# Management and Strategy

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>16.0</td>
<td>INTRODUCTION</td>
<td>540</td>
</tr>
<tr>
<td>16.1</td>
<td>ORGANIZATIONAL MATURITY</td>
<td>542</td>
</tr>
<tr>
<td>16.2</td>
<td>BUSINESS OBJECTIVES</td>
<td>554</td>
</tr>
<tr>
<td>16.3</td>
<td>KEY POINTS IN CHAPTER SIXTEEN</td>
<td>571</td>
</tr>
</tbody>
</table>
16.0 INTRODUCTION

In 1843, when telegraph technology was only seven years old, an amateur clock maker named Alexander Bain combined a telegraph machine with parts from old clock mechanisms. Bain received a British patent for “improvements in producing and regulating electric currents and improvements in timepieces and in electric printing and signal telegraphs.” At the time Bain’s invention was called the chemical telegraph, but today we regard it as the first fax machine.¹

Bain died in obscurity and poverty, and it was 100 years after his death before the fax machine was widely adopted as a business tool. Why wasn’t this invention taken up with the telegraph and telephone (an even later invention)?

It would be wrong to imagine a hopeful but naïve Bain trying to send faxes when no one had the capability to receive them. Bain and other early advocates of fax machines implemented the same business model for fax machines that was being used for telegraphs, with fixed office locations for city-to-city transmissions. This deployment architecture allowed fax machines to become an important means for distributing news photographs, but it didn’t provide much benefit to businesses. A network pattern, in which every business has its own fax machine, provides far greater benefit, but wasn’t possible as long as fax machines were expensive.

The fact that it took 130 years for Bain’s innovative technology for document exchange to succeed motivates us to write this final chapter to complete the story of Document Engineering. The dominant theme of this book so far has been how to understand documents and the business processes that use them. In this final chapter we look at the management and strategy concerns that cut across and frame the various phases and tasks in a Document Engineering effort.

While technology considerations are important, it is not the technology that primarily determines whether Document Engineering approaches will be successfully adopted within an enterprise or by two or more firms with business relationships. Other significant factors include the existence of industry standards or reference models, mechanisms that encourage technology adoption, the technological and
process maturity of the enterprises, their relative power in their relationships, and the extent to which they have complementary long-term business strategies.

Finally, the project has to make financial sense for the organizations or firms carrying it out because for almost every enterprise, one key measure of business success means making enough money to stay in business.

Putting a chapter on management and strategy concerns at the end of this book doesn’t mean that we should defer these concerns until the end of our project. Indeed, we should begin with these issues because they determine the goals and scope of our work, or even whether we should attempt to do it. But many of the concepts and examples in this chapter would be hard to understand if this chapter appeared earlier in the book.

And emphasizing these management and strategy dimensions of Document Engineering in a separate chapter doesn’t mean that we’ve ignored them up to now. For example, when we reviewed the big ideas of XML in Chapter 2, we noted that many apparently technical questions like how much validation to perform or the architectural locus of transformation were better answered by business and relationship factors. This theme, and the related idea that business models and technology continuously co-evolve, became more prominent in Chapters 4 and 5. There we presented a view of business in which patterns for processes and document exchanges function as building blocks both for improving existing business models and inventing new ones. Which pattern fits best is determined by both technical and business factors.

In Chapter 5 we described the co-evolution of technology and business to explain why a potentially disruptive technology can sometimes have little impact if it doesn’t fit into an existing model or pattern—or if the original technology proponents fail to identify that a relevant pattern exists.

Our review in Chapter 6 of interoperability challenges showed that mismatches between models in technology or syntax were far less detrimental than those resulting from a lack of shared context and goals between the parties in a business relationship. The importance of these business considerations shows why a purely technical perspective on business informatics, document exchanges, and web services is inadequate.
That’s why the Document Engineering approach that we presented in Chapters 7-15 began with the goal of understanding the requirements of the context of use. Defining the context identifies relevant organizational stakeholders and determines whether a more strategic or more tactical perspective is appropriate. Collecting and sampling the document inventory, taking stock of the existing information exchanges, and balancing the concerns of different stakeholders also require both technical and business insights. As we harvest components and develop conceptual models of the rules for our information requirements, we confront issues about the scope of analysis. Again, these are more often influenced by capabilities, management goals, and allocation of resources than by purely technical requirements. Finally, when we are ready to deploy new documents and services, their priority and organization will be influenced by business opportunities, relationships, competition, and strategic considerations that shape the business case.

### 16.1 ORGANIZATIONAL MATURITY

Throughout this text we have talked about standards and reference models and how they encourage the evolution and adoption of Document Engineering approaches and technologies. But while these are often necessary ingredients for success, they aren’t sufficient. Standards and business patterns are of no value unless an enterprise can recognize that they are relevant and can adapt them to close the gap between its current, As-Is models of documents and processes and its desired, To-Be ones. We should not even assume that an enterprise could understand that its current processes and documents might be inefficient or suboptimal and that there is a better way of doing things. Its ability to do this is dependent on its level of organizational maturity or capability.

One aspect of this capability is the pure resources needed: budget, time, technology, and available people. But a more important aspect of capability is the overall readiness of an organization to make a Document Engineering effort successful. Document Engineering is too new a discipline, and service oriented architecture too new as a domain in which to exercise it, for most companies to have any direct experience in doing it or doing things like it. So we won’t deal too much with generic resource concerns; instead we will focus on how capability and maturity affect Document Engineering.
16.1.1 MOTIVATING CAPABILITY ASSESSMENT

It can be expensive when a product or a project fails. It may be harmful or even fatal to a business. But while we can measure the performance of an organization after it builds a product or carries out a project, we can’t measure performance on something that hasn’t been done before. However, we can do a capability assessment to predict the likely success of a project.

A guiding assumption in capability assessment is that the maturity, predictability, and repeatability of the process used will determine the quality of the service produced. How these processes are managed matters just as much as what they are. We have developed a Document Engineering Capability Maturity model to guide this assessment.

A capability assessment can predict the likely success of a project.

16.1.2 THE CAPABILITY MATURITY MODEL

In the late 1980s, the Software Engineering Institute at Carnegie-Mellon University developed a Capability Maturity Model for Software that had a profound effect on software engineering practices throughout the world.³

The CMM describes the principles and practices underlying software process maturity and is intended to help software developers improve the maturity of their software processes in terms of a 5-level evolutionary path from ad hoc, chaotic processes to mature, disciplined software processes (see SIDEBAR).

For two decades the CMM has been used to assess the capabilities of software firms, and despite being officially retired by the Software Engineering Institute, it is still widely used to make contracting or outsourcing decisions. The most common rating (2004 data) for firms is still only Level 2, meaning that processes are repeatable but
not standardized. However, it is encouraging that the average capability rating is steadily improving.4

The CMM Levels

The typical characteristics of organizations at each of the 5 CMM levels are:

Level 1. Initial (heroics)

- The software process is ad hoc, and occasionally even chaotic.
- There is no stable environment for development and maintenance.
- Schedules are “backed in” and not based on quality.

Level 2. Repeatable (basic project management)

- Projects start from requirements that are subsequently tracked.
- Processes are established to manage cost, schedule, and functionality.
- Processes include version control, automated builds, and so on.

An effective process is one that is documented, trained, practiced, enforced, and capable of being improved.

Level 3. Defined (process standardization)

- The software process for both management and engineering activities is documented, standardized, and integrated into a standard software process for the organization.
- Typically some group or department is responsible for developing and standardizing processes.
- Training programs ensure that all staff and managers have required knowledge and skills.

Level 4. Quantitative (charts and graphs)

- Detailed measures of the software process and product quality are collected.
- Both the software process and products are quantitatively understood and controlled.
The idea that organizations can be classified according to their capabilities is a very sensible one, and we’ve used the CMM philosophy in many projects to assess and manage risk.

**16.1.3 THE DOCUMENT ENGINEERING CAPABILITY MATURITY MODEL**

The CMM was developed for the domain of software development, but it can also be useful for understanding an organization’s problems and prospects in other domains because we always need to understand current capabilities and perspectives to plan improvements. We can’t get from here to there unless we know where “here” and “there” are. The challenge lies in the fact that in any business ecosystem different people, even within the same organization, see “here” and “there” differently.

We adapted the concepts of CMM to Document Engineering and found it useful as a diagnostic tool to predict and understand problems and to communicate with high-level executives who don’t need to know about the nuts and bolts of their documents and business processes. But they do need to be sure of the effectiveness of their documents and processes that ensure quality and transparency of financial reporting, for example, as required by the U.S. Sarbanes-Oxley mandate (see Section 4.2.2.6).

We’ve made two important changes to the original CMM approach in adapting it for Document Engineering:
• We don’t believe it is useful to force assessments into levels.
• We need to distinguish between technology maturity and process maturity.

### 16.1.3.1 The Problem With Levels

Levels often encourage an adversarial character to the assessment. People fixate on the levels, and we want them to pay attention to a more nuanced assessment and recommendations. If an organization’s documents and processes are too informal or underspecified to enable an effective audit or successful automation effort, it is important to address the specific problems, not the summary evaluation that the organization has reached a particular maturity level.

### 16.1.3.2 Technology and Process Maturity

Technology maturity and process maturity are separable dimensions; we can have mature capabilities on one and not on the other. Understanding this distinction helps us portray our assessment more accurately and give more precise recommendations about what to do to improve capabilities and therefore the chances for success.

Cultures with mature procedures and processes take a strategic view of their business. They can usually adopt new technologies if they choose to do so. They have the skills and the tools to predict the business value of adopting new technologies and processes and to measure their progress in doing so. Of course, the organization may be disrupted by the new technology, but presumably that’s the point, and the organization understands how to systematize the new processes enabled by new technologies.

Organizations with low process maturity often don’t recognize the inefficiencies in how they do business and can’t adopt technology easily. Even when they recognize problems, they may put up with them because they don’t have the confidence in their ability to introduce new processes and technology that would eliminate them.
Organizations with low process maturity can’t adopt new technology easily

How a firm handles its procurement is a good indicator of its process and technology maturity. Procurement is a common business process where automation has substantial benefits. Most large enterprises have substantially automated the document exchanges in their supply chains for the direct procurement of the goods and materials that go into the products they make (often with EDI). But many companies, especially those with weak process maturity, have not yet automated their indirect procurement of the goods and services needed to run the business. These include office supplies, travel, maintenance and repairs, package shipment, temporary help, and many other categories with large numbers of low value transactions often initiated by employees other than the purchasing specialists who conduct direct procurement.

It may seem harmless for employees to disregard the company’s purchasing processes and buy office supplies during their lunch hour or make airline reservations that maximize their personal frequent flier miles. But the company then pays the retail price rather than a discounted corporate one that might be much less. Automating indirect procurement eliminates this maverick purchasing and lets businesses track and aggregate their purchases to negotiate volume discounts with suppliers. Furthermore, automated systems can encode and enforce corporate policies about preferred providers and spending limits and prove compliance to management and auditors.

Research or advanced technology organizations are chartered to explore new technologies and always have strong technology capabilities. But the people who work in these organizations probably don’t have it in their job description or their nature to systematize and measure processes. Firms with high process maturity recognize this gap and institute technology transfer processes to put new technologies into practice. In contrast, firms with low process maturity squander the work of their R&D labs and see it commercialized by competitors or by researchers so disappointed by the firm’s inertia that they quit to launch start-ups.
16.1.3.3 The CMM Framework and the Model Matrix

We can use our familiar Model Matrix to understand the application of the Capability Maturity framework to Document Engineering.

![Model Matrix Diagram](image)

Figure 16-1. Different Maturity Emphases on the Model Matrix

Figure 16-1 reminds us of the different skills sets and perspectives that come together in Document Engineering. It helps us understand how all of them contribute to achieving a complete understanding of an enterprise or organization that spans from its business model to the documents exchanged at the information level to carry it out. But such a complete perspective isn’t always possible or necessary. The capability maturity of the organization determines the amount of emphasis each of these perspectives will be given.

Some document automation, user interface, and even EDI projects have the explicit or implied goal of transforming printed documents or forms into electronic versions. If an organization’s primary capabilities are in these technology areas, it is not sur-
prising that their modeling activities take a more tactical perspective. They just aren’t used to considering the enterprise-level and business process context of their projects. This tactical perspective is also referred to as a bottom-up approach.

The business case for these projects is often a mandate by a dominant business in an asymmetric relationship for its trading partners to automate the exchange of information. Typically they will need to conform to the dominant company’s proprietary formats or implementation guidelines.

Here the goal is seen as taking an existing process (often, someone else’s) and encoding its rules directly in applications or electronic documents. Thus modeling is often viewed as needing little resources or, in the worst case, as irrelevant.

In contrast, organizations with strong process capabilities usually take a strategic approach that looks at organizational and high-level business concerns between an enterprise and its trading partners. This top-down approach provides a very coarse grained view focusing on business capabilities or competencies.

Process modeling concerns in these kinds of projects include the compatibility of business models, customer and supplier relationships, accounting practices, and acceptable business practices. No one pays much attention to the design of models for documents or their implementation technology as they assume these will become someone else’s (tactical) problem.

The Document Engineering modeling approach tries to balance the bottom-up and top-down approaches and depends on both process and technology capabilities. While technology and implementation decisions should always be secondary to business and information requirements, business processes and business documents are complementary and should receive the same level of engineering rigor. They can both be encoded so that computer applications can process them directly. But this requires bridging between the strategic and tactical perspective—what we call meeting in the middle.

Processes and documents are complementary and should be receive the same level of engineering rigor
As we discussed in Chapter 8, whether a project is more tactical or strategic defines what is and is not possible, how much work it will take, the scale or scope of technology, and the resources needed to implement a solution. A service oriented architecture is an example of a strategic platform that, once established, can be used to develop and deploy web services more incrementally as tactical projects.

However, even on tactical efforts there may be numerous conflicting project goals, so we should define them precisely and prioritize them to have a basis for deciding how to proceed when conflicts emerge. In some sense, this means that we have to take at least a somewhat strategic perspective when we carry out tactical projects.

### 16.1.4 Conducting a Capability Assessment

The tasks we collectively called analyzing the context in Chapter 8 often yield an informal sense of the capability maturity of the enterprise we’re studying as we initiate a Document Engineering project. But normally we wouldn’t describe this work as an assessment unless we ask explicit questions about technology and process maturity in the course of understanding the context.

A strategic, executive-sponsored project is more likely to contain an explicit activity to conduct an assessment, because senior management is more comfortable with bringing in outside consultants to take an objective look at things and is increasingly being required by law to do just that. Middle managers are less likely to be able to do this explicitly, but they can conduct a stealth or implicit assessment so that they can scope the project and manage the risks.

### 16.1.4.1 Questions That Assess Capability Maturity

We’ve found that a surprisingly small number of simple questions can tell us a lot about the capabilities that predict success or failure in Document Engineering efforts. The best questions yield unambiguous answers that can be supported by objective evidence. Obtaining evidence for answers is essential because when the assessment is an explicit activity, people often respond with what they think are the
right answers even when they don’t have the capabilities that the question is trying to evaluate. Good questions cut through what people do or say at the surface to expose deeper values or themes about how focused the firm is on understanding, measuring, and improving what it does to add value to its business and how capable it is of doing so.

Below is a list of some generic questions and some interpretation of possible responses in terms of capability maturity. In an assessment these questions need to be augmented by others that address factors specific to the context of use, but this list should convey the value of asking simple questions to diagnose capability maturity.

- Why is your organization considering this project?

Respondents in an organization with mature process capabilities will justify a project in terms of measurable business value, such as reduced operating costs or delivery times. If the organization also has mature technology capabilities, respondents might emphasize its innovative use of new technologies to achieve these process improvements. In contrast, if an organization has weak process capabilities and a bias toward technology, respondents might mention the goal “to explore new technology” without making an explicit connection to measurable business events.

Respondents in organizations with immature capabilities on both dimensions might describe the project as being imposed on them by upper management or by a dominant business partner and express some concern about their skills to carry it out or describe their efforts in heroic terms.

- What are the procedures for processing documents (in transactional contexts) or creating publications (in narrative contexts)?

Respondents in an organization with process maturity will describe an explicit and formal step-by-step process generally followed to complete transactions or produce publications, supported by current documentation. The process is completed on time and meets explicit quality criteria. If this organization has commensurate technology maturity, the processes will be automated and enforced by workflow or process management software and the quality criteria might involve validation or conformance to templates or schemas at each step.
Respondents in less mature organizations will describe their processes as less formal, and mention work-arounds and exception-handling activities that occasionally cause delays or lower-quality results. In the least mature organizations respondents might describe informal processes in which external actors or organizations control events: “we sometimes just go with the documents we have when the product is ready to ship or when the customer demands it.”

- How are the source files for important documents managed?

Organizations with process maturity have clear policies for document and data retention and might maintain a centralized repository with access and configuration controls. Technologically sophisticated organizations will automate backups and archiving to ensure conformance with these policies.

Less mature organizations delegate the management of documents to the organization or person who creates or processes them. Organizations with the weakest capabilities manage important documents informally or even outsource their storage; respondents might say “A contractor performs our customer support, and we don’t know what records are kept.”

- What kind of support is there in your organization for reuse of information that appears in more than one type of document or application?

Organizations with process maturity carefully analyze their information requirements and maintain libraries of reusable models, schemas, or document fragments and have governance policies for controlling versions and maintaining interoperability. The best of them use technology to encourage reuse, promote the idea of model based applications, and develop or follow industry or company standards to facilitate reuse both within the enterprise and with other companies.

Less mature organizations recognize the value of reuse but only encourage it informally. They don’t make the investments in modeling, training, or technology necessary to institutionalize the practice. The least mature organizations don’t have any cross-document or cross-application initiatives for reuse, and any reuse that happens is informal and ad hoc.
16.1.4.2 Capability Implications for Document Engineering

Conducting a capability assessment and identifying requirements for a context of use (Chapter 8) are closely related activities. Businesses with a process bias may be inclined to support a strategic, top-down business driven analysis and may be more willing to consider process redesign and reengineering. Businesses with a technology bias may favor more tactical, bottom-up approaches such as automating the documents that already exist in printed or legacy formats.

Businesses with good process capabilities are more likely to implement model based applications

Businesses with good process capabilities are more likely to have the discipline needed to implement model based applications. One of the benefits of using models is in reinforcing common patterns or standards, and creating and deploying standards requires a process perspective.

An enterprise that has immature processes and immature technology—running by ad hoc means with trailing edge technology—will undoubtedly resist Document Engineering efforts. They might view enterprise models with defined and measured processes as a threat to their current autonomy and operations. They might not know which applications or organizations would benefit from improvements in document processing or information architecture because they don’t have the system inventories, enterprise data dictionaries, or data warehouses that demonstrate a strategic focus on information technology. Their lack of processes for technology adoption might require remedial measures and hand-holding to deploy new models and applications.

In the Berkeley Event Calendar project, some aspects of the model and solution can be related to maturity considerations for the adopting departments.

The core model was kept small to impose minimal information requirements for describing an event. This meant that even organizations with informal processes and calendars that were not automated, would be able to comply with the event
The inventory-gathering activity is also shaped by the capability assessment. A mature organization can readily provide the appropriate information sources for analysis, but an immature organization may lack methods for responding to requests. In such organizations Document Engineers may need to use indirect means to capture requirements and business rules. And if interviews cannot be conducted without causing problems or provoking resistance, they must rely more on artifacts such as existing documents, forms, and program interfaces to determine information requirements.

When working in this kind of organization, we may need multiple stages of inventory gathering because the documents that are offered up first are sometimes the least useful. What makes them readily available is the fact that no one uses them. This is like being in a library where the most interesting books are always checked out. Richer artifacts take more careful analysis to discover.

### 16.2 BUSINESS OBJECTIVES

Every organization or enterprise, whether it is a commercial firm or a governmental, educational, and non-profit institution, has reasons for its existence. They will have different goals and carry out different activities to achieve them, but they have the common motivation of being successful enough at what they do to be able to continue doing it. So before they undertake Document Engineering, each must make a case
for it that identifies the business objectives and the likely return on investment for a project.

An enterprise with immature process capabilities will have a hard time justifying any project, Document Engineering or otherwise, because if its processes aren’t systematic and measured it can’t estimate the benefits of doing them differently. Nevertheless, the lack of systematic processes for initiating and managing projects causes some immature enterprises to suffer from the opposite problem of undertaking too many projects, some of which are redundant or lack clear business payoffs. In addition, if an enterprise can’t or won’t consider reducing its workforce because of legal, negotiated, or cultural constraints on staffing levels, it can’t easily capture the cost savings of automating manual processes and replacing employees with computers.

In contrast, an enterprise with mature process capabilities understands and controls its business processes, even if they aren’t fully automated. It measures its baseline costs and can determine what it needs to do to become more efficient or compete more effectively. It can also measure how close it is to where it wants to be and can modify its plans to deal with unforeseen requirements or events. What focuses all of these capabilities is the business case for the project.

16.2.1 MAKING A BUSINESS CASE

The business case for a tactical Document Engineering project like automating a document-intensive process, making such a process available as a web service, or integrating a document exchange with a business partner can be a straightforward and formulaic cost and benefit calculation:

• Compare the processing cost and time per document in the As-Is and To-Be applications.

• Estimate the value of other benefits that will emerge from the latter.

• Estimate the resources and time required to analyze, design, and implement the latter.
• Calculate when the new application will pay for itself by comparing the recurring benefits against the one-time costs.

• Decide whether the return is worth the investment.

The business case for a tactical project can be a formulaic cost and benefit calculation

This is a minimal approach to justifying the project, but it is pretty easy to do, and making a simplified business case is better than proceeding without one. Much of the time savings in document processing costs alone are sufficient to justify the project. But even in a small tactical project we think it is appropriate to attribute some value to the greater visibility and control that results, and to the improvements in organizational capabilities that will reduce development costs and improve productivity on subsequent projects.8

In contrast, even though they can yield substantial benefits, strategic projects are much harder to justify because they involve more complex and intangible factors on both the cost and benefit sides of the equation. For example, establishing a service oriented architecture as a unifying technical and business vision for an enterprise might:

• Involve many organizations and enable them to collaborate more effectively.

• Extend the lifetime and value of legacy systems and information sources.

• Enable the more rapid implementation and deployment of software functionality.

• Inspire new and more adaptable business models with both internal and external partners.

• Improve the usability or quality of the services it provides.

But precisely because of this broad impact, it is difficult to predict exactly which benefits will be the most important, when they will emerge, and which organizations will most effectively capture them. Many of them, like improvements in collaboration or business adaptability, are hard to quantify in monetary terms.
Strategic projects are much harder to justify because they involve more complex and intangible factors.

It is tempting to avoid these uncertainties by justifying a strategic initiative on a project-by-project basis, but this weakens the business case because it doesn’t account for the enterprise-scale investments in modeling, infrastructure, and organizational alignment whose payoffs emerge over time. Furthermore, a project-by-project focus necessarily distorts long-term goals to fit the shorter-term payoffs of each project.

For example, for a business to transform itself from a forecast-driven, make-to-inventory manufacturer to a demand-driven, make-to-order one, it must improve visibility and speed information flow throughout its supply chain, manufacturing, and inventory management processes. Each of these processes could be improved incrementally, but the new business model requires that all of them be improved with a coherent end-to-end perspective.

Similarly, it is possible to accumulate an enterprise information architecture from separate modeling efforts—but unless some of the modelers maintain an enterprise focus and iteratively evaluate and reconcile the modeling work done separately in projects, the whole will be less than the sum of the parts, with less consistency and reuse than desirable.

There is no perfect solution to the unavoidable tradeoffs between strategic and tactical projects. We recommend the approach advocated by Larry Downes in “The Strategy Machine.” Downes recommends creating a strategic project portfolio that includes projects with different time frames and risk profiles. This enables some tangible value to emerge earlier, which can protect the overall initiative from budget cuts or cancellation during economic downturns. Not every project in the portfolio will be successful, but focusing only on narrow tactical projects because they have fewer risks than strategic ones can be costly in lost business opportunities and missed productivity breakthroughs. We acknowledge, however, that effective management of the project portfolio requires mature capabilities for monitoring and measuring the impact of each project, decisively terminating those not likely to succeed, and reallocating resources to those that appear more promising.
16.2.2  A SAMPLE OF PROJECT JUSTIFICATIONS

In the following sections we discuss a variety of justifications for projects in the Document Engineering project portfolio. The list is illustrative, not exhaustive, and we begin with the most obvious reasons for projects—being able to do things cheaper, better, and faster.

16.2.2.1 Reduce the Processing Costs for Goods and Services

Potential cost savings resulting from automating the manual processing of paper documents drove the adoption of electronic data interchange in the 1970s and 1980s. In the late 1980s it was estimated that the cost of processing an average order or invoice document could be reduced by 75 percent if they were exchanged electronically, a saving of US$10 to $15 per document. The automobile industry interpreted this as a saving of $800 per car. However, even with these significant savings, the high costs of developing and operating suitable business interfaces with EDI limited its adoption to enterprises and supply chains with high transaction volumes.

A 2001 estimate of order-processing costs suggested the same 75 percent cost reduction through automation, but the savings were now $83 per document. Furthermore, the cost of document exchange efforts using the Internet and XML can be substantially lower than with traditional EDI. So while EDI remains important to companies with extensive legacy implementations, it is rarely the technology of choice for document exchange projects involving new business processes.

Since most enterprises spend between 50 and 80 percent of their total spending on external goods and services, the estimated 5 to 10 percent that can be saved by automating procurement can save tens of millions of dollars annually for large firms. For the U.S. government, whose US$305.5 billion in purchases of goods and services in 2003 made it the world’s largest buyer, even 5 percent cost reductions through the Integration Acquisition Environment initiative would save $15 billion.
In October 2004 Denmark mandated that all firms doing business with the government must send their invoices in the XML format of the Universal Business Language (see Section 4.3.2). For each of the more than 18 million invoices annually, a conservative estimate is that ten minutes of handling time can be saved for a cost savings of €94 million, (more than US$125 million at end of 2004 exchange rates). The Danish government is considering automating its procurement and reconciliation using the UBL purchase order, which would save another €66 million.17

As large as these cost savings are, Denmark is a relatively small country and these estimates assume the UBL-ification of only orders and invoices. The average international business transaction can involve up to 40 different types of documents. The preparation and handling of documentation to move goods across borders, and the delays caused by processing all those paper documents, adds an estimated 10 to 15 percent to the costs of the goods traded. If cross-border trading were made paperless, savings in trade between the countries in the Asia-Pacific Economic Cooperation region alone could be greater than US$60 billion annually.16

These huge estimates of potential cost savings through automating document exchanges are also emerging from banking, securities, insurance, health care, and other document-intensive industries. A 2003 study by Accenture estimated that a single bank with a 5 percent market share would derive between US$200 and $400 million in new revenue and save about $100 million in operating costs by automating all of its information exchanges and transactions that involve financial research.19

Huge estimates of potential cost savings are emerging from many document-intensive industries

**16.2.2.2 Improve Operational Visibility and Control**

Automated processes are more visible and measurable than non-automated ones, and they provide management with information about operations that can be used to further reduce costs and improve efficiency.
Many business problems with supply chains and distribution channels, including excess or insufficient inventory, demand variability, and high transportation costs primarily result from poor visibility and lack of collaboration. Technologies and business processes that speed information flow across the chain or that allow more information to be shared in controlled ways can substantially reduce these problems.

Many business problems result from poor visibility and lack of collaboration

For example, if a manufacturer and its supplier exchanged information about each other’s inventory, excess raw materials at the supplier might trigger a collaborating manufacturer to temporarily increase its own production. Likewise, impending shortages of critical components in the manufacturer’s inventory might cause the supplier to temporarily increase its own output to ensure that its customer could keep its production lines running.

Better information exchange for collaboration is also essential for satisfying social goals such as product traceability in safety recalls or for infection and contamination control. An exemplary initiative for the latter is the Notifiable Infectious Disease Information Messaging System (NIDIMS), developed in Hong Kong in response to the 2003 SARS outbreak. NIDIMS exchanges information about 28 infectious diseases between the Department of Health and various healthcare providers to increase surveillance with faster response time and fewer errors.

An important component of the benefit from automating document-intensive business processes is the elimination of errors when information flows from one document to another. As we’ve repeatedly shown, many business processes consist of a chain of related documents with overlapping content components; one estimate is that 75 percent of computer output becomes input to another system. Manually reentering this information is not just costly in time but also prone to error. For some clerical processes up to 30 percent of the effort may involve preparation and correction of information being passed through.

As much as 75 percent of computer output becomes input to another system
Information accuracy also affects cash flow. In 1986 the UK banking industry estimated that 50 percent of international Letters of Credit contain errors that require clerical intervention. The resulting delays resulted in the annual loss of US$100 million in interest on monies deposited. Automating the receiving and payment of invoices enables firms to reconcile orders and payments more quickly, allowing them to use early payment discounts and avoid late payment fees; the estimated savings has been put at 68 percent after five years.

In case anyone thinks these kinds of document exchange problems in supply chains or finance have no relationship to day-to-day experiences, consider the sobering report from the U.S. National Academy of Sciences Institute of Medicine called “To Err Is Human: Building a Safer Health System.” This report concludes that thousands of people die each year as a result of adverse drug effects and errors in medication, and 95 percent of the deaths would be avoided if doctors entered prescriptions using automated order entry systems. In one effort to improve the situation, WellPoint Health Networks, a leading health insurance company, is spending US$30 million to give 20 percent of the physicians it serves either a PC or a handheld computer so they can enter computer-readable prescriptions instead of scribbling them by hand.

**16.2.2.3 Accelerate Existing Processes or Enable New Ones**

Automated processes are also faster. A reduction in sourcing and procurement cycle time can lower inventories and enable firms to make more informed tradeoffs between maintaining inventory and reducing costs through bulk buying.

Faster access to information also enables better resource scheduling. When they are proactively notified of events relating to container and vessel movements, trucking companies can better schedule their vehicles and speed cargo deliveries. The wireless handheld package scanners used by FedEx drivers save ten seconds per package per stop, with estimated savings of at least US$20 million annually.

But while speeding up existing business processes is often an important goal, automation projects primarily provide an opportunity to rethink and redesign them. In fact, making existing processes faster may not be the right strategy. When we
introduced the Document Automation and Straight Through Processing patterns in Chapter 4, we cautioned against “paving the cow paths” and suggested that automation projects should consider implementing industry best practices and their enabling standards. So instead of merely accelerating invoice processing, why not notify the supplier when payment is authorized, initiate payment when the goods are scanned at the receiving dock, or adopt some other event-driven process that completely eliminates the need for the supplier to send an invoice document?

Conversely, accelerated cycle times can also enable entirely new business processes. Making it easier to get goods through customs with paperless trade administration has created new cross-border markets for smaller producers of perishable items like fruits, vegetables, and flowers. New online booking services for other kinds of perishable goods and services have been spawned by real-time inventory reporting for airline seats, hotel rooms, restaurant reservations, concert tickets, and so on. For example, if someone procrastinates in making restaurant reservations, they may still get a table at a posh restaurant that usually fills up weeks in advance by searching OpenTable.com.

Accelerated cycle times can create new business processes

We discussed several other new business patterns enabled by electronic information and document automation in Section 5.4, “New Business Models for Information Goods,” and Section 5.5, “From Forecast or Schedule-Driven to Demand or Event-Driven Models.”

**16.2.2.4 Make Publishing Processes Cheaper, Better, and Faster**

So far, we’ve emphasized benefits for the transactional end of the Document Type Spectrum, but we’d be remiss if we didn’t briefly mention that much of our discussion in the three previous sections about making processes cheaper, better, and faster also applies to narrative types of documents.

Lynda Brooks reviews three case studies of “applying a media-neutral publishing approach using XML technology” while reengineering traditional publishing processes and reports 25 to 40 percent operating cost savings with the additional
benefit of a shorter time to market. Instead of maintaining multiple versions of the same content and incurring redundant production costs, additional revenue results when multiple publications and formats are produced from the same base of structured content.  

Multiple publications and formats can be created from the same base of structured content

16.2.2.5 Reduce System Development, Maintenance, and Integration Costs

Just as the reuse of content is the primary basis for benefits in XML-based publishing, the reuse of type and class libraries and software frameworks is the basis for these benefits in contemporary software development practice. These two contexts converge when document and process models are encoded as XML schemas, which can be used just like programming language classes to guide the generation of software. The underlying economic justification is the same—amortizing the development and maintenance costs of the content, software artifact, or schema over multiple uses.

A new principle for reuse that we’ve introduced in Document Engineering is the emphasis on patterns and artifacts of a wider range of abstraction to include organizational and business process patterns as well as the more fine-grained patterns of documents and information components. We’ve also strongly emphasized the methods and artifacts needed to facilitate reuse during the analysis and design phases. While we advocate using models when we implement applications, we believe that the careful design of conceptual document and process models yields the biggest payoff. This is especially true in contexts involving information whose useful life is longer than that of the software that produces and consumes it.

Put another way, in software engineering fixing errors in designs is far more cost-effective than fixing them in implementations, and that rule also applies in Document Engineering. In a recent assessment of the benefits of a model-based architecture approach using UML and XML models, Martin Soukup reports “projects where the code generation saved person-years of effort, but the modeling errors found during the metamodel analysis phase saved tens of person-years.”
The careful development of conceptual document and process models yields the biggest payoff

Every software tool or application vendor makes claims about the productivity and quality benefits provided by their technology. Most of them emphasize how their products employ standards to counter their customers’ often-justified fear of being locked into a proprietary approach. But it is hard to differentiate the overlapping vendor categories of integration, collaboration, hub, portal, document management, middleware, enterprise infrastructure, and the new ones invented in each product marketing cycle. So we’re not going to repeat any of the specific percentages or return on investments in development, maintenance, and integration costs found in vendor case studies and white papers. The latest numbers can generally be found on the vendors’ websites.

Instead, we conclude this section with some caveats about the benefits attributed by vendors to the standards they support, which often differ depending on the business alliances they’ve made and the industries in which their customers predominate. Not every specification that is called a standard is equally likely to yield benefits.

Traction and sanction are two factors that steer a pattern or specification toward the status we recognize as a standard. Traction generates de facto standards, whose status is determined by adoption and popularity. Sanction creates de jure standards, where status is granted by a recognized authority. Internationally this means bodies such as the Internet Engineering Task Force (IETF), International Electrotechnical Commission (IEC), International Organization for Standards (ISO), or the International Telecommunication Union (ITU).33 However, there are many industry groups and regional bodies working at a more local level or outside these international bodies, and their credibility as standards-makers varies widely.

So any given pattern or specification, at any moment in time, has some degree of de facto and de jure standardization. Decisions by vendors or enterprises to adopt them are shaped by this mix. Vendors are usually more biased toward adopting de facto standards than de jure ones because of their customer focus, but governments, universities, and other institutions with longer time horizons are more biased toward de jure standards.34 Intellectual property terms also strongly affect standards adoption. Vendors and for-profit enterprises are more amenable to reasonable and nondiscrimi-
inatory (sometimes abbreviated to RAND) licensing terms than governments and open source advocates, for whom royalty-free terms can be essential prerequisites for adoption.\textsuperscript{35}

Ross Altman recently proposed a Standards Maturity Model analogous to the Capability Maturity Model that we discussed earlier in this chapter. In his model a Level 5 standard is functionally adequate, a product of a standards body, and ubiquitous in deployed platforms and applications. He cautions, however, that most standards never reach that level of maturity, and rates most web services standards as Level 3, that is, functionally adequate and published by a credible standards body but without much traction.\textsuperscript{36}

For most Document Engineering purposes the traction of adoption is more critical than sanction, especially for intraenterprise projects. Sanction is a means of encouraging traction, but a pattern, standard, or specification without industry adoption doesn’t offer many benefits.

\begin{quote}
A pattern, standard, or specification without industry adoption doesn’t offer many benefits
\end{quote}

\section*{16.2.2.6 Enhance Employee and Customer Satisfaction}

A final category of benefits that can emerge from Document Engineering projects involves the enhanced quality of the experiences for the employees or customers who interact with the systems or applications that implement new document or process models. The common theme for both employees and customers is the satisfaction that comes from doing higher-value activities instead of the routine or tedious ones that can be automated. It can be hard to put a direct monetary value on this benefit, but creative business cases might attribute reductions in employee turnover or absenteeism and higher customer retention to the increased satisfaction of using well-designed applications.

For example, with electronic documents, bank employees can spend less time checking errors on printed Letters of Credit and accounts payable clerks can spend less
time trying to reconcile orders and invoices. Truck drivers or patients can spend less
time waiting around. Authors can concentrate on creating and editing content and
rely on transformations to provide the various presentations and formats needed for
different devices or information products.

We’ve all experienced the satisfaction of being able to check on the status of our
order, payment, delivery, or other transaction with a self-service web application.
Being able to go to an Internet café, check a bank account balance, and transfer
money into it can save a vacation. But not all self-service applications increase cus-
tomer satisfaction, as we all know from our own painful encounters with imperson-
al, hard to use, or unreliable ones.

16.2.3

A SAMPLE OF PROJECT RISKS

We’ve discussed the most common justifications for projects, so we will now discuss
some of the common risks that can affect a project’s success.

The biggest risk in a Document Engineering project is attempting one that exceeds
the technology or process capabilities. This is why we advocate an explicit or implic-
it assessment before we start.

The following are some other risks that may have an impact.

16.2.3.1

The Comoditization of Business Relationships

Reducing the initial and recurring transaction costs of business relationships through
Document Engineering efforts is one of their most important justifications. But the
reduced costs are not always of mutual benefit. The flexible, plug-and-play vision of
service oriented architectures can enable a brutal Darwinism in business relationships.
Loosely coupled business relationships that are easy to create and inexpensive to main-
tain with little or no risk of proprietary lock-in are also easy to exploit or terminate.
The same technologies that can facilitate commitment and collaboration in a voice mode relationship can also enable an exit mode one. In the latter, a business can easily switch to alternate suppliers or outsourced service providers. From the perspective of the dominant partner, this is the benefit of transparent substitutability; from the perspective of the dominated one, it is a cost that shows reduced value for loyalty or continuity. The business relationship itself becomes a tradable commodity.

Business relationships can become a tradable commodity

For the dominant enterprise in an ecosystem to make the best use of its own capabilities, it almost has to exercise its market power to secure a larger portion of the value created by its business relationships. Companies have different reward structures for risk taking and innovation, which often amplify the conflicts of interest they always have with each other. For example, the standards to adopt when implementing a collaborative business process is a conflict typically resolved in favor of the most powerful party.

In a paper whose title cleverly asks, “When is Virtual Virtuous?” Chesbrough and Teece summarize the rule of business relationships as follows: “The most successful virtual companies sit at the center of networks that are far from egalitarian.” So from the perspective of the less powerful firm in asymmetric business relationships, the result of more efficient document exchanges is not always desirable.

16.2.3.2 Incomplete Automation and Zombies

In ambitious strategic initiatives that span one or more enterprises and that involve many processes and parties, some of the predicted benefits are likely to have an all-or-nothing character emerging only if all of the tactical projects that comprise the initiative are successful. And since large efforts must be carried out incrementally, automating one transaction or collaboration at a time, we must choose where to start.

It is typical to start where the largest incremental benefits can be obtained. In some firms, the process and document exchange project with the highest payoff might be part of order management or payment processing. For a firm acquiring another company, integrating the acquiree’s financial systems or product catalogs might add the most value.
In any case, a common risk is that one or more of the tactical projects to automate manual processes or replace expensive or brittle legacy automation technologies may not succeed. It isn’t necessarily a failure if a process can’t be completely automated; business processes that require expert analysis, tacit knowledge, and the interpretation of business policy can be made more efficient with document automation but a knowledge worker is still needed to perform the process. Nevertheless, a project that achieves partial automation is a failure if full automation is essential to satisfy project requirements—even if they were unreasonable.

Incomplete automation can leave the enterprise with a slow link in its information flow that nullifies most of the investments to improve other processes. The math is simple and brutal: for example, if automation eliminates 99 percent of the time taken to carry out nine of ten interconnected business processes, if the tenth manual task formerly took 10 percent of the total time, it now consumes 92 percent of it, and the end-to-end time is still 11 percent of what it was before automation.

A related bad outcome when a project doesn’t succeed in its automation goals is that both the original and the partly automated processes now run in parallel, with duplicate sets of costs and management overheads. The legacy and new systems live in a kind of half-dead state and are sometimes referred to as zombie systems.

**A Family of Zombie Projects**

What has been described as “possibly the most complex IT-based undertaking attempted in Australia” seems to be the latest in a family of zombie projects. The Export Integration (EXIT) system, developed in phases from the late 1980s to the late 1990s, was an EDI-based automation initiative to enable exporters, freight consolidators and forwarders, and airline and shipping companies to submit electronic documents to the Australian Customs Service (ACS). Unfortunately, only a minority of the intended users opted to use EXIT, and the ACS allowed the others to continue to submit paper forms.
The Australian government tried to kill the zombies by mandating electronic submission and by 2002 expected to replace the EDI-based EXIT with an XML-driven Integrated Cargo System (ICS) that used web forms. But ICS failed to meet its original goals and by late 2003 was a year late and more than AUD$100 million over its budget. Again the Australian government stepped in with legislation that would severely punish the IT vendors if ICS failed to meet a revised “go live” deadline in July 2004.

The Australian government amended the legislation and the “go live” date slipped again to October 2004 after a near-revolt by consultants and integration firms. They complained that the government mandate forced the ICS vendors to ship them code that was “not even worthy of alpha test status.”

In late 2004, ICS finally went live. Today the ACS proudly proclaims on its website that ICS is live and that EXIT is no longer available. But on the same page it lists “Communication options for unprepared clients” and acknowledges that “Export goods can be reported to Customs through an export agent, freight forwarder, Customs broker, bureau, or value added network.” So while Customs no longer has to handle manual processes, they’ve just been pushed onto others. The zombie (or son of the zombie) still lives.

Disaster stories for document exchange projects are unfortunately too common. We hope that Document Engineering will help make them rarer.

### 16.2.3.3 Unusable User Interfaces

Many business processes begin as user interactions with a printed document or web form, followed by automated processes that take place with little human involvement. These user interfaces are the document exchange equivalent of a telecommunication network’s “last mile.”

Throughout this book we’ve stressed the benefits of treating all kinds of document-model based interactions in the same way, emphasizing the commonalities between documents as interfaces for people and documents as interfaces to business processes.
It is essential for both user and process interfaces that they convey and capture the right information. But there is one difference that we can’t ignore. For user interfaces, usability also matters, adding an additional layer of requirements when we implement models in applications. As we pointed out in Chapter 15, user interfaces can sometimes be completely or partly generated from models. But we must often ensure that we provide an additional level of usability beyond what we can automate. Otherwise, as with incomplete automation, unusable user interfaces become a weak link in our information chain that undermines the benefits we created by automating the interfaces to other processes.

16.2.3.4 Unimplementable Models

There is an essential difference or gap between the real world being modeled and the conceptual domain of the model, or else the model would serve no purpose. Likewise, there is always a gap between a conceptual model and a physical implementation model, because a conceptual model is often most useful when it isn’t tied to specific or feasible implementations or technologies. But this means we can sometimes see what the current world looks like and what we would like it to be without being able to see how to get from one to the other. Our model might be unimplementable.

Some models can’t be implemented because of technology limitations. History is littered with designs like that of Leonardo Da Vinci’s “helical air screw,” which accurately embodied the principles of the helicopter in 1483 but couldn’t be tested until the early 1900s.41 A more recent example is the Sydney Opera House, whose award-winning sail-shaped concrete vaults couldn’t be built with the engineering technologies existing at the time its design was chosen. As a result, construction times tripled and its costs increased by a factor of 13.42

Some models can’t be implemented because of technology limitations
16.3 KEY POINTS IN CHAPTER SIXTEEN

- A capability assessment can predict the likely success of a project.

- Organizations with low process maturity can’t adopt new technology easily.

- Processes and documents are complementary and should be receive the same level of engineering rigor.

- Businesses with good process capabilities are more likely to implement model based applications.

- The business case for a tactical project can be a formulaic cost and benefit calculation.

- Strategic projects are much harder to justify because they involve more complex and intangible factors.

- Huge estimates of potential cost savings are emerging from many document-intensive industries.

- Many business problems result from poor visibility and lack of collaboration.

- As much as 75 percent of computer output becomes input to another system.

- Accelerated cycle times can create new business processes.

- Multiple publications and formats can be created from the same base of structured content.
• The careful development of conceptual document and process models yields the biggest payoff.

• A pattern or specification without industry adoption doesn’t offer many benefits.

• Business relationships can become a tradable commodity.

• Incomplete automation can leave the enterprise with a slow link in its information flow that nullifies most of the investments to improve other processes.

• Some models can’t be implemented because of technology limitations.