a set of decision drivers, which are listed below, ranging from strategic to tactical:

- Core competencies
- Value creation potential
- Critical success factors
- Architectural considerations
- Cost modeling

These decision drivers will be applied to the D&SS product development life cycle to determine how to treat each process.
5.2.1 Core Competencies
To determine whether to continue to perform the processes in house or whether to partner with another company, the decision matrix of section 2.3 is utilized as shown in figure 11. Based on the matrix, architecture and design should be kept in house. In addition, both processes have a high potential to add value for the customer through increased customer interaction and benefits from bundling of information. Conversely, based on the matrix, test and support should be outsourced and these processes will benefit little from additional customer interaction or information.

![Decision Matrices for D&SS](image)

5.2.2 Code and Production Processes
Since the proficiency in the code and production processes is high, but business values are low, other decision drivers will have to be considered. The conclusions of the outsourcing chapter recommended critical success factors for different types of relationships. Applying these factors to the nature of the software that the D&SS division intends to develop (i.e. mission critical products), the factors show an alignment with outsourcing (shaded boxes in table 10) for software coding. Also, the potential to increase customer value is not large, either through more customer interaction or adding more information. The fact that the software is evolving and not well defined implies that a RAD development model would be an appropriate choice. Software production is an unimportant process that could be kept in house or outsourced.
### Table 10: Critical Success Factors for D&SS

<table>
<thead>
<tr>
<th>Type of Relationship</th>
<th>Off the shelf or Contract</th>
<th>Risk/reward sharing</th>
<th>Keep in-house</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Software</strong></td>
<td>Simple Software</td>
<td>Complex or custom software</td>
<td>Evolving software</td>
</tr>
<tr>
<td><strong>Risks</strong></td>
<td>Low risk</td>
<td>Medium/high risk</td>
<td>Very high risk</td>
</tr>
<tr>
<td><strong>Time Frame</strong></td>
<td>Short</td>
<td>Long</td>
<td>Short</td>
</tr>
<tr>
<td><strong>Requirements/Expectations</strong></td>
<td>Well defined</td>
<td>Well defined</td>
<td>Not well defined</td>
</tr>
<tr>
<td><strong>Product</strong></td>
<td>Not innovative</td>
<td>Lots of innovation</td>
<td>Not innovative</td>
</tr>
<tr>
<td><strong>Technology</strong></td>
<td>Basic and non-dynamic</td>
<td>Very dynamic, not well understood</td>
<td>Very dynamic, well understood</td>
</tr>
</tbody>
</table>

#### 5.2.3 Concept and Integration Processes

Concept and integration are also difficult processes to evaluate. They are both, however, considered critical to the business. Both processes also present opportunities to involve the customer and add further value to the product with additional bundling of information. Furthermore, concept generation potentially involves a lot of innovation, so if these tasks are to stay in house, a large investment will be required to bring this skill up to world class standards and maintain it. The outsourcing evaluation cannot be finalized until the architecture of the virtual company is considered. If the architecture lumps processes together it may make sense to keep them within the same company or vendor.

#### 5.2.4 Architecture

As discussed in the section 4.4, the methodology used to develop an architecture goes from function to form using the concept of the virtual company. In this case the functions are the processes used to develop the final product. The product development can be decomposed using trees or layers (see figures 12 and 13). The objective is to create an architecture with minimal complexity and minimal interfaces. Unfortunately, by using the tree decomposition method, complexity is increased by the number of separate components and information flows that must be coordinated.
Alternatively, the layered approach, which lump processes together that interrelate, will satisfy the architectural objectives. Ideally, each layer should not be broken out and given to different companies, since there is so much interaction within the layer. In the case of D&SS, and based on the discussion in section 5.2, the systems engineering layer would stay in D&SS, while the software layer would be given to a partner company. The design process is the only process that D&SS would prefer to keep, since the business value is high and D&SS is good at this task. But to keep this task would increase the amount of interactions between the companies dramatically. Whether this is acceptable depends on the type of development
model being used (i.e. waterfall or RAD), the complexity and maturity of the software, and the experience of the two companies.

In systems architecture, various views are created in order to ensure that all interfaces have been considered, namely subsystem views and supersystem views. The subsystem views were created for the decomposition. Figures 14 and 15 show a section of the supersystem view of the virtual company including D&SS (systems engineering), customers (product...
developers), and other partners. The supersystem is shown with respect to its environment, including the market, corporate strategy, government regulation, and technology. Unattached arrows indicate relationships with other projects. The supersystem can be visualized as a dynamic network of specialized, component companies. Under the influence of its environment, the network is continually reconfiguring as projects are finished and new ones are needed. In order for this to work seamlessly, the interfaces between components must be compatible and consistent.

5.3 Selecting Relationship Types

Selecting a relationship type should be based on the analysis of the decision drivers, including core competencies and proficiencies, industry critical success factors, and architectural considerations. Also, the outsourced processes that interact with the customer require a strategic relationship to ensure high quality levels and avoid competitive vulnerability. D&SS, therefore, should keep system engineering processes in house and consider three options for the software processes:

1. Keep all software competencies in house and only outsource coding and test for surge control or only for commodity software products.
2. Outsource all software coding and test to one or more strategic partners
3. Outsource software design, code, and test to one or more strategic partners

Option 1 is only a temporary patch and appropriate only if D&SS were to bring all their software processes up to world class standards in the long-term. Option 2 separates design and code between two companies. So this option is inadequate since design and code are so highly interrelated, especially in a RAD environment where iterative development implies many exchanges of information between design and code as prototypes are developed. Therefore, it is best to pursue option 3 since it allows D&SS to focus on customer needs through systems engineering and it minimizes the amount of information that must flow between companies. Option 3 should be implemented as a virtual organization with multiple vendors to maximize innovation and reduce dependency on any one partner. The production and support processes will be outsourced.
Based on the research in section 3, in order to make option 3 work, D&SS and the partners must set expectations by establishing common goals and objectives to create a win/win relationship. The relationship must also be one with mutual reliance, a long-term focus, and a cultural fit. In the end, the partners must feel like they are an extension of D&SS.

5.4 Planning and Implementation

Once the strategy is formulated and clarified for D&SS, the outsourcing approach must be planned and implemented. Plans for developing capabilities, selecting a business model, and choosing the partners are needed. In order to develop capabilities, a three-phase approach is recommended for D&SS to bring up their software acquisition processes to a sufficient level and develop a communications and information infrastructure (IT) to support virtual software development. D&SS must appreciate the importance of IT in designing a network organization. Outsourcing without an appropriate investment in IT will eventually fail.

   Phase 1: Begin a pilot project to learn the lessons and requirements of outsourcing.
   Phase 2: Undertake a larger job with a single vendor to put the infrastructure in place.
   Phase 3: Develop a long-term strategic alliance with one or more companies.

D&SS will have to appoint a member of senior management to oversee the whole virtual company implementation to ensure: seamless coordination between departments; selection of safe partners; protection of sensitive information; and, maintenance of strategic direction.

5.4.1 Business Model Selection

The decision drivers for business model selection were summarized in section 4.8.7. Each driver is applied to D&SS in order to determine the most suitable business model.

<table>
<thead>
<tr>
<th>Driver</th>
<th>Applied to D&amp;SS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control</td>
<td>Mission critical product development needs control</td>
</tr>
<tr>
<td>Flexibility</td>
<td>Allows for more choice of partners and technology</td>
</tr>
<tr>
<td>Value creation</td>
<td>Value creation is critical for competitive advantage</td>
</tr>
<tr>
<td>Size of processes</td>
<td>Systems engineering and software development are large</td>
</tr>
<tr>
<td>Innovation</td>
<td>Innovation can be isolated to specific processes</td>
</tr>
<tr>
<td>Trust</td>
<td>Trust is required for close relationships</td>
</tr>
</tbody>
</table>

Table 11: Business Model Decision Drivers
Applying the control and value drivers to the graph in figure 8, the best business models would be either a centralized strategic alliance or an online community ("COIN"). The centralized strategic alliance would provide D&SS with sufficient control over all relationships to ensure high quality products for mission critical applications, but at the price of less flexibility. Conversely, the COIN model would give more flexibility in terms of partners and new technology, but with less control. To break the deadlock between control and flexibility, other drivers must be considered. Innovation is a good differentiator. If innovation is going to be system wide, then a virtual organization containing a flexible, tight group of equal partners is suitable. Alternatively, if innovation is limited to a few of the partners, then a model based on control is better. In the case of D&SS, innovation can be limited to within the partners since concept and system architecture innovations can be isolated to D&SS, while software development innovations can be isolated to the partners. Only once in a while there is a major change in software philosophy that will impact system engineering or vice versa - such as when software went from being function oriented to object oriented. In that case, the partners will have to rely on their close relationships to spread the new methods throughout the virtual organization. In conclusion, a strategic alliance model centralized around D&SS, is the best method to use. However, D&SS will not always be the head of the virtual company. With different products, leadership will move to different processes. Moreover, in the future as competition becomes more intense and more flexibility and systemic innovation is needed, COINs will become a more favorable business model. The move into a COIN will be a natural progression after developing outsourcing skills and processes with strategic alliances.

5.4.2 Partner Selection
Now that the partner skills have been identified, partners with world class capabilities for these specific skills must be found. As mentioned in section 4.6.3, other criteria also needed to further differentiate vendors and find a suitable partner. These criteria and how they apply to D&SS are summarized in table 12.

<table>
<thead>
<tr>
<th>Partner Selection Criteria</th>
<th>Criteria Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Capability/Experience:</td>
<td>□ Mission critical, embedded real-time SW&lt;br&gt;□ SW Languages, O/S, and platforms&lt;br&gt;□ Experience and capability of staff</td>
</tr>
<tr>
<td>Process Capability:</td>
<td>□ SEI CMM level 3 or greater&lt;br&gt;□ Six Sigma</td>
</tr>
</tbody>
</table>
5.5 Customer-Vendors Interface Definition

Excellent communication between partners is critical for a virtual organization. The easier and clearer the communication, the less chance for misunderstandings and disputes that hinder the development process. If an architectural perspective is used, interfaces between components (or companies) must be defined. The interfaces must, as in any system, be consistent and robust, and follow a defined protocol. As illustrated in figures 14 and 15, the nature of the interface will change with the business model being adopted. A strategic partnership will have more distance and controls than a community or COIN. This will be reflected in the nature of the contracts.

For a virtual company, the interface is more like the interface between two departments within the same company, than an external contract arrangement. It is essential that the connection should occur at multiple levels, ranging from senior management for strategic issues, down to lower levels for tactical and technical issues. The interface should be handled at a senior level by regular, full time relationship managers. At lower levels, the interface should be composed of Integrated Product Teams (IPTs) with members from different functions and different companies. Also, a good interface for a virtual organization allows relationships to be established quickly and with flexibility. Kirk Bozdogan of the MIT School of Engineering, advocates using “pre-sourcing” from a competitive preferred list of suppliers to get the partner involved in the product development earlier. The interface
requirements can be broken down into three phases; project inception, project management, and product delivery.

5.5.1 Project Inception
At the beginning of a project, the technical and non-technical goals and objectives need to be clearly defined. The best mediums for providing definition vary from company to company, but the SEI software acquisition guidelines recommends that D&SS would put out a solicitation package.¹ The package should contain the software requirements, a statement of work that details software-related tasks, and contract documentation, which contains management and test plans, and contract information such as contract type, incentives, acceptance procedures, and performance guidelines. The vendor would respond with a software engineering plan that describes how the vendor will carry out the software related tasks, a risk management plan, identification of metrics, details of work to be sub-contracted, and other contract details. Also, the vendor should include details of participants, responsibilities, processes, methodologies, techniques, and schedules that will be used.

The most important document is the software requirements. The software requirements establish a common and unambiguous definition of software related contractual requirements that is understood by all parties. Software related contracts consist of technical requirements and non-technical requirements. The technical requirements consist of system requirements that are allocated to software. The non-technical requirements consist of contractual agreements, conditions, and terms. All requirements must be managed throughout the length of the project to ensure that they are unambiguous, traceable, verifiable, documented, and controlled.

Next, a contract will be written for D&SS and its partners. In order to ingrain the spirit of a strategic partnership in the contractual process, an executive summary should be written, which states the intent of all partners very clearly and openly. This will mitigate any chance of misunderstanding the relationship's mission.³⁶ The contract must be set up so that both parties share project risks and rewards and it is perceived to be fair from both sides. Fairness is critical if a long term trusting relationship is to be tried. In order to facilitate this, an "open book accounting" policy may be followed to increase visibility into each others cost structures and verify that there is integrity on both sides.
Another essential characteristic of the contract must be flexibility. Flexibility will allow the parties to continue a relationship as the needs and technologies change. Recently, "master" arrangements have been used, which is designed to be an umbrella under which new agreements can be made without having to renegotiate general terms. Furthermore, Forrester Research recommends that shared penalties and incentives be built into the contract. As Keane's Barbara Wentzel advised, "each partner must have skin in the game." Also, exit strategies must be included in case the partner's performance becomes a problem.

5.5.2 Project Management
Once the project has been awarded and started, frequent interactions between partners are essential. Based on lessons learned from existing virtual companies, on site presence by at least one representative is critical. Dell likes to bring in large teams from suppliers to its company, while Nike has just one "ambassador" at each partner's plant. Also, experience has shown that local representation is essential for the success of the project, especially for complex mission critical software.

The project management of a virtual organization would more closely resemble the management of internal departments rather than contractors. As within normal companies, it is important to have periodic reviews to ensure that there is a continuing mutual understanding of the software requirements, open issues are resolved, corrective actions are handled according to plan, and software products will satisfy the requirements. Project management should also monitor costs, and identify issues and plan for their resolution.

Keane uses two documents throughout a project, a statement of work and a work breakdown structure. The statement of work is a management level document that communicates the expectations, boundaries, and deliverables of the project. The work breakdown structure is a technical document that is an overview of the objectives, tasks, and products for each phase of the product development.

5.5.3 Product Delivery
As the partner completes product deliverables, it is important to ensure that the products satisfy the contract requirements before acceptance for use in the next phase. This is achieved through a series of tests or walk-throughs to demonstrate that the products perform the defined functions. An important detail to be established before testing is to determine what actions are to be taken if a software product is determined to fall short of established
requirements. Keane uses a *system test plan*, which is a technical document based on the customer requirements from the statement of work. The system test plan is based on customer requirements and defines the customer acceptance criteria.

### 5.5.4 Cost/Performance Monitoring

In order to evaluate the progress of the project, assessments of the partners have to be made. Typical measurements include – effort expended, funds expended, progress towards completion of deliverables, number of items closed in the corrective action system, and completion of milestones. Measurement should be made continuously in order to stay abreast of any problems. Monthly reviews allow feedback to be given so that issues are identified and dealt with quickly. Measurement of benefits in the virtual organization will go beyond cost savings and into measurements based on new skills and revenue gains.

### 5.6 Infrastructure

The virtual company needs a robust IT infrastructure in order to support the organization. The IT infrastructure is composed of critical components such as secure communications, project management tools, collaborative computing tools, GroupWare, requirement management, corrective action management, knowledge database, and computer support. Many of these components need to be shared with the partners. For example, the requirement management tool will immediately reflect any changes made by systems engineering and can be incorporated in the software if the software partner has access to the tool. If there are doubts, issues, or questions, a communications infrastructure, such as email, telephone, or videoconferencing, needs to be available.

The companies discussed in the case studies (section 3.0), such as IT Solutions, use dedicated leased phone lines (T-1 and T-3) for domestic partners and satellite links to communicate with offshore partners. Although they are costly to set up and maintain, they do guarantee absolute security. Recently, with the advances in networking technology, the Internet has become an excellent infrastructure to base all communications. The Internet provides a robust and flexible backbone that does not have any single points of failure or dependant on any single point of control. The Internet also has the ability to perform across different platforms. Information flow between UNIX, Windows, and mainframes must be seamless. The drawbacks of the Internet are security, and for videoconferencing, speed. Security can be implemented through the use of security protocols or the use of virtual
private networks (VPN). VPNs encrypt messages and transmit them on the internet. In conclusion, the internet is providing a backbone for the tools and infrastructure that will allow virtual organizations to flourish.

Another aspect the infrastructure has to support is knowledge across the virtual organization. In traditional companies knowledge is transferred primarily through the use of bureaucracy. However, in a geographically and organizationally disparate situation, companies must rely on a communications infrastructure to capture and transfer knowledge. At British Petroleum (BP), knowledge is disseminated throughout their virtual organization with the use of individual home pages on their Intranet. BP’s home pages describe experiences from experts, share technical data, or forward on contacts and useful information on programs or processes. BP learned that virtual teamwork relied on people being open and cooperative and not possessive about information they know. BP found the benefits of the knowledge sharing resulted in less time to solve problems, reduction of rework because of more collaboration, and higher reliability because problems could be debugged remotely.

In conclusion, IT has to progress from being a support function to being integrated the overall corporate strategy so that IT can be leveraged to develop more value for the customer by developing a virtual product development environment, enhancing communications, and enabling knowledge sharing.
6 Conclusions

As competitive pressures increase, companies such as AlliedSignal's D&SS division need to find new ways to increase efficiency and innovations by working jointly with other organizations. This thesis provides a conceptual framework for formulating and implementing a product development outsourcing strategy for mission critical software. The research is based on field data and literature reviews, which were combined and used in a strategic context. Decision drivers and a logical sequence of steps were developed that a company can use to strengthen its chances of success in its outsourcing ventures. The thesis takes the implementation beyond traditional outsourcing and considers issues and methods for implementing a networked organization or a virtual company.

The research shows that a software product development outsourcing implementation must be preceded by a rigorous outsourcing planning and execution strategy that is aligned with the overall corporate strategy. The outsourcing strategy must be considered in a larger system context that considers the market and other companies including customers, competitors, and potential partners. The strategy must also have a long-term focus with an emphasis on balancing the tensions between controlling sensitive assets, developing flexibility, and nurturing trust.

![Diagram](image)

Figure 17: Strategic Forces in Tension in the Virtual Company

Determining what to outsource and to what degree the outsourcing relationship should be established are fundamental to the development of a virtual company strategy. Classic measures such as core competencies, company proficiencies, risk adversity, and control need to be used in conjunction with level of innovation, information flows, and virtual teamwork capability. Other outsourcing conclusions, developed primarily from field studies, are:
Interviews with various companies involved with outsourcing software revealed common keys to success. Both customers and vendors agree that clearly defining requirements and expectations, as well as having an open and honest relationship based on risk and reward sharing, are the most important contributors to a successful outsourcing relationship.

Critical success factors were developed that guide the choice between a contract relationship, a risk/reward partnership, and keeping the software in house. The factors are based on parameters such as the type of software, the level of definition, the amount of innovation, the amount of risk, and the degree of comfort with the technology.

In the future, as the virtual company concept becomes more prevalent, competitive forces will drive virtual companies to become more effective and successful. This is where system concepts can be used effectively as they are in any large system. Therefore, system architecture and system engineering concepts were used in this thesis to determine the design of the virtual company. By the considering the entire product development life cycle as a system, upstream and downstream influences were brought to bear at all levels from the supersystem to subsystem. The product development processes were decomposed using various philosophies that considered information flows, flexibility, and insulation of core competencies. In this way, a virtual company was designed that optimized the amount of interactions between partners and the ease of reconfiguring the company. Conclusions that were developed from researching virtual companies:

- A business model must be selected that is dictated by the amount of control required and the amount of value created. Tradeoffs have to be made for control versus flexibility and innovation versus intellectual property concerns. For D&SS, a centralized strategic alliance was the final choice since this model entailed a high degree of control that ensured high quality products for mission critical applications. It also allowed more protection for D&SS's core knowledge.

- Trust is an intrinsic component of the virtual company and must be nurtured for a long-term relationship to succeed. The trust is based on the knowledge that the relationships are based on a partner's ability to maintain top tier capabilities for the required skills and perform at a level that generates revenue for the virtual company.
Chapter 6

- Leveraging IT is essential to the success of a virtual company. IT has to move from purely support to being integrated in the overall strategy. IT has to complement the architecture of the networked organization by providing the latest technology to: the tools and infrastructure that facilitates virtual product development; produce an environment for creativity; and capture knowledge from throughout the organization.

If D&SS is successful in the future, going virtual will give the company new basic abilities which will further increase its competitiveness. D&SS will learn to see beyond their internal capabilities and recognize superior abilities in a potential partner. D&SS will also learn to manage and collaborate with geographically distributed organizations that have different objectives and cultures.

6.1 Reflections and Lessons Learned
Finding objective subjects to survey on this topic is difficult since negative connotations are often associated with outsourcing because of layoffs and workplace instability.

Combining systems engineering concepts with management principles resulted in a powerful method of analyzing a strategic management issue. The systems thinking facilitates the consideration of complexity, interactions, and the "big picture view".

6.2 Recommendations for Further Research
The field of study for virtual companies is very large, which gives the opportunity to go into greater depth in many areas. A few possibilities for further research are:

1. Performing interviews and surveys of managers in existing virtual companies in order to understand, in greater detail, the issues and vulnerabilities of those organizations.
2. Investigating effective metrics to measure the long-term impacts of outsourcing.
3. Applying more system concepts to the virtual company design to determine more trade-offs and to develop alternate innovative forms of the virtual company. Concepts such as the Design Structure Matrix (Eppinger, et. al) or axiomatic design principles (Suh) could be applied to further optimize the virtual company.
4. Carrying out additional investigations into tools and infrastructure that can further facilitate virtual product development.
Appendix A – Survey

Part A – Project Specific Questions

1. Is the outsourcing part of an overall strategy? If so, briefly what is the strategy?

2. How are the requirements for the software specified?

3. What type of software is outsourced (i.e. IT, real time embedded, etc.)?

4. How large is the software (lines of code)?

5. What metrics and statistics (i.e. delay, quality, SLOCs, etc.) are used to track the outsourcing activities?

6. How is the quality of the software measured? How is software tested, and by whom?

7. What actions are taken when software is determined to fall short of established requirements?

8. What are the additional costs from outsourcing? How are they quantified?

9. What are the additional cost-benefits from outsourcing? How are they quantified?

10. Please describe the nature of the relationship between client and vendor(s).

11. Please briefly describe the contract between the two companies.

12. What is the organizational structure between the client and the vendor?

13. What is the geographic distribution of this organizational structure?

14. How does the client’s process affect the outsourcing activities? Is it important?
15. How does the vendor’s process affect the outsourcing activities? Is it important?

16. How should IT be used to support the outsourcing activities?

17. What is the nature of the communication network? Regionally? Globally?

18. What are the most effective methods of communication between the client and the vendor?
   1. 
   2. 
   3. 

20. What is the flow of knowledge between the client and the vendor?

**Part B – General Questions**

1. What level of collaboration is optimal for a software outsourcing relationship?
   
   Contract basis – one vendor
   Contract basis – multiple vendors
   Risk/reward sharing
   Long term strategic partnership
   Other: __________________________

   What factors contribute to this selection?

2. What are the most important criteria for selecting a vendor?
   
   1. 
   2. 
   3. 

3. What are the important issues when considering outsourcing offshore?
4. How does outsourcing software add value to a client's enterprise?

1. 
2. 
3. 

5. What methods would be used to evaluate and quantify the total cost of outsourcing a unit of software versus coding it in house?

6. What metrics and statistics should a client use to monitor the following parameters:
   - Schedule performance of the vendor
   - Cost of the vendor
   - Quality of the product
   - Success of the relationship

7. What type of software is best to be outsourced? What type should not be outsourced? What level of complexity and size is appropriate to outsource (i.e. CSCI, CSC, CSU)?

8. In the future (1-5 years), what will be the most effective methods of communication between the client and the vendor?

9. What are the most important success factors for an outsourcing relationship?

1. 
2. 
3. 

10. What are the most important factors that lead to failure of an outsourcing relationship?
11. What are the largest risks of outsourcing?

1. 

2. 

3. 

12. How can these risks be mitigated?

13. Other comments?
## Appendix B – Consolidated Results of Survey

### Case Surveys Summary

#### PART A

<table>
<thead>
<tr>
<th>No</th>
<th>Question</th>
<th>AlliedSignal</th>
<th>Ford</th>
<th>Keane</th>
<th>IT Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>How are requirements specified?</td>
<td>Statement of work</td>
<td>Functional Specifications are developed by the planning organization. All project teams use a common format.</td>
<td>Detailed information on the software, users and service levels by type. Expectations are established with specific Metrics. Monthly reports &amp; meetings measure performance against metrics. Quarterly reviews evaluate trends, raise and resolve issues, and reset standards as needed.</td>
<td>Requirement Spec or Functional Spec or Product Spec document containing the functional, performance, security and other operational requirements</td>
</tr>
<tr>
<td>2</td>
<td>What type of s/w is outsourced?</td>
<td>IT</td>
<td>Customizations to our Metaphase PIM environment, which includes WEB interface, Orbix interconnect, C method code, MoDEL data model, scripts (MetaScript).</td>
<td>Applications (IT) Software</td>
<td>Web-based Solutions, Commercial Apps, System Software, ERP Solutions, Telecomm, Tools and Utilities</td>
</tr>
<tr>
<td>3</td>
<td>How large is the s/w?</td>
<td>Varies</td>
<td>We have approx. 1MIL lines of custom code written at this point for all Metaphase customizations for a wide variety of customer apps.</td>
<td>Varies</td>
<td>Size varies from a few thousand lines of code to over a million lines of code</td>
</tr>
<tr>
<td>4</td>
<td>Project management tools that are used?</td>
<td>MS Project</td>
<td>Microsoft Project is used (but not standardized) by Project Leads.</td>
<td>Comm. Strategy Periodic meetings minimum is quarterly Weekly, monthly and quarterly reporting Balanced Scorecard</td>
<td>Quarterly/Half-yearly reviews with the Customer Metrics reports at Program level</td>
</tr>
<tr>
<td>5</td>
<td>What metrics are used to track?</td>
<td>Delivery and quality rating</td>
<td>Quality and timely delivery are the main metrics applied.</td>
<td>Quantitative metrics: Response, repair and resolution times. Number and type of activities Number and type of delivered products Qualitative metrics: Client Satisfaction</td>
<td>Ontime Index: Projects delivered on schedule Review Efficiency: Number of defects found by ITS/( Number of defects found by ITS + customer) Effort Overrun</td>
</tr>
<tr>
<td>6a</td>
<td>How is the quality measured?</td>
<td>Testing</td>
<td>Code reviews are performed by internal Ford personnel, and a separate Quality</td>
<td>Quality standards are established up front.</td>
<td>Number of defects reported by the customer and an Error Prediction Model</td>
</tr>
</tbody>
</table>


<table>
<thead>
<tr>
<th></th>
<th></th>
<th>Assurance group (also internal) tests the functionality. Bugs are tracked in a database. No formal tracking mechanism is used for code reviews at this time.</th>
<th>Generally software must pass several levels and types of testing including user, unit and systems tests as well as most established quality requirements (i.e. documentation)</th>
<th>Unit Testing done by the programmer. System/Regression Testing done by the Test team. Field testing done by ITS at customer location and Acceptance Testing done by the customer.</th>
</tr>
</thead>
<tbody>
<tr>
<td>6b</td>
<td>Testing?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>What actions are taken when s/w falls short of requirements?</td>
<td>The planner involved raises any issues back to the customer contact to discuss work-arounds and other issues with the specified functionality as development progresses.</td>
<td>Rework the product until it passes. At some point, depending on the agreement the work is done at no cost to the client.</td>
<td>Corrective Action to resolve the problem. Preventive action to analyze the error so that it does not repeat in the future.</td>
</tr>
<tr>
<td>8</td>
<td>Nature of relationship?</td>
<td>Contract</td>
<td>Contractual with agreed metrics and long term (mullet-year) commitments.</td>
<td>Mutual dependence: It is critical that both organizations have skin in the game.</td>
</tr>
<tr>
<td>9</td>
<td>Contract description</td>
<td>Fixed cost for basic activities, surcharge for other activities outside the contract</td>
<td>See above</td>
<td>Long and complex. It must outline expectations and dissolution.</td>
</tr>
<tr>
<td>10</td>
<td>Organizational structure between parties?</td>
<td>Program Managers face off from both Companies, with Project Leads from the vendor reporting to Ford supervision.</td>
<td>Client should have a vendor liaison with enough power to set priorities, evaluate performance and clear roadblocks.</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Geographic distributions?</td>
<td>US</td>
<td>Local Detroit area presence, supplemented by resources from other geographic locations (including off-shore).</td>
<td>Varies</td>
</tr>
<tr>
<td>12</td>
<td>Impact of client's process</td>
<td>Make or break project, needs to be defined, specific, flexible, and measurable</td>
<td>Very important. It's essential that the client process mesh with the outsourcer activity; but the outsourcer vendor does not have to utilize the same client process in their own internal shop.</td>
<td>Must be in sync. Vendor/Client processes should be seamlessly integrated. You also need a cultural match.</td>
</tr>
<tr>
<td></td>
<td>Impact of vendor's process</td>
<td>How is IT used?</td>
<td>Communication network</td>
<td>Most effective methods of communication?</td>
</tr>
<tr>
<td>---</td>
<td>---------------------------</td>
<td>-----------------</td>
<td>-----------------------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>13</td>
<td>-</td>
<td>Telecom, infrastructure. Might be the end customer</td>
<td>Direct communication links and access to system resources will often play an important role in effective outsourcing (vendor accessing client equipment).</td>
<td>Email, scheduled in person meetings, telephone, formal process docs that are trained and available</td>
</tr>
<tr>
<td>14</td>
<td>Vendor should integrate their best practices with the clients.</td>
<td>IT's quality standards should be followed.</td>
<td>Regardless of the size and scope of the relationship, each organization should have an individual who is responsible to oversee the engagement. Communications networks grow as the size and complexity of the engagement grows.</td>
<td>On-site presence by the vendor Teleconference Video conference periodic weekly/monthly status reporting periodic monthly/quarterly metrics reporting periodic weekly/monthly and quarterly meetings quarterly/annual engagement reviews</td>
</tr>
<tr>
<td>15</td>
<td>The vendor's process is important. The vendor would typically have a process that caters to the outsourcing requirements. It is important that vendor processes are customizable for meeting each client's unique requirements.</td>
<td>Develop tools for status tracking of the work out-sourced Metrics Reporting and performance evaluation Change/Config. Management</td>
<td>VSNL Links- 1. US Locations : San Ramon 2. India Locations : Bangalore (South End road, Bull Temple Road) Chennai (Pycroft Gardens) 3. UK Locations : Bristol</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td>Face to face Video &amp; teleconference E-mail</td>
</tr>
</tbody>
</table>
### PART B

<table>
<thead>
<tr>
<th>No</th>
<th>Question</th>
<th>AlliedSignal</th>
<th>Ford</th>
<th>Keane</th>
<th>IT Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>Optimal level of collaboration?</td>
<td>Comb. Of risk/reward with partnership</td>
<td>Contract basis – one vendor</td>
<td>Risk/reward sharing Long term strategic partnership</td>
<td>Long term strategic partnership</td>
</tr>
<tr>
<td>1b</td>
<td>What factors effect the level of collaboration?</td>
<td>Definition of requirements Risk for non-performance Reward for delivery</td>
<td>In our environment a long term strategic partnership is already being addressed at the Corporate level for IT. Our internal initiative is better addressed by a contract that is periodically renewed.</td>
<td>Level of risk &amp; dependence on the success or failure of the vendor</td>
<td>Enables continuous learning and improvement in productivity and quality of services provided by the vendor.</td>
</tr>
<tr>
<td>2</td>
<td>Selection criteria?</td>
<td>Cost Flexibility Availability Soft skills</td>
<td>Technical capability Cost structure Relationship</td>
<td>Experience – been there done that - references Stability both organizational and financial Established repeatable processes with an emphasis on quality.</td>
<td>Flexibility Reliability Track record</td>
</tr>
<tr>
<td>3</td>
<td>Offshore issues?</td>
<td>Legal Proj. mgmt. Confidentiality</td>
<td>Same as #2 + size of company</td>
<td>All of the above and the ability to effectively communicate</td>
<td>Knowledge Transfer, Communication Cultural Issues, Security Process and Program Management Human Resources deployed and their retention</td>
</tr>
<tr>
<td>4</td>
<td>How is outsourcing value added?</td>
<td>1. Reduction of costs-higher margins 2. Focus on core competency 3. Improved quality by higher skilled workers</td>
<td>Ultimately we expect to save money by aligning with a single vendor, using economies of scale to reduce the total costs to Ford. We can also expand &amp; contract the development resources needed at any point in time by simply incrementing or decrementing the funding to the selected vendor, providing Ford with more flexibility and reduced internal overhead.</td>
<td>Improved processes resulting in higher productivity and/or quality . Improve management focuses on mission critical projects Reduced cost</td>
<td>Flexible software development capacity enabling quick roll-out of products Same or higher quality software developed at cost-effective rates as the software is developed by organizations whose primary business is software development</td>
</tr>
<tr>
<td>5</td>
<td>Methods to evaluate cost?</td>
<td>Direct+indirect+soft/transfer costs (i.e. telecom, travel, legal, time, etc.)</td>
<td>Our current outsourcing efforts have been with a few select companies that have expertise with the Metaphase technology. Development</td>
<td>You need to be able to assess the value that each organization brings to the table against the project's requirements. Examples include: Subject matter &amp;</td>
<td>The cost elements to be considered for in-house development are: Salary costs Hiring/Firing costs Hardware and software costs</td>
</tr>
</tbody>
</table>
resources from these companies are more expensive than if we were to train them in-house. However we are unable to retain internal resources since they will leave for more lucrative job offers elsewhere once they have experience. There are also travel expenses and hardware/software acquisitions that these companies must make to support Ford work that are charged back to Ford.

Outsourcing decisions should be made around a firm's internal competency and their level of performance. The most likely targets to outsource are those which are not related to the core competency of the business and the where the firm is performing below industry standards. These offer the greatest opportunity for improved quality/cost. It makes less sense to outsource things which are core competencies of the business that are performing a world class levels. There is no need to introduce risk here for minimal benefit.

Office space and facilities costs
Management overheads
The cost elements to be considered for outsourcing are:
Fixed price as per contract or work-order
Communication cost
Vendor development costs
Management overheads

### 6 What type of s/w is most suitable for outsourcing?

| Business apps. are best, but all s/w, depending on nature of product and definition of requirements. |
| Integration projects (interconnects between existing systems), projects with multiple 3rd party software involved are both NOT good candidates for outsourcing. Basic coding, Oracle work, other commodity software are easier to outsource. |
| Outsourcing decisions should be made around a firm's internal competency and their level of performance. The most likely targets to outsource are those which are not related to the core competency of the business and the where the firm is performing below industry standards. These offer the greatest opportunity for improved quality/cost. It makes less sense to outsource things which are core competencies of the business that are performing a world class levels. There is no need to introduce risk here for minimal benefit. |
| Maintenance and development projects with well defined requirements are best. The following type of software may be unsuitable for outsourcing:
Off-the-shelf Product software
Software requiring Rapid development Approach
Evolving requirements using a spiral model for development |

### 7a What metrics should be captured for schedule?

| Delivery/quality ratings | Timely delivery | Delivery as promised. This should be clearly defined and mutually agreed before the project starts. |
| Predefined cost equation (direct+indirect costs) | Hourly rate, travel costs, etc. | Delivery as promised. This should be clearly defined and mutually agreed before the project starts. |
| Quality rating | Bugs encountered | Delivery as promised. This should be clearly defined and mutually agreed before the project starts. |

### 7b What metrics should be captured for cost?

| On-time Index: Number of projects delivered on time/Total number of projects allocated. Schedule Overrun : Slippage/Planned days |
| Productivity: $ per Function Points delivered or $ per Bug fixed |

### 7c What metrics should be captured for quality?

| Error spill over: Number of defects detected by the client/(Number of defects detected by |
### Appendix B

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>7d</td>
<td><strong>What metrics should be captured for success?</strong></td>
<td>Predefined value statement</td>
<td>Functionality delivered as promised</td>
<td>Did the vendor properly set expectations and manage to them? If so you are probably satisfied even if there were many changes along the way.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Customer Satisfaction Index: Based on perception survey on service, quality and value-for-money provided by vendor Vendor Satisfaction Index: Based on perception survey on responsiveness, training and increased volume of business</td>
</tr>
<tr>
<td>8</td>
<td><strong>Future methods of communication</strong></td>
<td>Video</td>
<td>Virtual meetings using Net-meeting and related technology.</td>
<td>Personal relationships! We may use many tools to measure, monitor &amp; communicate information, but there must be a close working relationship where both share risks and rewards.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Video conferencing</td>
</tr>
<tr>
<td>9</td>
<td><strong>Success factors</strong></td>
<td>Cost control Commitment to deliveries Flexibility in relationship mgmt. (soft)</td>
<td>Consistent delivery of promised functionality Affordable cost structure On-going relationship</td>
<td>Cultural fit Clearly defined, mutually understood expectations Constant communication.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Delivering as per clients requirement Flexibility Openness and honesty</td>
</tr>
<tr>
<td>10</td>
<td><strong>Failure factors</strong></td>
<td>Scope creep Failure to test Retention of staff on both sides</td>
<td>Poor specifications Poor communications Technical complexity that overwhelms the vendor</td>
<td>Expectations not met. Failure to communicate Failure to deliver</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Not delivering as per requirement Not being responsive Lack of transparency</td>
</tr>
<tr>
<td>11</td>
<td><strong>Largest risks of outsourcing</strong></td>
<td>Knowledge loss at end of project Cost overruns Accountability of results</td>
<td>Loosing control of projects Quality becoming diminished Cost rising out of control</td>
<td>Failure of the vendor to deliver as promised. Responsiveness Instability of the vendor organization – disruption of the business. Misunderstandings due to poor communications</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Short term view of &quot;relationship&quot; will be eradicate the efficiencies A rushed front-end process may only uncover tactical gains Client/supplier personnel transitions could degrade the &quot;basis for the relationship&quot; established early on.</td>
</tr>
<tr>
<td>12</td>
<td><strong>How can risks be mitigated?</strong></td>
<td>Maintain an on-site presence of the vendor, internal QA, and regular monitoring of expenses to ensure alignment with budget.</td>
<td>Pick the right vendor and clearly define expectations</td>
<td>The Outsourcing relationship must be designed with the long term in mind – ideally 3-5 years. The fronted may be onsite to allow for knowledge building. As key managers and executives change the original intent of the relationship must be revisited and refined as necessary.</td>
</tr>
</tbody>
</table>
Bibliography

2 The Outsourcing Institute, “The Emergence of Application Development and Maintenance Outsourcing as a Tool for Maximizing IT Value”, 1996.


36 Presentation by Bozdogan, K., Professor at MIT, "The Impact of Early Supplier Integration on Systems Architecture: Some Lessons Learned", November 6, 1998.

