The Making of Information Systems

Karl E. Kurbel

The Making of Information Systems

Software Engineering and Management in a Globalized World

Springer

Karl Kurbel holds the Chair of Business Informatics at the European University Viadrina Frankfurt (Oder) in Germany

Prof. Dr. Karl E. Kurbel
European University Viadrina Frankfurt (Oder)
Chair of Business Informatics
15230 Frankfurt (Oder), Germany
kurbel.BI@uni-ffo.de

Chapter 9 was contributed by Brandon Ulrich, Managing Director of B2 International, a Budapest, Hungary, based software company

Brandon Ulrich
B2 International Ltd.
Madach ter 3, Em. III
1075 Budapest, Hungary
bulrich@b2international.com

ISBN: 978-3-540-79260-4 e-ISBN: 978-3-540-79261-1

DOI: 10.1007/978-3-540-79261-1

Library of Congress Control Number: 2008925552

Cover design: WMX Design GmbH, Heidelberg

Printed on acid-free paper

9 8 7 6 5 4 3 2 1

springer.com

Preface

Information systems (IS) are the backbone of any organization today. Practically all major business processes and business functions are supported by information systems. It is inconceivable that a business firm – or any other non-trivial organization – would be able to operate without powerful information systems.

This book deals with the question: Where do these information systems come from? In previous decades, the answer seemed fairly obvious: An organization would have their IT (information technology) group, and these people would develop information systems when the need arose. Most of the early books on business information systems started from this premise.

While inhouse IS development still has its role in large organizations, the number of options to obtain an information system has significantly grown. For example, an organization may choose to contract an external partner for the development. They may outsource their complete information systems development, or even their entire IT department, to such a partner. The partner can be located onshore or offshore. Many organizations establish captive centers, or they collaborate with offshore software companies in India, South America and Eastern Europe. Managing projects with globally distributed partners today creates additional challenges for the making of information systems.

Another significant change is that a good deal of large-scale information systems development (ISD) has moved from organizations whose core business *is not* software to those whose business *is* software. Fewer companies than previously actually develop individual information systems any more. In the business domain, large software vendors such as SAP, Oracle and Microsoft are providing standard software that already solves large portions of the problems for which individual information systems were developed before.

Since standard software never meets an individual organization's requirements one hundred percent, the customization of this software and its implementation in the organization have become major challenges. This means that much of the effort, time and money spent previously on information systems development now goes into customizing the standard software and adapting the organization to the software.

Taking into consideration an increasing number of already existing information systems, most organizations are facing the problem that any new system needs to be integrated into the existing IS landscape so that it smoothly collaborates with the other systems.

With the aforementioned factors in mind, this book examines and discusses the question of how information systems come into existence today. Chapter 1 describes typical information systems in a modern enterprise and which options in the making of information systems an organization faces. Chapter 2 discusses management issues and decisions regarding the launching of a project, inhouse or external development, outsourcing, offshoring as well as the costs and benefits of information systems.

The information systems architectures and platforms presented in chapter 3 play a pivotal role today. Since a new information system will most likely need to fit with the existing IS, an architecture may either be prescribed or to some extent need to be developed within the project. Flexible architectures have recently received much attention with the emergence of the SOA (service-oriented architecture) approach. Platforms provide the infrastructure for developing and running information systems.

In the fourth chapter, process models for information systems development are presented. Our investigation starts with the waterfall model and passes on to evolutionary development, prototyping, RUP (Rational unified process) and agile methodologies such as XP (extreme programming). Special attention is paid to the needs of off-shoring projects.

Chapter 5 focuses on two of the major stages in any development effort: analysis and design. Hardly any other area has received as much attention in research and practice as analysis and design. A vast body of methods and tools are available. We focus on the essential tasks to be solved in these stages, on modeling with the help of UML (unified modeling language) and on tools supporting the analysis and design activities.

In chapter 6, two more important stages are discussed: implementation and testing. In today's information systems development, these stages are largely based on automated tools such as IDEs (integrated development environments), program libraries and testing tools. There-

fore we discuss not only principles and methods but also typical tool support.

Chapter 7 covers the problem of selecting, customizing and implementing standard software in an organization. Since this standard software must collaborate with other software in one way or another, integration issues are discussed in a section on EAI (enterprise application integration). A particular problem in this context is integration with so-called legacy software – i.e. information systems that are not based on current software technology but must nevertheless continue to operate.

Like many business efforts, the making of an information system is usually done in the form of a project. The eighth chapter discusses project issues, in particular project management and project organization. Special consideration is given to the fact that many projects are just one out of many in a project portfolio, and that they may be performed by globally distributed teams.

Up-to-date tools for professional information systems development today are presented and evaluated in the final chapter. This includes tools that support the work of distributed project teams which have team members collaborating on several projects at the same time.

Before I started to work on this book, it seemed to be a clearly defined and overseeable project. However, as the work progressed, practically all topics revealed an abundance of facets asking for investigation. An empirical observation made in many ISD projects came true in this book project as well: "It takes longer than expected and costs more than expected."

I would not have been able to complete this book within a finite time without many people helping me. With sincere apologies to those whom I might have forgotten, my special thanks go to the following people:

Anna Jankowska developed a template for the layout and formatting of the manuscript and wrote many intricate Word macros to make it look like what the reader sees today.

Elvira Fleischer spent months of her life creating figures for the book and fighting with the template while formatting most of the chapters.

Brandon Ulrich deserves special thanks for many roles, including his roles as a contributor of essential ideas covered in the book, author of chapter 9, reviewer of the manuscript and proofreader as well.

Francesca Olivadoti was a valuable help in improving my English writing. Being actually a student of political science in the UK, she may have turned into an IS specialist after proofreading the manuscript twice.

Ilja Krybus helped me with various aspects of process models, modeling with UML and several tools to create reasonable examples (and figures) in chapters 4, 5 and 6.

Armin Boehm created among other things Visio diagrams and screenshots to illustrate standard software implementation and project management.

Ivo Stankov contributed his creativity in formatting several chapters using Anna's template and editing figures for the final layout.

Finally I would like to thank my wife *Kirti Singh-Kurbel,* not only for her patience and the time I could not spend with her during the past two years, but also because she contributed practical insights and experiences from large-scale software development, including project organization, project management and ISD offshoring, in many discussions at home.

Berlin, April 2008 Karl Kurbel

Table of Contents

1 The Digital Firm

The basic question we will answer in this book is: How can an organization today obtain the information systems it needs? What does it take to ensure that those systems are of a good quality and that they work together properly, supporting the needs of the organization?

Focus on business firms

The type of organization we have in mind is a business firm. However, the fundamental principles, methods and technologies for creating information systems discussed in this book are applicable to other organizations such as nonprofit organizations, government offices, and municipal authorities.

Initially, information systems development was mainly technical. It has since evolved into an activity with strong management involvement. Managerial-level decisions are required throughout the entire process. One reason for this is that many different ways to obtain an information system exist today. Managers have to decide which one to follow. For example, an organization may choose to:

Management decisions

1. develop the system inhouse if it has an IT department with a software development group,
2. contract an external partner, a software firm, to develop the system,
3. buy or license standard software and implement it within the organization, provided that standard software matching the firm's requirements is available on the market,
4. buy or license standard software if it satisfies at least some essential requirements, and extend that software with internally-developed components,
5. search the open-source market for complete information systems or for suitable components, and adapt that software to the needs of the organization,
6. search for web services available on Internet servers that would fulfill the desired tasks, and embed those services as part of a an overall solution tailored to the needs of the organization.

Software may be standardized, but organizations are not

Many more variations and diversifications of these approaches are possible, as are combinations of these approaches. One observation from the real world is that standard software rarely addresses the exact information needs of a particular organization. While the software is standardized, organizations are not. This is why standardized software must usually be customized to the organization – entailing minor or major changes to the software. Some important functionalities may be missing, while other features provided by the standard-software developers are superfluous to the implementing organization's needs.

Customization

Adapting standard software to the requirements of an individual organization is called *customization*. Customizing standard software has become a common approach to obtaining individual information systems in most companies today. Different approaches for customization are in use, e.g. parameterization and APIs (application programming interfaces). These approaches will be discussed in chapter 7.

Outsourcing

An even more fundamental management decision with long-term consequences is to entirely or partially *outsource* information systems development. This is a strategic decision because it influences the organization's future options on how to obtain new information systems and run them. *Outsourcing* means to contract out business functions or business processes to a different organization – in the context of this book usually to a software firm – or to a subsidiary.

Offshoring

When the outsourcing partner is located in a different country or continent, then this type of outsourcing is called *offshoring*. Transferring work to low-wage countries in general, and in the IT (information technology) field in particular, has recently received substantial attention. Many organizations hope to benefit from the global distribution of work by offshoring because it cuts costs. India, China and Eastern Europe are the preferred locations for offshoring IT work today. In chapter 2, outsourcing and offshoring with regard to information systems development are discussed in more detail.

1.1 The Role of Information Systems

Businesses rely on information systems

Information systems are the foundation of doing business today. Most business firms would not be able to operate without their information systems. In a similar way, nonprofit organizations, educational institutions, governments, public administrations and many other entities also rely on information systems.

The term *information system (IS)* derives from the fact that such a system deals with information – processing and producing information and making it available to people or other information systems that need the information to do their work. The information systems discussed in this book are used within organizations to support human task solving, automating some of this work where possible. In business informatics, information systems are often defined as socio-technical systems, or as "man – machine – task" systems. These terms indicate that an IS is a technical solution to a task in which human beings in an organization are involved, using the information produced, or providing information to be processed by the system.

Many definitions of the term "information system"exist

Definitions of the term "information system" vary. Depending on the backgrounds and viewpoints of the authors, some focus more on the technical perspective, others on the organizational and management aspects. In the field of management information systems (MIS), for example, an information system has been defined as a set of interrelated components that collect (or retrieve), process, store and distribute information to support decision making and control in an organization [Laudon 2007, p. 14].

In order to balance technical, organizational and management perspectives of IS, we give the following definition:

Definition: information system

> An *information system (IS)* is a computer-based system that processes inputted information or data, stores information, retrieves information, and produces new information to solve some task automatically or to support human beings in the operation, control and decision making of an organization.

The notion of a system implies that there are interrelated elements. In an information system, these elements may be programs or program modules, databases, data structures, classes, objects, user-interface forms or similar entities, depending on the perspective and on the abstraction level of the viewer. Taking a broader view, organizational units and hardware components may be included as well.

IS are composed of software elements and operate within organizations

In this book, we will consider information systems primarily as systems composed of software elements that are developed by and operate within organizations. The modeling and development of information systems, for example, will be discussed from the viewpoint that it is people who develop the software and use abstract models to do so. Therefore the final outcome is software that will be used by people in an organization.

Narrowing the perspective to some extent, we can say that an information system is a software system. Sometimes the two terms will be used interchangeably in this book. However, not every software system qualifies as an information system. Purely technical systems that do not have any organizational impact – for example software switching data packets in a GPRS (general packet radio service) network, a compiler or a cache manager – are not considered information systems even though computer scientists tend to call purely technical systems such as the last one "managers" or "management" systems.

Information systems are playing an increasingly important role in most organizations today. In many industries, companies depend heavily on their information systems. Information-intense industries such as insurance, banking and telecommunications could not survive without information systems. Some industries would not exist without IS, and electronic commerce would not have been invented. Firms such as Amazon, Yahoo, Travelocity, Hotels.com etc. would simply not have been created without powerful supporting information systems.

All industries depend on information systems

Also in traditional industries such as manufacturing and retail, there is a growing dependence on information systems. Firms need IS for every part of their business – for their daily operations, for controlling and reporting, for their strategic planning, and for maintaining their

supplier and customer relationships. It would be inconceiveable that General Motors, Siemens, Wal-Mart, Metro etc. are doing business today without efficient information systems.

Another reason why information systems are so important is that information technology (IT) accounts for a significant share of capital investment in modern economies. In the US, for example, investment in IT has become the largest single component of capital investment – 35 % of private business investment, and more than 50 % in information-intense industries such as the financial sector [Laudon 2007, pp. 5-6].

It is a well-known fact that efficient usage of information technology presupposes information systems able to utilize and exploit the features of the technology. Business productivity can increase substantially and firms can achieve strategic advantage over their competition by deploying information systems that support their strategic goals.

Nicholas Carr: "IT doesn't matter"

However, the importance of information technology as a differentiating factor in organizational performance has been challenged by Nicholas Carr in a famous article entitled "IT doesn't matter" [Carr 2003] and in his subsequent book "Does IT matter? Information technology and the corrosion of competitive advantage" [Carr 2004]. Carr argues that information technology may be bought by any company in the marketplace, so competitive advantage obtained through IT can be easily copied. Therefore, IT has become a commodity rather than a strategic factor.

Carr's theses have stirred-up an intensive discussion in the IS and management communities. Most IS experts disagreed with Carr's theses, yet one effect of the discussion was the significantly increased pressure on IT departments to justify the return on information-technology investments.

"Does software matter?"

With the question: "Does software matter?" Carr continued his argument and also classified software as a commodity that will mostly be developed in software factories in low-wage countries, bought off the shelf, or obtained as a service on a plug-and-pay basis [Carr 2005].

While some of Carr's observations are certainly correct, the situation in typical organizations around the world is more nuanced. Complete off-the-shelf software packages are suitable for standardizable products such as office programs but not for the heavyweight enterprise systems managing the business processes of a firm. Even if some components are purchased as ready-to-install modules or developed in India or Bangladesh, they still need to be integrated into and adjusted to the diversified information systems landscape in the organization.

Developing IS is not the core business

The situation is as outlined in the beginning of this chapter: Some IS may be developed inhouse, some bought off the shelf, and others purchased and customized. We definitely agree with one point of Carr's arguments: Typical business firms whose core business is not software have reduced the volume of internally developed corporate software dramatically, sometimes to the extent that they do not develop new software at all any more. The role of information systems development has changed – from developing entire new business solutions inhouse to implementing what others have developed, integrating that with the rest of the information systems in the organization, and perhaps developing some supplementary components.

1.2 Information Systems in the Enterprise

Stand-alone information systems

In the early times of business computing, most information systems were designed to solve specific problems or support a particular function, such as MRP (materials requirements planning), payroll or financial accounting. These were stand-alone systems, developed only to solve or support the task at hand. They were "islands" not connected with one another.

Integrated information systems

A typical enterprise today uses a large number of information systems. These systems tend to be integrated so that they can work together. All major business processes are represented in and operated with the help of information systems. Fewer and fewer companies use systems that they developed themselves. Instead they work with standard software, customized and extended to their needs.

Standard software, application package

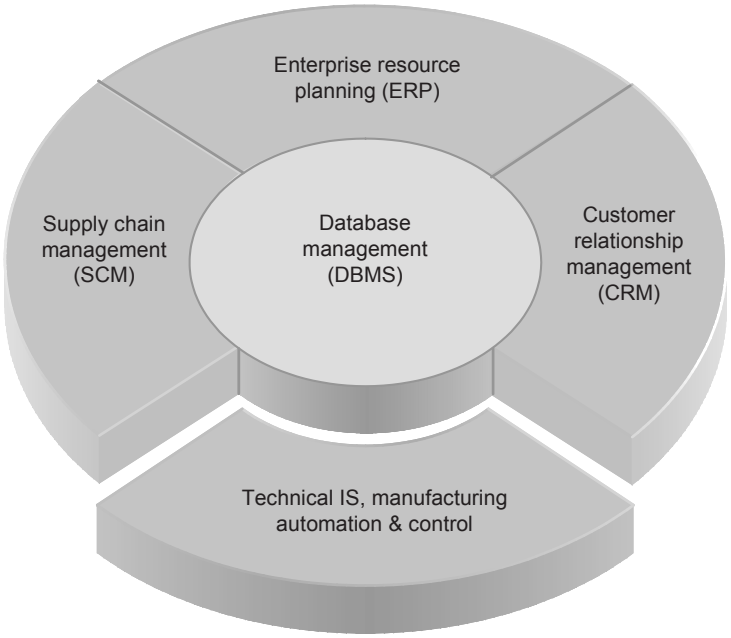
With *standard software*, also called standard packages, we denote a software system that was developed with the aim of being used by many organizations. Standard software exists for many problem areas: office programs, database management systems, enterprise resource planning etc. When business problems are underlying the software, the terms *business software* or *application package* are sometimes used.

Core information systems: ERP, SCM, CRM

A typical configuration of information systems in an enterprise comprises at least three large systems as figure 1-1 illustrates: An ERP (enterprise resource planning) system, an SCM (supply chain management) system and a CRM (customer relationship management) system.

All are built on top of one or more database management systems (DBMS) – ideally using the same logically integrated database.

Figure 1-1 **Core information systems in a typical enterprise**



The ERP, SCM and CRM systems are usually standard software that have been customized according to the requirements of the individual organization. Nowadays these three types of systems tend to be integrated: An SCM module, for example, will have access to information available in the ERP system.

CAD, CAP, CAM, CNC

If the company is a manufacturing firm, then technical information systems and software for manufacturing automation and control will form an equally significant share of the corporate software as the business systems. Ideally, technical systems such as CAD (computer aided design), CAP (computer aided planning), CAM (computer aided manufacturing) and CNC (computerized numerical control) will be well-integrated with the business systems, using the same logical database.

IS on a network

Information systems such as the above are designed to be accessed by many users at the same time. Previously these systems were run on

one computer, usually a large mainframe, and users were connected through terminals. Nowadays most processing is distributed to various computers connected by a network. Users access IS functionality as clients from personal computers, workstations, terminals and other end devices over the network. This means that the desired functionality must be available on network servers.

Application servers

A server providing access to information systems functionality is called an *application server*. Before the web age, that term referred to a server in a client-server based system. Nowadays application and web functionalities have become closely related. Therefore application servers and web servers are partly sharing the work, with some overlap, and many application servers are becoming web based. Well-known products include BEA WebLogic, Borland AppServer and IBM WebSphere Application Server. Open-source application servers are Apache Geronimo and JBoss.

IS development means development around the core systems

Organizations use more information systems than those depicted in figure 1. Dedicated systems for particular problem areas can be found in vast numbers. Yet the ones contained in the figure may be regarded as the core information systems on which today's companies operate. Information systems development today normally means development around those systems. The core systems are already there, limiting the degree of freedom for new systems or making additional systems unnecessary because the functionality is available in the standard software. What can be done and what has to be done is often determined or constrained by the requirements of the core systems. Any additional system must collaborate with the existing ones, in many cases providing data as input or processing information produced as output by the core systems.

1.2.1 Enterprise Resource Planning

An ERP system is an organization's IS backbone

The most fundamental information system in most organizations is the enterprise resource planning (ERP) system. An ERP system is a comprehensive information system that collects, processes and provides information about all parts of an enterprise, partly or completely automating business processes and business rules within and across business functions [Kurbel 2005]. ERP systems cover all major business functions and processes. They have reached a high degree of maturity

because they have been around for many years. ERP systems often originated from former MRP II (manufacturing resource planning) and MRP (material requirements planning) systems that go back as far as the late 1960s and early 1970s.

Is an ERP system a single system?

ERP systems are very large systems, so the question may arise whether an ERP system is actually one information system or many. ERP systems cover many areas and thus contain many modules. Originally these modules were more or less separate. A synonym for standard software was "modular program" because modules covering certain business functions were only loosely coupled and could be implemented separately. In this case, we might say that each module is an information system of its own.

Nowadays the degree of integration between the modules of an ERP system is so high that the systems appear as one system. For the user, an ideal ERP system will behave like *one* enterprise-wide information system with one database and one common user-interface. Therefore we consider an ERP system as one information system. Nevertheless such a system may be composed of many subsystems and many databases, as long as they are well integrated.

Market leader in ERP

The most common ERP system worldwide is SAP ERP (formerly SAP R/3). Its wide range of functionalities are illustrated in figure 1-2. Five comprehensive areas are covered by SAP ERP:

ERP application domains

- Analytics (support for strategic enterprise management and for reporting, planning, budgeting, analyzing most other areas)
- Financials (financial and managerial accounting)
- Human capital management (employee management, transactions involving employees, payroll etc.)
- Operations (logistics and production planning and control, inventory and warehouse management, procurement, sales etc.)
- Corporate services (services supporting employees in real estate management, incentive and commission management, travel management and more)

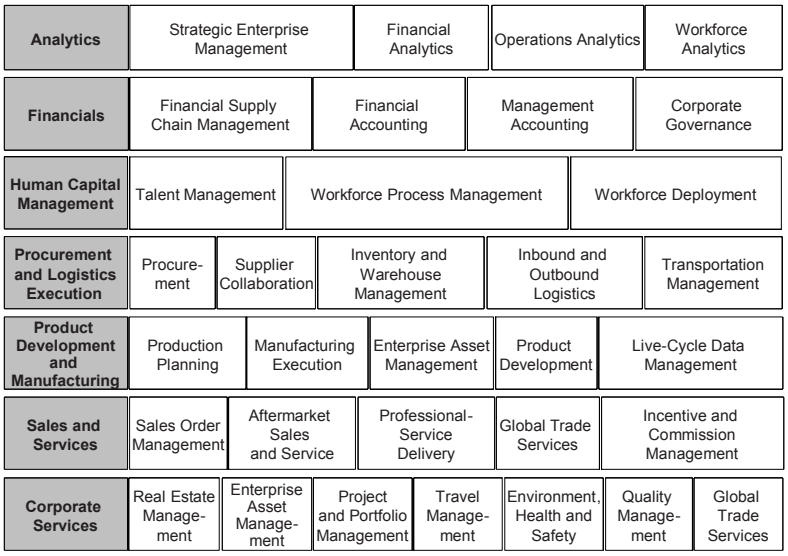
Figure 1-2 shows only the top-level domains supported by SAP ERP. Each section can be decomposed into many further sublevels. At the lowest sublevels, very detailed functions for each step of each business process are provided.

SAP, Oracle, Microsoft and open-source

On the market there are many ERP products offering similar functionalities although they may be arranged in different ways. However, their market shares are rather small. The big players are, after a round of

mergers and acquisitions at the beginning of the 21st century, SAP, Oracle (comprising former PeopleSoft and J.D. Edwards) and Microsoft. A number of ERP systems are available as open-source, including Compiere, ERP5, Openbravo ERP and OFBiz [Serrano 2006].

Figure 1-2 **Application domains and modules of SAP ERP [SAP 2007b]**



Since most systems provide the functionality required for enterprise resource planning, businesses usually do not develop information systems for ERP any more. However, if a company finds that some "standard" solution provided by the chosen ERP system does not reflect its individual requirements appropriately, then that company is likely to look for its own solution. This could be by developing or purchasing a dedicated information system for the specific problem, extending the ERP system, modifying its programs, or in other ways working its way "around" the ERP system.

Restrictions set by the ERP system

The new solution has to meet technological restrictions that are set by the ERP system. These restrictions could be the platform on which it runs, the programming language (if program code has to be modified), the database management system etc.

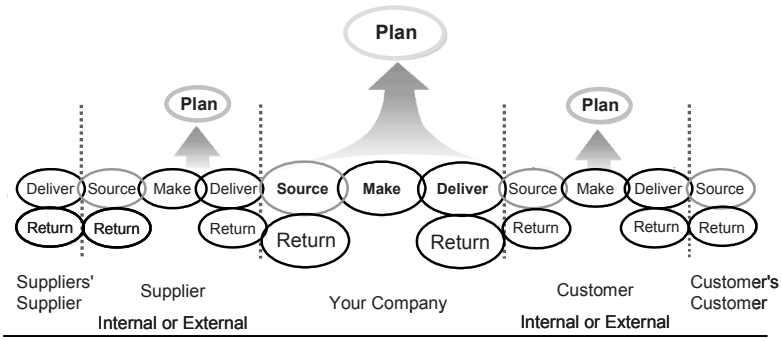
While the focus of ERP is to support internal business processes, business activities do not end at the boundaries of the company. Going beyond these limits is the task of supply chain management (SCM).

1.2.2 Supply Chain Management

Organizations collaborate in supply chains

The second of the core information systems is an SCM system. Organizations are collaborating in supply chains, creating networks of suppliers and customers over many levels, including the suppliers' suppliers and the customers' customers, as shown in figure 1-3.

Figure 1-3 **Supply-chain processes [SCC 2006]**



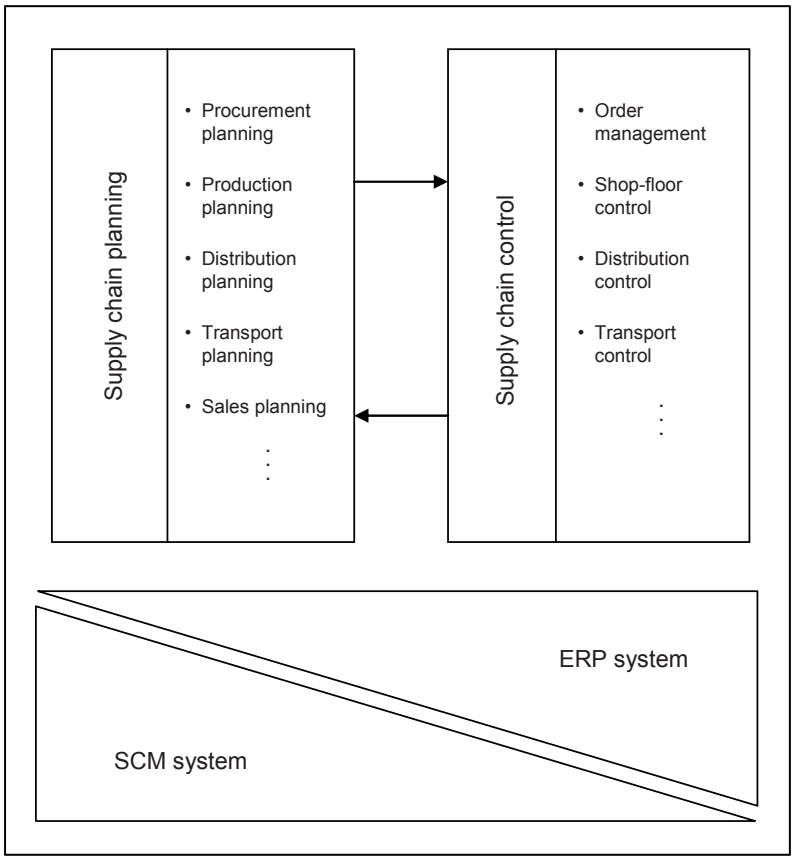
A firm's performance depends on its supply chains

Businesses have become increasingly specialized. They concentrate on their core competencies, outsource secondary activities and purchase assemblies rather than manufacture them themselves. Consequently, effective supplier-customer networks have become crucial for success. The performance of a firm depends heavily on the smooth functioning of the supply chains to which it belongs. No matter how efficient internal processes and the supporting ERP system are, if the supplier of a critical component, or that supplier's supplier, or a supplier further up in the chain fails to deliver properly, the company will not be able to perform as it thought it could. This effect is illustrated by figure 1-3.

SCM looks at the business partners

Therefore, a natural extension of ERP is *supply chain management* [Ayers 2001]. SCM considers the organization's business partners, in particular the suppliers and their suppliers. In addition, many methodological and technical shortcomings of ERP have been removed or at least improved in SCM. These improvements are known as *APS (advanced planning and scheduling)* [Meyr 2002] and are implemented in SCM solutions by SCM vendors.

Figure 1-4 **Relationship between SCM and ERP [Corsten 2001]**



SCM and ERP are connected

Supply chain management and enterprise resource planning are closely connected. This is due to two facts: In a supplier-customer network, many results provided by ERP are needed as input for the SCM system and vice versa. Secondly, the same functions are sometimes needed in both systems. There is a natural overlap between ERP and SCM functionality. In closely coupled solutions (e.g. SAP SCM [SAP 2005b] and SAP ERP), the SCM system may even invoke functions of the ERP system.

Planning and control levels in SCM

Like ERP systems, SCM systems support all levels of planning and control, from long-term strategic planning (such as setting up a supplier-customer network) to execution of daily operations. Figure 1-4 shows the relationship between ERP and SCM systems on the mid-range planning and control level. Dedicated planning functions are found in the SCM system, whereas control functions are often the same as in the ERP system. In addition, there is close interaction between the two systems because they often use the same data.

1.2.3 Customer Relationship Management

CRM: identifying, acquiring and retaining customers

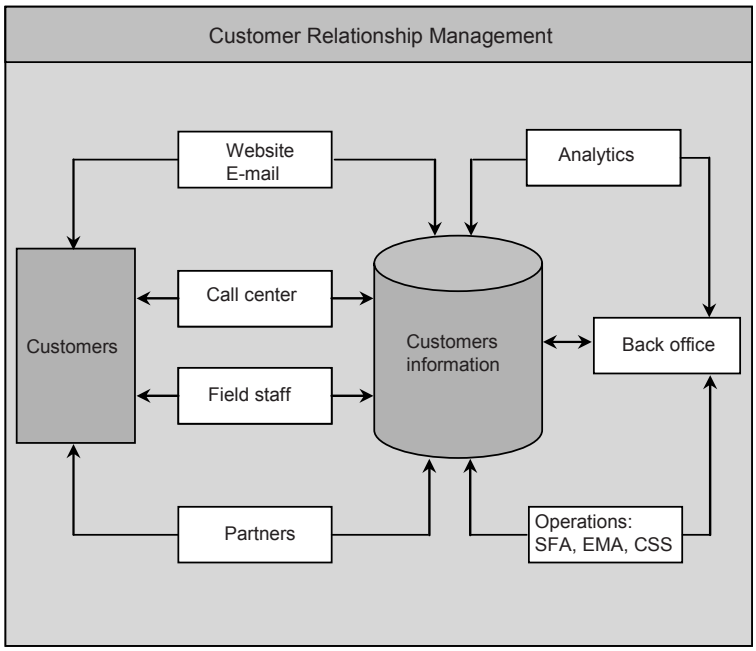
The most recent member of a typical business information systems suite is a customer relationship management (CRM) system. CRM is an integrated approach to identifying, acquiring and retaining customers [Siebel 2006]. Some authors consider good customer relations the most valuable asset of a business firm. While marketing and management have always placed high importance on customer relationships, the business's information systems have not supported this view until the late 1990s. Previously, valuable customer information was distributed and maintained in various information systems – in the ERP system, in e-commerce, call-center, customer-service systems, and more.

Managing customer interactions

The need to place the focus on customer relationships arose when marketing, sales and service departments developed new channels beyond traditional ones such as retail stores and field sales: websites (electronic shops), e-mail ordering, call centers, mobile commerce, push services etc. The number of sources of customer information grew. It became increasingly difficult to find, maintain and update customer information efficiently and consistently. Analyzing customer data for marketing in a unified way, in order to generate more value for the firm,

was not possible. By enabling organizations to manage and coordinate customer interactions across multiple channels, departments, lines of business and geographical regions, CRM helps organizations increase the value of every customer interaction and improve corporate performance.

Figure 1-5 **Sources and uses of customer information [Siebel 2006]**



Definition of a CRM system

A CRM system is an information system that is used to plan, schedule and control the presales and postsales activities in an organization [Finnegan 2007, p. 4]. The goal of CRM is to improve long-term growth and profitability through a better understanding of customer behavior. CRM includes all aspects of dealing with current and prospective customers: call center, sales force, marketing, technical support, field service etc. All customer information from these sources is collected and maintained in a central database as illustrated in figure 1-5. Marketing, sales and service departments access the same information.

"Back office" and "front office" systems

A typical "back office" system the CRM system is connected to is the company's ERP system. CRM systems are sometimes called "front office" systems because they are the interface with the customer.

CRM systems are composed of operational and analytical parts. *Operational CRM* includes in the first place support for:

Operational CRM: SFA, EMA, CSS

- SFA (sales force automation – e.g. contact/prospect information, product configuration, sales quotes, sales forecasting etc.)
- EMA (enterprise marketing automation – e.g. capturing prospect and customer data, qualifying leads for targeted marketing, scheduling and tracking direct-marketing etc.)
- CSS (customer service and support – e.g. call centers, help desks, customer support staff; web-based self-service capabilities etc.).

Analytical CRM

Analytical CRM consolidates the data from operational CRM and uses analytical techniques to examine customer behavior, identify buying patterns, create segments for targeted marketing, identify opportunities for cross-selling, up-selling and bundling, and separate profitable and unprofitable customers. This is done with business intelligence techniques such as OLAP (online analytical processing) and data mining, based on a data warehouse.

ERM, PDM

In addition to operational and analytical customer relationship management, many CRM systems include components for ERM (employee relationship management) and PDM (partner relationship management). This is due to the fact that employee performance and partner (e.g. dealer) performance are closely related with customer relationships.

Connections between CRM and various parts of enterprise resource planning are quite tight. That is why ERP vendors also provide CRM systems which interoperate with their respective ERP systems. It is not surprising that the long-time market leader in CRM, Siebel Systems [Siebel 2006], was bought by Oracle in 2006.

1.2.4 Database Management

All of the above information systems handle large amounts of data. Only in the early days of business information processing were these data stored in program-related data files. Early MRP (material require-

ments planning) systems, for example, had quite sophisticated file organization systems. However, today all non-trivial business information systems store their data in databases.

The roots of database management systems go back to the 1960s and 1970s, so it is not surprising that today's systems have reached a high level of maturity. The functionality of a modern DBMS comprises a lot more than just storing and retrieving data. For example, database schemata can be generated automatically from models. Visual tools for semantic data modeling, for creating graphical user-interfaces and for querying the database, as well as a workflow management system and much more are provided. In fact, Oracle's entire ERP functionality is largely based on tools around Oracle's database management system. This is not surprising as Oracle Corp. is one of the world's largest DBMS vendors.

Definition: database management system

A DBMS is an information system that handles the organization, storage, retrieval, security and integrity of data in a database. It accepts requests from programs or from end-users, processes these requests and returns a response, e.g. transferring the requested data.

Most DBMSs are relational

Most of today's database management systems are relational systems (RDBMS). With the emergence of object-oriented analysis, design and programming, RDBMS were extended to accommodate not only data records but also objects, thus realizing object persistence. Notwithstanding the existence of dedicated object-oriented DBMS, the majority of business information systems use RDBMS.

Database management systems on the market

There are many relational database management systems on the market. Oracle (Oracle Database), IBM (DB2), Microsoft (SQL Server) and Sybase (Adaptive Server Enterprise) have the largest market shares. MySQL and PostgreSQL are popular open-source products. A widely used DBMS for end-users, but not for large professional business systems, is Microsoft Access.

A major achievement of more than four decades of business information processing was the decoupling of application systems and database management systems. In earlier times the programs of an MRP II or ERP system, for example, referenced the DBMS directly. Since each vendor's DBMS implementation had its own extensions and modifications of the SQL (structured query language) standard, the application system and the database management system were tightly coupled. Portability of a database – and thus of an entire ERP system, for example – was a difficult, sometimes impossible task.

Interfacing with an RDBMS

Nowadays an RDBMS supports common interfaces with standard access methods. Programs now invoke operations provided by the interfacing technology instead of directly accessing the database

management system. Portability has significantly improved in this way. Standard technologies and access methods are:

- ODBC (open database connectivity), providing access to databases on a network for Windows programs,
- JDBC (Java database connectivity), allowing Java programs to access a relational database via the SQL language,
- Java EE/EJB (Java enterprise edition/Enterprise JavaBeans), giving higher-level access to a database than JDBC, using EJB entity beans,
- XML (eXtensible markup language) enabling, providing standard access methods for navigation and queries in XML. Data are extracted from a database and put into XML documents and vice versa.

The functionality of a professional DBMS is provided on a server. Like an application server for the business functionality, a database server is connected to a network. ERP, SCM and CRM functions access the server over the network. Human users such as database administrators and end-users also reach the server over the network.

1.2.5 Electronic Commerce and Electronic Business

With the explosive growth of the Internet, organizations began to employ the web to do business. Many organizations developed web-based systems to present themselves and to advertise and sell their products.

Web technology is different from conventional IS technology

This posed a major problem since web technology is quite different from the conventional information systems technology the back-office systems are based on. Web-based systems are written in HTML (hypertext markup language) and in software technologies extending HTML, whereas a typical ERP system is written in a language such as Java, C, C++, Cobol etc. and strongly relies on a database management system.

Two lines of development emerged: 1) dedicated web-based information systems and 2) web-based front-ends for the core back-office systems. In the beginning, web-based systems were stand-alone systems, not integrated with the business processes and the ERP/CRM systems of the company. This was not only a technological problem but also an organizational one.

Electronic commerce

Electronic commerce (e-commerce) refers to the process of buying and selling products or services over a digital network. Usually it is assumed that this network is the Internet and that the products or services are offered via the World Wide Web. An *electronic shop* (or a web shop) is an information system that presents products and services in a product catalog. It lets customers add products to a shopping cart and complete the purchase with a financial transaction. Product configuration, personalization and many more features may be included.

Data redundancy between ERP and e-commerce systems

A fundamental problem in the development of an electronic shop is that most of the data involved are available in the company's database or have to be stored in that database. Therefore, the shop system needs to access the database. Technologies to access a database from an HTML based user interface are available, for example invoking stored procedures of the database from ASP (Active Server Pages) or JSP (JavaServer Pages) scripts. Yet the script code is likely to contain redundant data-related functions that are implemented in the ERP system anyway. If the ERP system and the electronic shop are not integrated, this redundancy cannot be avoided. Many more problems may arise from the lack of integration.

Until today, e-commerce systems were often developed as individual solutions, without employing standard software. Ready-made shop solutions with tools for adaptation to company specific features are available, yet many organizations prefer tailor-made systems developed inhouse or by a web design agency.

Electronic business

Electronic business (e-business) takes the concepts and technologies of e-commerce into the inside of the business firm and into the business relations with partners. E-business is business performed with the help of digital networks, based on Internet, intranet, and web technology. E-business comprises all the business processes in the company, including processes for the internal management of the firm and for coordination with suppliers, customers and other business partners. E-commerce is a part of e-business.

One of the implications of e-business for information systems and their relationships is that system communication and interaction with users are now increasingly based on Internet protocols and languages instead of proprietary communication mechanisms. For example, a typical graphical user interface (GUI) of an ERP system in the past was based on forms that were generated with a tool provided by the ERP vendor. Using web technology in e-business now means that the user interface will not be created in a proprietary GUI technology but written in HTML or created with a tool that generates HTML forms. Likewise, data communication between systems or system modules is moving to

Internet technologies. Data are increasingly transferred in XML format, not only between web-based systems but also for accessing databases.

Coupling ERP, SCM, CRM with an enterprise portal

Another implication is that organizations provide portals for their employees, for business partners such as customers and suppliers, and for the general public. An *enterprise portal* is a website that serves as a single entry point or gateway to a company's information and knowledge base for employees and possibly for customers, suppliers, partners, or the general public. In modern architectures, access to the functionality and data resources of the core business information systems is also provided through portals. This means that systems for ERP, SCM, CRM etc. have to be coupled with a portal – another challenge where web technology and conventional software technology meet.

Web server and application server

Since electronic commerce and electronic business usually employ web technology, the basic pattern of client requests and server responses applies. This means that e-commerce/e-business information systems need a *web server*. If they are integrated with the core information systems that run on an application server, then both a web server and an application server will be present. The two servers communicate with respect to application functions and data. Since the functionalities of web and application servers are overlapping, a division of labor between the two has to be established.

E-commerce and e-business started as approaches employing cable-based networks and desktop computers. With the emergence of wireless networks and end devices capable of receiving, displaying, and transmitting data at reasonable speeds – mobile phones, PDAs (personal digital assistants), pocket PCs – a performance similar to that available on stationary computers was desired for mobile workers and their mobile devices.

Mobile commerce, mobile business

Mobile commerce (m-commerce) and *mobile business (m-business)* are the counterparts of e-commerce and e-business when the respective activities are based on the use of mobile appliances and wireless network technologies. Such technologies are, for example, UMTS (universal mobile telecommunication system), i-mode (an NTT DoCoMo technology [NTT 2006]), GPRS (general packet radio service), HSCSD (high speed circuit switched data) and GSM (global system for mobile communication).

Implementations of mobile-commerce and mobile-business systems vary significantly, depending on the type of network, the protocols available on the mobile devices, and the computing power of the devices. While early mobile phones were more or less "dumb" terminals, just capable of displaying simple data on WML (wireless markup language) cards, many modern phones have XHTML MP