TOWARDS AN EXTENDED ENTERPRISE THROUGH e-BUSINESS INTEGRATION

by

Nicolette Mostert

DISSERTATION

submitted in compliance with the requirements for the degree of

MAGISTER TECHNOLOGIAE

in

INFORMATION TECHNOLOGY

at the

FACULTY OF COMPUTER STUDIES

of the

PORT ELIZABETH TECHNIKON

Promoter

Dr Dalenca Pottas

December 2004

“ It is not the strongest of the species that survives,
nor the most intelligent - it is the one that is
most adaptable to change. ”
Charles Darwin

DECLARATION

I, Nicolette Mostert, hereby declare that:

- The work in this dissertation is my own work.
- All sources used or referred to have been documented and recognized.
- This dissertation has not previously been submitted in full or partial fulfillment of the requirements for an equivalent or higher qualification at any other recognized education institute.

_____

Nicolette Mostert

ACKNOWLEDGEMENT

The financial assistance of the Department of Labour (DoL) towards this research is hereby acknowledged. Opinions expressed and conclusions arrived at, are those of the author and are not necessarily to be attributed to the DoL.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

API	Application Programming Interface
B2B	Business-to-Business
B2Bi	Business-to-Business Integration
BPM	Business Process Management
BPMS	Business Process Management System
CLI	Call-Level Interface
COM	Component Object Model
CORBA	Common Object Request Broker Architecture
CRM	Customer Relationship Management
DCOM	Distributed Component Object Model
EAI	Enterprise Application Integration
ebXML	Electronic Business XML
EDI	Electronic Data Interchange
EJB	Enterprise JavaBeans
ERP	Enterprise Resource Planning
ESB	Enterprise Service Bus
ETL	Extract, Transform, and Load
ICT	Information and Communication Technology
IT	Information Technology
JDBC	Java Database Connectivity

JSP	Java Server Pages
JVM	Java Virtual Machine
MOM	Message-Oriented Middleware
ODBC	Open Database Connectivity
OLE	Object Linking and Embedding
ORB	Object Request Broker
PIP	Partner Interface Process
RMI	Remote Method Invocation
RNIF	RosettaNet Implementation Framework
ROI	Return On Investment
RPC	Remote Procedure Call
SCM	Supply Chain Management
SOAP	Simple Object Access Protocol
TP	Transaction Processing
UDDI	Universal Description, Discovery and Integration
WSDL	Web Services Description Language
XML	Extensible Markup Language

CHAPTER 1

INTRODUCTION

Chapter 1 serves as an introduction to the remainder of this dissertation. The reader is introduced to the problem domain and the problem addressed by this research project is stated. In addition, the objectives of this study are established.

In Chapter 2 the reader is introduced to the concept of an extended enterprise.

1.1 INTRODUCTION

In the global business environment organizations face intense competition from competitors and must constantly find new opportunities to increase efficiencies and increase profitability. One increasingly popular approach is to form extended enterprises where strategic relations between the organization and its suppliers, customers and other business partners help to provide a competitive edge over competitors (Active Software, 2000; Lublinsky, 2002). In order for the extended enterprise to be successful, it is important that relevant, up-to-date information is available to all participants at all times. Traditional ways of doing business, such as e-mail, faxes and voice mail, introduce delays and as a result, another solution must be found to share data, services and processes between the members of the extended enterprise. During these difficult economic times organizations are not willing to invest in new systems that will enable collaboration between the members of the extended enterprise. Instead, interest in integrating relevant existing systems between the members of the extended enterprise to allow for the sharing of data, services and business processes is growing (Reese, 2002).

The response to IDC's *Integration Drivers Study,* 2002 reflects the increasing importance of integration projects, as can be seen in Figure 1.1 (Hailstone, Vermeulen & Warmerdam, 2002). In addition, the growing importance of integration can be seen in the evolution that has taken place in organizations regarding who within the organization considered integration to be important. Instead of just the IT department feeling that integration is critical, the business leaders of the organization are increasingly looking to integration to improve key performance indicators (Reese, 2002).

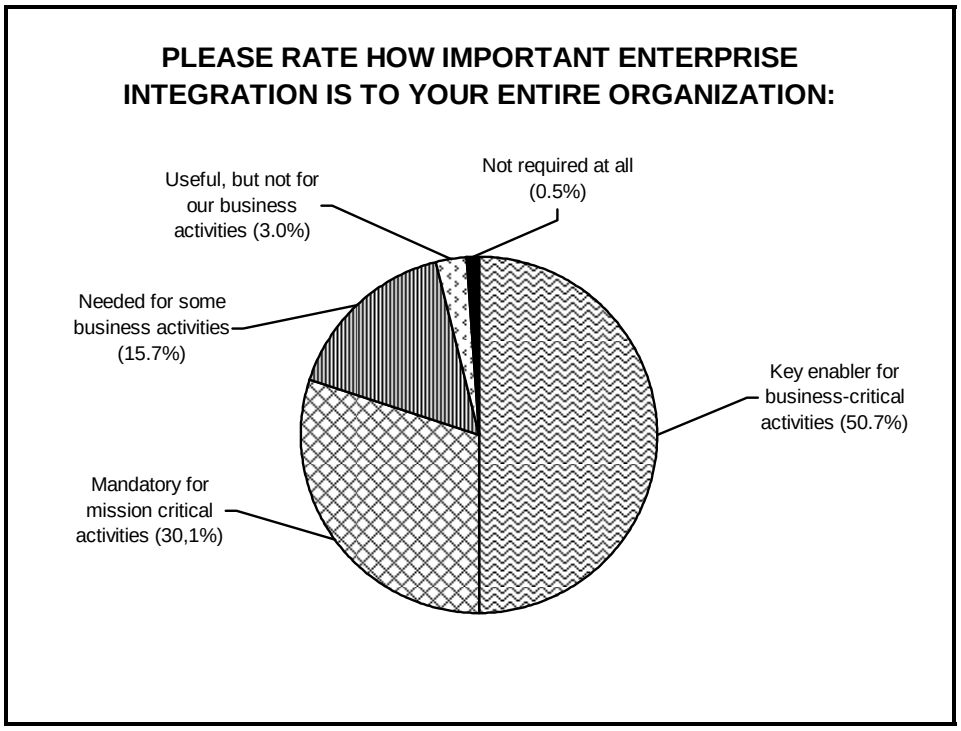

FIGURE 1.1: Criticality of integration to the organization (Hailstone, Vermeulen & Warmerdam, 2002).

The focus of this project will be on introducing the concept of an extended enterprise to business leaders, subsequently presenting e-Business Integration and the supporting role that it can play in the establishment of an extended enterprise. Various literature sources will be consolidated to describe the integration approaches and supporting integration technologies and standards that can be employed in establishing integrated communication between the members of the extended enterprise. Finally, a phased approach will be proposed that can be employed in supporting the establishment of an extended enterprise through e-Business Integration.

1.2 THE EXTENDED ENTERPRISE

The term *extended enterprise* represents the concept that an organization does not only consist of its employees, board members, and executives, but also its business partners, suppliers and customers. This collaboration is done to streamline business processes and enhance customer benefits (IFS, 2003). When considering existing EDI-based collaboration, the extended enterprise is not an entirely new concept, but what distinguishes the extended enterprise is that it aims for much more interactive services, higher flexibility, lower investments, and a considerably larger reach and richness in its collaboration (IFS, 2003). In order for the extended enterprise to be successful, it is important that relevant, up-to-date information is available to all participants at all times. To achieve this, the information systems of the organization must operate within a distributed application architecture (Cherry Tree, 2000).

At the core of the organization is the Enterprise Resource Planning (ERP) backbone, which is a primarily *inward* facing application that tracks the internal flow of information. To create an extended enterprise it is necessary that its information systems face *outward* to enable connectivity to business partners, suppliers and customers. Applications that extend the organization outwards include Customer Relationship Management (CRM) and Supply Chain Management (SCM) systems. These outward facing systems must be fully integrated with the ERP backbone for the organization to become a truly extended enterprise (Cherry Tree, 2000). Despite the value that can be derived from integrating the outward facing systems with the core of the enterprise, this final step in the creation of the extended enterprise is where most businesses fail. This is due to the fact that these systems were never designed to communicate with each other and the task of

integrating these systems remains elusive for even the most highly respected systems integrators (Cherry Tree, 2000). This integration challenge has resulted in the creation of disconnected applications throughout the organization, often referred to as *stovepipes*, *islands of automation*, or even *vertical application silos*. These islands of automation are unaware of events outside their domain and to solve this problem it is necessary to use various tools and technologies to integrate the entire extended enterprise (Cherry Tree, 2000).

1.3 e-BUSINESS INTEGRATION

Until recently, most integration projects have focused on *intra*-enterprise integration, or application integration that occurs within the firewall of the enterprise. This is known as Enterprise Application Integration (EAI). With the emergence of extended enterprises, *inter*-enterprise integration has become a priority. This involves the integration of applications with external business partners, suppliers and customers across the firewall. Integration across the firewall is also known as Business-to-Business Integration (B2Bi) (Linthicum, 2000).

The EAI and B2Bi markets are merging to form the e-Business Integration market (Aberdeen, 2003). A significant driver for this situation is the fact that you must be able to integrate your internal information systems before you can integrate with the external systems of your business partners, suppliers and customers.

This study will focus on both Enterprise Application Integration (EAI) and Business-to-Business Integration (B2Bi), which will be discussed in more detail in the following two sections.

1.3.1 ENTERPRISE APPLICATION INTEGRATION (EAI)

According to the EAI Industry Consortium, EAI can be defined as the process of integrating multiple applications that were independently developed, may use incompatible technology, and remain independently managed (2003[2]). EAI enables previously isolated systems within an organization to be integrated with related information and processes in other systems, thus enabling the organization to (Alpha Technologies, 2003):

- leverage valuable data in legacy and homegrown systems
- integrate legacy systems with newer technologies that will be important for current and future operations
- facilitate flexible data access and flow for ever changing business needs, and
- incorporate any unstructured data, such as e-mail, for business use.

It becomes clear that EAI can enable the organization to adapt rapidly to the ever-changing business environment. To position the organization for even greater success it is necessary to integrate the systems within the organization with the systems of all business partners, suppliers and customers of the enterprise. To achieve this, business-to-business integration is necessary.

1.3.2 BUSINESS-TO-BUSINESS INTEGRATION (B2Bi)

B2Bi refers to the controlled sharing of data and business processes among any connected applications and data sources across organizational boundaries using the Internet.

B2Bi allows organizations to link, for example, their SAP system to their supplier's Baan system and move information between them as needed to create an extended enterprise (Linthicum, 2000). In addition to the benefits of EAI, B2Bi can produce the following benefits for the organization (iSource, 2003):

- improved response times for inter-enterprise processes
- customer satisfaction due to improved purchasing process and shorter response times
- automation of high volumes of transactions between trading partners, and
- reduction in administrative costs.

1.4 PROBLEM DEFINITION

Increased collaboration is necessary between an organization and its suppliers, customers, and other business partners, collectively known as the extended enterprise, in order to stay competitive in the global marketplace. A higher level of collaboration can facilitate immediate reaction to current business events, as well as lead to considerable cost savings and improved organizational efficiency. e-Business Integration can be used to integrate relevant target and source systems to allow data, services, processes, and so forth to be shared, in order to support the establishment of this higher level of collaboration between the members of the extended enterprise. Despite the benefits that e-Business Integration can bring in supporting the establishment of an extended enterprise, business leaders are hesitant to invest in more technology solutions during slow economic times. The problem being addressed in this study relates to introducing business leaders to the concept of an extended enterprise, and then providing relevant information that can be understood by non-technical business leaders

on how e-Business Integration can support the establishment of an extended enterprise. Once business leaders understand the importance of e-Business Integration in the establishment of an extended enterprise it is necessary to provide them with an approach that they can work towards in the creation of an extended enterprise. This will typically involve achieving intra-enterprise integration first, and then advancing to inter-enterprise integration.

Effectively addressing the above problem would ensure that the objectives in section 1.5 are achieved.

1.5 OBJECTIVES

The principal objective of this study is to provide business leaders with an approach that can be followed when implementing e-Business Integration to support the establishment of an extended enterprise. In addition, it is also necessary to introduce business leaders to the concept of an extended enterprise and highlight the supporting role that e-Business Integration can play in establishing an extended enterprise. It is necessary to research and formulate the concept of an extended enterprise, as well as the concepts of intra- and inter-enterprise integration, which collectively forms the e-Business Integration market, in order to develop such an approach. In addition, it is necessary that these business leaders have a basic understanding of the various integration approaches that can be applied to integrate systems, as well as the enabling middleware technologies and standards employed in these integration approaches.

The following methodology will be followed in order to reach these objectives.

1.6 METHODOLOGY

An extensive literature study will form the basis of the research. The problem will be examined by discussing the following topics:

- the concept of an extended enterprise
- e-Business Integration, including Enterprise Application Integration (EAI) and Business-to-Business Integration (B2Bi)
- integration approaches, and
- integration technologies and standards.

Using the knowledge gained during this study, a model representing the phases that integration initiatives should undergo to establish integrated communication between the members of the extended enterprise will be developed.

1.7 LAYOUT OF DISSERTATION

Chapter 1 – Introduction

Provides some background information on the research area in order to define the problem.

Chapter 2 – The Extended Enterprise

This chapter will discuss the concept of an extended enterprise.

Chapter 3 – e-Business Integration

In this chapter intra- and inter-enterprise integration, which collectively form the e-Business Integration market, will be discussed.

Chapter 4 – Integration Approaches

The various integration approaches that can be employed to achieve either intra- or inter-enterprise integration will be described.

Chapter 5 – Integration Technologies and Standards

In this chapter several integration technologies and standards that can be employed to develop either intra- or inter-enterprise integration solutions will be discussed.

Chapter 6 – Towards an Extended Enterprise: The Extended Enterprise Integration Model

A model that proposes a phased approach to achieving integration between the systems of the members of the extended enterprise is introduced.

Chapter 7 – Employing the Extended Enterprise Integration Model

In this chapter an example is used to illustrate how the phased approach proposed by the Extended Enterprise Integration Model can be used to achieve intra-enterprise integration.

Chapter 8 – e-Business Integration: Benefits and Barriers

This chapter discusses the role that e-Business Integration plays in supporting the establishment of an extended enterprise, as well as the realities of employing e-Business Integration.

Chapter 9 – Conclusion

The research is concluded in this chapter.

1.8 CONCLUSION

The problems that will be addressed by this research project were defined and the objectives of the project stated in this chapter. The approach taken to solve the problems were discussed and the layout of the dissertation shown.

The concept of an extended enterprise will be discussed in the following chapter.

CHAPTER 2

THE EXTENDED ENTERPRISE

Chapter 1 establishes the necessity of the research documented in this dissertation and introduces the reader to the problem domain.

This chapter introduces the reader to the concept of an extended enterprise and includes a discussion on the difference between an extended- and a virtual enterprise. The extended enterprise is also evaluated from an Information Systems-based viewpoint.

In Chapter 3 e-Business Integration will be discussed and the role that it can play in solving communication problems between the members of the extended enterprise will be highlighted.

2.1 INTRODUCTION

The evolution of the Internet has a significant impact on the way that organizations operate. According to Kalakota and Robinson the, evolution of the Internet can be divided into phases (2000). During the first phase (1994 – 1997) the emphasis was on creating a web presence, while the second phase (1997 – 2000) focused on performing transactions over digital media. Today, with the Internet era in its third phase (2000 - ?), the focus is on how the Internet can impact profitability. With the abundance of product information available on the Internet, consumers can compare prices and services with the click of a button before making a purchasing decision. This leads to increased competition and shrinking profit margins. This phenomenon forces organizations to investigate other alternatives for increasing profits. Organizations are looking beyond their traditional boundaries to form dynamic relationships with the members of their extended enterprises in order to remain competitive and exploit profitable opportunities in a volatile economy (Power, Sohal & Rahman, 2001).

The *extended enterprise* is a term used to describe the collaboration and high level of interdependence that exists between an organization and its customers, suppliers and other business partners (Browne & Zhang, 1999). Competitive advantage not only involves internal efficiency in the extended enterprise, but the ability of the organization to share information and coordinate activity and requirements with the members of its extended enterprise as well. Dramatic savings of time and money can be realized through collaboration between the members of the extended enterprise (Nickerson, 2002). The real competition is no longer between individual organizations in current market conditions, but rather between extended enterprises (Alshawi, 2001; Surgency & IBM, 2002).

Consider the following example of an extended enterprise involved in the supply of potato chips, in order to explain the concept of an extended enterprise (adapted from (Alshawi, 2001)).

Even though there could be quite a number of organizations working together to manufacture potato chips, the most prominent members of the extended enterprise would include the following:

- the farmer who grows the potatoes
- the package manufacturer who supplies the bags that the potato chips are sold in
- the actual manufacturer of the potato chips
- the trucking company that delivers the product to supermarkets, and
- the supermarket where the product is sold.

Collaboration between these members of the extended enterprise is critical to its success as a whole. The farmer needs information about potato chip sales projections when planning his planting and harvesting of a potato crop, while the actual potato chip manufacturer needs information about crop conditions to plan his production schedule (Alshawi, 2001).

The above description of an extended enterprise may, to many people, seem to describe a virtual enterprise as well, but there is a subtle difference between an extended enterprise and a virtual enterprise. The difference between an extended enterprise and a virtual enterprise will be discussed in the next section.

2.2 EXTENDED AND VIRTUAL ENTERPRISES

There is a difference, which primarily relates to the purpose of the enterprise, while its description seems to relate to a virtual enterprise as well. The focus is on long-term relationships across the value chain in the extended enterprise, to enable the members of the extended enterprise to eliminate costs, streamline business processes, enhance mutual customer benefits and thus to gain strategic competitive advantage. The virtual enterprise signifies a more short-term, dynamic environment where relationships between organizations exist for a relatively short time to satisfy certain market demands quickly (Browne & Zhang, 1999). A virtual enterprise may be established to produce a particular product or to deliver a particular service and when the market for that product or service deteriorates it is dissolved.

Table 2.1 presents a further comparison between an extended enterprise and a virtual enterprise.

	EXTENDED ENTERPRISE	**VIRTUAL ENTERPRISE**
STRATEGIC ISSUE	Stronger long-term objective	Stronger short-time objective
MEMBERSHIP PURPOSE	Long-term business co-operation	Temporary co-operation to deliver new product or service
ORGANIZATIONAL STABILITY	Stable grouping of organizations across the product value chain	Dynamic grouping of organizations with core competences
MEMBER RELATIONSHIPS	Trust and mutual dependence for long term	Temporary and dynamic
BOUNDARIES	Full blurring for long term	Partly blurring for short term
ORGANIZATION TYPE	Product value-chain based to co-ordinate total product life-cycle	Frequently project or niche market based
CO-ORDINATION OF MEMBERSHIP	Usually the manufacturer	Frequently a broker manages the co-operation
INFORMATION AND COMMUNICATION TECHNOLOGY (ICT)	Facilitated and enabled by ICTs	Operation depends on sophisticated ICTs

TABLE 2.1: A comparison of the extended enterprise and the virtual enterprise (Browne & Zhang, 1999).

The major difference between these two enterprises lies in the relative stability of the extended enterprise versus the temporary and dynamic nature of the virtual enterprise. The requirement in the extended enterprise for organizational stability, trust and lasting relationships is high, while the focus in the virtual enterprise is on the temporary *coming together* of organizations to produce a particular product, or to deliver a particular service, quickly to satisfy a specific market need. Despite the differences mentioned, the similarity between the enterprises lies in the fact that both pursue enterprise partnerships in order to achieve success and remain competitive in a very volatile, competitive marketplace (Browne & Zhang, 1999).

While the success of both enterprises depends on information sharing, the virtual enterprise depends more on information technology to facilitate fast and accurate transactions, while in the extended enterprise the challenge of creating and sustaining it is more managerial, and concerned with the design and implementation of appropriate business processes. Despite this, the efficiency of the extended enterprise is to a great extent determined by the level of compatibility between the Information Systems of the members, as well as how effectively they can leverage these systems to maximize the sharing of information and the coordination of activities (Browne & Zhang, 1999). In the next section the extended enterprise will be discussed from an Information Systems-based viewpoint.

2.3 THE EXTENDED ENTERPRISE: AN INFORMATION SYSTEMS-BASED VIEW

2.3.1 INTRODUCTION

The term *extended enterprise*, as mentioned previously, refers to an organization that not only consists of its employees, board members and executives, but also its customers, suppliers and other business partners. In this network of relationships, traditional ways of doing business such as e-mail, faxes and voice mail introduce delays and require frequent re-entry of data into systems (Active Software, 2000). These delays and the opportunities created for errors to slip in when re-entering data are unacceptable in the extended enterprise where the need for speed, accuracy and detail is particularly high. This calls for the creation of a network that is accessible, integrated, scalable and resilient through the integration of inward-facing applications with applications that provide outward-facing connectivity to the other members of the extended enterprise (Stokes, 2001[2]).

Figure 2.1 provides an Information Systems-based view of the extended enterprise (Cherry Tree, 2000). The presentation of Information Systems in Figure 2.1 is merely an indication of the three major systems used to establish collaboration between the members of the extended enterprise and is by no means a representation of all Information Systems in the enterprise.

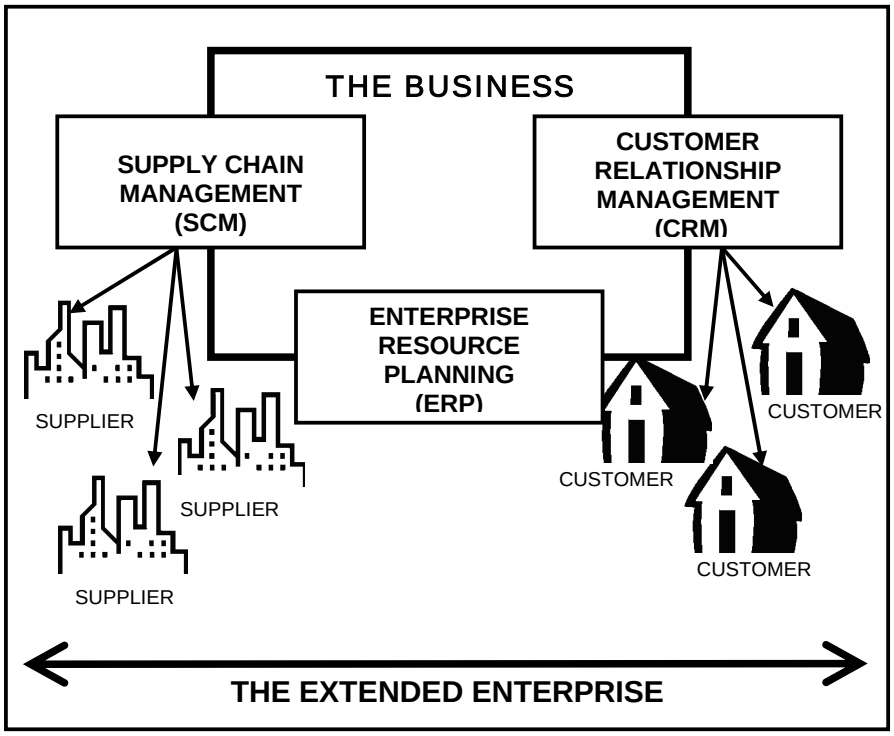

FIGURE 2.1: The extended enterprise - an Information Systems-based view.

The core of the organization is formed by primarily inward facing applications that track the internal flow of information. These applications include the Enterprise Resource Planning (ERP) backbone, as well as other core accounting, manufacturing and Human Resources applications. For the creation of a truly *extended* enterprise it is necessary that the Information Systems of the organization face outward to enable collaboration with other extended enterprise members. Applications that enable the organization to collaborate with its customers, suppliers and business partners include Customer Relationship Management (CRM) and Supply Chain management (SCM) systems (Cherry Tree, 2000). To ensure the uninterrupted flow of information and transactions, the core applications must be seamlessly integrated with the

applications that provide connectivity to other members of the extended enterprise (Stokes, 2001[1]; DataMirror, 2002; Stokes, 2001[2]; Cherry Tree, 2000). It is necessary, in order to achieve this, to integrate the inward facing applications with the outward facing applications through the use of intra-enterprise integration, which will be discussed in Chapter 3.

Following is a brief discussion of the three main types of Information Systems that contribute to the establishment of the extended enterprise (as highlighted in Figure 2.1).

2.3.2 ENTERPRISE RESOURCE PLANNING (ERP)

ERP systems were designed to integrate and track the internal flow of information in an organization. An ERP system can include modules for manufacturing, order entry, accounts receivable and payable, general ledger, purchasing, warehousing, transportation and human resources. ERP systems can be viewed as the backbone of the organization to which customer and supplier-facing applications, such as Customer Relationship Management (CRM) and Supply Chain Management (SCM) systems, may be linked to enable communication between the organization and other members of the extended enterprise (Cherry Tree, 2000).

2.3.3 CUSTOMER RELATIONSHIP MANAGEMENT (CRM)

CRM involves an integration of technologies *and* business processes to satisfy the needs of customers. It usually

involves the organization-wide acquisition, analysis and use of knowledge about customers to efficiently sell more products and services, while ensuring customer satisfaction (Bose, 2002). The two primary goals of a CRM system are to (Cherry Tree, 2000):

1. enable the organization to more effectively identify, contact, and acquire new customers, and to
2. leverage existing customer relationships.

2.3.4 SUPPLY CHAIN MANAGEMENT (SCM)

SCM systems are designed to link suppliers, manufacturers, distributors, and resellers to track and streamline the flow of materials and data through the manufacturing and distribution process. SCM systems improve production efficiency and flexibility by enabling greater data sharing between supply chain partners. The three primary goals of SCM systems are to (Cherry Tree, 2000):

1. decrease inventory costs by matching production to demand
2. reduce overall production costs by streamlining the flow of goods through the production process and by improving information flow between an organization, its suppliers, and its distributors, and to
3. improve customer satisfaction by offering increased speed and adaptability.

2.3.5 CONCLUSION

Since the extended enterprise is only as strong as its weakest member, it is important that the organization achieves intra-

enterprise integration before collaborating with other extended enterprise members (Active Software, 2000). Once this is achieved the organization can look beyond its firewall to the rest of the extended enterprise for new opportunities to eliminate costs and gain strategic competitive advantage (Mohindroo, 2002). In today's global, volatile economy the organization must collaborate more directly with members of the extended enterprise and respond more rapidly and intelligently to change (IBM, 2000). Inter-enterprise integration, which involves the integration of business-to-business applications with internal applications, is necessary to enable this level of collaboration (Mercator, 2000). Where intra-enterprise integration involves the integration of internal applications, inter-enterprise integration involves the integration of applications with external customers, suppliers and business partners (Microsoft, 2002). Intra- and inter-enterprise integration will be addressed in Chapter 3.

2.4 CONCLUSION

Organizations are no longer just competing against other organizations to deliver better services or products. In the global marketplace the focus has shifted to how effective they are at sharing information and business processes with customers, employees, and business partners to gain competitive advantage over the competition (Sturim, 1999; Information Builders, 1999). In a White Paper published by Surgency Inc., Bechmarking Partners and IBM, this trend is compared to the FIFA World Cup. All teams competing in the World Cup are made up of excellent soccer players, but in the end it is the team that can work together as one cohesive unit the best, that win the FIFA World Cup. No matter how great a single player is, he can not win the

World Cup alone. He can only win if he is part of a team that supports him, and if he in turn supports the other members of his team (Surgency & IBM, 2002). The same applies in the business world.

The success of an organization is no longer just determined by internal efficiencies, but also by the capability of their trading partners. A manufacturer competing in rapidly changing market conditions will be eliminated from the game very quickly if his suppliers have poor delivery records or deliver products of inferior quality. The limitations of trading partners will have a profound impact on the success, and in some cases even the survival, of the individual organization (Power, Sohal & Rahman, 2001).

Organizations that spent the greater part of the past decade purchasing expensive Enterprise Resource Planning (ERP), Customer Relationship Management (CRM) and e-Commerce applications, are now focusing on integrating these information silos, not only to improve internal efficiency, but also to facilitate collaboration with the members of their extended value chain (Oracle, 2002; Lawlor, 2003). Collaboration between an organization and its extended value chain provides opportunities to eliminate costs and gain strategic competitive advantage and has given rise to the concept of an *extended enterprise* where the organization is connected to its business partners, suppliers and customers.

The need for speed, accuracy, and detail is particularly high in the extended enterprise and the ability to meet these needs depends heavily on the ability to integrate and exploit all its information assets. In order for the extended enterprise to be successful, it is important that relevant, up-to-date information is available to all participants at all times.

While Figure 2.1 provides a simplified Information Systems-based view of the extended enterprise, there are a magnitude of other operational systems within the organization, which may include legacy-, manufacturing-, inventory-, shipping-, financial- and sales systems, to mention a few. It is estimated that the average Global 2000 organization consists of approximately 52 enterprise applications (META Group, 2001). Considering this, and taking into account that these applications must be successfully integrated within the organization before proceeding to integrating with other members of the extended enterprise, it becomes clear that establishing collaboration between the Information Systems of the members of the extended enterprise can be a complex task. In the next chapter the issues of intra- and inter-enterprise integration, which collectively forms the e-business integration market, are addressed.

CHAPTER 3

e-BUSINESS INTEGRATION

Chapter 2 describes the concept of an extended enterprise and establishes the importance of collaboration between an organization and its customers, suppliers and other business partners, collectively known as an extended enterprise.

In this chapter e-Business Integration, which includes intra- and inter-enterprise integration will be discussed. The role that intra- and inter-enterprise integration can play in solving communication problems between the members of the extended enterprise will also be highlighted.

Chapter 4 will describe various integration approaches that can be followed to achieve intra- and/or inter-enterprise integration.

3.1 INTRODUCTION

It was established that the value of an extended enterprise is to a great extent determined by the level of compatibility between the Information Systems of the members of the enterprise, as well as by how effectively these systems can be leveraged to maximize the sharing of information and the coordination of activities (Chapter 2). Through integration, both within the organization and between the members of the extended enterprise, business processes and data can be shared among connected applications and data sources, eliminating stovepipe applications, inconsistent data and inefficient business processes (Sun, 2002).

Until recently, most integration projects have focused on *intra*-enterprise integration, or application integration that occurs within the firewall of the organization (Linthicum, 2000). This is known as Enterprise Application Integration (EAI). *Inter*-enterprise integration has become a priority with the emergence of extended enterprises. This involves the integration of applications with external business partners, suppliers and customers across the firewall. Integration across the firewall is also known as Business-to-Business Integration (B2Bi) (Linthicum, 2000).

The emergence of the extended enterprise has not only necessitated B2Bi, but it has also led to the merging of the EAI and B2Bi markets to form the e-Business Integration market. Even though there is a magnitude of definitions for the term *e-business*, it is generally agreed that e-business enables collaboration between the organization and all of its relevant stakeholders. According to Rodgers, Yen and Chou, e-business uses “common electronic data standards with computer automation technology to electronically interconnect Information

Systems, integrate internal and external data streams, and automate business processes between trading partners" (2002). Even though Electronic Data Interchange (EDI) technology can be used to exchange data between organizations, it is a complex technology that does not provide process integration or the flexibility and maintainability that the extended enterprise demands (Themistocleous & Irani, 2001; IBM, 2000; Harris & Cantrell, 2002).

Since the extended enterprise is only as strong as its weakest member, it is important that the organization becomes more responsive to changing market conditions and customer needs and that timely, accurate information is available when needed to make critical business decisions (Power, Sohal & Rahman, 2001; Duffy, 2002; Evgeniou, 2002). This higher level of responsiveness required by organizations in current market conditions is driving the need for organizations to become more *agile*. The concept of an agile enterprise is discussed in the following section.

3.2 THE AGILE ENTERPRISE

An organization must be more than just efficient, effective, lean, customer-focused, able to add value, quality driven, and proactive rather than reactive to be considered *agile*. It must be able to respond to changes in demand in terms of both volume and variety and exploit profitable opportunities without delay, by using market knowledge gathered from all parts of the organization, as well as from members of the extended enterprise (Power, Sohal & Rahman, 2001).

It must be understood that the workforce, processes and technology in an organization have an impact on each other and the point at which agility occurs is where these factors intersect. This point is known as

the core of agility, as shown in Figure 3.1. At the core of agility is the knowledge gained by understanding the impact that the workforce, processes and technology have on each other (Duffy, 2002).

The core of agility represents the ability of the organization to deal with complexity, access and to share relevant information and knowledge, as well as its ability to sense and respond to the other members of the extended enterprise (Duffy, 2002; Power, Sohal & Rahman, 2001). In order for the technology factor of the organization to be adaptive and to contribute to the agility core, it is necessary that the organization link together internal applications and the internal business processes that interact with them, as well as mapping data between the various applications. This can be achieved through intra-enterprise integration (BEA Systems Inc., 2002; Evgeniou, 2002).

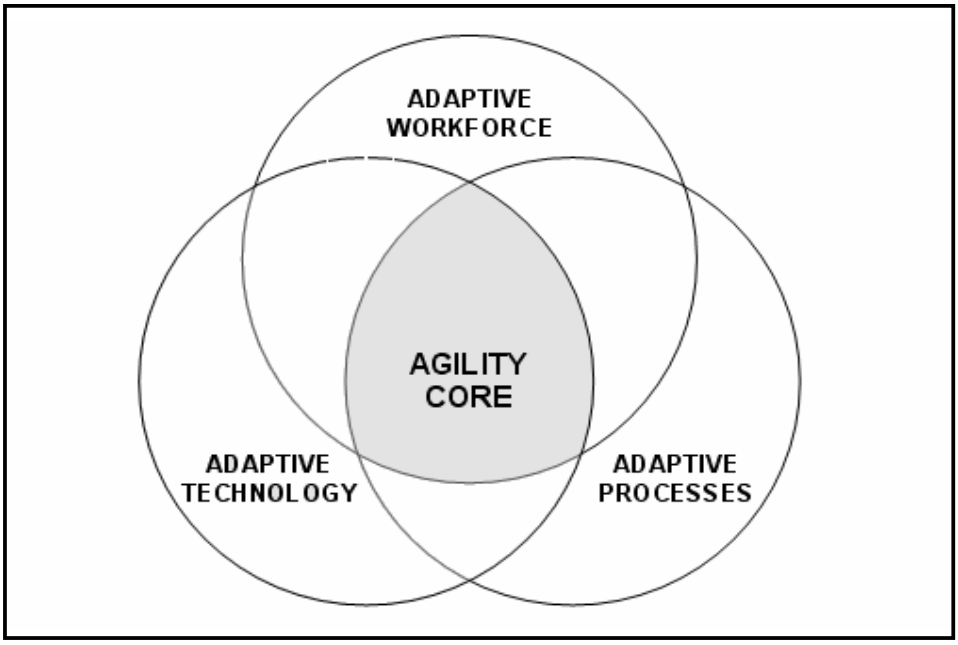

FIGURE 3.1: The core of agility (Duffy, 2002).

3.3 INTRA-ENTERPRISE INTEGRATION

Over the years organizations have made enormous investments in acquiring and deploying various applications to support specific business functions. They include custom-built applications and packaged best-of-breed applications (Microsoft, 2002; Linthicum, 1999). These individual applications support specific business functions and thus remain valuable to the organization but, as stated previously, communication between them has become increasingly important. They operate in isolation and are referred to as “application silos” and there are various reasons for their existence.

Separate applications have been acquired, and in various cases developed with the specific needs of the organization in mind, at different times and from different vendors to support specific business functions. In addition to this, various applications might have been introduced as a result of mergers and acquisitions. Many of these applications use incompatible and non-standard data storage and application development technology. All these factors contribute to the lack of communication between the various applications within the organization (Stokes, 2001[1]; Bussler, 2003; Linthicum, 1999). There is always the option of replacing them all with applications that use compatible technology, but the cost of doing so is generally prohibitive, especially when considering the fact that, when evaluated independently, they perform satisfactorily (Linthicum, 1999; Microsoft, 2002).

Bridging the communication gap between applications through manual processes where the output of one application is sorted, processed and input into another application, presents the organization with two challenges (Microsoft, 2002):

- Firstly, when employees spend time collecting and shuttling data between departments and then re-entering that data into other applications, it results in excessive administrative and operational costs. Unnecessary time and resources are consumed and the opportunities for human error are endless.
- Secondly, the impact on the organization is a lack of strategic advantage over its competition. When making use of manual processes it is harder to consolidate data from all areas of the organization to support faster and more insightful business decisions. Responding to competitive threats in a timely manner, and changing business processes to support new programs for customers and partners, also becomes increasingly difficult. Basically, the existence of application silos slows down business, drives up costs, and stands in the way of strategic goals.

Instead of using inefficient manual processes to bridge the communication gap between applications within the enterprise, Enterprise Application Integration (EAI) could be used (Linthicum, 1999; Microsoft, 2002; Gupta, 2003; Brown, 2001).

The EAI Industry Consortium defines Enterprise Application Integration (EAI) as “the process of integrating multiple applications that were independently developed, may use incompatible technology, and remain independently managed” (2003[2]). Linthicum defines EAI as the “unrestricted sharing of data and business processes among *any* connected applications and data sources in the enterprise” (Linthicum, 1999). According to Gupta EAI enables “multiple applications within an organization to share information and functionality to automate business processes” (Gupta, 2003). In essence, the goal of EAI is to ensure that the applications within the organization can communicate and share information in the most efficient way possible, which means

setting up the applications to send, receive, and react to information (Acharya, 2003; Mann, 1999).

EAI enables previously isolated systems within an organization to be integrated with related information and processes in other systems, thus enabling the organization to integrate valuable legacy systems with newer technologies that will be important for current and future operations. EAI also allows the organization to facilitate flexible data access and flow for ever-changing business needs, while also allowing the incorporation of any unstructured data for business use (Alpha Technologies, 2003). An important feature of EAI is the ability to integrate various applications that exist in application silos without having to replace, or make significant changes to applications that are performing satisfactorily (Gupta, 2003; Linthicum, 1999).

Once a satisfactory level of intra-enterprise integration is achieved the organization can integrate with other members of the extended enterprise through inter-enterprise integration. While many organizations strive towards both intra- and inter-enterprise integration, it is usually ideal to accomplish intra-enterprise integration before progressing to inter-enterprise integration for two reasons (Microsoft, 2002):

1. It is technically easier to facilitate communication between internal and external applications if internal applications are already integrated.
2. The returns on investment (ROI) of intra-enterprise integration projects are usually higher than inter-enterprise integration projects because of the remarkable and predictable cost reduction that is made possible by automating highly manual internal processes.

Intra-enterprise integration thus becomes the fundamental building block from which organizations continue to develop their more comprehensive inter-enterprise integration strategy.

3.4 INTER-ENTERPRISE INTEGRATION

Inter-enterprise integration extends the internal data and processes of the organization by integrating internal applications with business-to-business applications, enabling collaboration between the organization and other members of the extended enterprise (BEA Systems Inc., 2002; Mercator, 2000). Inter-enterprise integration can be achieved through the use of Business-to-Business Integration (B2Bi).

Where EAI involves the integration of internal applications, B2Bi involves the integration of applications with external customers, suppliers and business partners (Microsoft, 2002). B2Bi describes the set of software and processes required to enable the applications of different organizations to communicate and exchange data electronically. The Internet is typically used as the communication medium and although communication usually occurs without human intervention, human-to-machine and human-to-human communication is also possible (Morgenthal, 2001; Schulte, 2001). Even though Electronic Data Interchange (EDI) technology can be used to exchange data between organizations it is a complex technology that does not provide process integration or the flexibility and maintainability that the extended enterprise demands (Themistocleous & Irani, 2001). Even though B2Bi is a direct outgrowth of EAI, reusing many of the same approaches and technologies, there are some approaches to integration that occur within the firewall that are not transferable to integration initiatives that reach beyond the organization’s firewall.

3.5 EAI + B2Bi = e-BUSINESS INTEGRATION

The EAI and B2Bi markets are merging to form the e-Business Integration market (Aberdeen, 2003). A significant driver for this situation is the fact that successful business-to-business (B2B) communication requires the integration of internal systems (Carl, 2003). Organizations usually prefer to attain intra-enterprise integration first, before progressing to inter-enterprise integration. As mentioned previously, it is easier to integrate with external customers, suppliers and business partners if internal applications are already integrated. This merger of the EAI and B2Bi markets allow organizations to derive value from their integration efforts by using EAI as the building block upon which they can develop their long-term B2Bi strategy.

3.6 CONCLUSION

In the past organizations faced the daunting task of integrating all relevant applications, data and processes within their firewall to stay competitive, but with the emergence of the extended enterprise organizations now also need to integrate their applications, data and processes with those of other members of the extended enterprise. Organizations with solid intra-enterprise integration architectures find it easier to obtain the potential benefits from inter-enterprise integration efforts, resulting in organizations using EAI as the fundamental building block from which to develop their more comprehensive B2Bi strategies.

EAI and B2Bi are not simply applications that an organization can install to integrate applications, processes and data, but rather sets of reusable integration tools and services that can be combined in different ways to address specific integration issues related to the

requirements of the organization and other extended enterprise members (Microsoft, 2002). Each integration solution will require different approaches to sharing data, processes and application services between applications both within and beyond the organization's firewall. Even though approaches to application integration differ to a great extent, it is possible to create some general categories, which will be discussed in the next chapter.

CHAPTER 4

INTEGRATION APPROACHES

Chapter 3 describes the concepts of intra- and inter-enterprise integration, which collectively forms the e-Business Integration market.

Various integration approaches that can be followed to achieve intra- and/or inter-enterprise integration will be described in this chapter.

Several integration technologies and standards that can be employed to develop either intra- and/or inter-enterprise integration solutions will be discussed in Chapter 5.

4.1 INTRODUCTION

The integration issues that need to be addressed in each organization and extended enterprise vary considerably and there is no single integration solution available that solves all integration issues. Instead, there are several integration approaches that can be followed to share data, processes and application services between applications. Application integration approaches generally include the following (Hendrick & Hendrick, 2002; Jhingran, Mattos & Pirahesh, 2002; Christos, 2002; Vander Hey, 2000; Wipro, n.d.; Shi & Gandhi, 2001):

- Data-oriented integration
- User interface-oriented integration
- Application interface-oriented integration
- Service-oriented integration, and
- Process-oriented integration.

Certain integration approaches offer a more invasive integration solution, meaning that it is necessary to change the applications involved to allow the sharing of data, processes and application services, while other approaches are non-invasive (Raj & Ishii, 2004). In literature this is referred to as coupling versus cohesion, where coupled approaches could be either tightly or loosely coupled (Wagner, 2002; Linthicum, 2003[2]).

Coupling requires changes to source and target systems. Applications are bound in such a way that they are dependant on each other and any future changes to either application will require changes to the coupled applications as well. This makes the integration solution very vulnerable, since the failure of one application could bring down all coupled applications. The degree of coupling can vary from loosely- to tightly coupled, where loosely coupled applications are more

independent than tightly coupled applications. By loosely coupling applications the impact of changes to coupled applications are reduced. The service-oriented integration approach is an example of a coupled integration approach where applications are bound by shared services, as opposed to the simple exchange of data (Wagner, 2002; Linthicum, 2003[2]).

Cohesive integration solutions are non-invasive, where source and target systems are independent of each other. Changes to applications should not affect other applications directly. Cohesive integration solutions typically only provide simple data movement, making it unsuitable for instances where access to application services are necessary (Linthicum , 2003[2]).

When selecting an integration approach and enabling technologies it is very important to make sure that the integration problem faced is fully understood to ensure that the most appropriate integration solution is chosen. It is unnecessary to leverage an invasive service-oriented integration approach when simple data-oriented integration will satisfy the integration requirements. In the remainder of this chapter the five afore-mentioned integration approaches will be discussed.

4.2 DATA-ORIENTED INTEGRATION

Data-oriented integration deals with the simple exchange of data between target and source systems. Typically, data is extracted from the source, processed if necessary, and then updated in the target (Linthicum, 2003[1]). There are a number of potential scenarios, including (Bloor Research, 2004):

- single source, single target
- single source, multiple targets

- multiple sources, single target, and
- multiple sources, multiple targets.

A typical example of a single source, single target scenario would be moving data from one database to another, as seen in Figure 4.1.

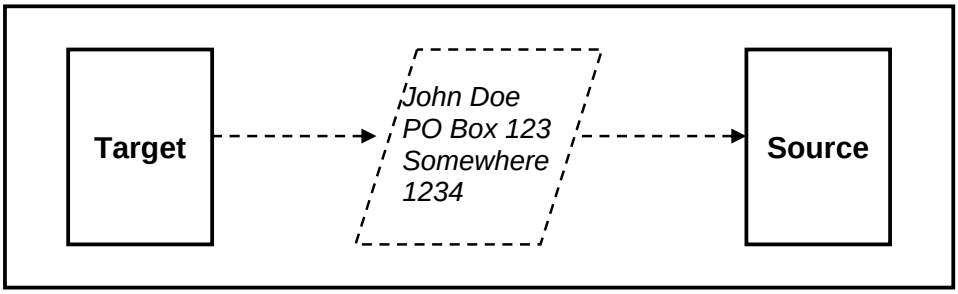

FIGURE 4.1: Data exchanged between a single target and source system.

The single source, multiple targets scenario might be applicable in a scenario where it is necessary to move data from a centralized implementation to a distributed one (see Figure 4.2).

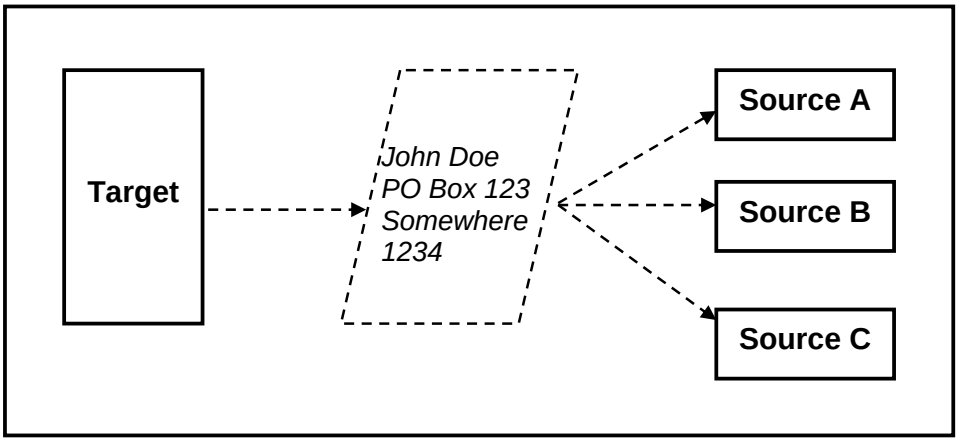

FIGURE 4.2: Data exchanged between a single target and multiple source systems.

Figure 4.3 illustrates the multiple sources, single target scenario, where an example would be consolidating multiple databases, or loading a data warehouse.

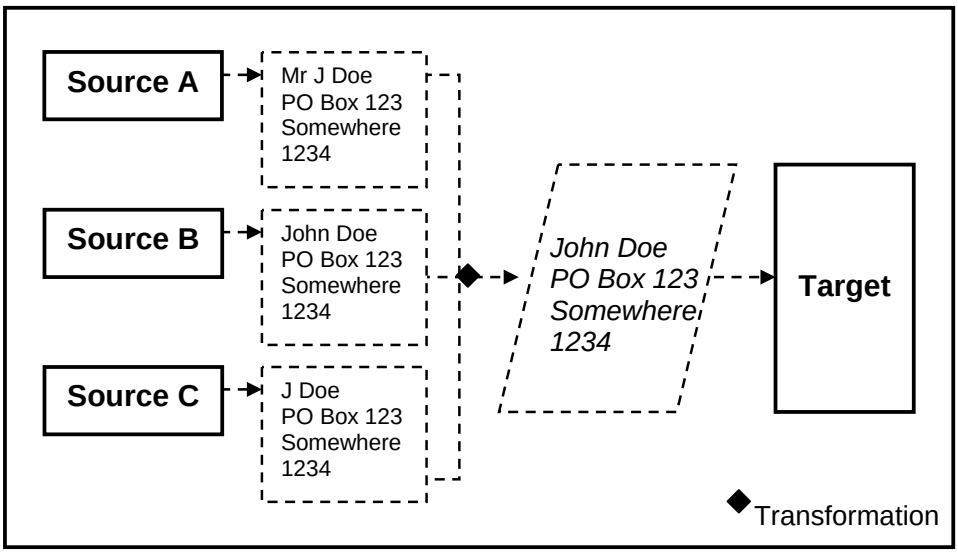

FIGURE 4.3: Data consolidated from multiple sources to a single target.

Lastly, the multiple sources, multiple targets scenario could be relevant in cases where it is necessary to extract data from multiple sources, consolidate the data and then present it to multiple sources, as illustrated in Figure 4.4.

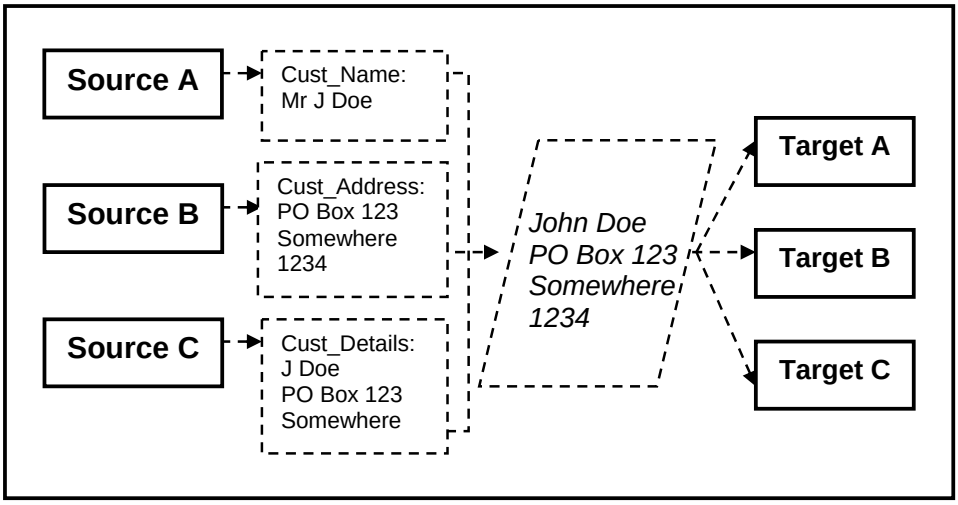

FIGURE 4.4: Data consolidated from multiple sources to multiple targets.

Data-oriented integration is considered to form the foundation for future application integration initiatives, not only because it is the most mature integration approach, but also because it is the most straightforward (Linthicum, 2003[2]; Hendrick & Hendrick, 2002; Fischer, 2001; Vander Hey, 2000; Christos, 2002). Despite the maturity and straightforwardness of this integration approach it is not a simple integration solution. Consider extracting data from as many as one hundred databases and several thousands of tables and then the possible transformation and application of business logic to the data before updating the target system (Linthicum, 2003[1]; Linthicum, 2003[2]). Complicating matters further is the heterogeneity of data. In addition to data in well-defined tables enterprises must also deal with unstructured content such as text, audio and video. Examples of text include e-mails and web pages, while audio might be call centre logs and video could include employee broadcasts, security camera footage and so forth. The fact that data is distributed over multiple locations in different organizations and questions about who owns and controls the data and access to the data further complicates data-oriented integration approaches (Jhingran, Mattos & Pirahesh, 2002).

What makes data-oriented integration solutions so prevalent is the relevant low cost of these solutions. Since this integration approach rarely, if ever, require any significant changes to the source or target systems there are no costs related to changing code, testing and deploying new versions of applications (Linthicum, 2003[1]). If the objective is the simple sharing of data then this integration approach is very effective in the extended enterprise where it is not always possible to make changes to applications that are beyond the control of one particular partner of the extended enterprise (Vander Hey, 2000; Linthicum, 2003[1]). Depending on the integration needs, other integration approaches might prove more valuable in the long run, but data-oriented integration provides a useful solution when integration needs are light and it is consistent with the *baby-step* approach most organizations are comfortable with when implementing integration solutions. Application integration initiatives are generally executed in a series of small, low-risk steps and data-oriented integration is often used as the first step on the road to application integration (Vander Hey, 2000; Linthicum, 2001).

There are various approaches to achieve data-level integration, which include (Shi & Gandhi, 2001):

- Extract, transform, and load (ETL)
- Data replication, and
- Data federation.

4.2.1 EXTRACT, TRANSFORM, AND LOAD (ETL)

Extract, transform, and load (ETL) tools transfer data from a source to a target system by performing three separate functions. Firstly the *extract* function reads the desired data from the source database. Next, the *transform* function works

with the extracted data to convert it to the desired state by using rules, lookup tables, or by combining it with other data. Finally, the *load* function writes the resulting data to the target database (Shi & Gandhi, 2001; De Montcheuil, 2004; TechTarget, 2003; Webopedia, 2001; Wagner, 2004).

ELT usually involves large amounts of data and could be used to (De Montcheuil, 2004; Webopedia, 2001; Wagner, 2004):

- acquire a temporary subset of data for a report or specific business application
- populate data warehouses, data marts or other operational data stores
- convert data from one database platform to another, and
- transfer data from one database to another.

ETL enables organizations to better leverage their information to make more informed business decisions and improve productivity (Wagner, 2004). A drawback of ETL is the fact that data is only as up to date as the extract and load, and thus not real-time (Shi & Gandhi, 2001).

4.2.2 DATA REPLICATION

Data replication involves the movement of data from one database to another database (or databases) in such a way that the data is consistent across all databases (see Figure 4.5) (Linthicum, 2001; Hopkins & Thomas, 1998; Laftsidis, 2004; Informix, 1998). These databases may be provided by the same or different vendors, and might even employ different database models. In such cases it is not sufficient to just move data from Database A to Database B – it is

necessary to transform the data based on business rules so that it is presented correctly to the target database(s) (Shi & Gandhi, 2001; Laftsidis, 2004).

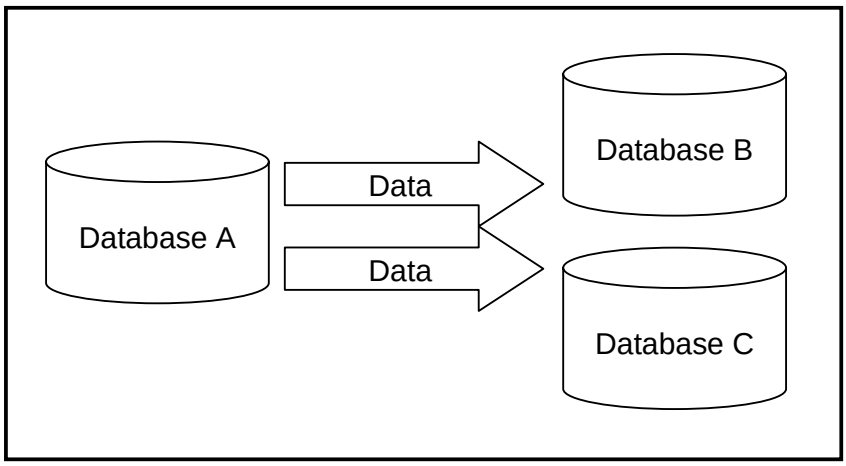

FIGURE 4.5: Data moved from one database to another.

There are two types of data replication, namely asynchronous and synchronous replication. With synchronous replication all replicated data is immediately updated when the source data is updated, thus all copies of data are kept exactly synchronized and consistent. Asynchronous replication, on the other hand, involves a delay between the modification of the source database and the update of the target database(s). This delay could be seconds, minutes, hours or even several days and the copies of the data will thus be temporarily inconsistent, but will eventually be synchronized in all databases (Informix, 1998; Hopkins & Thomas, 1998).

Data replication ensures availability of current data where and when needed. It is inexpensive and easy to implement since most relational database vendors provide replication services in their product offerings and there are various middleware

solutions on the market that offer replication services as well (Microsoft, 2004[5]; Informix, 1998; Linthicum, 2001).

4.2.3 DATA FEDERATION

Data federation allows the integration of multiple databases with diverse brands, models and schemas by creating a single, unified view of the databases. This is achieved by creating a virtual database that consists of many real physical databases. The virtual database allows applications to access data from the underlying databases using a single interface (see Figure 4.6) (Shi & Gandhi, 2001; Linthicum, 2001; Laftsidis, 2004).

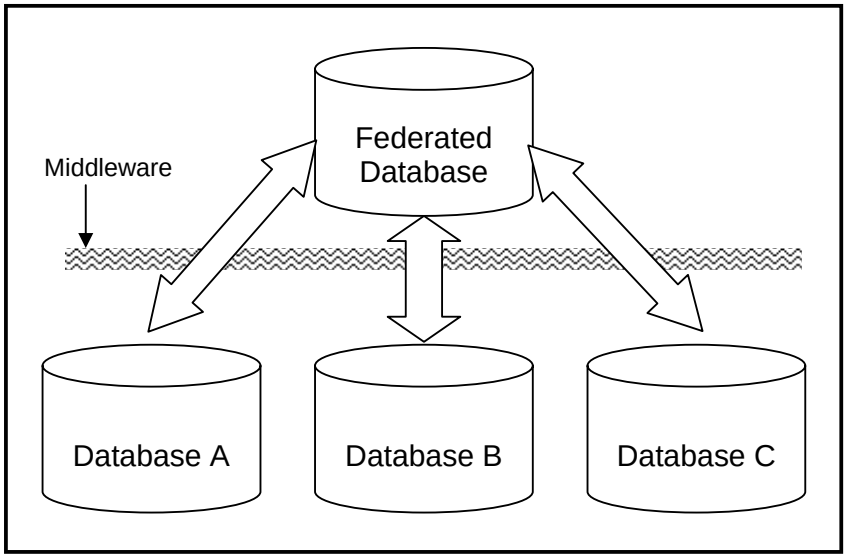

FIGURE 4.6: A virtual database creates a single, unified view of underlying databases.

This approach relies on database middleware to share data by placing a layer of middleware between the physical distributed databases and the applications that consume the data. Unlike

data replication, data federation does not require changes to be applied the source or target systems. The virtual database model only exists in software and only data is shared, the application logic that is embedded in the applications is not available to other systems (Shi & Gandhi, 2001; Laftsidis, 2004). The applications use this virtual database to access required data as needed and the database federation middleware handles the collection and distribution of data to the physical databases (Linthicum, 2001).

The advantage of data federation lies in its ability to bind various data types into a cohesive model that allows information exchange and access to any connected database in the enterprise through a single, well-defined interface (Linthicum, 2001).

4.3 USER INTERFACE-ORIENTED INTEGRATION

There are generally two approaches to user interface-oriented integration, namely screen scraping and content aggregation. Screen scraping is the most primitive of all integration approaches and makes use of the user interface as the general point of integration to extract information from screens. Basically, it involves creating an automated program imitating actual user actions – navigating through screens, emulating keystrokes, and reading screens into memory where the data is parsed, reformatted and transported to middleware layers and eventually sent to the target application. It is necessary to define how to get to the appropriate screens, locate the correct information on the screens, read the data from the screens and then process the data (Shi & Gandhi, 2001; Laftsidis, 2004; Linthicum, 1999; Fischer, 2001). Content aggregation is a more modern approach that is used for

collecting content from various sources and presenting it through a common interface, known as a portal. Through the portal, content from multiple sources within the enterprise are externalized and presented within a single interface, typically a web interface, to users (Laftsidis, 2004; Enosys, 2001; Roch, 2002; Linthicum, 2004[2]; Tarantella, 2004).

4.3.1 SCREEN SCRAPING

Even though other integration approaches might be more technologically appealing and more efficient there may be instances where screen scraping is the only solution. It is useful when integrating mainframe applications that cannot be accessed at the data-, method- or business process levels or in instances where original application code cannot be recovered or recompiled. In such cases the application can be accessed through the user interface without modifying the source or target application (Shi & Gandhi, 2001; Laftsidis, 2004; Linthicum, 1999; Bednar, 2002).

Before implementing screen scraping, it is necessary to understand the underlying data storage schema, the application logic, and how the data is presented to the user interface. Information presented to the user interface might be created by the application logic, such as calculated fields, and will thus not always map back to a database. Understanding the application thus requires reading the documentation, and in cases where the application is not well documented the source code will have to be read. Basically, the goal is to be able to trace all information that appears on the screen to the application logic and database schema information. Failure to understand the application will lead to misreading of the user

interface and, as a result, invalid data will be delivered to target applications (Linthicum, 1999). Once the source and target applications are understood, the information can be extracted from the user interface.

There are two basic techniques to extract information from screens, namely static and dynamic extraction. If the static approach is followed it means that information is accessed from a static location on the screen, for example row 5, column 3, position 7. It is thus not necessary to search the screen or employ conditional logic to find the information. This approach works best for applications that never, or rarely, change; otherwise the screen extraction process may extract either the wrong information or no information at all if the position where the data element resided has changed. The static extraction approach provides a straightforward extraction method that simply goes to the predetermined location and extracts the information. The dynamic extraction approach is more sophisticated than the static approach and uses conditional logic to scan the screen for information. Instead of going to a fixed location the screen is searched for a label that represents the required information. The label is then used as a reference point. While static extraction is not recommended for applications that might change, dynamic extraction has the ability to adapt to application changes and react to changes in the structure of the screen. Both these extraction techniques must ensure that only valid data is extracted by employing an error-checking routine that looks for problems, such as character data in numeric data fields, after the data is extracted. (Linthicum, 1999; Laftsidis, 2004). Once the information is extracted it can be presented to the target application as either screen data or screen objects.

When presenting the information as screen data it is presented as a simple stream of text which is parsed, identified, converted, and processed by the program that is responsible for processing the user interface information, for example the message broker or an adapter. Screen information is never tied directly to methods, and while the screens-as-data approach is very simple, there are instances where the fact that the methods that interact with the data are not tracked proves to be a disadvantage. In such cases the screens-as-objects approach is more beneficial (Laftsidis, 2004; Linthicum, 1999).

If information is presented as screen objects the information is transformed to an object and methods are created to manipulate the information. These screen objects can easily be placed into other environments that support objects where they can be dealt with like any other application objects. In addition, developers can extend the capabilities of the objects to meet integration requirements or the requirements of the target application (Laftsidis, 2004; Linthicum, 1999; Fischer, 2001).

While screen scraping does not have good scaling capabilities, it does offer a simple, low risk, low cost solution for instances where other integration approaches are not appropriate (Laftsidis, 2004).

4.3.2 CONTENT AGGREGATION

Content aggregation is a modern technique used to gather content from numerous sources and present it within a

combined interface, known as a portal, typically viewed in a Web browser. As opposed to screen scraping, content aggregation is a better user interface-oriented integration approach if the information consumers are humans and not applications (Laftsidis, 2004; Enosys, 2001). Using this approach, portals allow users to view and access a multitude of both internal and external resources through a single user interface (Enosys, 2001; Roch, 2002). These resources may be existing legacy systems, new packaged systems, custom developed systems or data, which could be structured data such as reports or unstructured data such as e-mails (iStart, 2003).

This approach avoids the back-end integration problem altogether by extending the user-interface of each application to a combined user interface. All participating applications are thus integrated through the common user interface although the applications are not directly integrated (Linthicum, 2003[1]). The portal application connects to each back-end application through a point of integration which could be a user interface, database, or maybe an application server, and must be able to control user interactions, capture and process errors, and control transactions from the user interface all the way to the back-end application (Linthicum, 2003[1]). By doing this the complexity and expense of traditional back-end integration is avoided (Linthicum, 2003[1]).

The use of portals has various advantages, which includes, as mentioned above, the fact that there is no need to integrate back-end systems directly within or between organizations. This results in a low cost, low risk, non-invasive solution, which is typically faster to implement than other integration

approaches. Other advantages include the fact that enabling technologies are mature and there are many portal-oriented integration solutions to learn from (Linthicum, 2003[1]; Laftsidis, 2004; Linthicum, 2004[2]; iStart, 2003).

Disadvantages include the fact that information does not flow in real-time and thus requires human interaction since systems does not react automatically to business events. Security is also a significant concern when data is being extended to users over the Web and the portal deployment also faces a lack of adoption. The portal solution is only successful if it is used by the users (Linthicum, 2003[1]). It must also be remembered that portals themselves are still applications that must be designed, built, and tested like any other application (Linthicum, 2004[2]).

4.4 APPLICATION INTERFACE-ORIENTED INTEGRATION

Application interface-oriented integration involves the use of application interfaces exposed by packaged or custom applications to gain access to business processes and information. The developers of these applications expose the interfaces to provide access to business processes and data contained within the applications. This allows other developers to access the services of these custom or packaged applications without having to make changes to the application or access the database directly, and without the need to invoke the user interface. By using these application interfaces applications can be integrated to share information and business logic and the only limitation developers face are related to the specific features and

functions of the interfaces. Some application interfaces allow access to business logic, some to data, and some to both (Linthicum, 1999; Laftsidis, 2004). Application interfaces are generally referred to as application programming interfaces (APIs).

The primary purpose of an API is to provide a set of functions, for example to get customer information, which can be used by developers. This eliminates the need to program everything from scratch. APIs are well-defined functions that are developed to connect to some sort of resource and allow developers to invoke the services of these resources (Wikipedia, 2004; Linthicum, 1999; Shi & Gandhi, 2001). For example, the function *GetCustomerInformation("1234")* might produce:

Customer Number: 1234
John Doe
PO Box 123
Somewhere
5678

Application interface-oriented integration is especially helpful to integrate packaged applications. Using application interfaces the packaged applications can be integrated with other applications in the enterprise. This is achieved by accessing and extracting business logic and data from the packaged application, transforming it to a format understandable by the target application and finally transmitting it to the target application. There are various types of technology available that can be applied for this purpose, but the most popular is message brokers (Linthicum, 1999; Laftsidis, 2004). When using application interfaces to integrate packaged applications it is useful to know which portions of the application will be accessed. This information is linked to the application architecture of the packaged application.

In general, packaged applications employ one of three application architectures, namely (Linthicum, 1999; Laftsidis, 2004; Raj & Ishii, 2004):

- centralized
- two-tier, or
- three-tier.

The most traditional of the three, centralized architecture, places data, business logic and user interfaces on the same machine, usually a mainframe or large minicomputer, from where it can be accessed by dumb terminals. The two-tier architecture separates the application into three components, namely the user interface, business logic, and data. These three components are distributed over two layers, namely the client and server, which are connected by a network. The client contains the user interface and the server the data. The business logic could be contained on either the client, in which case it is known as a *fat* client, or on the server with the database, in which case the client is known as a *thin* client. Most client/server systems employ the fat client approach. In the three-tier architecture, which is very similar to the two-tier architecture, the business logic is located on an application server which is placed between the client and the database. This implies that the client only deals with user interactions and the database only deals with data processing. Most packaged applications employ the three-tier architecture (Linthicum, 1999; Laftsidis, 2004).

As mentioned previously, the functions and features provided by application interfaces vary remarkably. The level of service provided by application interfaces can be categorized into business services, data services and objects. Business services include interfaces to business logic that exist within the packaged application and it is necessary to understand what each business service does, what the required data for each business service is, and what the expected outcome is. When

invoked, business services carry out preprogrammed functions such as adding a new customer or sales transaction. Business services also protect the integrity of data by providing a virtual gateway to the data contained within the packaged application. When adding a sales transaction directly to the database, for example, integrity controls could be disregarded and this could lead to inaccurate data. This could be avoided by using business services to access the data. Data services on the other hand provide direct access to data. Depending on the features provided, they could provide extraction and/or update capabilities, with or without integrity checks (Linthicum, 1999; Laftsidis, 2004; Raj & Ishii, 2004).

Objects are simply data services and business services joined as objects. Just as in object-oriented development, objects in packaged applications are the encapsulation of data and the methods that act upon the data. This eliminates the concern regarding data integrity since data cannot be accessed without invoking the appropriate method. Unfortunately these objects are normally not standard distributed objects, but rather proprietary objects defined by the packaged application vendors. As a result, these objects might not fit directly into all development environments. Fortunately many packaged application vendors acknowledge the need to expose these objects and are working on initiatives to provide standard object interfaces to existing packaged applications (Linthicum, 1999; Laftsidis, 2004).

Based on the level of services provided, application interfaces can be divided into the following categories (Linthicum, 1999; Laftsidis, 2004; Raj & Ishii, 2004):

- full-service interfaces
- limited-service interfaces, or
- controlled interfaces.

While full-service interfaces provide access to business services, data services and objects, limited-service interfaces typically only allow access to one service. Based on marketing and/or economic decisions from the vendor, controlled interfaces provide a bare minimum of features and functions. These interfaces are very proprietary and closed, offering limited access to business logic and data (Linthicum, 1999).

While packed application interfaces represent the majority of application interfaces there are thousands of additional application interfaces that should be recognized. Major categories of additional interfaces include vertical market application interfaces and application interfaces built into custom applications.

Vertical market application interfaces provide access to industry-specific applications, taking into account the needs of the specific industry. Industries such as industrial, health care, and finance all have very specific needs relating to the way that information is formatted, processed and transported (Linthicum, 1999).

Application interfaces can also be used to integrate custom applications, but unfortunately few custom applications were built with the ability to provide an application interface to other applications. This is because these applications were developed to serve a single purpose – not to share information and business logic with other applications. It is possible to build application interfaces for these custom applications by exposing business processes that already exist within the application. One way of exposing these, is through the user interface which provides access to the business processes. It might be necessary to change the applications in order to provide an application interface and the level of change required can vary greatly. In certain situations it might be necessary to rebuild the application completely –

one reason why some organizations choose less sophisticated integration approaches such as user interface or data-oriented integration (Laftsidis, 2004; Linthicum, 1999).

As mentioned previously, application interfaces vary greatly in the features and functions that they offer, as well as in the depth of services offered. Some applications offer no application interfaces at all. In such cases a user interface-oriented or data-oriented integration approach might be more appropriate. In cases where adequate application interfaces are available the application interface-oriented integration approach offers a simple integration solution that can ensure consistency of business data where access to both business logic and data are provided.

4.5 SERVICE-ORIENTED INTEGRATION

Through service-oriented integration, applications are allowed to share common application functions as well as data. For example, checking that a particular item is in stock is a process that might be performed by various applications both within and between enterprises. By sharing this, and other, application functions, various applications may access each other's application functions without having to rewrite each function within the various applications (Linthicum, 2003[1]; Linthicum, 2003[2]; Laftsidis, 2004; Linthicum, 1999; Cousins& Casanova, 2003).

Firstly, it is necessary to expose these functions as services. These services could be something simple (*getItemPrice*), more complex business functions (*scheduleDelivery*) or even system services (*authenticateUser*) (Channabasavaiah, Holley & Tuggle , 2004). The service-oriented integration approach is probably the most invasive

integration approach since it is necessary to change, and in some cases even rebuild, applications to expose the application functions as services. This makes it a very expensive integration approach as well, because in addition to changing the application it is also necessary to test, integrate and redeploy the application (Linthicum, 2003[2]; Laftsidis, 2004; Linthicum, 2001; Krass, 2003).

Basically, there are two approaches to sharing application services. A shared set of application services can be created and hosted on a central server, such as an application server, or existing application services can be shared using distributed method-sharing technology, such as distributed objects (Linthicum, 2003[2]; Linthicum , 2003[1]; Linthicum, 1999; Laftsidis, 2004; Krass, 2003). By utilizing these approaches composite applications can be created by integrating applications or application functions to create a new application with extended reach and functionality, matching the business processes of the enterprise more closely than any single packaged application could (Ulin, 2002; Krass, 2003). Composite applications are created using Web services.

The Web service standard defines the following, allowing the Web service to reside anywhere and be accessed from everywhere (Linthicum, 2003[2]; Gold-Bernstein, 2003[1]; Global Exchange Services, 2003; Ulin, 2002; Shi & Gandhi, 2001):

- Web Services Description Language (WSDL): A standardized interface describing the capabilities of the service, as well as how it can be accessed.
- Simple Object Access Protocol (SOAP): A standardized communication protocol for invoking Web services in a decentralized, distributed environment.

- Universal Description, Discovery and Integration (UDDI): A standardized repository for registering and discovering Web services described in WSDL.

To utilize Web services it is first necessary to change the relevant application so that it can expose its application functions as Web services – for example it must be able to respond to a SOAP request. Once described in WSDL the Web service can be published in a directory modelled for Web serviced (UDDI). Extensible Markup Language (XML) is a standardized message format, describing data in context by using XML tags to describe what a segment of data represents. SOAP and WSDL both rely on the XML standard's structure and definitions to provide specialized Web services functions (Linthicum, 2003[2]; Ulin, 2002). Web Services are based on open standards, indicating that there are no dependencies on specific platforms, application vendors, access device types, or locations (Ulin, 2002).

Service-oriented integration supports multiple platforms, making it possible for users to access business services and data through various types of end-user devices, including browsers and mobile devices such as pagers, cell phones and PDAs. In addition, development time is also decreased, since multiple developers can work on different services simultaneously. This is because each service is self contained and does not depend upon the state of another service (Gold-Bernstein, 2004).

4.6 PROCESS-ORIENTED INTEGRATION

Process-oriented integration supports the management and execution of common business processes that exist both within and between

organizations. Just about every activity in an organization can be described in terms of a business process (Linthicum, 2003[1]; Britton, 2003). Consider the potato chip manufacturing example used in Chapter 2. The primary business process for the actual manufacturer of the potato chips might be *Manufacture Potato Chips*, which in turn could consist of sub-processes such as *Wash Potatoes*, *Peel Potatoes*, *Cut Potatoes* and so forth. Other essential supporting processes might include *Buy Supplies*, *Take on New Employee*, *Pay Taxes* and various others. While this is a very simplified example, the execution of one process might ripple through the entire organization and trigger the execution of processes in other systems and other organizations as well. The process of placing an order for a book with Amazon.com will lead to the execution of processes to confirm that the book is in stock, and if not to order it from the relevant supplier, and to deal with the payment of the order, picking the book from the warehouse, confirming the shipping address and shipping the book. The process of buying the book will not only involve processes internal to Amazon.com, but also stretch out to, for example, the enterprise handling the payment of the order (PayPal) and the enterprise shipping the product (FedEx). Chris Britton states that “all a business does is operate processes, manage processes, sell and market process deliverables, and plan how to change processes in the future” (2003).

Process-oriented integration supports the execution of business processes by automating the flow of data and the execution of sub-processes to complete the business process (Linthicum, 2003[2]; Shi & Gandhi, 2001; Wagner, 2002). This must be done in the correct order, with accurate information, control sequences, state maintenance, durability, and the ability to handle exceptions (Linthicum, 2003[2]). From the discussion it can be deduced that process-oriented integration basically encapsulates all the other integration approaches by

providing process management at the data, application interface, user-interface and service levels (see Figure 4.7). Various points of integration could be utilized – in some instances the data-oriented approach will be used to gain access to data and in other instances the application interface or user-interface will provide the point of integration (Linthicum, 2003[1]; Linthicum, 2003[2]).

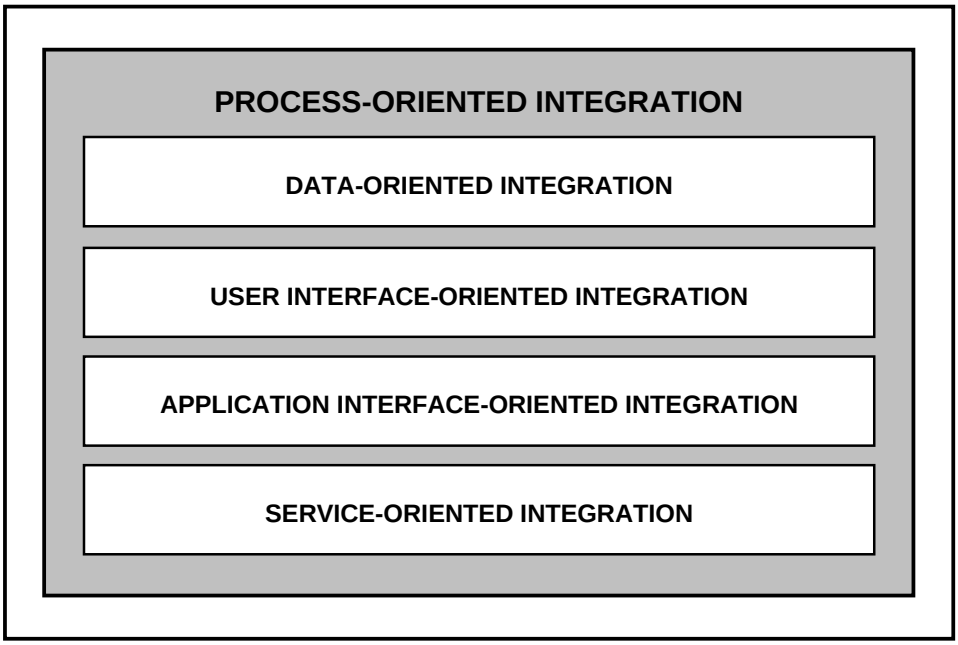

FIGURE 4.7: Process-oriented integration encapsulates other integration approaches.

Through Business Process Management (BPM) business processes are defined and modelled (both automated and manual steps) and the tasks and resources that make up the business process are coordinated through the integration of human and electronic resources within the organization and its network of extended enterprise partners (McDaniel, 2001; Roch, 2002; Gold-Bernstein, 2003[1]; Alodar Systems, 2003). Once the business process is defined, a Business Process Management System (BPMS) enables the organization to model, deploy and manage business processes that span multiple

applications, departments and extended enterprise partners (Smith & Fingar, 2003; Petrassi, 2004). Since it is required that the BPMS model, deploy and manage any business process across any application, department or partner it is necessary that the system meet three important requirements. It must be highly flexible and scalable, and since it manages processes both intra- and inter-enterprise it must also offer advanced security features (Smith & Fingar, 2003).

The process-oriented integration approach compliments the other integration approaches by supporting the flow of data and control logic between relevant processes found both intra- and inter-enterprise. This is achieved by placing a layer of control logic on top of the integration technology, allowing the control logic to bind the underlying applications into a single unified multi-step business process (Linthicum, 2003[2]).

4.7 CONCLUSION

Depending on the circumstances, integration initiatives can implement various integration approaches to solve intra- or inter-enterprise integration problems. These approaches were discussed in this chapter. Integration initiatives are inherently complex and might include a combination of approaches to achieve the desired results. It is not possible to satisfy all integration requirements with a single product or technology and it is thus often necessary to employ various integration technologies. In the following chapter various integration technologies are discussed.

CHAPTER 5

INTEGRATION TECHNOLOGIES AND STANDARDS

Chapter 4 describes various integration approaches that can be followed to achieve intra- and/or inter-enterprise integration.

While Chapter 4 discusses various approaches to achieve both intra- and inter-enterprise integration, this chapter will discuss the various integration technologies and standards that support the development of integration solutions.

Chapter 6 will introduce a model that advocates a phased approach to achieve intra- and/or inter-enterprise integration.

5.1 INTRODUCTION

In the previous chapter the various approaches that can be followed to achieve intra- or inter-enterprise integration were summarized. When pursuing these approaches it is necessary to employ integration technologies to produce the desired level of communication between the relevant target and source systems. A detailed discussion of the inner-workings of these technologies is beyond the scope of this project. The focus will rather be on providing an overview of the manner that these technologies enable communication between systems. The middleware technologies and standards that facilitate communication between systems will be discussed in the remainder of this chapter (Linthicum, 2003[1]).

5.2 MIDDLEWARE

Middleware allows communication between dissimilar entities, which could be databases and/or applications, both within and between enterprises (Linthicum, 2003[1]; Alodar Systems, 2003; Linthicum, 2003[2]; Carnegie Mellon, 1997[3]). Middleware utilizes two types of communication models, namely synchronous and asynchronous.

Synchronous communication requires that a source system halts processing when making a request from a target system. Processing only continues once a response from the target application is received. The synchronous communication model involves tight coupling between the middleware and the relevant communicating systems – indicating that problems with the middleware halts application processing. While this is a drawback of synchronous communication, it is useful in situations where it is necessary to know whether a

message was received and processed by the target system, such as cases where a transaction must be completed before processing can continue. Alternatively, the asynchronous communication model does not halt application processing when communication occurs between systems. The source system is not dependent on a response from the target application for application processing. It typically involves placing a message in a queue, resuming processing, and then obtaining a response from the queue at a later time. This communication model is of use in situations where high throughput and high speed communication is required (Linthicum, 2003[1]; Alodar Systems, 2003).

There are various types of middleware that can be used to enable communication in particular situations. The various types of middleware available include (Carnegie Mellon, 1997[3]; Raj & Ishii, 2004; CSK Software, 2003; Britton, 2003; Linthicum, 2003[2]; Linthicum, 2003[1]):

- Database-oriented middleware
- Remote procedure calls
- Distributed objects
- Transactional middleware
- Message-oriented middleware, and
- Broker-based integration.

These types of middleware will be discussed in the following sections.

5.2.1 DATABASE-ORIENTED MIDDLEWARE

Database-oriented middleware enables database-to-database, as well as application-to-database communication, and is

typically used to extract data from both local and remote databases (Linthicum, 2003[1]; Britton, 2003). Communication occurs regardless of the database model employed or the platform upon which the database exists (Linthicum, 2003[1]; Raj & Ishii, 2004). Database-oriented middleware provides the ability to convert the application language into something that the target database will understand (for example an SQL-query) sends the query to a local or remote database, processes the query on the target database and then moves the results of the query (the results set) back to the requesting application. In addition, the response set is converted into a format that the requesting application will understand (Linthicum, 2003[1]).

Database-oriented middleware can generally be categorized as being native database-oriented middleware, call-level interfaces (CLIs) or database gateways (Linthicum, 2003[1]; Raj & Ishii, 2004). These database-oriented middleware categories will be discussed in the following sections.

5.2.1.1 Native database-oriented middleware

Native database-oriented middleware is created for a specific database, for example middleware provided by Sybase to access a Sybase database. Native middleware provides access to native database features and functions such as stored procedures and triggers. Since the middleware has been created for a specific database it offers the best performance, but a limitation is the fact that it only offers communication to one type of database. Once communication is established to a database using native database-oriented middleware, considerable adjustments are

required to convert to another database (Linthicum, 2003[1]).

5.2.1.2 Call-level interfaces (CLIs)

Call-level interfaces (CLIs), such as Microsoft's ODBC and OLE DB, and JavaSoft's JDBC, provide access to multiple databases through a well-defined common interface. CLIs transforms requests into something that the target database can understand and then transforms the results set into a representation understandable to the requesting application (Linthicum, 2003[1]; Raj & Ishii, 2004). The ODBC, OLE DB, and JDBC CLIs are very briefly described below.

ODBC

Microsoft's Open Database Connectivity (ODBC) provides a standard API that facilitates access to various relational data sources. ODBC exposes the capabilities of the data source, while ODBC drivers implement the functions in the ODBC API. ODBC drivers also accommodate differences between data sources. Using ODBC an application can gain access to any data source for which it has an ODBC driver (Britton, 2003; Linthicum, 2003[1]; Microsoft, 2004[2]; Microsoft, 2004[3]; Microsoft, 2004[6]; Lublinsky, 2002).

OLE DB

While ODBC provides access to mostly relational data sources, OLE DB enables access to data sources that include relational databases, documents, spreadsheets, files and e-mail. Basically, OLE DB is a set of COM-based interfaces that provide access to a

variety of diverse data sources. A basic OLE DB implementation consists of a data provider and a data consumer. The data provider responds to OLE DB calls to provide data to a data consumer (Britton, 2003; Linthicum, 2003[1]; Microsoft, 2004[4]).

JDBC

JavaSoft's Java Database Connectivity (JDBC) provides access to various relational data sources from any Java-enabled application, including applets, servlets, Java Server Pages (JSP), Enterprise JavaBeans (EJB), and standalone applications. While JDBC is functionally equivalent to ODBC, it is designed specifically for Java programs (Linthicum, 2003[1]; Fallahi, 2004; JavaSoft, 1997; Lublinsky, 2002).

5.2.1.3 Database gateways

Database gateways provide access to data locked inside larger systems, such as mainframes. They can also remap older database models such as flat files, ISAM and VSAM so that they appear more conventional. Database gateways are typically used in situations where it is necessary to access data residing in environments that are not easily accessible (Linthicum, 2003[1]).

Access to data is often a key requirement in application integration solutions, making database-oriented middleware essential. Fortunately database-oriented middleware is not only very mature and well tested, but there are currently also a multitude of solutions available to retrieve data from a database, as well as place data in a database (Linthicum, 2003[1]).

5.2.2 REMOTE PROCEDURE CALLS

Just as the name implies, Remote Procedure Calls (RPCs) allows a client system to invoke a procedure on a server system by making a function call (see Figure 5.1). RPCs enable communication between different applications, which could be located on the same or remote machines. RPCs are not only the easiest middleware to understand and use, but also the oldest (Linthicum, 2003[1]; Linthicum, 2003[2]; Britton, 2003; Carnegie Mellon, 1997[4]). RPCs allow applications to maintain the integrity of data it owns, since other applications can only access or modify the data by directly requesting the owner to perform the desired procedure.

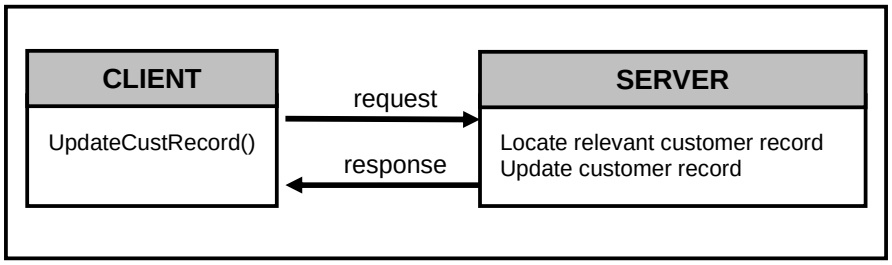

FIGURE 5.1: Client system invoking a procedure on server system through a function call.

RPCs employ a synchronous communication mechanism, since execution of the client program is blocked when making a remote procedure call. Execution only continues once a response from the server system is received. If the procedure call or response is lost in the network, the server is slow, or the server stops while processing the request, the client is left waiting – this has lead to RPCs being known as *blocking middleware* (Linthicum, 2003[2]; Britton, 2003; Carnegie Mellon, 1997[4]).

RPCs require more bandwidth than other middleware solutions, due to the high level of network communication between the client and server systems. In addition, RPCs require immense levels of processing power to function adequately. Despite their simplicity, RPCs are not always the best solution to solve integration problems in an extended enterprise. Blocking application processing during a remote procedure call diminishes the performance of the application and the high bandwidth required makes it unsuitable for use over slower networks, such as the Internet (Linthicum, 2003[1]; Linthicum, 2003[2]).

5.2.3 DISTRIBUTED OBJECTS

Distributed objects are not only used in application development, but can also facilitate communication between applications (Linthicum , 2003[1]; Raj, 1998). Through a process known as *application wrapping* an existing application can be changed to appear as a distributed object to both local and remote applications. By *wrapping* an application its business processes are exposed as methods that can be invoked by

other, local or remote, applications (Linthicum, 2003[1]; Raj, 1998). For example, in the extended enterprise there might be independent applications owned by the various members of the extended enterprise that have some methods such as *Add_Customer()*, *Edit_Customer()* and *Delete_Customer()* in common. To share these methods a set of distributed objects could be created, housing these common methods (Linthicum, 2003[1]). Linthicum defines distributed objects as "small application programs that use standard interfaces and protocols to communicate with one another" (Linthicum, 2003[1]). If two distributed objects are created using the same standard and a standard communications protocol, then those objects should be able to exchange data and carry out application functions by invoking each other's methods, regardless of whether one runs on a UNIX server and the other on an NT server (Linthicum, 2003[1]).

There are three major distributed object middleware standards, namely CORBA, COM/DCOM and Java RMI (Carnegie Mellon, 2004; Britton, 2003; Raj, 1998). These standards will be discussed in the sections below.

5.2.3.1 CORBA

The Common Object Request Broker Architecture (CORBA) specification was developed by the Object Management Group (OMG), a consortium of industry vendors representing computer manufacturers, software vendors, academic-, and government organizations. CORBA is a specification of an industry standard architecture for Object Request Brokers (ORBs). A standard architecture allows vendors to develop ORBs that facilitate object re-use regardless

of location, hardware platform, operating system, or programming language (Segue Software, 2004; Linthicum, 2003[1]; Carnegie Mellon, 1997[1]; Britton, 2003; Object Management Group, 2004).

Object Request Brokers (ORBs) enable communication and data exchange between objects, allowing developers to develop applications by leveraging various objects, locally or over a network, that communicate via the ORB (Linthicum, 2003[1]; Carnegie Mellon, 2004). The ORB provides a directory of distributed objects available, managing access to and from objects, linking them to other objects, monitoring their function, tracking their location and managing communication with other ORBs. While it may appear as if the object is local to the client it may reside on another machine, communicating through the ORB over a network.

By using a CORBA-compliant ORB a client can invoke a method without being aware of the location of the method, or the programming language, operating system or hardware platform used. It is the responsibility of the ORB to intercept the call, locate an object that can implement the request, pass the necessary parameters to the distributed object and then invoke its method and return the results of the invocation to the client (Carnegie Mellon, 1997[1]; Raj, 1998).

5.2.3.2 COM/DCOM

Component Object Model (COM) is a specification developed by Microsoft that enables software components to communicate by allowing systems to invoke services provided by COM-compliant components (COM objects) (Microsoft, 2005; Linthicum, 2003[1]; Carnegie Mellon, 2001). Developers use the COM specification and associated implementation to create re-usable software components, which can then be linked together to build new systems. COM enables communication between these re-usable components, for example COM Object Linking and Embedding (OLE) technology allows Microsoft Word documents to dynamically link to data in Microsoft Excel spreadsheets (Carnegie Mellon, 2001; Microsoft, 2005). OLE is a set of standard COM interfaces that allows components to be linked and embedded into container applications.

To support communication among COM objects on different computers COM was extended to Distributed Component Object Model (DCOM) (Carnegie Mellon, 2001; Linthicum, 2003[1]; Laftsidis, n.d.). DCOM allows network-based communication among COM objects and enables developers to extend their re-usable components over a network. These components operating on different platforms can interact as long as DCOM is available within the environment (Carnegie Mellon, 2001; Linthicum, 2003[1]).

COM/DCOM provides services for component interaction on a single platform, as well as across heterogeneous networks (Carnegie Mellon, 2001). While COM/DCOM supports various non-Windows platforms it is still very much Windows-bound, making CORBA a better choice for most integration initiatives (Carnegie Mellon, 2001; Linthicum, 2003[1]).

5.2.3.3 Java RMI

Java Remote Method Invocation (RMI) allows remote Java objects to be invoked from other Java Virtual Machines (JVMs), which could be on the same machine or located on a machine accessed over a network (Baclawski, 1998; Sun, 2004[2]; Sun, 1997). Java RMI has the ability to pass objects as parameters to methods, unlike other remote invocation middleware, such as RPC, which require parameters to be primitive data types or structures composed of primitive data types. These objects could even be object types that the JVM has never come across before (Baclawski, 1998; Reilly, 2000; Laftsidis, n.d.). Java RMI can work on many diverse operating system platforms, as long as there is a JVM implementation for that platform (Raj, 1998).

5.2.4 TRANSACTIONAL MIDDLEWARE

Transaction-oriented middleware coordinates the movement of data and the sharing of services between many different resources and requires that applications be divided into a set of transactions. The transactions are then controlled by the

transactional middleware from their beginning to their end (Linthicum, 2003[1]; Linthicum, 2003[2]). A transaction is a unit of work that must either end up completed (committed) or it must be completely undone (rolled-back) (Britton, 2003). Consider withdrawing money from an ATM. If an error occurs before you receive the money you would expect the entire transaction to be cancelled, in other words your account balance must remain unchanged. Thus, a transaction must either be completed entirely, or entirely undone. Many tasks performed in an organization are transactional. For example, when submitting an order form it is required that all order lines are submitted, not just some.

Transactions must obey the ACID properties where *A* stands for *Atomic*, *C* for *Consistent*, *I* for *Isolated* and *D* for *Durable*. *Atomic* refers to the requirement that transactions are always completed entirely, *Consistent* to the fact that the system is always in a consistent state regardless of whether the transaction is completed or not, *Isolated* to the ability of the transaction to work independently of other transactions that run in the same environment and *Durability* refers to the fact that transactions can survive system failures once they are committed (Linthicum, 2003[1]; Britton, 2003). These properties ensure a high degree of application integrity when utilizing transaction-oriented middleware.

Transactional middleware, including transaction processing (TP) monitors and application servers, controls transaction applications by centralizing application processing. A centralized server processes transactions and provides a location to share methods between applications and to run application code (Linthicum, 2003[1]; Carnegie Mellon, 1997[5]).

Transactional middleware can also be used to update a number of databases at the same time. All connected databases are updated with the newer data and the success or failure of each update is returned. This ensures that data across all databases are consistent, since the updates to the databases are not committed if the update fails in one of the databases. In such cases all the updates are rolled back to ensure consistent data across all connected databases (Linthicum, 1999).

While transactional middleware ensures the effective coordination of data movement and method sharing between resources, the invasiveness of this middleware approach makes it unsuitable for most inter-enterprise integration initiatives. In most cases it is necessary to make changes to applications to leverage transactional middleware, something that is not always possible in inter-enterprise integration initiatives where applications are not owned and controlled by one extended enterprise member (Linthicum, 2003[1]; Linthicum, 2003[2]).

5.2.5 MESSAGE-ORIENTED MIDDLEWARE

Message-oriented middleware (MOM) uses messages that have both structure (a schema) and content (data) to move information between target and source systems, employing either a point-to-point or message queuing model. When using the point-to-point model, both the source and target systems must be active to exchange messages. The message queuing model does not require all participating systems to be available. If the receiving system is busy or offline then the

message is stored in a temporary storage area, known as a queue, until the target system is available to process the message (Linthicum, 2003[1]; Carnegie Mellon, 1997[2]; Shi & Gandhi, 2001).

The message queuing approach is the most prevalent, allowing the requesting system to place a message on the queue and proceed, getting a response from the queue at a later stage if required (Linthicum, 2003[1]). Messages are guaranteed to reach their destination since the queue can be put on disk so that it can be recovered in the case of a system failure. The queue manager repeatedly attempts to deliver messages in the queue to their intended destinations (Linthicum, 2003[1]; Shi & Gandhi, 2001; Britton, 2003). When a message is put in the queue during a transaction the queue manager can work with a transaction manager so that the message can be removed from the queue if the transaction is not completed (rolled back) (Britton, 2003).

MOM allows systems to operate in an asynchronous mode, meaning that program execution is not stopped while the source system waits for a response from the target system (CSK Software, 2003; Davydov, 2000; Roch, 2002; Linthicum, 2003[1]; Shi & Gandhi, 2001). While this is considered an advantage it is also a disadvantage in the sense that it does not guard against overloading a network. A source system can keep putting messages in a queue, while the target system might not be able to keep up (Carnegie Mellon, 1997[2]).

MOM provides a standard API that shields systems from differences in hardware, operating system, platforms and networks (Shi & Gandhi, 2001; Linthicum, 2003[1]). MOM is

most beneficial in situations where systems are not always up and running, over networks that are not always dependable or where bandwidth is limited (Linthicum, 2003[1]).

5.2.6 BROKER-BASED INTEGRATION

Broker-based integration solutions typically implement either a hub-and-spoke or a bus model to facilitate communication between target and source systems. The hub-and spoke model is typically implemented in the form of a message broker, also known as an integration broker, while the bus model is implemented in the form of an enterprise service bus (ESB) (Microsoft, 2004[1]; Friedman, 2003).

Message brokers extend the basic functionality of message queues by adding routing and transformation capabilities (Laftsidis, n.d.; Linthicum, 2003[1]; Shi & Gandhi, 2001; Longo, 2001; Roch, 2002). Messages are moved from a source system to a broker, which then translates the message into a format understandable by the target system, and then passes the message on to the target system (Raj & Ishii, 2004; Microsoft, 2004[1]; Linthicum, 2003[1]; CSK Software, 2003; Shi & Gandhi, 2001). Instead of directly communicating with each other, participating systems communicate with the message broker, which directs communication between the systems. Message brokering technology follows a loosely coupled approach. It is thus not necessary to make major changes to participating systems to facilitate communication and changes in participating systems do not affect communication. A drawback includes the fact that all messages must pass through the broker, which could cause a bottleneck and also

provide a single point of failure. If the broker is down communication is not possible (Microsoft, 2004[1]; Shi & Gandhi, 2001; Laftsidis, n.d.).

An enterprise service bus (ESB) extends the concept of a message broker by supporting, amongst others, XML and Web Services standards (Cape Clear, 2004; Craggs, 2003; Friedman, 2003). The ESB is a standards-based integration solution. Participating systems publish messages across the ESB while other systems monitor the ESB and grab messages that relate to them. It is no longer necessary to send messages to a centralized broker for transformation and routing, transformation services are provided in the bus to translate messages when needed (Cape Clear, 2004; Van Huizen, 2003; Friedman, 2003; Thomas & Buckley, 2004).

In addition to the middleware discussed above, there are various standards that support integration between systems. A few of these standards will be discussed in the following section.

5.3 STANDARDS

A standards-based approach to application integration can assist in assuring a successful implementation (Sun, 2002; Fiorano, 2003). There are an extensive variety of standards available to aid both intra- and inter-enterprise integration. It is beyond the scope of this project to discuss all of these standards; instead four of the more recent standards that promote communication both intra- and inter-enterprise will be discussed in the following sections.

5.3.1 XML

Extensible markup language (XML) is a standard that enables information exchange both intra- and inter-enterprise. XML contains data, known as metadata, which describes the content of the actual data (Linthicum, 2003[1]; Lublinsky, 2002; Harris & Cantrell, 2002; Shi & Gandhi, 2001; Britton, 2003; Bergholz, 2000). The data elements are surrounded by start and end tags, as can be seen in the simplified example below:

```
<name>
<first>  John  </first>
<last>  Doe  </last>
</name>
<address>
<pobox>  PO Box 123  </pobox>
<city>  Somewhere  </city>
<code>  1234  </code>
</address>
```

XML allows source and target systems to exchange data by describing the data in such a way that it can be read by many different types of systems. Problems related to incompatible data formats and different hardware- and software platforms are thus eliminated, since the XML data is stored in plain text format (W3Schools, 2004; Shi & Gandhi, 2001; W3C, 2004).

5.3.2 RosettaNet

RosettaNet is a non-profit consortium of major high technology organizations that are working together to create, implement and promote industry wide, open e-business process standards and services. These standards serve to align processes between global trading partners in the high

technology industry (Linthicum, 2003[1]; RosettaNet, 2004[3]; RosettaNet, 2004[2]; Skinstad, 2000; Bussler, 2003). RosettaNet fundamentally allows trading partners to agree on how to execute standard business processes (Linthicum, 2003[1]). This is accomplished by developing common Partner Interface Processes (PIPs) and common dictionaries.

RosettaNet PIPs are specialized system-to-system dialogues that define business processes between trading partners. RosettaNet dictionaries, such as the RosettaNet Business Dictionary and the RosettaNet Technical Dictionary, provide a common vocabulary to reduce confusion when conducting e-business, since each trading partner uses uniquely defined terminology. For example, the RosettaNet Technical Dictionary provides common terms to define products and services, since the various manufacturers, distributors, and resellers all describe products and services in different ways. The RosettaNet Implementation Framework (RNIF) permits quick and efficient implementation of RosettsaNet standards by providing exchange protocols, such as specifying information exchange between trading partner servers using XML (RosettaNet, 2004[2]; RosettaNet, 2004[1]; Linthicum, 2003[1]; Skinstad, 2000; Bussler, 2003).

5.3.3 BizTalk

Microsoft BizTalk, supported by a wide range of organizations, enables participants to integrate and manage business processes by exchanging business documents, such as invoices, between systems both intra- and inter-enterprise. The BizTalk Framework specifies how to design and develop

XML-based solutions to enable communication between these systems. This enables organizations to exchange XML documents with extended enterprise partners and internal systems regardless of the platform, operating system or underlying technology (TopXML, 2004; Micrsoft, 1999; Skinstad, 2000; Linthicum, 2003[2]). Unlike standards such as RosettaNet and ebXML, which are process-oriented, BizTalk is information-oriented (Linthicum, 2003[1]).

5.3.4 ebXML

The electronic business XML initiative (ebXML) is a joint project between the United Nations Centre for Trade Facilitation and Electronic Business (UN/CEFACT) and the Organization for the Advancement of Structured Information Standards (OASIS) (Sun, 2004[1].; Bussler, 2003; Linthicum, 2003[2]). The goal of the venture is to develop a set of specifications to enable organizations to conduct business over the Internet and to use XML to exchange business data. ebXML provide a standard method to exchange business messages, conduct trading relationships, communicate data in common terms, and define and register business processes (Bussler, 2003; Skinstad, 2000; ebXML, 2004; Linthicum, 2004[1]).

5.4 CONCLUSION

While Chapter 4 discussed the various approaches to achieve both intra- and inter-enterprise integration, this chapter discussed the various technologies that support integration. While the technologies

discussed here are by no means an extensive representation of the various technologies available, it does represent the technological approaches often discussed in literature regarding intra- and inter-enterprise integration initiatives (Carnegie Mellon, 1997[3]; Raj & Ishii, 2004; CSK Software, 2003; Britton, 2003; Linthicum, 2003[2]; Linthicum, 2003[1]). The next chapter introduces the Extended Enterprise Integration Model, a model that proposes a phased approach to achieve either intra- or inter-enterprise integration.

CHAPTER 6

TOWARDS AN EXTENDED ENTERPRISE: THE EXTENDED ENTERPRISE INTEGRATION MODEL

Chapter 5 discusses the various integration technologies and standards that can be employed to develop intra- and or inter-enterprise integration solutions.

This chapter introduces the Extended Enterprise Integration Model. This model represents the phases that integration initiatives should undergo to establish integrated communication between the members of the extended enterprise.

Chapter 7 utilizes an example to illustrate how the Extended Enterprise Integration Model can be deployed to support the establishment of an extended enterprise.

6.1 INTRODUCTION

The previous chapters introduces the concepts of intra- and inter-enterprise integration and then discusses the various integration approaches that can be followed to achieve varying levels of integration, as well as several integration technologies that can be employed to achieve the actual integration of source and target systems. All organizations differ, and it is thus not possible to suggest one single integration approach and appropriate integration technologies to employ in order to solve the communication problems between the members of the extended enterprise. While the integration approaches and requirements of the various members of the extended enterprise will differ, it is possible to suggest general phases that intra-enterprise integration initiatives, and ultimately inter-enterprise integration initiatives, should embark on to solve communication problems between the various members of the extended enterprise. There are no models presented in literature that support such a phased approach to intra- and/or inter-enterprise integration initiatives. As a result of the research done up to date this chapter introduces the Extended Enterprise Integration Model, a model representing the phases that integration initiatives should undergo to establish integrated communication between the members of the extended enterprise.

6.2 THE EXTENDED ENTERPRISE INTEGRATION MODEL

A diagrammatic representation of the Extended Enterprise Integration Model is presented in Figure 6.1 at the end of this chapter. While intra-

and inter-enterprise integration initiatives involve the same phases, the issues to consider when undertaking each phase will differ. It will be necessary to consider aspects specific to intra- and inter-enterprise integration initiatives, such as the increased need for security when communication extends beyond the borders of the organization (inter-enterprise integration). As mentioned in previous chapters, it is highly recommended that intra-enterprise integration is achieved first, before organizations embark on inter-enterprise integration initiatives.

In the subsequent sections of this chapter the various phases of the Extended Enterprise Integration Model will be discussed. At the end of each Phase it is necessary to evaluate the outcome of that Phase to ensure that the integration initiative stays on track. When concerns are identified during these evaluation phases it is necessary to return to certain preceding phases, depending on the phase that is being evaluated and the issues discovered, and to work through the subsequent phases again in order to resolve these issues.

6.3 REALIZATION OF INTEGRATION NEED

Before an organization invests time and money in deploying the Extended Enterprise Integration Model to achieve either intra- and/or inter-enterprise integration, there will typically be some problem or event that triggers the realization that some form of integration is necessary to solve communication problems or improve certain business processes. This trigger could be the result of an external event, or the success of a recent integration initiative may trigger the need for further integration initiatives. This will typically lead to various informal discussions, meetings and even the gathering of background information on e-Business Integration. These discussions

will in turn drive the decision to embark on an integration initiative by deploying the Extended Enterprise Integration Model.

6.4 PHASE 1: DETERMINE INTEGRATION READINESS

6.4.1 DETERMINE WHETHER THERE IS A NEED FOR INTEGRATION

It is first and foremost necessary to determine whether integration is, in actual fact, needed. There is no value in undertaking an integration initiative for purely technical reasons unless there are clear business problems that will be addressed by the integration solution (Babu, 2003). Business problems that could indicate that integration is needed may include (Gartner Holdings, 2003; Hindi & Kuhbock, 2003):

- Manual and un-integrated processes may affect the organization's competitive advantage negatively. For example, real-time processing of orders could prevent the organization from losing sales to competitors who can process orders much faster.
- Mergers and acquisitions may necessitate that duplicate systems and processes be integrated to coordinate activities across the organization and ensure access to current, complete data that are consolidated across the organization.
- Compliance with government regulations may necessitate that certain transactions are integrated to ensure that they are processed in less time with fewer errors.

- Supply chain issues could also require that inventory, production and order management systems and processes be integrated within the organization and with business partners. Integrating these systems could provide real-time visibility, allowing the organization to better match incoming orders with what is in inventory and what is being produced. This could lead to benefits such as reduced inventory levels and improved order tracking.
- Customer relationship concerns may require that CRM systems be integrated with other systems, such as the order management system. This will allow customer service employees to respond to customer enquiries faster and more efficiently, since accurate, complete, up-to-date information on each customer will be available at the touch of a button.

Once it is determined that there is a definite business need for integration, it is necessary to determine whether the organization is prepared for the impact that the integration solution will have on the organization.

6.4.2 DETERMINE POSSIBLE IMPACT OF INTEGRATION

While it is impossible to establish the exact impact of any integration initiatives at this stage, it is necessary to consider the impact that an integration initiative may have on the organization and possibly on other members of the extended enterprise as well. With intra-enterprise integration initiatives the solution will typically impact various departments

throughout the organization, while inter-enterprise integration initiatives will have a much broader impact on various members of the extended enterprise. The business problems that drive the need for integration could indicate the possible impact of an integration initiative – the more stakeholders affected by the business problems the greater the possible impact.

6.4.3 IDENTIFY BUSINESS ADVISORS AND PROJECT CHAMPION

Due to the immense impact that integration initiatives could have on the organization, both internally and across organizational borders, it is important that the initiative is supported by organizational business leaders (Bridges, Bond & Torrance, 2001; Rodgers, Yen & Chou, 2002; LaFaire, 2003; Maverick, 2003). It is necessary to identify an organizational leader that will act as project champion to drive the integration initiative from the top (Brown, 2003). By studying the business problem it is also necessary to identify business leaders, subject matter experts, and various other employees and stakeholders who will be involved during the next phase to determine the business goals of the integration initiative. When identifying these business advisors it is important that all the stakeholders who will be impacted by the integration initiative are identified.

6.5 PHASE 2: DETERMINE BUSINESS GOALS

6.5.1 DETERMINE BUSINESS GOALS

The business goals that should be achieved through the integration initiative must be determined during this phase. Following an integration approach and implementing several integration technologies to integrate various source and target systems can only be successful if it is clear what the business goals of the integration initiative are (Wagner, 2002; Gosain & Thillairajah, 2002; Schmidt, 2003; Gupta, 2003). Applying integration technology to an integration problem does not necessarily solve the problem if it is not aligned with the business goals. Creating an enterprise portal where various shared methods can be accessed will not solve the problem if the only business goal was to simply share data between System A and System B.

The business goals should be determined by the business leaders and subject matter experts, not by the IT experts (Babu, 2003). It must be remembered that the integration initiative serves to *support* the establishment of the extended enterprise – integrating intra- and inter-enterprise applications does not *create* the extended enterprise. Determining the goals of the integration initiative to support the establishment of the extended enterprise should thus be done by the business leaders and subject matter experts who use the systems that require integration. The subject matter experts are employees working in areas such as sales, marketing, finance, production, and so forth who use various IT systems

in their daily work. They can be from various departments and enterprises, depending on whether it is an intra- or inter-enterprise integration initiative.

It is not required to identify the exact systems that should be integrated during this phase. The business goal could be a simple statement such as "*We want our customers to be able to check the availability and current price of our products on our web-site*," or more complex such as "*When a customer places an order for a product on our web-site we want the process of checking that the product is in stock automated, and if the product is not in stock the relevant systems of our suppliers and production department should be informed automatically to start production of the product.*" Without going into such detail as exactly which systems to integrate, the business leaders and subject matter experts need to determine which problems they need to solve through application integration, such as the afore-mentioned examples (Babu, 2003).

6.5.2 EVALUATE BUSINESS GOALS

Setting clearly defined goals at the start of an integration initiative is one of the most important contributing factors to the success of the initiative and will provide a measure to assess the success of the final integration solution (Microsoft, 2002). It is thus very important that the business goals are evaluated before advancing to Phase 2. The business goals indicate what the users expect from the integration initiative and essentially affect the integration requirements of the project

(Pinkston, 2001). When evaluating the business goals it is necessary to determine that the goals represent the true integration problem that needs to be solved through e-Business Integration. If it becomes apparent that the business goals are not acceptable, then it is necessary to return to Phase 1.3 to ensure that the appropriate business advisors were identified to determine the business goals. Once the correct business advisors are identified the business goals can be resolved.

6.6 PHASE 3: SPECIFY INTEGRATION REQUIREMENTS

6.6.1 SPECIFY FUNCTIONAL INTEGRATION REQUIREMENTS

During this phase the IT experts will work with various subject matter experts to identify the systems that will have to be integrated to realize the business goals determined during Phase 2. The subject matter experts will be able to guide the IT experts on the business links between various systems (Babu, 2003; LaFaire, 2003). These subject matter experts are highly proficient in specific areas such as processing claims or production planning, and should fully understand the business goals of the integration initiative to ensure that appropriate requirements are gathered (LaFaire, 2003). Their knowledge on how specific processes are executed within or between enterprises is invaluable to determine which systems to

integrate. Interviews and workshops with these subject matter experts will determine the functional requirements of the integration initiative, such as "*In order for customers to check the availability and price of our products it is necessary for System A (availability) and System B (price) to share product data.*"

While functional requirements determine the specific features and functions that the integration initiative must deliver to realize the business goals determined during Phase 2, there are also non-functional requirements that must be determined (LaFaire, 2003).

6.6.2 SPECIFY NON-FUNCTIONAL INTEGRATION REQUIREMENTS

Non-functional requirements could include the following (Spiers, 2001; Ravishankar, 2003):

- Simplicity - While the integration solution must be able to work with various hardware, operating system and network platforms it must still be simple to understand, use and maintain.
- Scalability - The integration solution must be able to accommodate an increase in demand, such as increasing volumes of work and an increase in the number of connected users.
- Manageability - The implemented solution must be relatively easy to manage.

- Security - A clear security policy is necessary to manage access rights between the various systems and services that comprise the integration solution. This requirement is especially important in inter-enterprise integration initiatives.
- Reliability - Fault-tolerance is of utmost importance. Data loss should be minimized, failed transactions should be reported and rolled-back, and hardware and communication failures should be detected and corrected.
- Re-use - The integration solution must make efficient use of existing resources such as legacy systems. The solution must also be able to develop and grow with the enterprise as the enterprise changes.

While there are various integration approaches and relative integration technologies available to solve integration problems, they are only beneficial once it is clear what the goals and requirements of the integration initiative are (Gosain & Thillairajah, 2002). For that reason Phases 2 and 3 are very important to the success of the integration initiative, making it clear what should be accomplished through integrating various systems. From Phase 4 onwards the IT experts will become more involved to create the integration strategy, choose the appropriate integration approach, deploy the integration solution, and so forth.

6.6.3 EVALUATE INTEGRATION REQUIREMENTS

Once the functional and non-functional requirements are identified it is necessary to verify that the requirements

actually represent the true integration requirements of the enterprise. Poor communication between the IT experts, business leaders and subject matter experts could lead to the specification of inaccurate requirements, and ultimately to the development of an integration solution that does not satisfy the needs of the enterprise. If problems are encountered it is necessary to either return to Phase 3.1 and 3.2 to verify that all functional and non-functional requirements are specified to satisfy the goals defined during Phase 2, or to Phase 1.3 to ensure that a suitable project champion and business advisors were identified.

6.7 PHASE 4: DEFINE INTEGRATION STRATEGY

While the business goals and integration requirements determined during Phases 2 and 3 outlines what the outcome of the integration initiative should be, it is also necessary to define *how* these goals will be reached. This is done during this phase by defining the integration strategy.

6.7.1 DETERMINE WHETHER INTRA- AND/OR INTER-ENTERPRISE INTEGRATION IS NECESSARY

By this stage the business goals and integration requirements should make it clear whether intra- and/ or inter-enterprise integration is necessary.

6.7.2 DETERMINE INTEGRATION STAGES

When defining the integration strategy it is necessary to determine how the goals and requirements defined during Phases 2 and 3 can be reached in the best way possible. Instead of satisfying all goals and requirements with a single deployment of the integration solution it is recommended that integration solutions be deployed in various stages (Spiers, 2001; Wagner, 2002; Bridges, Bond & Torrance, 2001; Gold-Bernstein, 2003[2]; Carzoli, 2002). New functionality can be added incrementally, with each successive stage extending the scope of the initial integration solution. The integration strategy must be developed in such a way that future requirements can be incorporated in successive deployment stages without requiring major redesign (Spiers, 2001). Inter-enterprise integration initiatives will practically always necessitate that the integration solution be deployed in various stages, while smaller intra-enterprise integration initiatives may not require a staged deployment approach.

6.7.3 CONSIDER TECHNOLOGICAL FACTORS

When developing the integration strategy there are various technological factors in the enterprise integration environment that the IT experts should take into account (Patil & Simha, 2004):

- There are multiple systems that will need to be integrated, including packaged applications and applications developed in-house.

- There are various techniques to access the services and data controlled by these systems.
- These systems offer varying levels of support for integration. Some have been developed with the assumption that it will never be necessary to integrate them with other systems.
- The systems run on a multitude of different platforms and operating systems.
- There are varying models in place to handle security aspects and transactions.
- Application availability requirements vary – some applications are expected to be up and running continuously, while it may not be that critical for other applications.
- Data formats and network protocols differ.
- The performance and response requirements of the various systems will differ; some systems may require real-time response.

Once it is determined whether intra- and or inter-enterprise integration is necessary and whether it is necessary to deploy the integration solution in various stages, the IT experts and subject matter experts can develop an integration strategy by considering the business goals and integration requirements, as well as the technological issues highlighted above. The integration strategy should identify all the issues that need to be addressed to fulfill the objectives set out during the previous phases and should indicate when specific stages of the integration initiative will be deployed.

6.7.4 EVALUATE INTEGRATION STRATEGY

Once the integration strategy is defined it should be evaluated to ensure that the integration solution will be developed and deployed in manageable stages, and that the timing margins are reasonable. It must also be evaluated to ensure that the deployment of the integration strategy will satisfy the business goals and integration needs defined during Phases 2 and 3. If problems are picked up and they relate to integration stages or timing issues it is necessary to go back to Phase 4.1 and 4.2 to ensure that an appropriate integration strategy is defined. Bigger issues may necessitate a return to Phase 1 to ensure that the all the relevant stakeholders who will be impacted by the integration initiative are identified and that appropriate business goals and integration requirements are defined.

During the next phase an integration approach should be selected to compliment the integration strategy.

6.8 PHASE 5: SELECT INTEGRATION APPROACH AND VENDOR

6.8.1 SELECT INTEGRATION APPROACH(ES)

During Phase 2 the basic goals of the integration initiative were determined and then extended to include the functional and non-functional requirements during Phase 3. Phase 4 requires that the goals and requirements defined during Phases 2 and 3 were considered to define the integration

strategy - a strategy that considers short- as well as long-term objectives before clearly defining how the objectives of the current integration initiative will be reached. Once that is determined an integration approach can be selected that will meet these objectives, while keeping long-term objectives in mind as well. This choice will typically be made by either in-house IT experts or IT consultants contracted to the integration initiative. These IT experts need a solid understanding of the features and drawbacks of the various integration approaches discussed in Chapter 4, and how these approaches can best meet the objectives of the enterprise (Microsoft, 2002).

Depending on the integration strategy and requirements it may be necessary to employ more than one integration approach. It is important to select an approach, or combination of approaches, that will satisfy all, or as many as possible, of the integration requirements as effectively as possible (Hohpe & Woolf, 2004). The approach selected must take long-term integration objectives into account and be flexible and extensible enough to adapt to ever-changing business conditions and needs (Microsoft, 2002). Once an appropriate integration approach is identified a vendor that will provide the appropriate technologies to implement the approach should be selected. Depending on the integration approach(es) selected it may be necessary to utilize integration technologies from multiple vendors.

6.8.2 SELECT INTEGRATION VENDOR(S)

When selecting a vendor there are various factors that should be considered (Tibco, 2002; Fiorano, 2001; Microsoft, 2002):

- Longevity - The vendor must be able to provide reliable support during all the phases of project deployment, including post-sales support. It is thus necessary to select a vendor with an established track record and long-term stability. Long-term stability is important to ensure that the vendor will still be in business and able to provide continuous support in the event that it is necessary to modify and extended the solution when business needs change.
- Proven results - The vendor should have a track record of satisfied customers who deployed similar solutions with proven results in relevant industries.
- Choice of technology - The vendor should not narrow the choice of technologies to certain technology solutions favored by the specific vendor. It is important though, that the solution employ technologies that are widely accepted and understood. Uncommon technologies will require specialized skill sets to develop, implement and maintain the solution, resulting in unnecessary high costs and longer development time.

Depending on whether integration with packaged applications are required or not, it is advisable to select a vendor that can provide integration to key packaged applications, such as ERP and CRM systems. This necessitates close partnerships between the vendor and the providers of these packaged

applications, such as SAP and PeopleSoft (Tibco, 2002). The vendor should also be able to guarantee a reasonable speed of deployment. Extended deployment phases increases costs and reduces return on investment, while also impacting business agility negatively (Microsoft, 2002). In certain instances it may be necessary to make use of several vendors to develop the integration solution. In such cases it must be ensured that the solutions developed by the various vendors will be able to collaborate to satisfy the overall business goals and integration needs of the integration initiative.

6.8.3 EVALUATE SELECTED INTEGRATION APPROACH(ES) AND VENDOR(S)

Before the selected integration approach is followed by the vendor to develop the integration solution, it is necessary to verify that an appropriate integration approach was selected that corresponds to the integration strategy and that will support the development of a solution that will satisfy the business goals and integration requirements. When selecting the integration approach the long-term goals of the integration initiatives should be taken into account as well to ensure that the solution is scalable. It is also important to ensure that an appropriate vendor, or vendors, are selected to develop the integration solution. If it becomes apparent that the selected integration approach is inappropriate, then it is necessary to return to Phase 5.1 to decide on an alternative integration approach to employ. In such a case it is also necessary to verify that the vendor who was selected in Phase 5.2 is still suitable for the

alternative integration approach. If the only issues are with the chosen vendor then it is necessary to execute Phase 5.2 again to select another vendor.

6.9 PHASE 6: DEPLOYMENT

6.9.1 IMPLEMENT PILOT PROJECT

Before the vendor selected during Phase 5 deploys the integration solution it is advisable that a pilot project is implemented. The pilot project can reduce the risks inherent in such major investments by allowing the enterprise to validate their choice of integration approach and to test the performance of the integration solution developed by the chosen vendor. By selecting a pilot project that represents a realistic scenario it can indicate whether the objectives of the integration initiative are met and challenges can be detected and resolved before they delay the major solution deployment (Microsoft, 2002).

6.9.2 EVALUATE PILOT PROJECT

If major challenges are encountered during the pilot project it is necessary to revisit Phases 2 to 5 to verify the following:

- Phase 2: Do the business goals determined during this phase represent the true business goals?
- Phase 3: Do the functional and non-functional requirements determined during this phase cover all the

requirements that must be satisfied to reach the business goals determined during Phase 2?

- Phase 4: If it was determined that intra- and/or inter-enterprise integration was necessary, was it the right verdict? Was the integration solution divided into appropriate stages?
- Phase 5: Was a suitable integration approach, or combination of approaches, selected? Was an appropriate vendor, or vendors, selected? Were all technological factors considered and addressed in the integration strategy?

By revising the outcomes of these stages it is possible to locate the origin of any problems, and necessary steps can be taken to solve the problems.

Once the enterprise has ensured that all their objectives will be met by the chosen integration approach and that all problems highlighted by the pilot project are resolved, the vendor can deploy the integration solution throughout the enterprise.

6.9.3 DEPLOY INTEGRATION STAGE

The enterprise can prepare for the first deployment stage by identifying training requirements and preparing training material where necessary (Carzoli, 2002). While implementation is a major milestone for the integration initiative it is not the end of the initiative. As each integration stage is deployed it is necessary to evaluate the deployed

stage to determine whether future deployment stages require changes.

6.9.4 EVALUATE INTEGRATION STAGE

After each stage of the integration solution is deployed it is necessary to evaluate the deployed stage to ensure that potential problems are identified in time. Although most problems should be resolved once they are identified during the pilot project there will still be ongoing modifications and extensions as new problems are discovered and the needs of the enterprise changes (Microsoft, 2002).

Deploying and evaluating the various integration stages will be repeated until the entire integration solution is deployed. Once the integration solution is deployed the organization will also be able to evaluate the integration solution to determine whether their objectives are met and where they can improve future integration initiatives.

6.10 PHASE 7: EVALUATION

While it is necessary to evaluate the success of each phase as it is executed and draws to completion, it is also necessary to evaluate the overall success of the integration solution once it is deployed and in use.

6.10.1 DETERMINE WHETHER BUSINESS GOALS AND INTEGRATION REQUIREMENTS ARE SATISFIED

It is necessary to determine whether the deployed integration solution actually realizes the business goals and integration requirements defined during Phases 2 and 3.

6.10.2 DOCUMENT PROBLEMS ENCOUNTERED

Throughout the project note should be taken of difficulties encountered, so that these issues can be resolved and avoided in future integration initiatives.

As the enterprise realizes the benefits that the integration initiative brings about it will lead to new goals surfacing, which could either lead to extending the existing integration initiative, or starting a new integration project to cater for different objectives. For example, the initial integration initiative may have followed a data-oriented integration approach to integrate data residing in various databases to enable customers to check the availability and price of products on the organization's web-site. As the organization realizes the benefits of the implemented integration solution, for example fewer calls from customers requiring product details, the organization may decide to extend the solution to include a customer portal where customers can check their account details and change their personal details themselves. This will significantly reduce the workload of employees who normally handle account enquiries and the multitude of faxes, e-mails, letters and telephone calls from customers who want to change

their address, telephone number or other personal details. It will be possible to just extend the initial integration initiative, since the solution is still data-oriented. If however, the organization decides to include functionality on the portal that allows customers to order products or pay accounts online it may not be possible to just extend the initial integration initiative. Placing an order may require that various services be invoked to complete the order process. In this instance a data-oriented integration approach is no longer suitable. While the data-oriented integration solution can provide a solid foundation for future integration initiatives, a service- or even process-oriented integration approach may be more appropriate in this instance. These long-term goals should be covered in the integration strategy to ensure that the current integration solution is scalable. If the integration strategy does not cover these issues, then the integration solution could lack long-term viability.

6.11 SUPPORT FUNCTIONS

Communication is very important during all phases to ensure that all stakeholders understand what integration entails and the impact that it will have on the organization (Microsoft, 2002; Carl, 2003). Since integration initiatives generally impact various internal departments or extended enterprise partners, depending on whether it is an intra- or inter-enterprise integration initiative, resistance to change is one of the main barriers to application integration (Themistocleous & Irani, 2001; Maverick, 2003). The integration solution will change the way that these departments or partners perform certain processes and empowering them with knowledge on the impact that the integration solution will have on their area of knowledge could reduce resistance to change. While communication and change management are

important support functions in any project, it is beyond the scope of this research to discuss it in detail.

6.12 CONCLUSION

This chapter introduces the Extended Enterprise Integration Model. This model represents the phases that integration initiatives, both intra- and inter-enterprise, undergo to establish communication between the various members of the extended enterprise.

While the integration needs of all organizations differ, this model represents generic phases that can form the foundation of any integration initiative and ultimately lead to the establishment of an extended enterprise. When working through these phases it will be necessary to consider and include issues that are specific to that integration initiative.

In the next chapter the example of an extended enterprise introduced in Chapter 2 will be used to illustrate how an organization can follow the phases of the Extended Enterprise Integration Model to support the establishment of an extended enterprise.

External trigger

Realization of Integration Need

PHASE 1: Determine Integration Readiness	
1.1	Determine whether there is a need for integration
1.2	Determine possible impact of integration
1.3	Identify business advisors and project champion

PHASE 2: Determine Business Goals	
2.1	Determine business goals
2.2	Evaluate business goals

PHASE 3: Specify Integration Requirements	
3.1	Specify functional requirements
3.2	Specify non-functional requirements
3.3	Evaluate integration requirements

PHASE 4: Define Integration Strategy	
4.1	Determine whether intra- and/or inter-enterprise integration is necessary
4.2	Determine integration stages
4.3	Consider technological factors
4.4	Evaluate integration strategy

PHASE 5: Select Integration Approach and Vendor	
5.1	Select integration approach(es)
5.2	Select integration vendor(s)
5.3	Evaluate selected integration approach(es) and vendor(s)

PHASE 6: Deployment	
6.1	Implement pilot project
6.2	Evaluate pilot project
6.3	Deploy integration stage
6.4	Evaluate integration stage

PHASE 7: Evaluation	
7.1	Determine if business goals & integration requirements are satisfied
7.2	Document problems encountered

SUPPORT FUNCTIONS

FIGURE 6.1: The Extended Enterprise Integration Model

CHAPTER 7

DEPLOYING THE EXTENDED ENTERPRISE INTEGRATION MODEL

Chapter 6 presents the Extended Enterprise Integration Model consisting of various phases that integration initiatives should undergo to achieve either intra- or inter-enterprise integration, ultimately leading to the creation of an extended enterprise.

This chapter utilizes the example of an extended enterprise introduced in Chapter 2 to illustrate how the Extended Enterprise Integration Model can be deployed to support the establishment of an extended enterprise.

In Chapter 8 the benefits of utilizing e-Business Integration in the establishment of an extended enterprise will be highlighted and some of the obstacles that can be expected when utilizing e-Business Integration will also be pointed out.

7.1 INTRODUCTION

The previous chapter introduced the Extended Enterprise Integration Model consisting of various phases that integration initiatives should undergo to achieve either intra- or inter-enterprise integration, ultimately leading to the creation of an extended enterprise. In this chapter the example of an extended enterprise introduced in Chapter 2 will be used to illustrate how this model can be deployed during an intra- and inter-enterprise integration initiative.

7.2 THE EXTENDED ENTERPRISE EXAMPLE

Chippie's Chips is a brand of potato chips that is sold in various supermarkets throughout the country. Even though there are quite a number of organizations working together to manufacture the potato chips, the most prominent stakeholders include the following:

- *John Brown*, a farmer with seven farms in various regions of the country who grows the potatoes used in the manufacturing of *Chippie's Chips.*
- *Bag It!*, the packaging manufacturer that supplies the bags that the potato chips are sold in.
- *Chippie's Chips*, the actual manufacturer of the potato chips.
- *Speedy Transport Services (PTY) LTD*, the transport company that delivers the product to various supermarkets.
- The supermarkets where Chippie's Chips are sold.

Chippie's Chips is a relatively new brand and the organization currently communicates with the various stakeholders mentioned above through more traditional means such as telephone calls, faxes, e-mail and even snail mail. The potato chips are becoming increasingly popular though,

and sales figures are soaring. Since sales figures are soaring, manufacturing also needs to increase. While the sales department at Chippie's Chips always generates accurate sales projections, the manufacturing department struggles to plan their production schedules since John Brown cannot always supply them with timely estimates on how many tons of potatoes he will be able to supply on certain dates.

7.3 REALIZATION OF INTEGRATION NEED

Lester Philips, a manager at Chippie's Chips, has recently read a document entitled *Towards an Extended Enterprise through e-Business Integration*. This document describes the concept of an extended enterprise, a term used to describe the high level of interdependence that exists between the organization and its customers, suppliers and other business partners. In the case of Chippie's Chips its extended enterprise partners would include John Brown, Bag It!, Speedy Transport Services (PTY) LTD, and the various supermarkets where the potato chips are sold. The document goes on to describe how e-Business Integration can improve communication between the members of the extended enterprise, and also includes an overview of the various integration approaches available that can be followed to integrate target and source systems, as well as an overview of several technologies that can be implemented to realize the actual integration of systems. Finally, the document introduces the Extended Enterprise Integration Model, a model that proposes a phased approach that can be pursued to achieve either intra- or inter-enterprise integration, ultimately leading to integrated communication between the various members of the extended enterprise. After various discussions about the concept of an extended enterprise with other managers, the management of Chippie's Chips decides that it could benefit the organization to involve the various afore-mentioned

stakeholders in establishing an extended enterprise. Since their biggest communication problem is currently the lack of communication between John Brown and the production department of Chippie's Chips, they decide to deploy the Extended Enterprise Integration Model to try and resolve the problem.

7.4 PHASE 1: DETERMINE INTEGRATION READINESS

7.4.1 DETERMINE WHETHER THERE IS A NEED FOR INTEGRATION

The management of Chippie's Chips decides to schedule a meeting at their head-office with farmer John Brown and their production planner to discuss the communication problems they are experiencing. At the meeting they discuss the situation with John Brown to try and determine where the problem lies. When the production department receives the sales projections from the sales department, they need to know whether John Brown will be able to supply them with *x* tons of potatoes on specific dates, so that they can plan their production. The production planner usually phones John Brown to inquire whether he will be able to supply *x* tons of potatoes on a specific date. John Brown can never give him an immediate answer, since he needs to contact the managers of his seven farms first to determine whether they will be able to supply the *x* tons of potatoes.

He usually phones the farm managers on their mobile phones, but they are normally out working on the farm and not able to give him a figure immediately. At the beginning of the season the farm managers take the crop conditions and weather patterns of the region into account to estimate how many tons of potatoes they will be able to harvest per week. They store these estimates in databases on their desktop computers, and as the season progresses and crop conditions and weather patterns change they adjust the figures. Since they are normally not at their computer when Mr. Brown phones, they usually send the figures through to him when they get home at night, either through faxes or e-mails.

John Brown then needs to look at all these figures to determine whether he will be able to deliver *x* tons of potatoes to Chippie's Chips on the specified date, and also whether the whole batch can come from one farm, or whether several of the farms need to send batches to make up *x* tons. Once he has determined this he phones the production planner of Chippie's Chips to confirm delivery. This whole process normally takes three to four days, sometimes even longer.

After discussing possible solutions to the problem they agree that none of them seem efficient enough to really solve the problem. After a lunch break the management of Chippie's Chips decides to discuss the concept of an extended enterprise with Mr. Brown. He seems apprehensive at first, especially when they discuss how e-Business Integration can solve various communication problems. He has never heard of this *extended enterprise* concept, let alone *e-Business Integration*. Up until a few months ago his teenage son still had to help him when he wanted to send e-mails! While this all

seems very complicated to him he is willing to learn more, and the management of Chippie's Chips decides to schedule a follow-up meeting. They decide that Lester Philips, the manager that initially introduced the concept of an extended enterprise to the other managers, will prepare a presentation on what an extended enterprise is, and they also decide to find an IT consultant that specializes in application integration to come and give a presentation on intra- and inter-enterprise integration. They also agree to invite the farm managers of John Brown's seven farms to the meeting.

At the follow-up meeting Lester Philips gives his presentation on what an extended enterprise is and the IT consultant gives a presentation on what exactly intra- and inter-enterprise integration is in terms that the farmers and managers of Chippie's Chips can understand. After the presentation there are various questions, especially from John Brown, and the consultant answers them as best he can without going into too much technical detail. After much discussion everyone agrees that working towards an extended enterprise can benefit everyone, and if they make use of e-Business Integration to solve some of their communication problems it can also make life easier for everyone and save lots of time. Even John Brown realizes that it might not be necessary anymore for him to spend a whole night with his calculator and faxes and e-mails from his farm managers to try and determine whether he will be able to supply *x* tons of potatoes to Chippies Chips on a specific date.

After further discussions the management of Chippie's Chips, along with John Brown and his farm managers, agree that

they are willing to embark on an integration initiative to try and solve their communication problems.

DETERMINE WHETHER THERE IS A NEED FOR INTEGRATION:

After various solutions to the communication problems between the production department of Chippie's Chips and farmer John Brown were discussed it was concluded that the only viable solution will necessitate integration of certain systems. These systems will be identified in subsequent phases.

7.4.2 DETERMINE POSSIBLE IMPACT OF INTEGRATION

While it is impossible to determine the exact impact of an integration solution at this stage, it is relatively apparent that an integration solution will most likely have an impact on the production department of Chippie's Chips, John Brown and his farm managers. An integration solution will typically impact these stakeholders, since the communication problem exists between them.

DETERMINE POSSIBLE IMPACT OF INTEGRATION:

Since the exact integration solution is not determined yet, it is impossible to determine the exact impact of the solution, but the solution will most likely impact the production department of Chippie's Chips, farmer John Brown and his farm managers.

7.4.3 IDENTIFY BUSINESS ADVISORS AND PROJECT CHAMPION

The root of the communication problem between the production department of Chippie's Chips and John Brown is the lack of timely and efficient communication between John Brown and his farm managers. It is decided that Lester Philips will be the project champion that will drive the development of an integration solution that will solve the communication problems. He will be assisted by John Brown to solve the communication problems on his side. His farm managers will need to be consulted to develop a solution, and depending on the solution chosen the production planner of Chippie's Chips may also need to act as a business advisor.

Since the farm managers will need to be present to determine the business goals of the integration initiative, and it is difficult for them to travel from all over the country to attend meetings, they decide to move on to Phase 2 of the integration initiative after the lunch break.

IDENTIFY BUSINESS ADVISORS AND PROJECT CHAMPION:	
Project Champion:	Lester Philips, assisted by John Brown.
Business Advisors:	The seven farm managers of John Brown's seven farms. The production planner of Chippie's Chips (provisionally).

7.5 PHASE 2: DETERMINE BUSINESS GOALS

7.5.1 DETERMINE BUSINESS GOALS

After the lunch break Lester Philips, the management of Chippie's Chips, the production planner and production manager of Chippie's Chips, and John Brown and his farm managers discuss the problem further and determine that the overall business goal is to consolidate the estimates from all the farm managers, so that John Brown can easily determine whether he will be able to supply Chippie's Chips with x tons of potatoes on a specified date, and also whether the whole batch will come from one farm, or whether it should come from various farms to make up x tons. In addition the production planner of Chippie's Chips would like to access this data electronically, without having to contact John Brown every time he needs this information. John Brown is not comfortable with the option of allowing the production planner access to this data, and they agree that currently it will be sufficient if he can access the data immediately and give the production planner an immediate answer.

John Brown decides that he will hire the IT consultant that gave the presentation on e-Business Integration to help him with this intra-enterprise integration initiative, and schedules a meeting at his farm with the consultant and his seven farm managers to specify the integration requirements of the integration initiative.

Lester Philips schedules a follow-up meeting between himself, John Brown, the production planner, and the production manager to evaluate and discuss the business goals further to ensure that everyone's communication needs are satisfied.

DETERMINE BUSINESS GOALS:

John Brown must be able to access the crop estimates of his seven farms instantaneously and be able to see whether he will be able to supply *x* tons of potatoes on a certain date. He must also be able to determine whether the whole batch will come from one farm, or made up from separate batches from various farms.

7.5.2 EVALUATE BUSINESS GOALS

At the follow-up meeting the afore-mentioned business goals are discussed, and while John Brown is satisfied with the business goals, the production planner is unconvinced that these goals will lead to the development of a solution that will solve the communication problems. After further discussions and negotiations it is decided that the business goals should be revised to include functionality that will allow the production planner to confirm electronically, without having to always contact John Brown directly, whether or not *x* tons of potatoes could be supplied on a certain date.

EVALUATE BUSINESS GOALS:

REVISED BUSINESS GOALS:

John Brown must be able to access the crop estimates of his seven farms instantaneously and be able to see whether he will be able to supply *x* tons of potatoes on a certain date. He must also be able to determine whether the whole batch will come from one farm, or made up from separate batches from various farms. In addition, the production planner should be able to confirm electronically whether or not John Brown will be able to supply *x* tons of potatoes on a specific date.

7.6 PHASE 3: SPECIFY INTEGRATION REQUIREMENTS

7.6.1 SPECIFY FUNCTIONAL REQUIREMENTS

Since the business goals of the integration initiative were revised, the meeting between John Brown, his farm managers, and the IT consultant is rescheduled to include Lester Philips and the production planner and production manager of Chippie's Chips. After a short meeting to discuss the objectives of the meeting the IT consultant has two separate meetings, first with John Brown and his farm managers, and secondly with the production planner and production manager of Chippie's Chips. The purpose of these separate meetings is to determine how these stakeholders currently execute the processes that cause the communication problems.

At the meeting between the IT consultant and the farmers they explain to him in detail how they currently capture their estimates and what they do when they receive a call from Mr. Brown enquiring how many tons of potatoes they will be able to supply on a specific date. John Brown also explains how he consequently processes the various faxes and e-mails to determine whether or not he will be able to supply the required amount of potatoes to Chippie's Chips on a specific date, and also how he determines whether the whole batch will come from a single farm, or whether various farms will supply potatoes to make up x tons. Since John Brown performs all his calculations manually the IT consultant determines that the only systems available to integrate are the databases of the seven farm managers that they use to store their estimates on how many tons of potatoes they will be able to supply on specific dates.

The meeting between the IT consultant and the production planner and manager is relatively short, since the production planner normally just phones John Brown with the details of how many tons of potatoes he needs on a specific date and then waits three to four days for John Brown to confirm whether or not he will be able to supply. Once the production planner has received this confirmation he can plan his production schedules accordingly.

With the help of the IT consultant Lester Philips, John Brown, the farm managers, the production planner, and the production manager determine the following functional requirements: the integration initiative must integrate the databases of the seven farm managers into a single solution that processes the estimates contained in the databases and generate a report

for John Brown indicating whether or not he will be able to supply *x* tons of potatoes on a specific date, and also whether the whole batch will come from one single farm, or various farms. Without exposing sensitive data or allowing the production planner to access John Brown's crop estimates, he should be allowed to confirm electronically whether or not John Brown will be able to supply a certain amount of potatoes on a specific date.

SPECIFY FUNCTIONAL REQUIREMENTS:

The integration initiative must integrate the databases of the seven farm managers into a single solution that processes the estimates contained in the databases and generate a report for John Brown, indicating whether or not he will be able to supply *x* tons of potatoes on a specific date, and also whether the whole batch will come from one single farm, or various farms. The production planner should be able to confirm delivery of *x* tons of potatoes on a specific date at the touch of a button.

7.6.2 SPECIFY NON-FUNCTIONAL REQUIREMENTS

Non-functional requirements include the following:

- The solution must be simple to understand, use and maintain, since John Brown is not very computer literate.
- John Brown is considering buying more farms throughout the country, and the solution must be scalable enough to accommodate additional databases.
- Security is also important since John Brown does not want his competitors to gain access to this sensitive data and

the production planner should only be able to confirm delivery, not view confidential data such as crop estimates.

- The solution must make use of the existing databases used by the seven farm managers.

SPECIFY NON-FUNCTIONAL REQUIREMENTS:

- The solution must be simple to understand, use and maintain.
- The solution must be scalable enough to accommodate additional databases.
- Security is important.
- The solution must make use of the existing databases used by the seven farm managers.

7.6.3 EVALUATE INTEGRATION REQUIREMENTS

Once the functional and non-functional requirements are determined the stakeholders present at the meeting decide to evaluate these requirements after a short break, since it is difficult for the farm managers to attend a follow-up meeting to evaluate the requirements.

The integration requirements are evaluated against the business goals that were determined during Phase 2, to ensure that the integration requirements will lead to the development of an integration solution that will satisfy the business goals. It is concluded the integration requirements represent the true requirements of the current integration initiative.

EVALUATE INTEGRATION REQUIREMENTS:
The integration requirements are evaluated against the business goals and it is determined that the integration requirements represent the true requirements of the current integration initiative.

Armed with the business goals and the integration requirements the IT consultant is now ready to define the integration strategy.

7.7 PHASE 4: DEFINE INTEGRATION STRATEGY

After considering the business goals and the integration requirements the IT consultant defines the integration strategy to determine how these goals and requirements will be satisfied.

7.7.1 DETERMINE WHETHER INTRA- AND/OR INTER-ENTERPRISE INTEGRATION IS NECESSARY

Since the systems of the farm managers need to be integrated first it is clear that intra-enterprise integration should be achieved first on the side of John Brown, before the solution can be extended to inter-enterprise integration.

DETERMINE WHETHER INTRA- AND/OR INTER-ENTERPRISE INTEGRATION IS NECESSARY:
The solution will firstly employ intra-enterprise integration, which will eventually be extended to inter-enterprise integration.

7.7.2 DETERMINE INTEGRATION STAGES

The IT consultant, after consulting the various stakeholders, determines that the solution should be developed and deployed in two stages. It is firstly necessary to integrate the systems of the farm managers, and secondly, to extend the integrated solution to allow the production planner to confirm delivery. Since both these stages are relatively small-scale projects it is determined that the intra-integration stage should be completed within four months and the inter-enterprise integration stage within two months. The entire integration project should thus be completed within six months.

DETERMINE INTEGRATION STAGES:
The solution will be developed and deployed in two stages and completed within six months.

7.7.3 CONSIDER TECHNOLOGICAL FACTORS

The IT consultant analyzes the systems used by the seven farm managers to determine whether the systems offer support for integration and how the data controlled by these systems can be accessed. He also determines the platforms and operating systems that the systems run on.

CONSIDER TECHNOLOGICAL FACTORS:

The IT consultant determines the level of support for integration that the systems offer, as well as how the data controlled by these systems can be offered. The platforms and operating systems that the systems run on are also determined. He will need to consider these technological factors when he chooses an integration approach and integration technologies.

7.7.4 EVALUATE INTEGRATION STRATEGY

The integration strategy is evaluated and it is confirmed that the integration solution will firstly involve the intra-enterprise integration of the systems used by the farm managers and that the solution will then be extended to an inter-enterprise solution to allow the production planner to confirm delivery at the touch of a button. In addition, it is confirmed that developing the solution in two stages will improve the manageability of the project, and that six months is a reasonable time in which to develop and deploy the solution.

EVALUATE INTEGRATION STRATEGY:

The decision to develop the integration solution in two stages is confirmed and it is agreed that the solution will be developed and deployed within six months.

The IT consultant is now ready to select an integration approach to implement the integration strategy.

7.8 PHASE 5: SELECT INTEGRATION APPROACH AND VENDOR

7.8.1 SELECT INTEGRATION APPROACH(ES)

After considering the various integration approaches discussed in Chapter 4, the IT consultant concludes that the data-oriented integration approach will be the best solution for the current integration initiative. The solution will consolidate various source systems into a single target and then process the consolidated data to produce the desired reports. After further investigation the consultant determines that data federation would be the best data-oriented integration approach to follow.

SELECT INTEGRATION APPROACH(ES):
The data-oriented integration approach will be followed and a solution will be developed that employs data federation.

7.8.2 SELECT INTEGRATION VENDOR(S)

The consultant contacts various vendors specializing in data federation and finally selects a vendor that has a track record of satisfied customers who deployed similar solutions with proven results. The vendor will firstly develop and deploy the intra-enterprise integration solution, and then extend the solution to an inter-enterprise integration solution within the given time.

SELECT INTEGRATION VENDOR(S):

After contacting various vendors the IT consultant chooses *DataFed Corporation* to develop the integration solution.

7.8.3 EVALUATE SELECTED INTEGRATION APPROACH AND VENDOR

The IT consultant evaluates the selected integration approach to ensure that the chosen approach will lead to the development of an integration solution that will satisfy the business goals and integration requirements of the integration initiative. He also contacts some of the vendor's previous clients to determine whether they were satisfied with the integration solutions that the vendor developed and deployed for them.

EVALUATE SELECTED INTEGRATION APPROACH AND VENDOR:

The IT consultant confirms that an appropriate integration approach and vendor were selected.

7.9 PHASE 6: DEPLOYMENT

7.9.1 IMPLEMENT PILOT PROJECT

The intra-enterprise integration stage of the integration initiative is deployed as a pilot project to ensure that the

solution is working before it is extended to an inter-enterprise solution.

IMPLEMENT PILOT PROJECT:
The intra-enterprise stage of the integration initiative is deployed as a pilot project.

7.9.2 EVALUATE PILOT PROJECT

The pilot project were deployed without any problems and it is determined that it satisfied all business goals and integration requirements.

EVALUATE PILOT PROJECT:
The pilot project satisfies all business goals and integration requirements.

7.9.3 DEPLOY INTEGRATION STAGE

The inter-enterprise integration solution is developed and deployed.

DEPLOY INTEGRATION STAGE:
Once the intra-enterprise integration is finalized, the inter-enterprise integration stage is deployed.

7.9.4 EVALUATE INTEGRATION STAGE

The deployed inter-enterprise integration stage is evaluated to determine whether it satisfies all business goals and integration requirements related to this stage of the deployment.

EVALUATE INTEGRATION STAGE:
The inter-enterprise integration stage is evaluated and it is determined that all related business goals and integration requirements are satisfied

7.10 PHASE 7: EVALUATION

7.10.1 DETERMINE WHETHER BUSINESS GOALS AND INTEGRATION REQUIREMENTS ARE SATISFIED

Once the solution is deployed John Brown finds that he can immediately log into the new system and confirm whether he will be able to supply *x* tons of potatoes on the specified date. He can then contact the relevant farm manager or managers, if more that one farm needs to make up the batch, and instruct them to deliver the potatoes to Chippie's Chips.

The management of Chippie's Chips is also extremely pleased, since their production planner can now confirm delivery of x tons of potatoes on a specified date by just stipulating the amount of potatoes and the date. The system

then accesses the integrated system of John Brown to confirm whether he will be able to supply the potatoes.

DETERMINE WHETHER BUSINESS GOALS AND INTEGRATION REQUIREMENTS ARE SATISFIED:

All business goals and integration requirements are satisfied.

7.10.2 DOCUMENT PROBLEMS ENCOUNTERED

Since it was a relatively small-scale integration initiative there were no major problems encountered. An issue that did cause problems was the intervals at which the farm managers updated their databases. While this issue was not related to the development of the integration solution, it did have an impact on the efficiency of the deployed solution. If up-to-date information on the availability of potatoes was not available it had a negative impact on the efficiency of the integration solution. The problem was solved by making the farm managers aware of the importance of updating their databases at regular intervals when crop conditions changed.

John Brown is so impressed with his intra-enterprise integrated systems that he has contacted the IT consultant to determine whether the solution can be extended to include data about his farm employees. It takes him days at the end of the month to work out all the wages of his farm workers. His farm managers usually make printouts of the hours worked from their databases (the same databases that contain the crop estimates), and then fax them to Mr. Brown. He calculates the wages with the aid of these faxes and a calculator.

7.11 CONCLUSION

The purpose of this chapter is to illustrate how the Extended Enterprise Integration Model can be employed to support the establishment of an extended enterprise. While the example used is very simplified it illustrates how the various phases of the model work together to discover the objectives, choose an integration solution and finally deploy the integration solution to solve communication problems within the extended enterprise.

CHAPTER 8

e-BUSINESS INTEGRATION: BENEFITS AND BARRIERS

Chapter 7 illustrates how the Extended Enterprise Integration Model can be deployed to achieve intra- and inter-enterprise integration and solve communication problems between the members of the extended enterprise.

Chapter 8 will highlight the benefits of utilizing e-Business Integration in the establishment of an extended enterprise. In addition, some of the obstacles that can be expected when utilizing e-Business Integration will be emphasized.

Chapter 9 concludes the research presented in this dissertation.

8.1 INTRODUCTION

Even by following the phased approach introduced in Chapter 6, to integrate either intra- or inter-enterprise applications proves a daunting task. Despite the complexities, intra- and inter-enterprise integration, collectively known as e-Business Integration, plays an important role in supporting the establishment of an extended enterprise. This chapter investigates the role that e-Business Integration plays in supporting the successful establishment of an extended enterprise, as well as some of the realities of implementing an e-Business Integration strategy.

8.2 THE BENEFITS OF IMPLEMENTING AN e-BUSINESS INTEGRATION STRATEGY

As mentioned previously, the integration of applications between the members of the extended enterprise contributes significantly to the success of the extended enterprise. Thus, creating the extended enterprise inevitably involves the adoption of an e-Business Integration strategy to integrate the applications of the various members of the extended enterprise. The integration of applications, both on the intra- and inter-enterprise level, proves beneficial to the members of the extended enterprise for a number of reasons. A successfully integrated enterprise can bring about significant savings in cost, resources and time, while also improving information usage (EAI Industry Consortium, 2003[1]).

8.2.1 COST SAVINGS

While system management and maintenance costs are lowered, the addition of new applications to the organization also becomes easier and more economical (Shi & Gandhi, 2001). Automating manual processes can also reduce transaction costs. It is no longer necessary for employees to enter, process, and re-enter data into different applications, which also greatly reduces the occurrence of errors associated with the high degree of human interaction involved in processing transactions manually (Mohindroo, 2002). Other cost savings include lower operational, production and inventory costs. Automating administrative processes, such as procurement, results in significant operational cost savings. It is estimated that the automation of procurement by integrating order systems directly with suppliers can drive down handling costs by up to 85 percent, while also cutting the cost of supplies. Poor communication between designers and subcontractors often results in production delays and budget overruns in manufacturing, which drives up costs. These delays can be eliminated and production costs reduced by integrating and automating business processes that support the manufacture and supply of products and components. Inventory levels are influenced by a number of factors, including market trends, sales projections, pipeline status, production planning and supplier reliability. By sharing this information in real-time with customers, distribution channels and suppliers, inventory planning can be improved and unnecessary high inventory levels reduced, which lowers inventory costs (Lublinsky, 2002).

8.2.2 RESOURCE SAVINGS

More value can be derived from legacy- and other existing reliable applications by integrating them with the rest of the organization. These applications do not become obsolete and there is no need for expensive application replacement plans (Ravishankar, 2003). The acquisition of new applications should also not compromise functionality. It is possible to buy the best suitable product for the targeted business process and the organization, since it can be effectively integrated with existing applications (Ravishankar, 2003). The automation of manual processes also leads to increased employee efficiency and productivity – their time can be spent on more important issues than entering, processing, and re-entering data into different applications (Gartner Holdings, 2003). Integration between the members of the extended enterprise also allows the optimization of resource usage due to improved forecasting, reduced inventory levels and shorter cycle times (eQuarius, 2003). Expansion and growth through mergers and acquisitions can also be handled more efficiently when the applications are already integrated, since it is easier to then integrate the new applications with existing ones. This results in an easier transition and the benefits of the merger or acquisition can be realized without delay (Gartner Holdings, 2003).

8.2.3 TIME SAVINGS

The integration of Information Systems result in faster, more efficient, and more adaptive business processes, which enable the organization to react to dynamic market conditions faster

than the competition. The organization is also able to do the same thing faster than the competition, which results in a greater capacity to efficiently do more, for example accept more orders. This allows either the reduction of prices (to increase order volume) or the ability to retain higher margins for reinvestment in the organization (Gartner Holdings, 2003). By integrating information across the entire extended enterprise landscape, the organization can enhance the stickiness of their web sites with interesting, up-to-date and relevant information, which not only improves customer satisfaction, but also makes the web site attractive to advertisers (Hayes, 2002). Customers can use the web site to access account information, view product catalogues, place orders and resolve problems. The customer's order can also be confirmed within hours instead of days. Except for the fact that transactions can be handled faster when automated, there is also the added advantage of easily adjusting to customer's needs in a timely manner (Shi & Gandhi, 2001).

In the highly competitive global economy it is unacceptable to leave customers waiting for products because the products are not in stock or suppliers are not able to deliver on time. It is a significant problem for organizations operating over the Internet, where the competition is just a mouse-click away. When ordering products over the Internet customers demand immediate confirmation on product availability. By integrating relevant applications across the entire extended enterprise, the customer can receive a real-time response regarding configuration options, pricing, and availability. Integrating all customer information and feedback enables personalized customer interactions and leads to a more efficient buying

process where unnecessary delays are eliminated and customer loyalty increased (Lublinsky, 2002).

The integration of Information Systems also leads to a faster time-to-market with new products and services. Bringing a new product or service to market involves many highly collaborative processes between researchers, production partners, distribution partners and customers. Time is often lost when communication breaks down between the organization and any of these stakeholders. By automating these collaborative processes and sharing information such as customer requirements, forecasts, production schedules and inventory levels, the organization is able to understand market needs and trends better, which allows earlier initial orders, better production planning, and higher margins (Lublinsky, 2002). Launching new products and services ahead of competitors also increases competitive advantage.

The organization can also take advantage of emerging opportunities more quickly by evaluating information across the entire extended enterprise landscape. Streamlining the flow of information between people, processes and applications also eliminates unnecessary delays (Hayes, 2002).

8.2.4 IMPROVED INFORMATION USAGE

Managers are able to make better business decisions and more accurate forecasts, since timely and accurate information that is integrated across the extended enterprise is readily available. Increased productivity is also possible when

up-to-date information is available to employees, customers, suppliers and other business partners as, and when they need it – whether they access the information from a personal computer, laptop or mobile phone (Laroia & Sayavedra, 2003). Since the manual entry, processing and re-entry of data between applications are no longer necessary, the quality of information is also improved. In addition, the integration of information from all areas of the extended enterprise increases the speed, efficiency and responsiveness of customer service. All customer information is integrated, therefore the customer can receive personalized service and the organization is able to respond to changing customer demands and resolve any problems that may occur in a timely and cost effective manner (Tibco, 2002). It is also possible to reach, and serve, a broader customer base over the Internet by integrating new Web applications with core business applications. In addition, effective communication between the members of the extended enterprise improves agility, which allows the organization to promptly act on new opportunities to increase revenue and reduce costs (eQuarius, 2003).

Improved information usage also leads to increased supply chain efficiency by allowing the members of the extended enterprise to improve forecasting, reduce inventory, and cut cycle times (Serve, Yen, Wang & Lin, 2002). This is the result of automated processes, better visibility into the entire supply chain and the ability of all members to share information and act on that information, as well as the ability to respond to any event in the extended enterprise in a timely manner (Mohindroo, 2002; Pinkston, 2001). Examples of improved supply chain efficiency include the following:

- Automation of the inventory management process can result in reduced inventory requirements, fewer errors from manual handling of data, less time required to manage and track inventory manually and longer-term inventory planning (Gartner Holdings, 2003).
- Faster product delivery is possible by automating the order management process, integration between areas such as manufacturing and shipping, and through effective communication and integration with suppliers and other business partners (Hayes, 2002). The shorter order fulfilment cycle time not only improves customer satisfaction, but the organization is also able to accept a higher volume of orders (Gartner Holdings, 2003).

From the prior discussion it can be derived that there are various benefits associated with the establishment of an extended enterprise though the integration of the Information Systems of the members of the extended enterprise. In summary, from a business perspective, collaboration between the Information Systems of the members of the extended enterprise can lead to the following advantages:

- increased competitive advantage
- increased revenue
- increased supply chain efficiency
- improved ability to handle growth
- faster time-to-market with new products and services, and a
- better sales process.

In spite of this, e-Business Integration is by no means the silver bullet that will solve all extended enterprise communication problems. e-Business Integration remains a daunting task that requires thorough

planning and commitment from all the members of the extended enterprise.

8.3 THE REALITIES OF IMPLEMENTING AN e-BUSINESS INTEGRATION STRATEGY

While Figure 2.1 (Chapter 2) provided a simplified Information Systems based view of the extended enterprise, there are a magnitude of other operational systems within the organization, which may include legacy, manufacturing, inventory, shipping, financial and sales systems, to mention a few. It is estimated that the average Global 2000 organization consists of approximately 52 enterprise applications (META Group, 2001). In addition, when integrating these incompatible systems it is necessary to consider the integration requirements along with the constraints and requirements of the existing systems (Themistocleous & Irani, 2002). Considering this, and taking into account that these applications must be successfully integrated within the organization before proceeding to integrating with other members of the extended enterprise, it becomes clear that implementing an e-Business Integration solution can be a complex task.

Technology is evolving rapidly with new integration products and developments introduced every day. This fact, along with the high cost and risk associated with an e-Business Integration solution, has made many executives and managers apprehensive of it, which makes it difficult for enterprises to justify investing in these projects. For many enterprises an e-Business Integration solution is seen as a luxury, and not a necessity.

Another concern when exposing information assets through e-Business Integration is that of security. As enterprise information is transmitted electronically, the chance of being attacked by hackers increases and it is thus necessary to implement technology that can improve security in all stages of the e-Business Integration solution (Rodgers, Yen & Chou, 2002).

Despite the various benefits of applying e-Business Integration in the establishment of an extended enterprise cited in the previous sections, it must be remembered that integration projects remain daunting and expensive tasks in a world where technology is changing rapidly. Executives and managers need to be aware of the necessity of e-Business Integration initiatives to ensure competitive advantage for the extended enterprise. The enterprise should analyze the long-term impacts of an integration strategy and determine which e-Business Integration solution will be most beneficial to the enterprise. According to the META Group, cost is not the only selection criteria when evaluating integration solutions. Some of the factors to consider include the infrastructure readiness of the enterprise to support integration, the flexibility and scalability of the chosen integration solution, the willingness of all stakeholders to support the integration solution, and so forth (META Group, 2001). All stakeholders need to share the same vision of the expected outcome to ensure that a solution most beneficial to the extended enterprise as a whole is chosen.

8.4 CONCLUSION

While there are various benefits associated with applying e-Business Integration, enterprises must be realistic about their integration initiatives. e-Business Integration projects are complex long-term

initiatives that not only have far-reaching effects on the individual organization, but on the whole extended enterprise as well. When embarking on an integration project it is necessary to consider all the integration products and developments in the field before choosing the tools necessary to deliver a total integration solution that meets the specific needs of the enterprise.

The use of e-Business Integration in the establishment of an extended enterprise provides a clear competitive advantage, but the commitment of all stakeholders in the enterprise is of the utmost importance to ensure the success of a potentially lengthy project. The complexity of the project must be understood, and thorough planning and a sound technical approach are necessary to ensure that the project can be completed on time, within the budget, and, ultimately, result in a fully integrated extended enterprise where it is easier to extend to customers, suppliers and other business partners. Overcoming the challenges facing an e-Business Integration initiative will result in a highly responsive extended enterprise with a shared information infrastructure that is easier to enhance and build upon at a future date.

CHAPTER 9

CONCLUSION

Chapter 8 discusses the benefits and barriers of utilizing e-Business Integration in the establishment of an extended enterprise.

This chapter concludes the research presented in this dissertation and suggests some areas suitable for future research.

9.1 INTRODUCTION

The research performed for this project results in the development of the Extended Enterprise Integration Model. In addition to the development of the model, the following objectives are established in Chapter 1:

1) ***"The principal objective of this study is to provide business leaders with an approach that can be followed when implementing e-Business Integration to support the establishment of an extended enterprise."***

   Chapter 6 presents the Extended Enterprise Integration Model. This model represents the phases that integration initiatives, both intra- and inter-enterprise, undergo to establish integrated communication between the various members of the extended enterprise. In Chapter 7 an example is used to illustrate how the Extended Enterprise Integration Model can be used to improve communication between members of the extended enterprise.

2) ***"It is also necessary to introduce business leaders to the concept of an extended enterprise and highlight the supporting role that e-Business Integration can play in establishing an extended enterprise."***

   Chapter 2 introduces the concept of an extended enterprise and discusses the difference between an extended- and virtual enterprise. The extended enterprise is also discussed from an Information Systems-based viewpoint. In Chapter 3 e-Business Integration, which includes both intra- and inter-enterprise integration, is discussed. This chapter highlights how the integration

of various systems within and between enterprises can ease communication problems amongst the various members of the extended enterprise. Chapter 8 investigates the role that e-Business Integration plays in supporting the successful establishment of an extended enterprise, and includes a discussion of the realities of implementing an e-Business Integration strategy.

3) ***"In addition, it is necessary that these business leaders have a basic understanding of the various integration approaches that can be applied to integrate systems, as well as the enabling middleware technologies and standards employed in these integration approaches."***

Chapter 4 discusses the various integration approaches that can be employed to achieve either intra- and/or inter-enterprise integration and Chapter 5 discusses the integration technologies and standards that can be employed in the actual integration solutions. The aim of these chapters is not to provide a complete discussion of technology implementations, but rather to describe the concepts behind the integration approaches, technologies and standards in order to provide business leaders with a basic understanding of theses concepts.

It becomes apparent from the above discussion that all of the objectives established at the beginning of this research project were accomplished. The following section will provide a brief overview of each chapter contained in this dissertation.

9.2 CHAPTER OVERVIEW

The purpose of this section is to provide the reader with an overview of the work presented in this dissertation.

Chapter 1 – Introduction

Chapter 1 provides an introduction to the problem domain and states the problem definition and objectives of this research project.

Chapter 2 – The Extended Enterprise

This chapter introduces the concept of an extended enterprise and discusses the difference between an extended- and a virtual enterprise. The extended enterprise is also discussed from an Information Systems-based viewpoint.

Chapter 3 – e-Business Integration

e-Business Integration, which includes intra- and inter-enterprise integration, is discussed. In addition, the role that intra- and inter-enterprise integration can play in solving various communication problems between the members of the extended enterprise is highlighted.

Chapter 4 – Integration Approaches

This chapter describes the various integration approaches that can be employed to achieve either intra- and/or inter-enterprise integration.

Chapter 5 – Integration Technologies and Standards

In this chapter several integration technologies and standards that can be employed to develop either intra- and/or inter-enterprise integration solutions are discussed.

Chapter 6 – Towards an Extended Enterprise: The Extended Enterprise Integration Model

This chapter introduces the Extended Enterprise Integration Model. This model proposes a phased approach to integrate intra- and/or inter-enterprise systems to solve communication problems between the members of the extended enterprise.

Chapter 7 – Employing the Extended Enterprise Integration Model

In this chapter an example is used to illustrate how the phased approach proposed by the Extended Enterprise Integration Model can be used to support the establishment of an extended enterprise.

Chapter 8 – e-Business Integration: Benefits and Barriers

This chapter discusses the role that e-Business Integration plays in supporting the establishment of an extended enterprise, as well as the realities of employing e-Business Integration.

Chapter 9 – Conclusion

The research is concluded with this chapter.

Future research directions will be discussed in the following section.

9.3 FUTURE RESEARCH

Possible directions for future research (related to the Extended Enterprise Integration Model) include the following:

- The status of integration initiatives in South Africa. Such a project should conduct a study of the status of integration initiatives in the South African context. Difficulties encountered during these projects can be identified and the capacity of the

Extended Enterprise Integration Model to ease these difficulties, tested. This could lead to further enhancements to the model.

- Implementation of the Extended Enterprise Integration Model. The Extended Enterprise Integration Model can be used to establish either intra- and/or inter-enterprise integration within and/or amongst organizations. Such a study will determine the viability of using the model in industry and will highlight phases of the model that require modifications.

9.4 CONCLUSION

This chapter concludes this dissertation and illustrates that all of the objectives established at the beginning of this research project were accomplished. An overview of the information covered in the various chapters of this dissertation was provided and future research directions suggested.

In the global marketplace internal efficiency no longer ensures the survival of an organization, and strategic relations with customers, suppliers and other business partners have become increasingly important. In this extended enterprise there is a need to share relevant information, business processes and services with these stakeholders. While communication between these stakeholders is nothing new, the technology that enables the communication has evolved vastly. Since the beginning of time the communication mechanisms between organizations have evolved from smoke signals to pigeons to snail mail to telephones to faxes to e-mails to EDI, and so forth. Today, there is a need for real-time, or near real-time, communication between organizations to share relevant information, business processes and services to achieve a competitive edge over competitors. To achieve this level of collaboration it is necessary to integrate relevant systems

within and between organizations to share these business processes, services and information. e-Business Integration enables this integration, and the Extended Enterprise Integration Model presented in this dissertation proposes a phased approach to deploy e-Business Integration in support of the establishment of an extended enterprise. David S. Linthicum, an internationally known application integration and e-Business expert, feels so strongly about the importance of integration between the members of the extended enterprise that he makes the following statement in his book titled “B2B Application Integration – e-Business Enable Your Enterprise” (Linthicum, 2003[1]):

> *“Either businesses are getting on board and automating their common business events, inter- and intra-organization, or they are preparing to exit their market. There is no in-between.”*

BIBLIOGRAPHY

Aberdeen. (2003). *Worldwide Enterprise Business Integration Spending: Forecast and Analysis 2002–2006*. Boston: Aberdeen Group.

Acharya, R. (2003). EAI: A Business Perspective. *eAI Journal*, April 2003, pp. 37-44.

Active Software. (2000). *Business-to-Business Integration for the New Network Economy*. Santa-Clara, CA: Active Software.

Alodar Systems. (2003). *Enterprise Application Integration (EAI) Glossary*. Retrieved June 17, 2003 from http://www.alodar.com/glossary.html.

Alpha Technologies. (2003). *Frequently Asked Questions – What is EAI?* Retrieved April 9, 2003 from http://www.alphatechnologies.net/faq.htm.

Alshawi, S. (2001). Logistics in the Internet Age: Towards a Holistic Information and Processes Picture. *Logistics Information Management*, Volume 14 Number 4, pp. 235-241.

Babu, M. (2003). Four Questions Business Managers Should Ask About EAI. *eAI Journal*, April 2003, pp. 14-16.

Baclawski, K. (1998). *Java RMI Tutorial*. Retrieved November 09, 2004 from http://www.ccs.neu.edu/home/kenb/com3337/rmi_tut.html.

BEA Systems Inc. (2002). *BEAWebLogic Integration: Introducing BEA WebLogic Integration*. Place: BEA Systems Inc.

Bednar, E. (2002). Adapting to Your Environment: Self-Generating Adapters to Change EAI Landscape. *eAI Journal*, October 2002, pp. 18-20.

Bergholz, A. (2000). Extending Your Markup: An XML Tutorial. *IEEE Internet Computing*, July/August 2000, pp. 74-79.

Bloor Research. (2004). *Data Integration Volume 2 - Data Federation*. Retrieved October 07, 2004 from http://www.gii.co.jp/english/bl19484_data_integration.html.

Bose, R. (2002). Customer Relationship Management: Key Components for IT Success. *Industrial Management and Data Systems Journal*, 102/2, pp. 89-97.

Bridges, C., Bond, J. & Torrance, S. (2001). *The insiders' guide to B2B integration: A report from the European Integration Summit*. Netmarkets Europe 2001.

Britton, C. (2003). *IT Architectures and Middleware*. Boston: Addison-Wesley.

Brown, L. (2001). *What is Enterprise Application Integration?* Retrieved June 17, 2003 from http://www.systeminnovations.net/definingEAI.htm.

Brown, T.M. (2003). A Letter to Your CEO. *Business Integration Journal*, July 2003, pp. 1-2.

Browne, J. & Zhang, J. (1999), Extended and Virtual Enterprises - Similarities and Differences. *International Journal of Agile Management Systems*, 1/1, pp. 30-36.

Bussler, C. (2003). B2B *Integration: Concepts and Architecture*. Berlin: Springer.

Cape Clear Software Inc. (2004). *Enterprise Service Bus (ESB) Solution*. Waltham, MA: Cape Clear Software Inc.

Carl, A. (2003). 10 Keys to Successful EAI. *eAI Journal*, March 2003, pp. 8-10.

Carnegie Mellon Software Engineering Institute. (1997[1]). *Common Object Request Broker Architecture (CORBA)*. Retrieved October 18, 2004 from http://www.sei.cmu.edu/str/descriptions/corba.html.

Carnegie Mellon Software Engineering Institute. (1997[2]). *Message-Oriented Middleware*. Retrieved October 18, 2004 from http://www.sei.cmu.edu/str/descriptions/momt.html.

Carnegie Mellon Software Engineering Institute. (1997[3]). *Middleware*. Retrieved October 18, 2004 from http://www.sei.cmu.edu/str/descriptions/middleware.html.

Carnegie Mellon Software Engineering Institute. (1997[4]). *Remote Procedure Call*. Retrieved October 18, 2004 from http://www.sei.cmu.edu/str/descriptions/rpc.html.

Carnegie Mellon Software Engineering Institute. (1997[5]). *Transaction Processing Monitor Technology*. Retrieved October 18, 2004 from http://www.sei.cmu.edu/str/descriptions/tpmt.html.

Carnegie Mellon Software Engineering Institute. (2001). *Component Object Model (COM), DCOM, and Related Capabilities*. Retrieved October 18, 2004 from http://www.sei.cmu.edu/str/descriptions/com.html.

Carnegie Mellon Software Engineering Institute. (2004). *Object Request Broker*. Retrieved October 18, 2004 from http://www.sei.cmu.edu/str/descriptions/orb.html.

Carzoli, M. (2002). *Executing an EAI Project Strategy*. Retrieved August 21, 2003 from http://www.ebizq.net/topics/int_sbp/features/1744.html.

Channabasavaiah, K., Holley, K. & Tuggle, E.M. (2004). *Migrating to a Service-Oriented Architecture*. Somers, New York: IBM Corporation.

Cherry Tree. (2000). *Extended Enterprise Applications*. Edina, Minnesota: Cherry Tree.

Christos, V. (2002). *Web Services as an Application Integration Tool: Creating an Evaluation Framework*. Middlesex, UK: Brunel University.

Cousins, P. & Casanova, I. (2003). *Service-Oriented Integration: A Strategy Brief*. Dublin: IONA Technologies.

Craggs, S. (2003). *Best-of-Breed ESBs: Identifying Best-of-Breed Characteristics in Enterprise Services Buses (ESBs)*. Place: Saint Consulting Limited.

CSK Software. (2003). *Financial EAI With X-Gen*. Frankfurt: CSK Software.

DataMirror. (2002). *In Pursuit of the Extended Enterprise: Integrating CRM, ERP and e-Business Applications*. Markham, Canada: DataMirror Corporation.

Davydov, M.M. (2000). *Getting ERPs on the Same Page*. Retrieved June 10, 2003 from http://www.intelligenterp.com/feature/2000/10/davydovOct20.shtml.

De Montcheuil, Y. (2004). *Data Integration Without the Hassle*. Place: Technical Enterprises, Inc.

Duffy, J. (2002). The Agile Organization. *eInform*, Volume 4 Issue 1, pp. 1-13.

EAI Industry Consortium. (2003[1]). *Enterprise Application Integration Definition*. Retrieved April 9, 2003 from http://www.eaiindustry.org/.

EAI Industry Consortium. (2003[2]). *What is EAI?* Retrieved April 9, 2003 from http://www.eaiindustry.org/ page.php?content=about_faq.

ebXML. (2004). *About ebXML*. Retrieved November 22, 2004 from http://www.ebxml.org/geninfo.htm.

Enosys Software. (2001). *Enterprise Information Integration: Integrating the Real-time Enterprise*. San Jose, CA: Enosys Software.

eQuarius. (2003). *B2B Solutions*. Bellevue, WA: eQuarius.

Evgeniou, T. (2002). *Building the Adaptive Enterprise: Information Strategies for Successful Management of Complex, Global Corporations in Times of Change*. France: INSEAD.

Fallahi, B. (2004). *What is JDBC?* Retrieved November 09, 2004 from http://www.devx.com/tips/Tip/5736.

Fiorano Software, Inc. (2001). *Redefining the EAI Landscape*. Los Gatos, CA: Fiorano Software, Inc.

Fiorano Software, Inc. (2003). *Demystifying Enterprise Service Bus Technology*. Los Gatos, CA: Fiorano Software, Inc.

Fisher, P. (2001). Renew, Don't Rewrite, Legacy Systems. *ADTMAg.com*, December 2001. Retrieved May 19, 2003 from http://www.adtmag.com/article.asp?id=5751.

Friedman, T. (2003). *Assessing the Retailer's Payback from Application Integration*. Newton Upper Falls, MA: Retail Systems Alert Group Inc.

Gartner Holdings. (2003). *Going Beyond IT ROI—Estimating the Business Value of Process Integration Solutions*. Ireland: Gartner Holdings.

Global Exchange Services. (2003). *Web Services White Paper*. Gaithersburg, Maryland: Global Exchange Services.

Gold-Bernstein, B. (2003[1]). *Business Integration Strategy*. Retrieved October 14, 2003 from http://www.ebizq.net/filelib/2929.html.

Gold-Bernstein, B. (2003[2]). *Creating an Integration Strategy*. Retrieved October 14, 2003 from http://www.ebizq.net/topics/int_sbp/features/2928.html/?3107nl.

Gold-Bernstein, B. (2004). *Integration Best Practices: Beyond The SOA, Composite Apps And Web Services Hype*. Retrieved October 27, 2004 from http://www.ebizq.net/topics/int_arch/features/4894.html?related.

Gosain, S. & Thillairajah, V. (2002). *EAI: The Business Drivers and Technical Challenges*. Retrieved August 21, 2003 from http://www.ebizq.net/topics/int_sbp/features/1766.html.

Gupta, M. (2003) *Elements of a Successful Enterprise Integration*. Troy, MI: Syntel.

Hailstone, R., Vermeulen, P. & Warmerdam, M. (2002). *IT and Business Integration: Moving Towards the Agile Organization*. Framingham, MA: IDC.

Harris, J.G. & Cantrell, S. (2002). Enterporise Solutions Integration Technology. *Next Generation Enterprise Solutions*, Issue 3, March 2002, pp. 1-4.

Hayes, H.A. (2002). *Creating a Flexible Infrastructure for Integrating Information*. San Jose, CA: IBM Corporation.

Hendrick, S.D. & Hendrick, K.E. (2002). *Vendor Needs and Strategies*. Framingham, MA: IDC.

Hindi, F. & Kuhbock, M. (2003). *Eight Warning Signs Of Poor Business Integration*. Retrieved August 21, 2003 from http://www.ebizq.net/topics/eai/features/2054.html.

Hohpe, G. & Woolf, B. (2004). *Enterprise Integration Styles*. Retrieved November 23, 2004 from http://www.informit.com/articles/article.asp?p=169483.

Hopkins, M. & Thomas, D. (1998). *Data Replication With InterBase*. Retrieved September 30, 2004 from http://www.ibphoenix.com/main.nfs?a=ibphoenix&s=1094768346:252050&page=ibp_howto10.

IBM Corporation. (2000). *Business-to-Business Integration With Trading Partner Agreements*. Hampshire, UK: IBM Corporation.

IFS. (2003). *A Step-by-Step Guide to the Extended Enterprise*. Retrieved February 11, 2003 from http://www.ifsde.com/produkte/applications/pdf/step_by_step_guide.pdf.

Information Builders. (1999). *The Critical Role of Business Intelligence in e-Business*. New York: Information Builders.

Informix. (1998). *Enterprise Replication: A High-Performance Solution for Distributing and Sharing Information*. California: Informix.

iSource. (2003). *Enterprise Application Integration 101*. Retrieved April 9, 2003 from http://www.isourceonline.com/article.asp?article_id=3010.

iStart. (2003). *Portals - Know What You're Getting Into*. Retrieved October 07, 2004 from http://www.istart.co.nz/index/HM20/PC0/PV21902/EX245/AR23173.

JavaSoft. (1997). *What is JDBC?* Retrieved November 09, 2004 from http://java.sun.com/docs/books/jdbc/intro.html.

Jhingran, A.D., Mattos, N. & Pirahesh, H. (2002). Information Integration: A Research Agenda. *IBM Systems Journal*, Volume 41 Number 4, pp. 55-62.

Kalakota, R. & Robinson, M. (2000). *e-Business 2.0 – Roadmap for Success*. Boston: Addison-Wesley.

Krass, P. (2003). Composite Applications: Easy-to-build Software That Works the Way You Do. *CFO IT*, Winter 2003 Issue. Retrieved October 07, 2004 from http://www.cfo.com/article.cfm/3010947/c_3046602?f=insidecfo.

LaFaire, D. (2003). *Are you Ready for B2B?* Retrieved June 25, 2003 from http://www.eaijournal.com/Article.asp?ArticleID=156&DepartmentId=2.

Laftsidis, A. (2004). *Enterprise Application Integration*. Retrieved September 30, 2004 from http://www.ida.his.se/ida/kurser/informationsystems_engineering/kursmaterial/forelasningar/Chapter15.pdf.

Laftsidis. A. (n.d.). *Middleware in EAI*. Sweden: IBM Corporation.

Laroia, A., & Sayavedra, L. (2003). EAI Business Drivers. *eAI Journal*, February 2003, pp. 27-29.

Lawlor, A. (2003). Integration in an Era of Tight IT Budgets. *eAI Journal*, April 2003, pp. 57-58.

Linthicum, D.S. (1999). *Enterprise Application Integration*. Boston: Addison-Wesley.

Linthicum, D.S. (2000). *EAI Application Integration Exposed*. Retrieved January 27, 2003 from http://www.softwaremag.com/archive/2000feb/EAI.html.

Linthicum, D.S. (2001). Chasing the Tail of the Beast. *Software Magazine*, February/March 2001. Retrieved September 16, 2004 from http://www.softwaremag.com/archive/2001feb/BeastsTail.html.

Linthicum, D.S. (2003[1]). *B2B Application Integration: e-Business-Enable Your Enterprise*. Boston: Addison-Wesley.

Linthicum, D.S. (2003[2]). *Next Generation Application Integration: From Simple Information to Web Services*. Boston: Addison-Wesley.

Linthicum, D.S. (2004[1]). *ebXML's Move To SOA: Clearly A Good One*. Retrieved June 28, 2004 from http://www.ebizq.net/hot_topics/soa/features/4323.html.

Linthicum, D.S. (2004[2]). The Next Wave: Integrating Applications With a Rich Client Interface. *Business Integration Journal*, July 2004, p. 17.

Longo, J.R. (2001). The ABCs of Enterprise Application Integration. *eAI Journal*, May 2001, pp. 56-58.

Lublinsky, B. (2002). Approaches to B2B Integration. *eAI Journal*, February 2002, pp. 38-47.

Mann, J.E. (1999). Workflow and EAI. *eAI Journal*, September/October 1999, pp. 49-53.

Maverick, G. (2003). EAI Project Management. *Business Integration Journal*, November 2003, pp. 48-50.

McDaniel, T. (2001). Ten Pillars of Business Process Management. *eAI Journal*, November 2001, pp. 30-34.

Mercator. (2000). *Transforming the Commerce Chain Through B2B Integration*. Wilton, Connecticut: Mercator Software.

META Group. (2001). *Understanding the Role of Integration in e-Business Success*. Stamford: META Group.

Micrososft Corporation. (1999). *Microsoft Announces Finalized BizTalk Framework*. Retrieved November 22, 2004 from http://www.microsoft.com/presspass/press/1999/Dec99/BIZTALK1.0pr.asp.

Microsoft Corporation. (2002). *Enterprise Application Integration – A Business Decision*. New York, NY: Microsoft Corporation.

Microsoft Corporation. (2004[1]). *Message Broker*. Retrieved November 18, 2004 from http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnpag/html/archmessagebroker.asp.

Microsoft Corporation. (2004[2]). *Micrososft ODBC Drivers*. Retrieved November 09, 2004 from http://msdn.microsoft.com/library/default.asp?url=/library/en-us/odbc/htm/odbcodbc_drivers_overview.asp.

Microsoft Corporation. (2004[3]). *ODBC*. Retrieved November 09, 2004 from http://msdn.microsoft.com/library/default.asp?url=/library/en-us/vccore98/HTML/_core_odbc.asp.

Microsoft Corporation. (2004[4]). *OLE DB*. Retrieved November 09, 2004 from http://msdn.microsoft.com/library/default.asp?url=/library/en-us/oledb/htm/dasdkoledboverview.asp.

Microsoft Corporation. (2004[5]). *Q&A: Replication in Microsoft Access for Windows 95*. Retrieved September 30, 2004 from http://msdn.microsoft.com/archive/default.asp?url=/archive/en-us/dnaraccessdev/html/ODC_ReplicationinAccess95.asp.

Microsoft Corporation. (2004[6]). *What is ODBC?* Retrieved November 09, 2004 from http://msdn.microsoft.com/library/default.asp?url=/library/en-us/odbc/htm/odbcodbc_drivers_overview.asp.

Microsoft Corporation. (2005). *COM: Component Object Model Technologies*. Retrieved January 3, 2005 from http://www.microsoft.com/com/default.mspx

Mohindroo, A. (2002). *Deriving Value from an Extended Enterprise with Oracle9iAS Integration - An Oracle Business White Paper*. Redwood Shores, CA: Oracle Corporation.

Morgenthal, J.P. (2001). Implementing B2Bi in the Age of Increased Internet Security. *eAI Journal*, November 2001, pp. 8-10.

Nickerson, B. (2002), Six Steps for Collaborative Success. *eAI Journal*, June 2002, pp. 44-45.

Object Management Group. (2004). *CORBA Basics*. Retrieved November 09, 2004 from http://www.omg.org/gettingstarted/corbafaq.htm.

Oracle. (2002). *Deriving Value from an Extended Enterprise with Oracle9iAS Integration*. Retrieved Month Date, Year, from http://www.oracle.com/ip/deploy/ias/docs/oracle9ias_integration_bwp.pdf.

Patil, S. & Simha, N. (2004). Integration Approaches: Web Services vs Distributed Component Models. *Web Services Journal*, Volume 3 Issue 4. Retrieved September 16, 2004 from http://www.sys-con.com/webservices/article.cfm?id=508.

Petrassi, J. (2004). How BPM Impacts Consulting Services. *BP Trends*, April 2004, pp. 1-4.

Pinkston, J. (2001). The Ins and Outs of Integration – How EAI Differs From B2B Integration. *eAI Journal*, August 2001, pp. 48-52.

Power, D.J., Sohal, A.S. & Rahman, S. (2001). Critical Success Factors in Agile Supply Chain Management - An Empirical Study. *International Journal of Physical Distribution & Logistics Management*, Volume 31 Number 4, pp. 247-265.

Raj, G.S. (1998). *A Detailed Comparison of CORBA, DCOM and Java/RMI*. Retrieved November -09, 2004 from http://my.execpc.com/~gopalan/misc/compare.html.

Raj, P. & Ishii, N. (2004) *The Value of CORBA in Enterprise Application Integration (EAI)*. Retrieved October 11, 2004 from http://www.sei.cmu.edu/str/descriptions/clientserver_body.html.

Ravishankar, P. (2003). *EAI – Viability for an SME*. Pittsburgh, PA: Business Functions.

Reese, A.K. (2002). *Enterprise Application Integration 101*. Retrieved May 19, 2003 from http://www.isourceonline.com/article.asp?article_id=2922.

Reilly, D. (2000). *Introduction to Java RMI*. Retrieved November 09, 2004 from http://www.javacoffeebreak.com/articles/javarmi/javarmi.html.

Roch, E. (2002). *Application Integration Business Case and Technology Trends*. Palo Alto, CA: Tibco.

Rodgers, J.A., Yen, D.C. & Chou, D.C. (2002). Developing e-Business: A Strategic Approach. *Information Management & Computer Security*, 10/4 [2002], pp. 184-192.

RosettaNet. (2004[1]). *e-Business Standards Initiative*. Retrieved November 21, 2004 from http://www.rosettanet.org/RosettaNet/Rooms/DisplayPages/LayoutInitial?Container=com.webridge.entity.Entity%5BOID%5BD81335C9911BDB4592BF89D954A7B183%5D%5D.

RosettaNet. (2004[2]). *RosettaNet General FAQ*. Retrieved November 21, 2004 from http://www.rosettanet.org/RosettaNet/Rooms/DisplayPages/LayoutInitial?Container=com.webridge.entity.Entity[OID[22107C9E15B9684D94D836A18BB2CE80.

RosettaNet. (2004[3]). *RosettaNet Home*. Retrieved November 21, 2004 from http://www.rosettanet.org/RosettaNet/Rooms/DisplayPages/LayoutInitial.

Schmidt, J.G. (2003). *EAI Methodology: The Theory of Application Integration*. Calgary, Alberta: EAI Industry Consortium.

Schulte, R. (2001). Understanding the Convergence of B2B and Internal Application Integration: Questions and Answers. San Jose, CA: , Gartner Inc.

Segue Software. (2004). *A Corba Primer*. Retrieved October 25, 2004 from http://www.omg.org/news/whitepapers/seguecorba.pdf.

Serve, M., Yen, D.C., Wang, J.C. & Lin, B. (2002). *B2B-enhanced Supply Chain Process: Toward Building Virtual Enterprises*. Business Process Management Journal, Volume 8 Number 3, pp. 245-253.

Shi, P. & Gandhi, S. (2001). *Enterprise Application Integration Viewpoint*. Chicago, IL: Center for Technology Innovation.

Skinstad, R. (2000). *Business Process Integration Through XML*. Vallentuna, Sweden: Netfish Technologies, Inc.

Smith, H. & Fingar, P. (2003). *Business Process Management - The Third Wave*. Tampa, Florida: Meghan-Kiffer Press.

Spiers, R. (2001). *An Integration Framework*. London: Clarity Integration Ltd.

Stokes, N. (2001[1]). EAI and Beyond: A Multi-Level Flow Model. *eAI Journal*, May 2001, pp. 43-48.

Stokes, N. (2001[2]). The Extended Enterprise Comes of Age. *eAI Journal*, September 2001, pp. 42-46.

Sturim, R. (1999). *Achieving Competitive Advantage through Supply Chain Integration*. Sunnyvale, CA: Vitria Technology, Inc.

Sun Microsystems. (1997). *Definition of Terms: Java Remote Method Invocation*. Retrieved November 17, 2004 from http://java.sun.com/j2se/1.5.0/docs/guide/rmi/spec/rmi-objmodel3.html.

Sun Microsystems. (2002). *Integration for the Extended Enterprise*. Santa Clara, CA: Sun Microsystems.

Sun Microsystems. (2004[1]). *Business Process Integration*. Retrieved October 10, 2004 from http://wwws.sun.com/software/sunone/docs/arch/chapter6.pdf.

Sun Microsystems. (2004[2]). *Java Remote Method Invocation (Java RMI)*. Retrieved November 17, 2004 from http://java.sun.com/products/jdk/rmi/.

Surgency Inc. & IBM Corporation. (2002). *Competing for the World Cup in Business: Qualifying as Players, Winning as Teams*. New York: IBM Corporation.

Tarantella Inc. (2004). *Portal Integration*. Retrieved September 16, 2004 from http://www.tarantella.com/portals/.

TechTarget. (2003). *Extract, Transform, Load*. Retrieved October 07, 2004 from http://searchdatabase.techtarget.com/sDefinition/0,,sid13_gci519447,00.html.

Themistocleous, M. & Irani, Z. (2001). Benchmarking the Benefits and Barriers of Application Integration. *Benchmarking: An International Journal*, Volume 8 Number 4, pp. 317-331.

Themistocleous, M. & Irani, Z. (2002). Novel Taxonomy for Application Integration. *Benchmarking: An International Journal*, Volume 9 Number 2, pp. 154-165.

Thomas, N. & Buckley, W. (2004). *The Enterprise Service Bus: A Developer-Friendly Integration Engine*. Retrieved November 18, 2004 from http://www.sys-con.com/webservices/article.cfm?id=668.

Tibco Software Inc. (2002). *Five Proven Strategies for Extracting Top Value From Business Integration*. Palo Alto: Tibco Software Inc.

TopXML. (2004). *What is BizTalk?* Retrieved November 22, 2004 from http://www.topxml.com/b2b/articles/biztalk_unleashed/3_whatisbiztalk.asp.

Ulin, D. (2002). *Business Process Improvement with Composite Applications*. Scottsdale, Arizona: Optura.

Vander Hey, D. (2000). One Customer, One View. *Intelligent Enterprise Magazine*, Volume 3 Number 4. Retrieved June 17, 2003 from http://www.intelligententerprise.com/000301/feat1.shtml.

Van Huizen, G. (2003). ESB Architecture Eases App Integration. *Network World*, 06/02/03. Retrieved November 19, 2004 from http://www.nwfusion.com/news/tech/2003/0602techupdate.html.

W3C. (2004). *Extensible Markup Language (XML)*. Retrieved November 22, 2004, from http://www.w3.org/XML/.

W3 Schools. (2004). *How can XML be Used?* Retrieved November 21, 2004, from http://www.w3schools.com/xml/xml_usedfor.asp.

Wagner, P. (2002). *Enterprise Application Integration Success Factors*. Sydney, Australia: eServ Global.

Wagner, M.J. (2004). *What is Spatial ETL?* Retrieved October 07, 2004 from http://www.safe.com/spatialetl/.

Webopedia. (2001). *Extract, Transform, Load*. Retrieved October 07, 2004 from http://www.webopedia.com/TERM/E/ETL.html.

Wikipedia. (2004). *Application Programming Interface*. Retrieved September 30, 2004 from http://en.wikipedia.org/wiki/API.

Wipro Technologies. (n.d.). *Enterprise Integration (EI) Framework*. Chennai: Wipro Technologies.

APPENDIX A

TOWARDS AN EXTENDED ENTERPRISE THROUGH E-BUSINESS INTEGRATION

by

Nicolette Mostert and Dalenca Pottas

Paper presented at the Post-graduate Research Symposium of the SAICSIT 2003 Conference held in Johannesburg, South Africa in September 2003.

TOWARDS AN EXTENDED ENTERPRISE THROUGH E-BUSINESS INTEGRATION

This research paper provides an overview of the integration problem facing the extended enterprise today. E-Business integration is also discussed as a possible solution to the problem. A brief overview of Enterprise Application Integration (EAI) and Business-to-Business Integration (B2Bi), which collectively forms the e-business integration market, is also presented.

Categories and Subject Descriptors: H.1.1 [Models and Principles]: Systems and Information Theory - *Value of Information;* K.4.3 [Computers and Society]: Organizational Impacts - *Computer-supported collaborative work;* K.4.4 [Computers and Society]: Electronic Commerce - *Electronic data interchange (EDI)*

General Terms: Economics, Management, Performance

Additional Keywords and Phrases: Business-to-Business Integration, e-Business Integration, Enterprise Application Integration, Extended Enterprise, Extended Value Chain

1. INTRODUCTION

The evolution of the Internet has a significant impact on the way that enterprises operate. According to Kalakota and Robinson the evolution of the Internet can be divided into phases [2000]. During the first phase (1994 – 1997) the emphasis was on creating a web presence, while the second phase (1997 – 2000) focussed on performing transactions over digital media. Today, with the Internet era in its third phase (2000 - ?), the focus is on how the Internet can impact profitability. Enterprises of all

sizes communicate with their extended value chains, which include business partners, suppliers, customers and government agencies on a daily basis. In general, this communication takes place through the use of telephones, faxes, e-mail or Electronic Data Interchange (EDI). To stay competitive in this new Internet era there is a growing need for enterprises to accurately and speedily gather and analyze relevant facts form its extended value chain to service customers effortlessly and cost-effectively. If the value chain is not fully integrated it could take weeks or months to gather the information, by which time the market and business conditions have changed and the information is irrelevant.

The Internet has grown into an electronic platform that enables communication between the enterprise and its extended value chain inexpensively by radically reducing the cost of doing business while improving sales and service [Stokes 2001]. Enterprises that spent the greater part of the past decade purchasing expensive Enterprise Resource Planning (ERP), Customer Relationship Management (CRM) and e-Commerce applications, are now focusing on integrating these information silos and looking to their extended value chain for opportunities to eliminate costs and gain strategic competitive advantage [Oracle 2002]. This collaboration between an enterprise and its extended value chain has given rise to the concept of an *extended enterprise* where the enterprise is connected to its business partners, suppliers and customers.

In the extended enterprise the need for speed, accuracy, and detail is particularly high and the ability to meet these needs depend heavily on the ability to integrate and exploit all information assets of the extended enterprise. This paper discusses the integration problem facing an extended enterprise today, as well as e-business integration as a possible solution to the problem.

2. THE EXTENDED ENTERPRISE

The term *extended enterprise* represents the concept that an enterprise does not only consist of its employees, board members, and executives, but also its business partners, suppliers and customers. This collaboration is done to streamline business processes and enhance customer benefits [Industrial and Financial Systems 2001]. In order for the extended enterprise to be successful, it is important that relevant, up-to-date information is available to all participants at all times.

At the core of the enterprise is the ERP backbone, which is a primarily *inward* facing application that tracks the internal flow of information. To create an extended enterprise it is necessary that its information systems face *outward* to enable connectivity to business partners, suppliers and customers. Applications that extend the enterprise outwards include CRM and SCM systems. These outward facing systems must be fully integrated with the ERP backbone for the enterprise to become a truly extended enterprise [Cherry Tree 2000]. Despite the value that can be derived from integrating the outward facing systems with the core of the enterprise, this final step in the creation of the extended enterprise is where most enterprises fail. This is due to the fact that these systems were never designed to communicate with each other and the task of integrating these systems remains elusive for even the most highly respected systems integrators [Cherry Tree 2000].

This integration challenge has resulted in the creation of disconnected applications throughout the enterprise, often referred to as *stovepipes*, *islands of automation*, or even *vertical application silos*. These islands of automation are unaware of events outside their domain and to solve this problem it is necessary to use various tools and technologies to integrate the entire extended enterprise [Cherry Tree 2000].

3. E-BUSINESS INTEGRATION

Until recently, most integration projects have focused on *intra*-enterprise integration, or application integration that occurs within the firewall of the enterprise. This is known as Enterprise Application Integration (EAI). With the emergence of extended enterprises, *inter*-enterprise integration has become a priority. This involves the integration of applications with external business partners, suppliers and customers across the firewall. Integration across the firewall is also known as Business-to-Business Integration (B2Bi) [Linthicum 2000].

The EAI and B2Bi markets are merging to form the e-Business Integration market [Aberdeen 2003]. A significant driver for this situation is the fact that you must be able to integrate your internal information systems before you can integrate with the external systems of your business partners, suppliers and customers. A brief overview of these two segments of the e-Business Integration market follows.

3.1 ENTERPRISE APPLICATION INTEGRATION (EAI)

According to the EAI Industry Consortium, EAI can be defined as the process of integrating multiple applications that were independently developed, may use incompatible technology, and remain independently managed [2003]. EAI enables previously isolated systems within an enterprise to be integrated with related information and processes in other systems, thus enabling the enterprise to integrate valuable legacy systems with newer technologies that will be important for current and future operations. EAI also allows the enterprise to facilitate flexible data access and flow for ever changing business needs, while also allowing the

incorporation of any unstructured data for business use [Alpha Technologies 2003].

It becomes clear that EAI can enable the enterprise to adapt rapidly to the ever-changing business environment. To position the enterprise for even greater success it is necessary to integrate the systems within the enterprise with the systems of all business partners, suppliers and customers of the enterprise. To achieve this, business-to-business integration is necessary.

3.2 BUSINESS-TO-BUSINESS INTEGRATION (B2Bi)

B2Bi refers to the controlled sharing of data and business processes among any connected applications and data sources across enterprise boundaries using the Internet. B2Bi allows enterprises to link, for example, their SAP system to their supplier's Baan system and move information between them as needed to create an extended enterprise [Linthicum 2000]. In addition to the benefits of EAI, B2Bi can improve response times for inter-enterprise processes leading to customer satisfaction due to an improved purchasing process and shorter response times. The automation of high volumes of transactions between trading partners also leads to a reduction in administrative costs [iSource 2003].

4. CONCLUSION

For an enterprise to stay competitive in the global marketplace it is crucial for the enterprise to be connected with all its business partners, suppliers and customers to facilitate immediate reaction to current business events. This need for up-to-date information, as well as the

need for transactions to be handled immediately, calls for the integration of all information systems in the extended value chain. For enterprises to create a cost-effective extended enterprise, an integration solution is needed that easily shares one common integrated view of information as it flows throughout the entire enterprise value chain. Future research will focus on developing a model for the establishment of an extended enterprise through e-business integration. To develop such a model it will be necessary to investigate all existing and emerging integration tools and technologies, since such a solution will necessarily include multiple technology approaches to solve both the intra- and inter-enterprise integration problem. The research will also include establishing what the integration requirements of an extended enterprise are.

5. REFERENCES

ABERDEEN. January 2003. *Worldwide Enterprise Business Integration Spending: Forecast and Analysis 2002–2006*. Aberdeen Group, Boston.

ALPHA TECHNOLOGIES. 2003. *Frequently Asked Questions – What is EAI?* http://www.alphatechnologies.net/faq.htm

CHERRY TREE. 2000. *Extended Enterprise Applications*. Cherry Tree & Co, Minnesota.

EAI INDUSTRY CONSORTIUM. 2003. *Enterprise Application Integration Definition*. http://www.eaiindustry.org/

INDUSTRIAL AND FINANCIAL SYSTEMS. 2001. *A Step-by-Step Guide to the Extended Enterprise*. http://www.ifsde.com/produkte/applications/pdf/step_by_step_guide.pdf

iSOURCE. 2003. *Enterprise Application Integration 101*. http://www.isourceonline.com/article.asp?article_id=3010

KALAKOTA, R. AND ROBINSON, M. 2000. *e-Business 2.0 – Roadmap for Success.* Addison-Wesley, Boston.

LINTHICUM, D.S. March 2000. *EAI Application Integration Exposed*. http://www.softwaremag.com/archive/2000feb/EAI.html

ORACLE. November 2002. *Deriving Value from an Extended Enterprise with Oracle9iAS Integration*. http://www.oracle.com/ip/deploy/ias/docs/oracle9ias_integration_bwp.pdf

STOKES, N. 2001. The Extended Enterprise Comes of Age. *eAI Journal*, September 2001.