1

Background, Terminology, Opportunity, and Challenges

As indicated by its title, this first chapter will provide the foundations upon which the approaches, technical discussion, solutions, and other material found throughout this book will be built. Readers who have some amount of experience and accomplishments in one or more of the book's major subject areas—data warehousing, customer relationship management (CRM), or e-commerce—may wish to skim this chapter, while those readers new to all of these various disciplines will likely want to spend a bit more time with the material contained in this introductory chapter.

We *strongly* recommend, however, that all readers spend at least a little time with this chapter before moving on to the more in-depth, solutions-focused content found in subsequent chapters, for several reasons:

- We are in the early stages of a "great convergence" of the most significant technology disciplines of the 1990s, and, as we'll discuss shortly, there are a number of lessons to be learned from what we did right in the 1990s *as well as what we did wrong* that are important to understand with regard to the foundation for e-commerce-based data warehousing and data management.
- Terminology in all of the underlying disciplines we'll discuss—particularly data warehousing, but also in e-commerce and CRM—has evolved to a hodgepodge of multiple, often conflicting meanings for commonly used terms and phrases. For the sake of consistency and common understanding, we will be very specific in this chapter about the terms we will use throughout this book and the context in which we will use them.

- As we'll discuss near the end of this chapter, there are some significant challenges with regard to organizations' efforts to apply data warehousing principles and technology to their respective e-commerce initiatives. Understanding these challenges—and how to overcome them—is every bit as important as understanding the concepts, architecture, and technology that we'll discuss in this chapter and throughout this book.

Background: A Look Back at the 1990s

The period from 1990 through 1999 was a remarkable decade in which many core computing and communications technologies and developments from the prior decade all came together and transformed the way in which business in the United States and throughout much of the world was done. Information systems broke out of the back office, where, for the most part, earlier generations of computer applications had been used primarily for permitting repetitive business processes to be accomplished faster and to encompass significantly larger volumes of data than if manual processes had still been used. The major product and architectural advances of the 1980s—desktop computers (PCs, as well as larger, more powerful workstations), networking technology (particularly local area networks, or LANs), and relational database management systems (RDBMSs)—converged to permit whole new classes of information systems and applications to be created and successfully deployed. At the same time, core telephony and other communications technology saw similar dramatic advances, and likewise converged with computer technology to provide a much deeper set of tools from which advanced information systems could be constructed.

But what was most notable about information systems throughout the 1990s was not just the continuing advancement in and synergies among different types of computer and communications technologies, but rather how that technology led to the birth and widespread acceptance of certain classes of applications. Specifically, we saw the following occur:

1. Enterprise Resource Planning (ERP) applications take hold.
2. CRM catches on.
3. Organizations pursue data warehousing to provide business intelligence.
4. The Internet evolves to a phenomenally successful e-commerce engine.

Let's discuss each of these points in turn.

ERP Applications Take Hold

From the earliest days of corporate computing systems until the end of the 1980s, nearly all large and mid-sized organizations pursued a mix-and-match, highly heterogeneous (read: disjointed) approach to their core back-office systems (finance, human resources, purchasing, inventory management, etc.), consisting of often poorly integrated custom and multivendor commercial applications. The 1990s saw, for the first time, many companies embrace product offerings from companies such as SAP AG, Peoplesoft, and Oracle that provided (or at least intended to provide) a single integrated package framework upon which most or all of a company's core business processes could be implemented, deployed, and used throughout the enterprise. By the mid-1990s, when belated awareness of the impending Y2K problem finally caught the attention of corporate information technology strategists and planners, the ERP "movement" got a boost from companies who chose to pursue a strategy of replacing non-Y2K compliant legacy systems rather than fixing them. For many organizations, ERP implementation and deployment *was* their Y2K fix.

Although many organizations had less-than-satisfactory experiences with their ERP package efforts—there were more than a few high-profile, high-cost failed ERP efforts—one significant fringe benefit did come out of a decade's experience with complex ERP integrated packages. In the late 1980s and into the first few years of the 1990s, if an organization attempted to develop and deploy a complex, enterprise-scale client/server system, failure was just about guaranteed. Many studies in the 1992–1993 time frame proclaimed the failure rate for client/server projects to be anywhere from 70 to 85%, and for a brief period of time there was such a backlash directed at client/server computing that the entire distributed systems approach to computing was branded by many as a failure, and reversion to centralized, highly inflexible computer architectures built around nonintegrated legacy applications was seen by those same doomsayers as inevitable.

Fortunately, despite the high-profile ERP failures, enough successes were widely documented that organizations began expanding the breadth and reach of their core applications beyond internal business processes (e.g., finance, human resources, etc.) toward new classes of applications that crossed enterprise boundaries and involved multiple corporations. For example, supply chain automation applications began appearing, and as we'll discuss later in this book, business-to-business (B2B) e-commerce applications such as electronic procurement (e-procurement) and buyer-to-seller electronic marketplaces are directly descended from these first-generation cross-enterprise supply chain applications, which in turn owe a large portion of their growth to the tenacity of ERP proponents who persevered throughout the decade and made *successful* large-scale, complex distributed computing systems a reality.

CRM Catches On

As we noted at the beginning of this section, the 1990s saw computing technology break out of its home in corporations' back offices and become "customer-facing." Two business areas in which customer-facing applications became common, almost ubiquitous, were

- Sales force automation (SFA) applications - A company's sales force functions of contact management, lead generation and tracking, and related business processes were linked together and managed by a single integrated environment. Whereas in the pre-SFA world computer technology was typically applied to after-the-fact rudimentary business processes such as managing accounts payable following a sale, SFA applications supported customer interaction and relationships *during* the sales cycle.
- Call center applications - Requests from a company's customers and prospects for product information, postsales product support, general customer service—or pretty much anything that might be requested over the phone—were managed in an increasingly sophisticated manner. Features were supported such as automatically routing an incoming call to a particular call center agent by identifying the incoming telephone number based upon looking up a customer's "value" (e.g., identifying high-net-worth customers of a bank as compared with "the average guy"). As was noted earlier, the convergence of communications technology played an important role in development and deployment of call center packages and systems. (A commonly heard term with regard to call center projects is CTI, or "computer-telephony integration.")

These two classes of applications—SFA and call center—we will call CRM *core systems*, terminology we'll discuss further in the "Terminology" section of this chapter. What is important to realize, though, is that as with ERP applications, CRM core systems evolved during the 1990s from a basic concept at the start of that decade to, by decade's end, a nearly indispensable part of most corporations' application portfolios. Not only were SFA systems and call centers here to stay, but they were also positioned for their next step along their evolutionary path, as we'll discuss later in this chapter: convergence with Internet technologies and e-commerce business models.

Organizations Pursue Data Warehousing to Provide Business Intelligence

In the 1990s, the database management systems (DBMS) world was facing a crisis. DBMS vendors such as IBM, Digital Equipment, Oracle, Ingres, and others had spent much of the latter part of the 1980s trying to develop distributed versions of their respective core database products. With the explosion of personal computers and minicomputers during the 1980s, corporate data assets were increasingly dispersed among hundreds or even thousands of different platforms throughout the enterprise. The idea behind a distributed DBMS (DDBMS) product was that a single enterprise-wide data management layer would provide various types of transparency services (e.g., location transparency, platform transparency, and data format transparency) and treat these physically dispersed stores of data as if they were really a single, logically centralized, and homogeneous database. For example, a single query could be executed against the DDBMS layer that would, using its own directory and metadata (a database term for "data about data") information, determine that three different databases would need to be accessed at execution time to merge and organize the requested information and present the combined results back to the user or requesting application.

Without going into a lot of detail, DDBMS technology failed, and organizations entered the 1990s facing an ever-worsening "islands of data problem."[1] Data management strategists began looking at alternatives to the failed DDBMS approach to dealing with this situation, and the idea of "data warehousing" was born.[2] Basically, data warehousing took a "something old, something new" approach to the islands of data problem: if it was too difficult to reach out at execution time to many different distributed, heterogeneous stores of data throughout the enterprise, why not preload (e.g., copy) selected groups of data from different databases and file systems into a single new database, where that content would be consolidated, "cleansed," and staged, ready for use? The "something old" portion of this approach is that most organizations were doing something like this already in the form of extract files, in which they would extract data from their legacy systems and move that data into a flat file for simple querying or generation of standard reports.

Data warehousing took off, though, for a couple of reasons:

- Whereas DDBMS technology had been thought of as a solution for both transactional and informational/analytical applications, organizations who built and deployed data warehouses typically focused their usage on the informational/analytical side to generate reports, analyze trends, and so on. Eventually, the term "business intelligence" came to represent the spectrum of different analytically focused usage and interaction models for an underlying data warehouse.

- Instead of flat files, data warehouses were typically built on top of either a relational database (taking advantage of the maturation and increasing acceptance of RDBMSs as successors to earlier pointer-based, relatively inflexible database models) or a new generation of proprietary "dimensional" database products (e.g., IRI's Express or Arbor's Essbase) that were specially architected for data analysis instead of transaction processing. While many data warehousing professionals became caught up in the relational versus proprietary database wars of the mid-1990s, the reality was that both were vast improvements over flat extract files, helping to facilitate the growth and acceptance of data warehousing.

We'll discuss more about the growth of data warehousing later, particularly as the discipline began converging with and expanding the capabilities of ERP and CRM systems. But it's important to note that by the end of the 1990s, data warehousing—like ERP and CRM—was here to stay.

The Internet Evolves to a Phenomenally Successful e-Commerce Engine

As momentous as ERP, CRM, and data warehousing are, all three of those areas can arguably be categorized as belonging to the realm of "professional computing." That is, someone from the general populace is likely to be unaware of (or maybe only peripherally aware of) sales force automation, data warehousing and business intelligence, supply chain management, or the other aspects of one or the other of these various disciplines.[3]

Most people, though, are at least familiar with the Internet and have at least a rudimentary understanding of how certain aspects of the Internet are changing aspects of their lives. Many analysts and observers have documented the dramatic evolutionary path of the Internet from a text-dominated research network at the beginning of the 1990s to a home for "billboardware" (e.g., simple advertising-oriented graphical Web sites) in the middle of the decade and then, by the end of 1999, a full-fledged force for revolutionary economic change and opportunity.

It's this most recent and current role of the Internet—economic enabler—that is most important to us with regard to the subject matter of this book. Pick any term or phrase you prefer: e-commerce, e-business, new economy, digital economy (in the next section we'll talk about terminology and sometimes conflicting meanings, as well as those definitions that are hype-laden). The indisputable fact is that an entire new and very powerful toolbox is now available to every current and potential business in the world. And, later in this chapter, we'll discuss how organizations are just beginning to use that toolbox's contents to extend the dramatic accomplishments of the 1990s that we discussed in this section—ERP, CRM, data warehousing and business intelligence, and the Internet—with dramatic "1 + 1 + 1 + 1 = 1,000" synergy.

Terminology and Discussion

Before we discuss the dramatic "1 + 1 + 1 + 1 = 1,000" synergy, though, we need to briefly cover the topic of terminology. As we noted several times in the preceding section, terminology in three of the areas in particular—data warehousing, customer-facing applications, and the Internet—is decidedly nonstandard, resulting in not uncommon misunderstandings and misconceptions about how these various disciplines can and will interact with one another.

Where we believe it's important to do so, we'll provide brief explanations behind the reason for multiple overlapping terminology. But rather than getting bogged down in this subject, for the most part we'll introduce (or reintroduce) key terms and phrases and indicate *precisely* what we mean when we use that terminology throughout this book.

Data Warehousing Terminology

Data warehouse: A logically consolidated store of data drawn from one or more sources within the enterprise and/or outside the enterprise.

Discussion: Note that the commonly used, classical (circa early 1990s) definition[4] of a data warehouse—"subject-oriented, nonvolatile, time-variant" (basically, a read-only database that contains historical information and is organized for easy access)—presupposes a particular architecture and set of capabilities (e.g., "nonvolatile" presupposes periodic batch updates and one-directional flows of content into the data warehouse). As data warehousing grew and matured throughout the 1990s, organizations began implementing data warehouses that supported real-time flows of data into the data warehouse; "feedback loops" of content from the data warehouse back to source transactional systems; and other complex cross-system interactions. In the discipline of electronic commerce, this next-generation approach to data warehousing is critical, particularly in managing data content across both the Internet and other more traditional, non-Internet channels.

Note also that a data warehouse is a *logically* consolidated store of data. While most data warehouses are currently implemented in a single database instance, it will become increasingly common to find larger, more complex stores of data implemented across multiple databases that are logically related to one another as part of a data warehousing *environment*.

Data mart: A subset of content drawn from a data warehouse—and possibly enhanced with additional content drawn from non–data warehouse sources—that is typically used to support a specific set of required business functionality.

Discussion: As with the term "data warehouse," there is no official definition for a data mart. Complicating the matter is that data marts are usually defined in the context of a data warehouse (e.g., "smaller than a data warehouse"). A further complication is that many people use the terms "data warehouse" and "data mart" interchangeably; for example, a data mart may acquire all of its content directly from source applications rather than from an already existing data warehouse.

For the most part, we will avoid using the expression "data mart" in this book unless we're discussing either (1) the architecture of a particular software vendor that includes a data store labeled as a "data mart," or (2) a secondary, physically separate subset of content extracted from an existing data warehouse. To help avoid confusion, we won't use the two terms interchangeably; rather, we will use "data warehouse."[5]

Business intelligence: A portfolio of informationally focused or analytical applications established on top of a data warehouse.

Discussion: Many people with cursory familiarity with data warehousing think of a data warehouse as being *both* (1) a store of data and (2) a set of simple reporting and analysis capabilities provided as part of the data warehouse. While this approach may well have described early- and mid-1990s data warehousing, during the last half of the 1990s an entire spectrum of different types of uses for a data warehouse evolved. Therefore, it's useful to make a distinction between the data warehouse (the store of data itself) and how that warehouse's content is used (e.g., for one or more business intelligence purposes).

We find the following classification of business intelligence applications to be useful:

- Simple reporting and querying - "Tell me what happened" functionality that provides point-in-time results, comparisons between current results with those of past periods, or other rudimentary, relatively static reports or ad hoc queries.
- Online analytical processing (OLAP) - "Tell me what happened, *and why*." Processing that enables users to further analyze results of reports and queries by drilling

into the underlying details, looking at results in different ways, or otherwise performing manipulation of report and query results.

- Executive information systems (EISs) - Systems that "tell me lots of things, but don't make me work too hard to get that information." Whereas OLAP functionality can provide a wealth of information at summarized levels as well as the underlying details, tools in that category have traditionally been complex to use. EISs typically take the form of electronic online briefing books, "executive dashboards," online scorecards, or some other form in which a limited set of key business indicators (KBIs) are made available to executive users in an extremely easy-to-use fashion.
- Data mining - Predictive "tell me what might happen" capabilities that sometimes also take the form of "tell me something interesting, even though I don't know which specific questions to ask." Data mining is often thought of as the modern successor to artificial intelligence technology—neural network technology is a commonly used data mining technique—but more often than not, data mining is instituted through the use of "heavy statistics." Statistical methods such as regression analysis, association, classification, and other techniques are used to build models through which either (depending on the methodology of the techniques) large volumes of data or data samples are applied, looking for patterns, hidden relationships, and so on.

In the context of this book's subject—data warehousing applied to e-commerce—it's important to note that *all* of the above types of business intelligence are applicable. However, one particular business-to-consumer (B2C) situation might require only data mining techniques for purposes of customer segmentation (as discussed in Chapter 2); another B2C situation might be better served by implementing and deploying the entire spectrum of business intelligence applications for a number of purposes throughout the organization. Where applicable, we'll refer to particular forms of business intelligence in our discussion and examples throughout the book, and we'll be consistent and specific with our recommendations.

Extraction, transformation, and loading (ETL) and data warehousing middleware: Classical data warehousing is, as we've mentioned, batch processing based. During the 1990s an entire class of tools evolved that handle part or all of the technical processes required to

- extract data from a source application's database or file system
- perform necessary data transformations such as correcting rudimentary data errors, summarizing data, converting and unifying data codes (e.g., source

application #1 uses "M" for male and "F" for female, while source application #2 uses an integer of 1 for male and 2 for female; these would typically be converted to some common code in the data warehouse as data is consolidated)

- load the transformed and "cleansed" data content into the data warehouse's database

Hence, the phrase "extraction, transformation, and loading"—or ETL for short—became widely used to describe those processes. However, data warehousing ETL can also be viewed as a subset of cross-system *middleware*, or integration-focused services used to link applications, facilitate data sharing, and so on. With regard to data warehousing for e-commerce, a mix of traditional, batch-focused ETL services together with more highly functional middleware capabilities are needed. And, with regard to middleware, there will be a mix of general-purpose middleware along with other middleware products that are exclusively focused on Internet-based integration and interaction. Therefore, we will be very specific throughout this book in explaining exactly what types of middleware we are referring to in various scenarios: traditional ETL tools, general-purpose middleware, or Internet-specific middleware.

Customer Relationship Management Terminology

Customer Relationship Management (CRM): A class of applications that includes *three different subclasses*, each of which will be discussed below:

- Traditional CRM core systems
- CRM analytics
- e-CRM systems

Discussion: As was noted earlier in this chapter, the term "CRM" has been widely used to refer to those applications that are primarily "customer-facing." However, many people think of what we'll term "CRM core systems" when they hear the term "CRM." It is important to make a distinction among the three subclasses listed above and described below, so we will avoid using the all-encompassing term "CRM" in this book; rather, we will specifically mention the class of CRM to which we're referring.

Traditional CRM core systems: "Offline" (i.e., non-Internet based) applications for either SFA or call centers.

Discussion: The primary purpose of SFA and call centers is to effectively and efficiently manage *customer contacts*. SFA systems have traditionally been used in a business-to-business setting (e.g., managing interactions between an electronics company's sales force and other companies that represent their customer base and prospects). Conversely, call centers have been used primarily in business-to-customer settings: product support, telemarketing, and so on. (Note, however, that SFA can be applied as necessary to business-to-consumer situations just as call centers can be used for business-to-business support.)

Some companies have integrated their SFA and/or call center systems with other internal applications; for example, an order recorded in an SFA application is linked in a real-time (or near-real-time) manner with back-office applications such as accounts payable and inventory management. More often, though, SFA and call center environments have been loosely coupled with other applications, relying on periodic batch transfers or even manual-based work processes to exchange information.

The common thread between traditional SFA and call centers is that in both settings, customers are *not* connected with the systems themselves. That is, if you were to create an end-to-end process flow diagram covering both offline and online processes for an SFA or call center environment, those processes in which the internal company environment interfaces with the customers are typically manual: a customer placing an order with a salesperson; that same customer requesting a copy of orders from that salesperson he or she placed over the past six months; or an individual placing a phone call to the call center to discuss a billing problem.

The entire CRM discipline has evolved with the Internet, but we find it's important to make a distinction between traditional CRM core systems and, as we'll discuss shortly, Internet-driven CRM, or e-CRM.

CRM analytics: Supporting applications that draw data from CRM core systems for the purpose of "better understanding customer behavior" by producing reports, segmenting and classifying customers based on purchase patterns, and similar analytically focused functionality.

Discussion: Recall our discussion about business intelligence as a portfolio of applications sitting on top of a data warehouse, which in turn draws its contents from source applications. CRM analytics are, in effect, special-purpose data warehouses and business intelligence applications that, instead of supporting inventory analysis or financial reporting, are designed to draw content from the CRM core systems (the data sources), reorganize and restructure that content as necessary, and support special-purpose reporting, analysis, and data mining.

As will be discussed later in this chapter, CRM analytics range from simple one-to-one "data marts" (e.g., data is extracted from a single SFA application into a separate database, where segmentation, profiling, and other analysis occurs) to more complex environments in which multiple SFA applications and multiple call centers need to be considered as data sources to the database(s) upon which the CRM analytics will be applied. The latter setting is commonly found in larger corporations, especially those in which acquisitions of other companies, each with their own CRM core systems, has occurred. And, as we'll discuss, the technical processes necessary to support CRM analytics become much more complex when Internet-based CRM (e.g., e-CRM, discussed next) is included in the mix.

Finally, it's important to note that some commercial software packages that support CRM analytics support more than just analysis and reporting. For example, a package might perform customer segmentation and then, based on targeted marketing directives and company strategy, provide capabilities to support *campaign management* functions such as (1) mailing, faxing, or e-mailing offers (e.g., special promotions) to targeted customers or prospects, or (2) initiating and managing a program to reacquire former customers. Therefore, unlike traditional data warehousing and business intelligence in which the reporting and analytic activities tend to be ending points with regard to online activity, CRM analytics increasingly feed the results of those analytical processes into other transactional business processes.

e-CRM: Internet-based systems that manage customer contacts online.

Discussion: Unlike traditional CRM core systems that, as discussed above, facilitate "offline" interaction with customers, e-CRM moves *both* families of CRM core systems—sales contact management and customer support—online and, using Internet technologies, directly links the customers and prospective customers into the systems themselves.

"Pure-play" Internet companies (e.g., a B2C or B2B dot-com company that has no preexisting non-Internet business operations) would likely implement newer-generation e-CRM systems that include not only support for the core, transactionally focused functions (again, sales contact management and customer support) but also tightly integrated analytical and informational functions such as customer segmentation and profiling—a relatively straightforward proposition that often can be based on a single application package (or set of modules from a single vendor). However, companies with *both* Internet-based and non-Internet operations will inevitably find themselves needing all three classes of CRM systems discussed above, as we'll discuss throughout this book. Therefore, while next-generation e-CRM packages provide higher levels of integration between transactional and analytic business processes, "click-and-mortar" companies (a commonly used term to describe a company with a mix of non-Internet and Internet-based business)[6] will find themselves facing significantly more complexity than with earlier, mid-1990s-class CRM environments. And, as we'll discuss in the "Challenges" section, understanding this complexity (or, more precisely, *not* appreciating and understanding that complexity) can be a major impediment to successfully building data warehousing environments to support e-commerce.

Internet Terminology

Electronic commerce (e-commerce): Using the Internet as the foundation for many different types of business processes related to commerce: purchasing, online payment, order fulfillment, supply chain management, and customer support, among others.

Electronic business (e-business): Synonymous with e-commerce.

Discussion: A common point of confusion for many is the distinction—or lack thereof—between the terms "e-commerce" and "e-business." The former term was the first to appear during the dramatic growth of the Internet during the 1990s, but by the latter part of that decade the term "e-business" had come into vogue. Some consultancies and product companies define e-business as the overall framework and infrastructure of Internet-based business, while in their view e-commerce refers only to the actual online transactions (e.g., placing an online order and using your credit card for automatic payment).

We believe that the term "e-transaction" should be used for online transactions, and for all intents and purposes "e-commerce" and "e-business" can be used interchangeably. As with data warehousing and CRM, there are no official definitions provided by a standards body that explicitly and authoritatively define the meanings of these two terms. For consistency, we will use "e-commerce" throughout this book, unless we are referring to product or service offerings from a vendor who explicitly uses the phrase "e-business."

Business-to-consumer (B2C): e-Commerce services that link a business with consumers (e.g., current customers and prospects).

Business-to-business (B2B): e-Commerce services that link businesses with one another.

Consumer-to-consumer (C2C): e-Commerce services that link consumers with one another (e.g., online auction services); a subset of B2C e-commerce.

Consumer-to-business (C2B): e-Commerce services that link consumers with one another and, subsequently, to one or more businesses; another subset of B2C e-commerce.

Discussion: We introduced the terms B2C and B2B earlier in this chapter. As we'll discuss in Chapter 2, though, we believe that data warehousing services and environments in support of e-commerce need to vary slightly between those needed for traditional B2C companies (e.g., an online retailer, or "e-retailer"[7]) and those needed for the less common—but still real—C2C and C2B subcategories of B2C.

e-Marketplace: An Internet-based environment in which buyers and sellers "meet," exchange information, and buy and sell products.

Discussion: Three points are worth noting. First, e-marketplaces are more commonly thought of in a B2B context (e.g., vertical, industry-specific marketplaces, as we'll discuss in Chapter 4), though arguably e-marketplaces first gained a foothold in the C2C subset of the B2C marketplace: online auction sites such as eBay (*www.ebay.com*).

Second, adding an *e* in front of a commonly used word to create a term such as "e-marketplace" is extremely common in the world of e-commerce (yet another term created the same way). Rather than list and define all the commonly used terms such as "e-supply chain," "e-procurement," "e-marketing," "e-selling," and dozens (or more likely hundreds) of others, we will as necessary define other "e" terms as we introduce them; most, however, are self-explanatory.

The third point is that even though "e" terms are commonplace in the world of the Internet and e-commerce, so too are "i" terms (e.g., preceding a commonly used word with an *i* instead of an *e,* with *i* usually standing for "Internet"); this is particularly common in company names (e.g., iVillage—*www.ivillage.com*). Unless there is a very precisely defined "i" term that we need to use, we will avoid the confusion of intermixing "e" and "i" terms in subsequent chapters.

Opportunity

So far we've discussed the background and terminology of the various pieces of this book's subject matter. What is the significance?

Simply stated, the opportunity facing nearly every company is to apply various forms of business intelligence, as instantiated through a combination of classical and next-generation data warehousing, to their e-commerce efforts in an increasingly competitive marketplace. Or, as we put it earlier, combining data warehousing and business intelligence with CRM, selected ERP functionality (e.g., those related to business-to-business, cross-enterprise transactions) and e-commerce with "1 + 1 + 1 + 1 = 1,000" synergy.

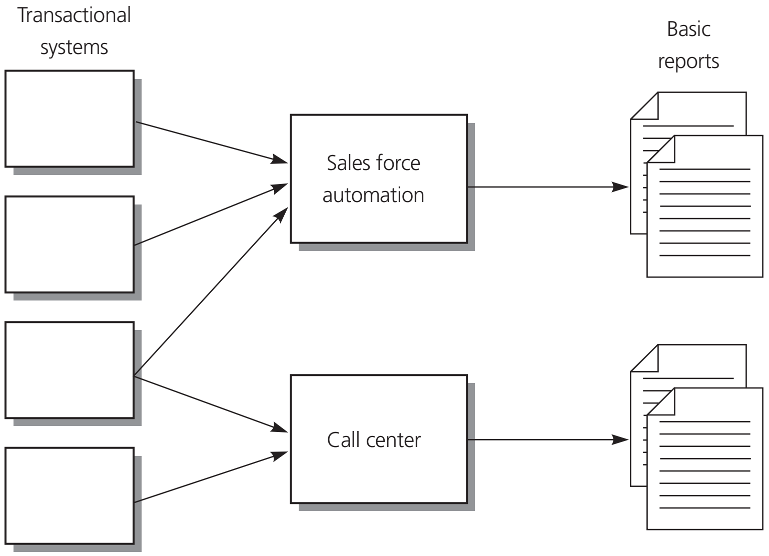

Figure 1.1 Phase 1: initial deployment of CRM core systems.

Let's look at the CRM space as an illustration of how that discipline has been slowly—but steadily—evolving to include aspects of data warehousing. Figure 1.1 illustrates an organization with two first-generation CRM core systems, one each for SFA and call center support.

Note that in Figure 1.1 there is extremely limited reporting functionality available for each of the CRM core systems: basically, a few rudimentary package-produced reports that show weekly and monthly counts for new sales staff contacts (from the SFA package) and inbound customer calls (from the call center package). No advanced analytical capabilities (e.g., drill-down analysis) or data mining was supported. In effect, the CRM core systems were early 1990s incarnations of old-fashioned mainframe-based legacy systems: they captured data from transactions and then "locked the data away" rather than use that information for business intelligence purposes.

As illustrated in Figure 1.2, the next step taken by most organizations was to extend (or at least attempt to extend) the basic reporting capabilities of the packages through custom coding. As reporting functionality increased, though, many of the systems-based reasons for data warehousing (e.g., reports using large volumes of data adversely impacting the performance of the transactional database) began impacting the SFA and call center applications.

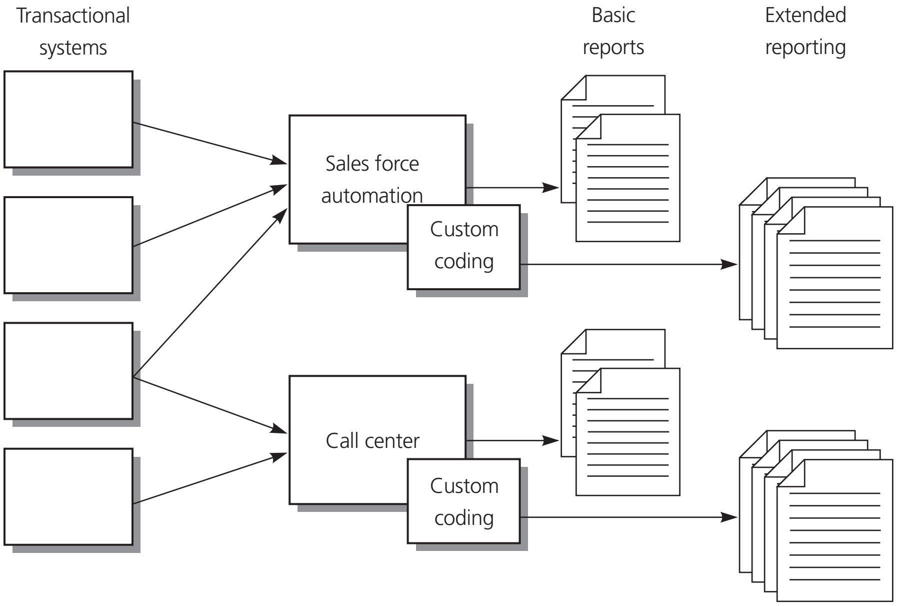

Figure 1.2 Phase 2: extending CRM core systems' reports through custom coding.

Attempting to overcome performance problems, the next step taken by most organizations was to extract data from their SFA and call center packages' databases, copy that content into a separate database, and move the reporting capabilities from the core systems to the new copy of the data. Basically, even though the term wasn't commonly used, these organizations were creating functionality-specific data marts (see Figure 1.3).

Note that the evolution to CRM data marts typically was problematic, for several reasons. Most organizations found themselves facing a mix of reporting capabilities, with some remaining against the CRM core systems due to difficulty of migration and with newer reports and, finally, some advanced analytics (e.g., OLAP functionality) applied against the CRM data mart.

A more significant problem, though, was that many larger organizations typically found themselves by the mid-1990s with *multiple* call center and SFA systems. Whether due to corporate mergers or organizations within a corporation simply choosing to pursue their own, nonintegrated CRM initiatives, most companies found themselves faced with an environment like that shown in Figure 1.4.

Faced with the desire to obtain a "whole customer view" (i.e., a comprehensive picture of customer interactions and behavior), the next step taken by

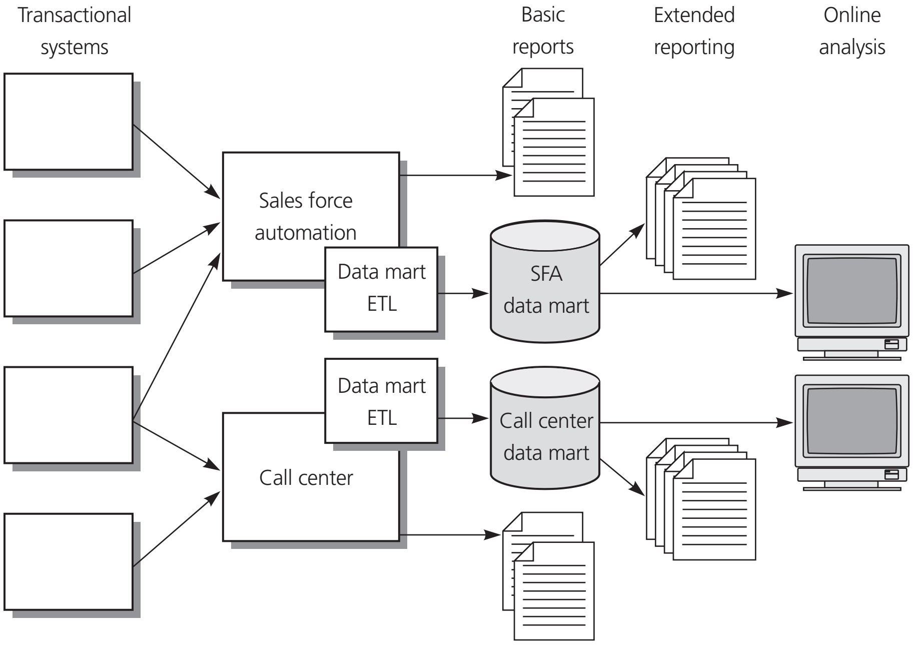

Figure 1.3 Phase 3: creating functionality-specific data marts for CRM support.

forward-looking (some would say "courageous," others might use the phrase "foolishly optimistic") organizations was to evolve from their hodgepodge of CRM data marts to a single integrated customer data warehouse. Two different approaches were popular, as illustrated in Figure 1.5. Some organizations chose to create a new customer-oriented data warehouse with the sole mission of supporting CRM analytics—not only basic reporting and analysis, but also customer segmentation, profiling, and so on. Alternatively, other organizations chose to extend an existing data warehouse that already contained customer data with content from the various CRM core systems.

Though both of the approaches illustrated in Figure 1.5 appear to be a straightforward, desirable state of integrating CRM-produced data for reporting and analytical purposes, it should be noted that most organizations' experiences with this step along the evolutionary path are usually only partially successful—not for reasons of technology, but because of organizational issues such as system ownership and lack of incentive to move from an environment in which "subenterprise" organizations control their own systems with a minimum of necessary cooperation with other organizations. The reason it's important to note these difficulties is that, as shown in

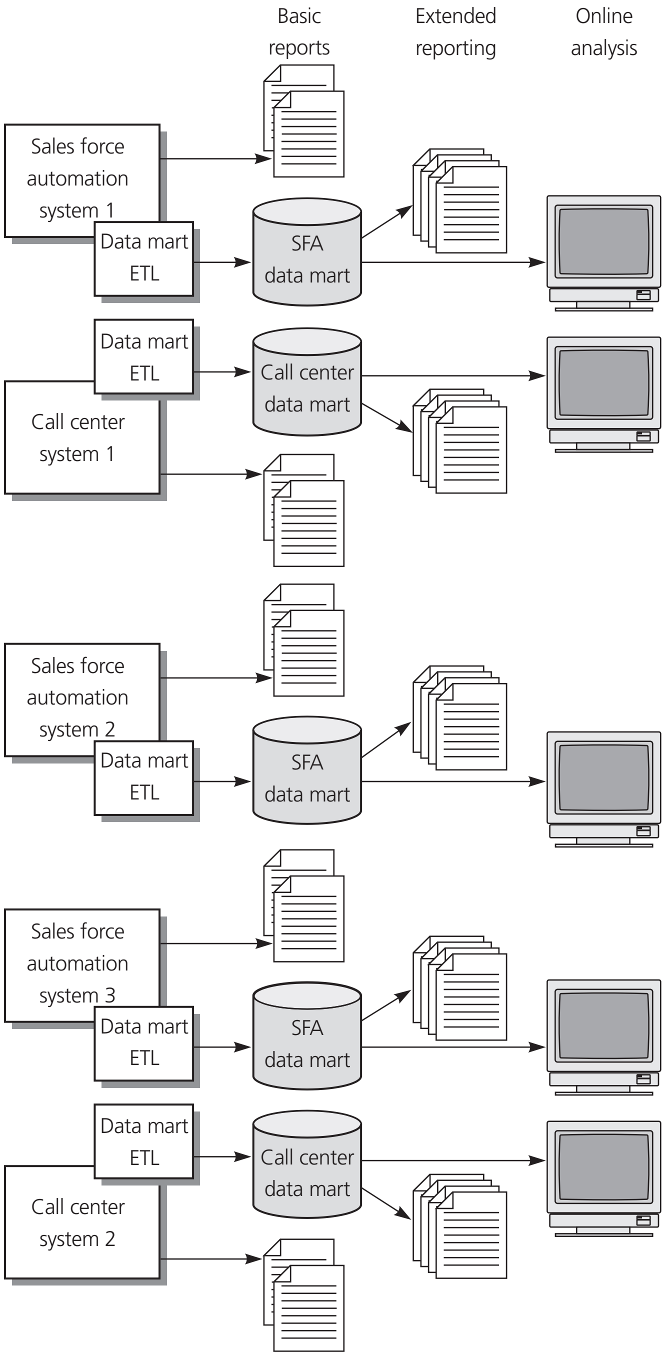

Figure 1.4 Phase 4: multiple CRM core systems and nonintegrated data marts.

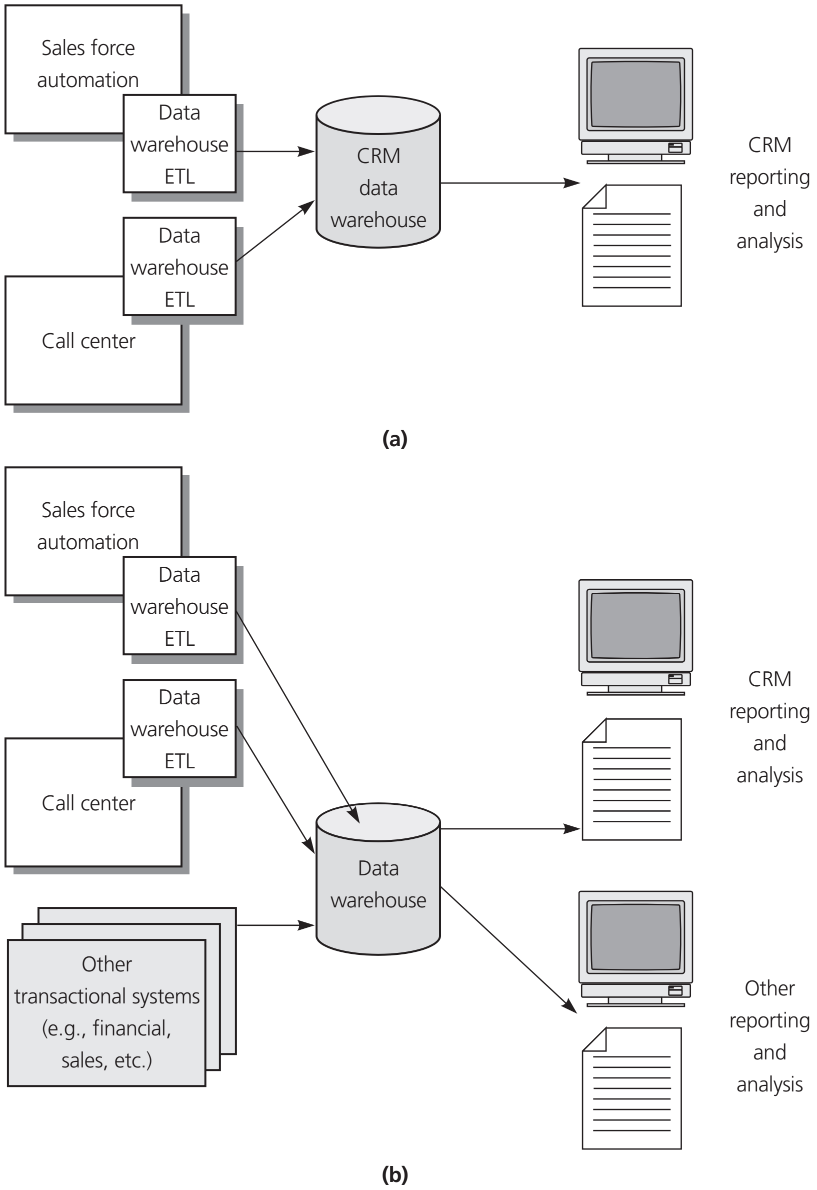

Figure 1.5 Phase 5: two approaches to integrated CRM-focused data warehousing: (a) CRM-specific data warehouse and (b) CRM data added to a general-purpose data warehouse.

Figure 1.6, along comes the Internet and e-commerce . . . and an entirely new set of CRM integration problems.

Figure 1.6 represents the current state of affairs with regard to most click-and-mortar corporations' CRM systems. Whereas an Internet pure-play company such as an online B2C retailer might have an environment like that shown in Figure 1.5(b), a click-and-mortar company typically finds itself backing away from whatever successes it was able to attain in building integrated whole-customer views with a data warehouse for the purpose of supporting CRM analytics. Why? The frenzied pace at which "old economy" companies (e.g., those that existed before the late 1990s and that contain non-Internet sales channels) rushed to get onto the Internet was often accomplished by either (1) chartering an internal e-commerce organization to build and deploy e-commerce capabilities as quickly as possible and deferring plans for integrating their e-CRM capabilities with already existing CRM core systems and CRM analytics until "some time in the future," or (2) spinning off an entirely separate company (as was common during 1999 and early 2000, when dot-com divestitures were all the rage). In the latter case, integration between e-commerce and existing non-Internet channels wasn't deferred; they weren't part of the business model.

And now we come to the critical juncture that provides the foundation for this book's subject matter. It would be natural for a company to take a step back, look at an environment such as that shown in Figure 1.6, and initiate a program to integrate their e-CRM information (e.g., online customer contacts, requests made through an online customer support system, etc.) with corresponding content from traditional CRM core systems into a single, consolidated *traditional* data warehouse. By "traditional" we mean a data warehouse that is stocked in a batch-oriented manner using traditional ETL processes, and whose primary purpose is supporting the various types of business intelligence discussed earlier in this chapter (e.g., basic reporting, OLAP, EIS, and data mining). Figure 1.7 illustrates this approach.

While success with the approach shown in Figure 1.7 would certainly provide high levels of integration across e-commerce and traditional channels, that integration would typically be limited to *after-the-fact* business intelligence. This is an important distinction to make. As illustrated in Figure 1.8, a "diversion" on the CRM analytics evolutionary path shown in the previous figures is using customer data not only for after-the-fact analytics but also applying the results of those analytics to "quasi-transactional" business processes such as targeted marketing and campaign management.

The step-by-step, loosely coupled, and typically batch-oriented business processes illustrated in Figure 1.8 may be satisfactory for traditional customer interaction channels, but at the "Internet speed" of e-commerce, they are woefully

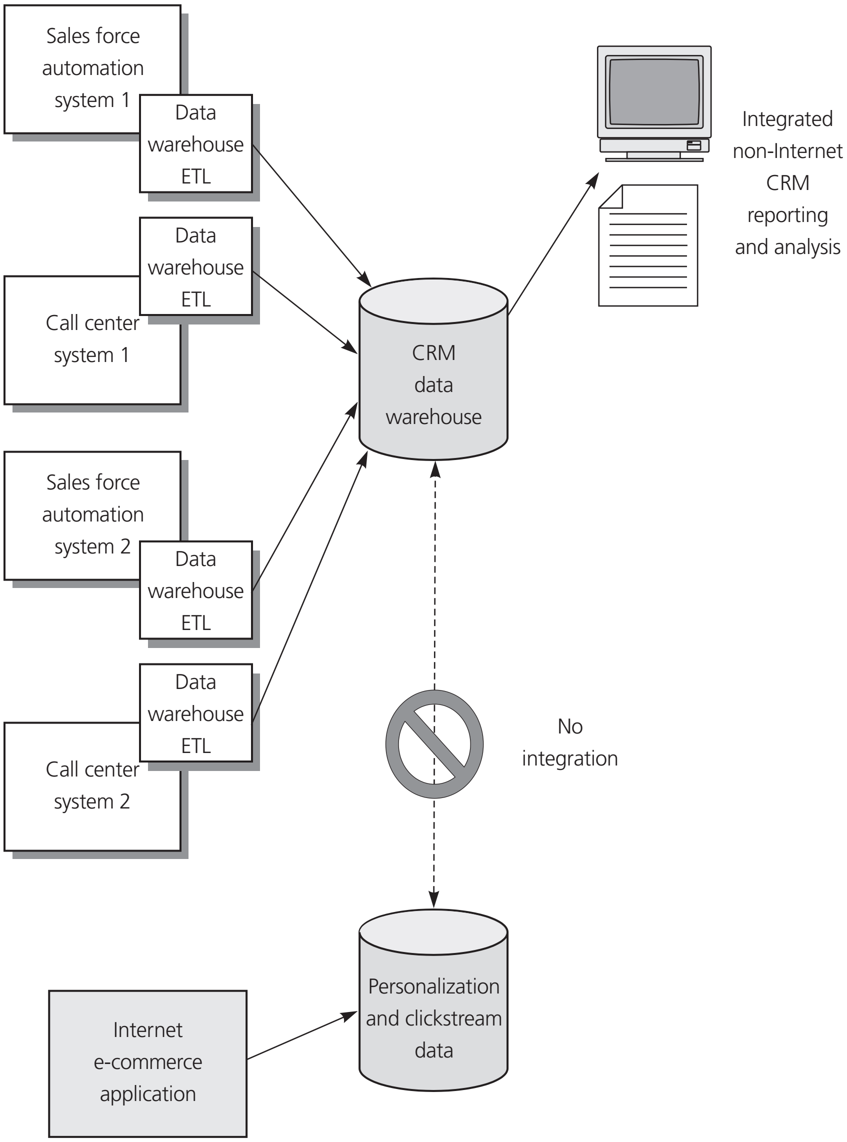
Sales force automation system 1
Data warehouse ETL
Data warehouse ETL
Call center system 1
Sales force automation system 2
Data warehouse ETL
Data warehouse ETL
Call center system 2
CRM data warehouse
Integrated non-Internet CRM reporting and analysis
No integration
Internet e-commerce application
Personalization and clickstream data

Figure 1.6 Phase 6 (current state): CRM-specific data warehouse with nonintegrated e-commerce and traditional CRM analytics.

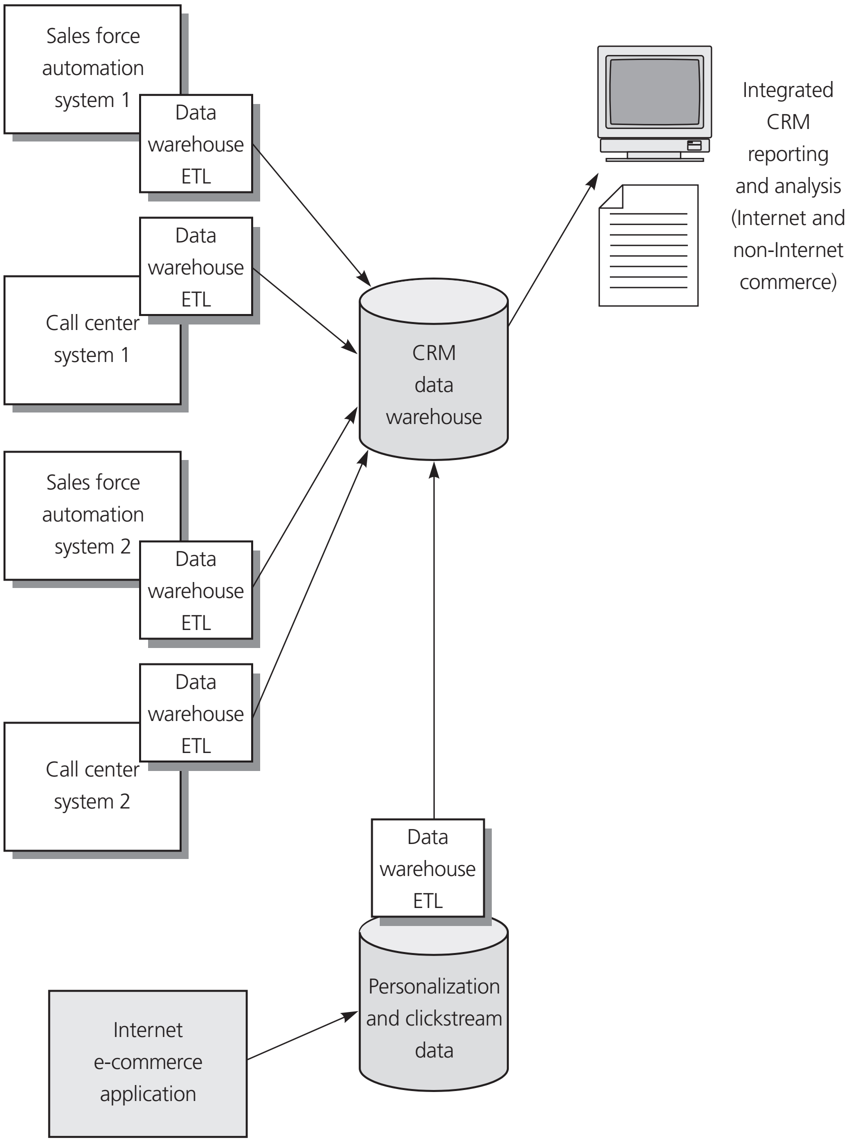
Sales force automation system 1
Data warehouse ETL
Data warehouse ETL
Call center system 1
Sales force automation system 2
Data warehouse ETL
Data warehouse ETL
Call center system 2
CRM data warehouse
Integrated CRM reporting and analysis (Internet and non-Internet commerce)
Data warehouse ETL
Personalization and clickstream data
Internet e-commerce application

Figure 1.7 One approach to integrating e-CRM and traditional CRM data content.

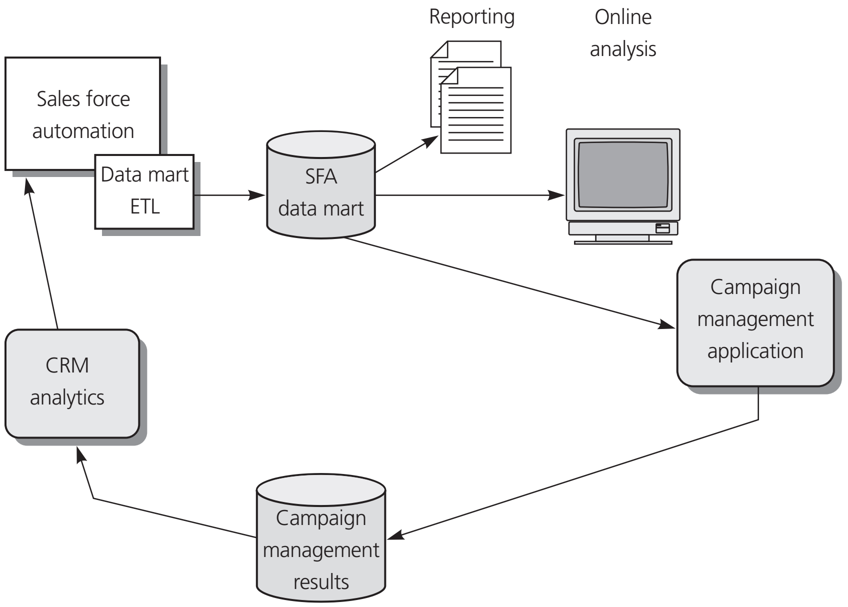

Figure 1.8 Using content provided by CRM analytics in customer-facing transactional processes.

inadequate. *And this presents the opportunity:* rather than simply graft "old economy" business processes onto Internet-produced customer data, organizations are taking a fresh look at innovative ways of applying business intelligence to various aspects of e-commerce (in this thread of examples, CRM-focused processes and systems). We'll take an in-depth look in subsequent chapters at many examples. As a transition to that subsequent discussion and as a conclusion to this section, one next-generation example of applying business intelligence to e-commerce in a B2C scenario is using a "whole-customer profile" created from online and offline channels and, in real time, combining that static business intelligence content with a customer's online shopping activity—what he or she is putting into the online shopping cart, what items are being removed from the shopping cart, what items are being browsed and not selected for purchase, and so on—and making shopping-time recommendations for complementary products, alternative products, as well as other shopping-time suggestions. Figure 1.9 illustrates this type of environment.

As will be discussed in subsequent chapters, traditional, batch-oriented data warehousing products and capabilities are *not* adequate to completely support an

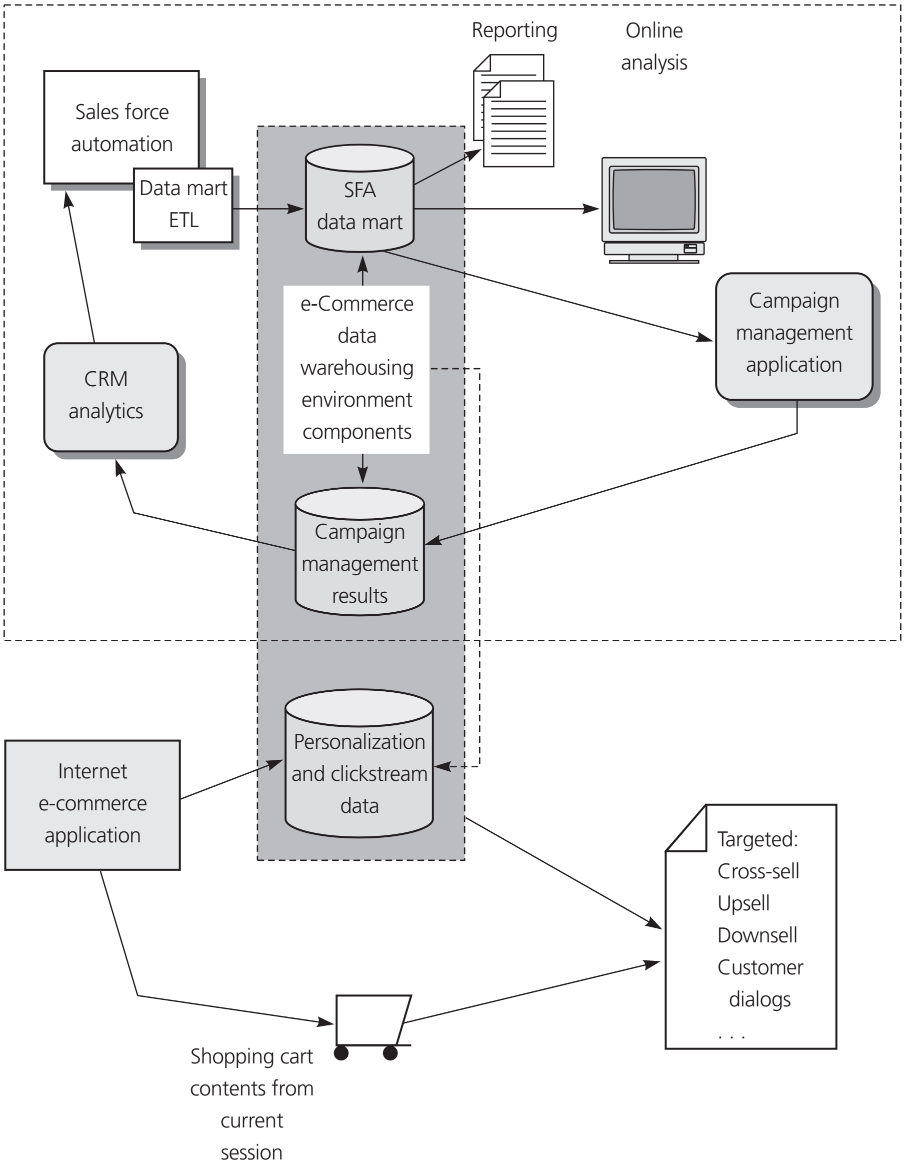
Reporting
Online analysis
Sales force automation
Data mart ETL
SFA data mart
e-Commerce data warehousing environment components
Campaign management application
CRM analytics
Campaign management results
Personalization and clickstream data
Internet e-commerce application
Targeted:
Cross-sell
Upsell
Downsell
Customer dialogs
. . .
Shopping cart contents from current session

Figure 1.9 An example of combining traditional data warehouse content with other e-commerce data.

environment like that shown in Figure 1.9. Real-time access to various types of data is needed; bidirectional flows of data are required; and robust applications logic, far more complex than that required for traditional after-the-fact business intelligence or even offline campaign management, is a necessity.

Summarizing the opportunity: instead of just letting the natural CRM evolution proceed to include e-commerce, an organization should jump-start its approach to using customer data in business processes, both on the Internet as well as through traditional non-Internet channels, if applicable (i.e., for a click-and-mortar company). Doing so extends far beyond the simple-to-understand B2C e-retailing example illustrated above; innovative ways of applying current and next-generation data warehousing and business intelligence to e-commerce are applicable to B2B environments (see Chapter 4), as well as the C2C and C2B variations of B2C that we'll discuss in Chapter 3.

And, as we'll discuss later, it's essential to ensure that business processes are modernized and kept in concert with their underlying technology. Therefore, implementing the models and architectures we'll discuss throughout this book *must* be done as part of an overall end-to-end business strategy that encompasses both e-commerce and traditional channels and the necessary touch points throughout those respective sets of processes.

Challenges

There are demonstrable and self-evident business advantages to making data warehousing and business intelligence capabilities an integral part of an organization's e-commerce efforts. However, it needs to be noted that there are three primary challenges in doing so, as will be discussed in this section:

- "Discipline-centric" views among corporate strategists, consultants and systems integrators, and product vendors
- "Dot-com" spin-offs as separate companies
- The need to deliver products and solutions in "Internet time"

Discipline-Centric Views

Simply stated, there exists an artificial, unnatural, and unhealthy demarcation between the discipline of data warehousing and that of e-commerce. Refer back to the sequence of evolutionary steps in the CRM space throughout the 1990s that was described and illustrated in the previous section. Looking back, it's natural to wonder why, back when organizations were first embracing and deploying SFA and call center capabilities, they didn't recognize that implementing CRM core systems without a close integration with corresponding CRM analytics wasn't desirable. Why didn't they implement both sides of CRM—transactional and informational—together, or at least in lockstep with one another?

Many reasons exist, such as companies' budgets and tight development schedules that typically permitted implementation of CRM transactional capabilities. Additionally, CRM product vendors faced intense pressure to focus their development energies on their primary products—initially, the CRM core systems—in an ever-competitive marketplace and to defer analytical extensions to "some future release."

The above reasons are understandable. The same marketplace, time, and budgetary pressures were evident in the ERP arena, and it wasn't until the latter part of the 1990s that ERP vendors began providing data warehousing extensions to their core products (e.g., SAP's Business Information Warehouse—BIW, also known as Business Warehouse or BW).

A more ominous part of the problem, though, is that many consultants and systems integrators who specialized in CRM dismissed data warehousing and business intelligence as irrelevant to the management and use of customer data. Even when clients and prospective clients expressed an interest in coupling their SFA or call center functionality with robust reporting and analytical capabilities, many CRM-centric consultants proposed and attempted to implement solutions such as those illustrated in Figures 1.2 or 1.4—solutions that were problematic for reasons of performance, lack of necessary data integration, or both.

Correspondingly, many data-warehousing-oriented consultants and integrators were unfamiliar with emerging product-provided CRM analytical capabilities, and instead implemented custom-built and often error-prone solutions using general-purpose querying and reporting tools rather than integrating a CRM analytics package with an underlying data warehouse.

While the barriers between CRM and data warehousing have finally crumbled with the introduction and acceptance of CRM analytically focused products from vendors such as E.piphany *(www.epiphany.com)* that contain an underlying data mart to support campaign management and other functionality, *history is repeating itself in the e-commerce world.* Many consultants and systems integrators who specialize in e-commerce have a lack of understanding of data warehousing, and instead see e-CRM as being all that's necessary for business intelligence purposes. While this may be true in pure-play Internet companies, click-and-mortar companies with a combination of traditional and e-commerce channels *need* a mixture of traditional data warehousing and business intelligence with the analytical capabilities provided by e-CRM. And, likewise, other consultants and systems integrators who specialize in data warehousing still see their discipline in an early-1990s context, based solely on batch processing and simplistic after-the-fact reporting and analysis.

Dot-com Spin-Offs

A second complication is that organizations themselves are inadvertently contributing to the problem through their e-commerce spin-off activities, as discussed earlier in this chapter. While a dot-com spin-off may provide (at least temporarily) high returns through stock market valuation of the separate company, integration of "new economy" and "old economy" channels then becomes extremely problematic. As a separate company, an e-commerce spin-off's technology and business strategists often have little or no incentive to integrate their online customer data with that of their offline cousin's systems. For example, a company embracing the trend toward mixed-channel customer processes—for example, ordering a product online but picking it up at the local mall rather than having that product shipped to the customer's home—will face many problems trying to retroactively "reintegrate" e-commerce and non-Internet customer activity data to report on and analyze mixed-channel activity (e.g., tracking the relationship between additional purchases in a store when picking up a product ordered online and the relationship between complementary in-store product sales and subsequent online purchases).

Operating at "Internet Time"

The third challenge is the need for every entity involved in e-commerce—companies selling products to the general public (i.e., B2C companies); companies frantically trying to build online marketplaces with one another (i.e., B2B companies); the venture capital and investment community; consultants and systems integrators; and product vendors—to deliver functioning results (systems, products, online customer interactions, etc.) as quickly as possible . . . that is, in what has become known as "Internet time."

A direct relationship exists between time-to-results and amount of functionality. That is, the need to deliver results extremely quickly *will* result in reduced functionality (or at least less functionality than if more time had been available). The consequence of "Internet time" driving functionality has been the challenges discussed above: demarcation between e-commerce and data warehousing, as well as the lack of an up-front consolidated strategy covering integration between Internet and non-Internet channels.

Overcoming the Challenges

Throughout this book we'll present solutions that take the challenges discussed above into consideration. That is, we won't make recommendations and present architectures for overwhelmingly complex three-year "Big Bang" business-to-consumer projects in which a team locks itself away for two or three years, performs requirements analysis and then design and development, and then finally delivers its initial comprehensive functionality long after the window of business opportunity has closed. Rather, we'll focus on iterative and incremental approaches in which up-front strategy and architecture needs to be comprehensive to prevent architectural dead ends, yet at the same time components of functionality can be delivered in 90–120-day increments, with each component building on what has already been delivered.

This approach, however, does require an awareness of the challenges discussed above and a conscious effort by all those involved to overcome them. Of particular note is the first challenge discussed, the artificial demarcation between data warehousing and e-commerce. It is imperative that whoever is driving an initiative within a particular setting—whether internal person or outside consultant—acknowledges up front the necessity of making data warehousing and business intelligence an integral part of an e-commerce initiative.

Summary

In this initial chapter, we have provided the foundations and framework for the material in the chapters that follow. The background material and terminology presented for the various disciplines such as CRM, ERP, and data warehousing that are now converging is, we believe, very important to understand to ensure that past missteps of insufficient integration (which in turn results in less-than-optimal solutions) aren't repeated in the e-commerce arena.

By way of transitioning from this introductory material to concepts and architecture, we'll next discuss applying data warehousing to business-to-consumer settings in Chapter 2, since most readers are likely to have had personal experiences with various B2C companies through product browsing, purchasing, and customer service.

Endnotes

1. A more detailed—and somewhat irreverent—discussion of how DDBMS failure evolved into 1990s-style data warehousing can be found in A. Simon, *Data Warehousing for Dummies,* IDG/Dummies Press, 1997, Chapter 1.
2. Most data warehousing professionals recognize Bill Inmon as having first coined the phrase "data warehouse," though IBM's DDBMS effort that was still under way at the time was called their "Information Warehouse."
3. Of course, nearly everyone has had experiences with call centers, both on the inbound side (e.g., calling for customer support) and on the outbound side (e.g., receiving one of

those aggravating trying-to-sell-you-something calls, usually at the most inconvenient time possible).

4. There is no *official* definition of a data warehouse, that is, a standard definition supported by a standards committee such as the American National Standards Institute (ANSI).
5. More information about data mart architecture can be found in A. Simon, *90 Days to the Data Mart,* John Wiley & Sons, 1998.
6. Another less frequently used term is "clicks and bricks."
7. Also referred to as "e-tailer."