The Approach in a Nutshell

This chapter and beyond...

This chapter summarizes the overall methodology employed by successful process improvement and application development projects. This approach—a process for process improvement—is organized into four phases:

1. *Frame the process*—includes identifying a business process, clarifying its boundaries, performing an initial assessment, and establishing goals for the redesigned process.

2. *Understand the current (as-is) process*—includes modeling its workflow, and performing a more specific assessment.

3. *Design the new (“to-be”) process*—includes devising potential improvements, assessing them, selecting the main characteristics of the process, and finally designing the new workflow.

4. *Develop use case scenarios*—makes the transition into system requirements analysis by describing how process actors would interact with a system to complete tasks.
This chapter also introduces the various modeling techniques and frameworks that are employed by this approach. Process workflow modeling is the central technique, but the others are necessary to establish context for the process, to ensure that other critical factors in addition to workflow are addressed, and to define information systems requirements.

This is your guide to the rest of the book—a one-chapter summary of everything else we will cover, intended to clarify how the individual parts make up the whole. After some brief notes on the method and the intended audience, we will get into the specifics.

Why it works
This methodology works because it uses modeling techniques, frameworks, guidelines, phases, and steps that make it complete, repeatable, and learnable. Just as important, it is a practical response to where projects actually go wrong rather than a theoretical exploration of how things ought to be done. Many process improvement projects stumble badly or fail outright, and key lessons from these experiences have been incorporated in this approach, and verified in practice. The methodology is conceptually similar to many others, but it differs substantially in four important ways:

1. It includes frameworks and guidelines to ensure that real business processes are identified with clearly delineated boundaries—failure to do so is a common denominator of troubled projects.

2. It uses a process modeling technique that is simple to read and highlights the role played by individual participants—this increases participation, buy-in, and accuracy.

3. It includes methods for addressing all of the factors (enablers) that support a process, and the environmental factors that constrain it—otherwise, projects tend to zero in on the obvious enablers, workflow design, and information technology.

4. It integrates process-oriented and IT-oriented efforts by including techniques such as use case scenario analysis that support the transition from process improvement to requirements definition activities.

Whom it's for
We developed and organized this material specifically for people working on or responsible for a process improvement project, including:

1. Business analysts, consultants, and project leaders, whether internal to the organization or employed by a consulting firm;

2. Systems analysts defining IT requirements;

3. Business managers with a mandate to fix their processes;

4. Other professionals with an interest in the topic, especially those participating in executive M.B.A. or continuing education programs.

Many of the examples are drawn from medium to large enterprises, but there is nothing in the material that prevents using it in smaller organizations—some of our most successful projects with this approach have been in companies with fewer than 40 people. On the topic of examples, we have avoided including one big case study and instead have drawn upon a variety of illustrations. We did this because a single example cannot represent the range of issues encountered in practice, and tends to encourage a focus on the specific example rather than the general message. Besides, a life-size case study would be too large for the book, and our feedback was that the reader would quickly tire of a simplified example.

The book concentrates more on how-to and less on justification, case studies, or management issues. The book will not, however, cover everything that goes into a complete process improvement undertaking. Issues such as the design of organization structures, developing compensation schemes, and managing change are beyond our scope. Others have already done a fine job of covering subjects like those.1

This chapter, once again, is an overview of the rest of the book. It introduces various concepts, methods, frameworks, and tools, but at the core is the business process, so we will begin with a definition.

1. One book that does an excellent job of addressing some of these issues is Improving Performance: How to Manage the White Space on the Organizational Chart by Gary A. Rummler and Alan P. Brache. New York: Jossey-Bass Publishers, 1995. This is an essential title in the library of anyone interested in improving business processes.
Processes—results, not work

The term “business process,” or simply “process,” is used in many contexts, with varying meanings. Our definition is “a collection of interrelated work tasks, initiated in response to an event, achieving a specific result for the customer and other stakeholders of the process.” It’s awkward, but it includes the main elements in establishing the scope of a process—the event at one end, and at the other, the result and the customer expecting it. Do not be confused by the term “customer”—it doesn’t just refer to a customer purchasing goods or services. The customer of a process could be a government agency, a supplier, an employee, a trade association, or any internal or external person or organization expecting a result from a process.

The result is the most important part of the definition—without a result or output, what reason is there for a process to exist? And the result is not a vaguely defined service or condition—individual results must be specifically identifiable and countable. For instance, a client called to report problems with his company’s process improvement project. The team was having trouble controlling scope, and could not seem to get all of the activities into a single process model. The root of the problem was that the scope was the logistics process, which violates the rule about specific, countable results—it is impossible to identify specific occurrences of logistics or answer a question like “How many logistics did we do today?” because logistics isn’t a process, but a function that participates in multiple processes. Deliver Shipment, however, is a manageable process because you can identify each specific shipment delivered, count how many have happened, and identify the process’s customer.

Without a clearly defined process or subprocess as its scope, a project will suffer “scope creep,” or implement improvements that do more harm than good. A more precise definition of business process is in Chapter 4, and techniques for process identification are in Chapter 5. The essence, though, is a triggering event, a defined sequence of tasks and decisions involving multiple actors, and a countable result for the primary customer and each of the process’s other stakeholders.

“Event—tasks—result” is a framework for defining a process, one of several frameworks employed in this methodology.

Frameworks in general

A framework is a structure for discovering, organizing, and presenting ideas or information. They are as simple as the two-by-two matrices loved by consulting organizations for categorizing products, or as complex as a multidimensional structure for financial analysis. Ours are typically straightforward—a set of categories to be filled in or questions to be answered. For example, the simple framework we use to state clear and actionable objectives is the three Ts—topic, target, and time frame. Improve customer service is a fine overall goal, but vague, until the framework is used to break it into measurable objectives:

- Topic: telephone hold time for calls to the customer service line;
- Target: 60 seconds or less 98% of the time;
- Time frame: within 45 days.

Frameworks manage complexity by reducing vague or complex topics to a set of simpler questions. They organize work, maintain focus during interviews or facilitated sessions, ensure coverage of all aspects of a topic, and provide a standard format to document the findings. One of the most important frameworks is the one we use to consider all dimensions of a process.

A framework for process enablers

We defined a process as a set of tasks, but there is more to it than that, specifically the important concept of enablers. An enabler is a factor that helps a process to achieve its intended results and meet performance targets within the applicable constraints. The two that we are most concerned with are workflow design and IT support, but the framework employed here includes a total of six, as illustrated in Figure 3.1:

1. Workflow design;
2. Information technology;
3. Motivation and measurement;
4. Human resources;
Motivation and measurement

Motivation and measurement encompass the explicit and implicit reward systems of the organization. Their concern is how people, organizations, and processes are measured and assessed, and the associated consequences—reward and punishment. Experience shows that people do what they are measured on, and if the measures do not align with the goals of a redesigned process, failure is virtually certain.

Human resources

The human resource enabler covers the knowledge, skills, and experience of the workforce, training, organizational structure, job definitions, and so on. A process requires the right people in the right job with the right skills.

Policies and rules

This includes the rules and policies established by the enterprise to guide or constrain business processes, as well as applicable laws and regulations. Clearly, a policy such as “any requisition over $1,500 requires approval from a vice president” has a substantial impact. In practice, many processes include work to enforce rules or regulations that are obsolete, contradictory, or overly complex.

Facilities

Facilities are the workplace design and physical infrastructure such as equipment, furniture, lighting, air quality, and ambient noise. There is a growing trend to recognize the importance of facilities as enablers to effectiveness, productivity, and well-being. This may not be of concern in your process, and might be replaced by other, more relevant, enablers.

All too commonly, we see new processes with a well-designed workflow and supporting information systems fail because other enablers, especially motivation and measurement and human resources, were not adequately considered.

A framework for putting processes and systems in context

Information systems are of particular interest in this book, and we use a framework for putting the analysis of business processes in context with analysis of information systems requirements. We will build this
framework up progressively, beginning by recognizing that neither processes nor systems exist in isolation—their sole purpose is to support the aims of the enterprise. Once the enterprise has clarified its mission, strategy, goals, and objectives, business processes that support them can be developed. In turn, information systems can be developed or acquired which support the aims of the business processes. Thus, we arrive at the following three-layer framework for organizing analysis and design activities:

Mission, strategy, and goals
Business process
Information system

Conversations about the top layer are often ambiguous because the most commonly used terms—mission, strategy, goals, and objectives—take on multiple definitions. We attach a specific meaning to each:

**Mission:** The mission articulates the essential nature of the business—it states the products or services provided, and the markets or customers served. Generally, there is also an indication of the “style” of the enterprise—how it wishes to be perceived by stakeholders such as customers, employees, investors, and the community.

**Strategy:** This term is used to describe everything from a specific action plan through to the competitive space the enterprise wishes to occupy. We use a definition closer to the latter, in which strategy describes how an enterprise differentiates its products and services (e.g., low price, customization, and so forth), and therefore, why a customer would choose it over alternatives. As we will see, this concept is just as applicable in government agencies or internal services where there doesn’t appear to be a choice.

**Goals and objectives:** The overall improvement directions established by an enterprise are its goals, backed up by measurable performance targets, which are its objectives.

Individual processes have their own mission strategy and goals, which should align with their enterprise counterparts, and which provide important guidance when making design decisions. Other factors, such as culture and core competencies, are also important. Collectively, we refer to these as the environment within which a process operates. Chapter 7 expands on the topic of a process's environment.

In practice, we use a five-layer framework, which we arrive at by expanding on the information system layer. Information systems are not monolithic, but are themselves made up of layers (tiers) of interacting components. In the three-tier information systems architecture, these layers are named according to the service they provide—presentation services, application logic services, and data management services:

**Presentation**
**Application logic**
**Data management**

This is probably the most widely used framework in the world of systems analysis and design. If we take this three-tier framework, and use it in place of the simpler information systems layer in the preceding framework, we end up with this five-tier framework:

Mission, strategy, and goals
Business process
Presentation
Application logic
Data management

This five-tier approach is at the heart of the methodology we employ for studying business processes and information systems requirements. Each tier in the three-tier systems architecture bears further explanation.

**A closer look at the three-tier architecture**

**Presentation**
A system (or any automated device) requires mechanisms to capture data and instructions from a person (e.g., for an ATM, “Withdraw $200 from the checking account for this customer/password combination”) and a way
to convey information back to that person (e.g., "Sorry, insufficient funds in account."). This is the service provided by the presentation layer, also known as the user interface, or UI. It includes any mechanisms by which people (or other systems) interact with an automated system. In our context, the user interface is usually a graphical user interface (GUI) running on a desktop, like the familiar Windows, Macintosh, or Web browser interfaces, but it could be just about anything—a character-based computer terminal, an interactive voice response (IVR) system, a kiosk's touchscreen, a bar code scanner, a badge reader, a hand geometry scanner, or some new device technology just over the horizon. We can also consider it to include printed output, such as receipts or reports. The presentation layer for an ATM includes a variety of mechanisms—a display screen, command buttons, a numeric keypad, a card reader, an envelope taker, a cash dispenser, a receipt printer, and a security camera.

**Application logic**

Everyone seems to have a different name for this layer—process logic, application, business logic, business rules, and transactions, among others. Whatever the name, the concept is the same—programmed transactions containing logic to enforce the rules of the business and properly update files and databases.

In the withdraw cash example, application logic receives a message from the presentation layer and then checks that the PIN is valid for the account number, verifies that the account has adequate funds, determines applicable service charges, and so on. It also coordinates updates to the data management layer, updating the account balance and logging details of the transaction. Depending on the technology, the programmer might implement the logic in programs called stored procedures, methods, components, or modules. As analysts, our concern is only with defining the business rules and data updates.

A major strength of the three-tier architecture is that the same transaction (programmed application logic) may be invoked by different presentation mechanisms. For instance, the same transfer cash functionality may be accessible via an ATM, via a terminal at a teller's workstation, over the telephone via an IVR system, or over the Internet via a Web browser. This gives us the flexibility to add or modify user interfaces without reprogramming transactions, or to change the rules of a transaction without having to change all of the user interfaces.

**Data management**

The major strength of most computers in a business setting is not in the computing they perform, but in their ability to store and retrieve data. The data management layer provides this service, maintaining the records of people, places, things, events, and so on that are necessary for an enterprise to operate. Commonly, a relational database management system (DBMS) is used, but any of a number of database or file system technologies could be employed. Since business computers are primarily data processors, and since processes act on objects that are represented as data, data is considered throughout our analysis.

Each of these three tiers, as well as business processes, is analyzed and documented using specific modeling techniques. Before introducing these, we offer a few observations about models in general.

**Models in general**

A model is an abstraction or representation of some subject matter, and is one of two types. An aircraft mockup used for wind tunnel testing, like an architect's model of a building, is an iconic model—it resembles the physical object it represents, except that it is smaller and simplified. A mathematical model of the economy is a symbolic model—a representation of some physical (i.e., the weather) or conceptual (i.e., a budget or a project plan) subject matter that isn't intended to look like that subject. Symbolic models often represent concepts that can't actually be observed. Our workflow models are made up of boxes, lines, and text that bear no resemblance to the people, documents, and in-boxes that they represent.

Whatever the type, useful models generally meet the following criteria:

- They highlight facets of interest while masking unnecessary detail, and employ conventions for adding progressive amounts of detail.
- It is more convenient, less expensive, and safer to manipulate a model than it is to manipulate the corresponding real-world objects. In our case, a process workflow model supports understanding and assessing a process design without actually implementing and then observing it.
Modeling is not an end in itself, and people frequently need to be reminded to stop modeling when the purpose has been achieved. An as-is workflow process model is used to understand why the current process behaves the way it does—modeling should stop as soon as that understanding has been reached. On the other hand, a model of the to-be process is used to guide implementation, and will continue to be maintained through implementation and revision, and therefore requires much more detail and precision.

The modeling techniques we employ

As noted, there are associated modeling techniques for each of the lower four tiers of the five-tier framework.

Business process: process workflow models

Process workflow models (or workflow process models) as illustrated in Figure 3.2 are known by many names, but because of their appearance are most commonly referred to as “swimlane diagrams.” A swimming pool might be divided lengthwise into swimlanes for racing or swimming laps. Just as each swimmer is expected to stay in his or her own lane, each “actor” with a “role” in the process has his or her own swimlane. A box represents a task or step in the process, and is placed in the swimlane of the responsible actor. Arrows connecting the boxes indicate the sequence and flow of the steps. This type of diagram can be used to show both current (as-is) and proposed (to-be) process workflow, from a simplified overview down to very detailed steps. This particular form of workflow process model has become the de facto standard for depicting business processes because of its merits:

- They are self-explanatory. While other techniques may require considerable training before they can be read, swimlane diagrams are immediately understandable by almost everyone. (That’s not to say they’re easy to build, but that’s what this book is for.)
- They show individual actors, and emphasize their tasks and interactions with other actors. Participants can identify with the model, so it’s easier to produce a complete and accurate representation.

Several chapters are devoted to process workflow models, beginning with Chapter 8.

Presentation: use case scenarios

Eventually, while we are designing the to-be process, we begin considering how information systems will help an actor complete a task. Swimlane diagrams are not appropriate for this kind of detail, so we switch gears and employ “use case scenarios,” a variant of the popular “use case” technique. Essentially, a use case scenario models a dialogue—the back and forth—between an actor and a system for a particular scenario. The scenario is like a single test case, with defined actors, conditions, and data values.

Why is this necessary? Well, think of all the times you’ve used a system and wondered, “What in the world were they thinking?”—Web sites with torturous navigation, interactive voice response systems that never get where you want to go, and ATMs that take you through 15 steps before concluding that you made a mistake in the second step and must start over. By depicting the interaction between actor and machine before the system is designed and developed, use case scenarios help us avoid this sort of problem.

---

2. This modeling style was popularized by Rummler and Brache in *Improving Performance*. 

---

Figure 3.2 A workflow process model or “swimlane diagram.”
Use case scenarios are introduced in Chapter 16, along with information on how they are used as a springboard into more detailed analysis.

**Application logic: various techniques**

This is the most complex of the three layers to analyze, requiring a number of techniques including event identification, state transition modeling, and transaction specification. The key points are covered in Chapter 16, in the context of case analysis.

**Data management: data models**

The heart of a data model is an entity-relationship diagram (ERD) made up of three types of components:

1. The distinct things ("entities") about which information is needed (e.g., Order, Customer, Facility, Product, Part, Supplier, etc.);
2. The associations ("relationships") among those entities (e.g., "Part is used in Product," "Product is requested on Order," and "Order is placed by Customer");
3. The facts ("attributes") about each entity that must be recorded (e.g., the attributes of Part include Part Number, Description, Unit Weight, and Unit Price).

A definition is produced for each entity, to ensure consistent interpretation by all project team members. The first time you go through this step, you'll be amazed at the different interpretations for common terms like "product" or "customer." This makes data models a powerful tool early in a process-oriented project—they improve communication and consistency, because the entity names and definitions provide a standard vocabulary for the things the process deals with. Later in the project, data models are essential for database design and for describing data-oriented system requirements such as user interface behavior or program logic.

Examples can be seen in Chapter 15, which is an overview of data modeling.

**Five-tier thinking**

To summarize, this framework encapsulates the essence of our approach—it organizes the analytic techniques we employ on all process improvement and application development projects. It lets us look at a business, including the processes and systems that support it, as a whole. As we have noted before, processes and information systems are inseparable, and both exist to support the aims of the enterprise.

We have found this five-tier approach very useful for explaining the analysis steps that a project will undertake. Often, during a project kickoff, we will present a variation on Figure 3.3, and review each layer while making the following points:

An enterprise has a mission, strategies, and goals

...which are supported by business processes. Actors in a business process are in turn supported by information systems that they interact with via

![Figure 3.3 A five-tier framework for business analysis.](image-url)
...the presentation layer, which communicates commands to the
...application logic layer, which enforces business rules and coor-
nidates updates to the
...stored records in the data management layer.

Participants, from senior executives to technical specialists, consistently express support for this framework because it shows the interrelationships between their respective areas of interest.

A workflow-driven methodology—the process of studying processes

Finally, let's review the methodology we will explore in the remainder of the book. We describe it as a starting point, because there are unique aspects to every project—the world of business processes is filled with politics, sensitivities, conflicting goals, old habits, seemingly unshakable paradigms, resource constraints, and a host of other pitfalls, so you'll probably have to tailor the approach to your environment.

As outlined earlier, the approach proceeds through four phases:

1. Frame the process;
2. Understand the current (as-is) process;
3. Design the new (to-be) process;
4. Develop use case scenarios.

This approach is summarized in Figure 3.4.

Frame the process
This is arguably the most important phase, because, while quick, it prevents many common problems later. When we are called in to assist with a project that is in trouble, we invariably have to go right back to the beginning and work through framing the process. This includes:

* Identify a set of related processes, including the target process to be improved, by developing an overall process map. This clarifies what's in and what's out of scope by showing the process to be studied, plus touch points to other processes.
Establish the scope of the target process, using a framework for identifying a process and its boundaries:

Process name, initiating event, customer and result, other stakeholders and the result they expect, approximately five to seven milestones within the process, actors, timing, and frequency.

- Review or document enterprise mission, strategy, and goals.

- Perform an initial process assessment:
  Stakeholder-based: customer, performer, owner/manager;
  If necessary, consider other stakeholders: supplier, regulator, and so forth;
  Summarize in the process case for action. 3

- Determine process vision and performance objectives:
  Describe how different stakeholders will perceive the new process, and specify the improvement dimension;
  Summarize in the process vision.

- Develop glossary of terms and definitions (the start of the data model);

- Summarize—construct and distribute a poster;

- Optionally, begin documenting noteworthy observations of culture, core competencies, and management systems.

Understand the as-is process

Now that the goals for the process are clarified, you will need to understand why those goals are not being met—many so-called improvements are implemented that don't actually improve anything because other factors are the root cause. The key here is not to exhaustively document the current process, but to understand it.

- Map the current process workflow to show who does what, when:
  Develop swimlane diagrams;

Use progressive levels of detail (we will define them later), stopping when process behavior is understood—don't get bogged down in detail!

- Investigate all other enablers (current use of IT, motivation and measurement, and so forth).

- Record initial thoughts on strengths and weaknesses of the current process, especially leverage points where significant improvement is possible.

- Perform an enabler-based final assessment of the as-is process:
  Enablers: workflow, IT, motivation and measurement, policies and rules, HR, facilities (and other).
  At this point, you will also start to collect ideas for the to-be process, so in a way, you have a foot in both camps—the as-is and the to-be.

- Optionally, assess each individual step (e.g., is it necessary, is it done by the right performer, and so forth).

- Document important aspects of culture, core competencies, and management systems.

Design the to-be process

We divide this part into two stages. The first, characterize the to-be process, determines a set of improvements or design characteristics that will work in concert to achieve process goals. We do this stage specifically because teams often leap into the design of a new workflow that incorporates "improvements" that actually work at cross-purposes or are inconsistent with other enablers. Once the team is satisfied that it has identified a cohesive, effective set of characteristics for the new process, the second stage begins—design the to-be workflow.

A) Characterize the To-Be Process

- Decide on direction—abandon, stay as-is, improve, redesign, or outsource?

- Develop ideas for characteristics or features of the new process. Techniques include:
  Identify improvements that would address leverage points;
  Challenge assumptions underlying process steps;

3. This phrase was introduced in Hammer and Champy's book.
Brainstorm by enabler (this may, in effect, have been done during assessment).

- Assess promising ideas in context (with respect to other enablers) using a matrix format (e.g., a change in workflow may require a change in job definition that may in turn require changes to recruiting and training, as well as changes in compensation and performance measurement).

- Based on the assessment, select the key features of the to-be process.
- Develop/revise conceptual data model.

B) Design the To-Be Process Workflow
This is actually quite straightforward if all the previous steps have been worked through.

- Draw the to-be workflow:
  Progress through the different levels of detail;
  Assess and check viability at each level;
  Revise, or move on to the next level of detail;
  Iterate.

Analyze use cases
With use cases, we make the transition to looking at how systems can support the process.

- Identify use case scenarios:
  Review the to-be process workflow to identify individual use cases.
  Identify a set of process scenarios (usually around 5 to 10) that will exercise the main paths through the new workflow.
  For each process scenario, identify the individual use case scenarios (typically 7 to 15), and document the preconditions, major decisions, and outcomes for each.
- Develop individual use case scenarios:
  First pass—focus on the dialogue;
  Second pass—add data, transactions, and interface objects.

From use case analysis, we can proceed to other activities such as detailed data modeling, business rule specification, and user interface design. A detailed treatment of these is beyond the scope of this book.

Applications of the approach
Our approach could be characterized as process-focused and workflow-driven, but should you always use a process-focused, workflow-driven approach? Here's a short answer—a tautology, in fact. If work flows at all, you should consider using workflow, and if it flows far, you must use workflow. Let's clarify that.

Anytime the application or business improvement project under consideration involves multiple actors (e.g., job titles or organizational units), and the handling of an event flows between them, workflow should be considered. It may not be essential until at least four or five actors are involved, but you can always give it a try. If you are able to develop a good workflow process model easily, without any significant discoveries or major issues arising, it probably was not necessary, but you won't have wasted much time.

In other cases, the problem at hand is entirely transactional—an event is handled almost completely by one or two actors in a single transaction. These types of projects are becoming less common, but in those cases a workflow-driven approach may not make sense. An example is an application to automate the basic transactions of a video store—place a reservation, book a rental, and so forth. In this case, there is not a multistep, multi-actor process, so we'd define the scope and objectives and then jump right into data modeling and use case analysis.

Finally, there are other cases that we have not alluded to where elements of this approach have proven to be extremely useful. So far, we have described a method in which a new process is designed, and then a supporting information system is developed. Often, this is not the sequence of events. Instead, a major information system is acquired, such as:

- An enterprise resource planning (ERP) application (e.g., from SAP, PeopleSoft, or Oracle);
- A customer relationship management (CRM) application (e.g., from Siebel, Clarify, or PeopleSoft via their Vantive acquisition);
A supply chain management (SCM) application (e.g., from i2 or Manugistics).

Recent experience shows that when these implementations don’t take a business process orientation, the result is often disastrous. One approach is to develop process workflow models for your desired (to-be) business processes to use in selecting and configuring a purchased application. Alternatively, many commercial applications are designed around an ideal (best practices) business process. If the necessary expertise (usually from the vendor or a consulting firm) is available, a model of the process the application supports can be developed. This will clarify whether a particular offering supports your goals and objectives, and assist you in identifying the changes necessary to implement it.

We have also used these techniques when integrating enterprise applications with preexisting legacy and custom applications. A new field called enterprise application integration (EAI) provides methods and technology for doing just that. The application of EAI is much more effective when coupled with process-oriented approaches. Some of the available EAI tools focus specifically on expediting business processes, which of course requires that the processes be identified and mapped. This is such a critical aspect of EAI that we feel the field should have been named “enterprise process integration.” A particularly important use of EAI is in e-commerce applications, where it is used to coordinate existing applications in support of a Web-based front end. As of this writing, two good sources for EAI information are www.eaijournal.com and www.eaiquadrant.com.

One of the hallmarks of a powerful tool, technique, or approach is that it proves itself useful in a wide variety of situations, some of them quite unexpected by the originators. The process-oriented approach and the techniques that come with it fall into this category. Armed with this overview, let’s dive into the details and see how it can work for you.
Just What Are Processes, Anyway?

The trouble with "process"

We've said it before, and we'll say it again—identifying the wrong collection of activities as a process is the most common, and arguably most serious, problem we encounter. A business process is not just a random collection of activities—it meets certain precise criteria. Time and again on consulting assignments, when a development or process redesign project is in serious trouble, we find that a true business process was not correctly identified or its scope was not clearly defined. Jumping into a process without putting it into context with other business processes or confirming that you have, indeed, initiated work on a true business process is sure to lead to trouble. Some of the problems will show up fairly early as difficulties in constructing a workflow model. Others will arise later, perhaps not until your new process is implemented and performance turns out to be worse instead of better.

Consider three examples:

1. Senior management specifically instructed a project team to develop cross-functional business processes during the implementation of a large, purchased information system. With the
assistance of an international consulting firm, the team determined what their processes were, and began mapping and assessing them. After a month’s work, the team sensed that all was not well, and we were called in to provide some assistance. We had the team use our guidelines and methods to identify a set of processes. This made it very evident that the processes the team had already identified were anything but cross-functional. In fact, the processes had *exactly the same boundaries as the existing organization structures*. We advised the team that they were not going to achieve the benefits that their executives were expecting unless they proceeded with the set of cross-functional processes they had just identified. The project managers, however, opted to continue with the original processes because of the work that had already been done. A year later, the problems from this decision were apparent to everyone, and the team had to go back and redo most of the work of the past year using the “real” cross-functional processes. Moral: It’s the Process, stupid.

2. A project team attempted to model the workflow for a major area they had improperly described as a single process—Supply Chain Management. Eventually, the effort collapsed amidst finger pointing, frustration, and missed deadlines. Their process model became too confused to follow (or complete!). The reason was clear. Because their scope actually included some five processes, it was impossible to express in a single diagram. There was no clear beginning point—there were many—and there was no clear ending point—there were many. It was impossible to trace a path (a workflow) through all of the included activities, especially because of timing issues. Some tasks were part of transaction-oriented processes that happened hundreds of times per day, others were part of ad hoc processes that occurred several times a month, and others still were monthly or quarterly. Try getting all that on one diagram, and then improving it! Moral: Too many Processes spoil the broth.

3. One organization enthusiastically embraced process improvement, with good reason: customers, suppliers, and employees found the company’s processes slow, inconsistent, and error-prone. Unfortunately, they were so enthusiastic that *everyone* defined his or her own work, or the work of his or her small department, as a process. Of course, each of these was in fact a function within a small number of overall processes. You can guess what happened. Within the boundaries of each of these processes, work was efficient *from the perspective of the performer*. However, these mini-processes were efficient largely because they had front-end requirements that made work easier for the performer, but imposed a burden on the customer. The attendant delay and effort meant that the real processes behaved even more poorly than they had before. Moral: One department’s improvement is another department’s bottleneck.

Our goal is to be sure your project gets off to a good start and avoids problems like these by defining a project scope that is equal to a complete business process—as we will define the term. This maximizes the potential benefits, and is easier to work with. If all projects are defined in this manner, then overlap, underlap (gaps), and duplication will be apparent and avoided.

On the other hand, your intent may not be to work with a complete business process—your scope may be some other set of tasks, like the work done by a particular job function, or perhaps it is just a stage within a process because you lack the sponsorship or resources to take on the whole thing. That’s fine, and is in fact the most common situation, but understanding what constitutes a process is still important. Otherwise, you might optimize a part of a business process only to cause a deterioration of the whole. There are ways to avoid this, but only if you recognize the situation in advance.

In this chapter, we will give some very specific guidelines on what constitutes a business process. In the next chapter, we will put those guidelines to work with specific techniques for discovering business processes and clarifying their scope.

**The trouble with terminology**

The early part of your project, and this book, is concerned with defining business processes properly: not too small (by task), not too big (an area),
not by specialty (a function), and not by organizational structure (such as a division or department). Along the way, we'll use some common English words, such as "process," "function," "task," and "activity," that have multiple and overlapping meanings. These terms have taken on a specific interpretation in the context of process and workflow analysis, and business in general, so we must carefully distinguish among them so as not to confound the concepts.

English words—dictionary definitions

Except where noted, definitions are from [1].

A process is a particular course of action intended to achieve a result (synonym: procedure) [2]; a series of actions or operations conducting to an end; especially: a continuous operation or treatment especially in manufacture.

A function is a professional or official position: Occupation.

An activity is a natural or normal function: as a process (as digestion) that an organism carries on or participates in by virtue of being alive; a similar process actually or potentially involving mental function; an organizational unit for performing a specific function; also: its function or duties.

A task is a usually an assigned piece of work often to be finished within a certain time; a duty or function.

So a task can be a function, and an activity can be a process or a function. To further complicate things, the word "procedure" shows up too. It usually means a defined series of steps and decisions to accomplish some task or activity. Yikes! About the only thing we can conclude is that some work (or is that activity?) is involved!

But if the English terms are confusing, information systems jargon is downright contradictory.

Information systems terms

In the later part of our project, we will become increasingly concerned with information systems, as automated implementation considerations come to the fore. Then business terms can become confused with computer jargon, which uses some of the same terms in a different way. At that point, we will really need to be precise in the meanings of these terms. Let's look at some computerese.

A process is an executing program. A process consists of the program code (which may be shared with other processes which are executing the same program), and some private data [3].

A function is a computer subroutine; specifically; one that performs a calculation with variables provided by a program and supplies the program with a single result [1]; or a set sequence of steps, part of larger computer program: subprogram, procedure [2].

So processes are programs, which may incorporate functions. But the terminology for the information engineering methodology introduces another scheme: functions are composed of processes, and an activity is either a function or a process [4]. In another scheme, functions are composed of processes which are composed of activities which are composed of tasks. In yet another, activities are composed of processes. A process can even be "a generic term that may include compute, assemble, compile, interpret, generate, etc." [5]

Similar conflicting definitions can be given for the other terms: tasks can compose, or be composed of, processes, and if we cared to bring in procedure, operation, or, worse yet, system, we could easily construct a huge circular definition that would require a computer to process—and you can be sure we'd become confused in the process.

So what is a process?

Business process terms

You keep hearing "make sure your business processes aren't functions," which, using normal English or computer definitions, doesn't make sense. But we will precisely define these two terms such that it does make sense, and provides guidance on how best to scope your project. The definition of business process is meant to be prescriptive, not descriptive: you should strive to make your project focus on a business process as we will define it. Addressing a complete business process offers the potential for greater benefits, and is easier to analyze using tools like workflow modeling. And even if you can't take on a complete business process, recognizing the actual process boundaries can save a lot of grief, so let's try to precisely define our terms.

Essentially, a process is a way for an enterprise to organize work and resources (people, equipment, information, and so forth) to accomplish its
aims. Historically, an enterprise would organize its work and resources into specialties or functions. Nowadays organizations try to organize work and resources to achieve a specific output—a result—for a specific customer. This is process orientation, and gives rise to this definition:

“A business process is a collection of interrelated work tasks, initiated in response to an event, that achieves a specific result for the customer of the process.” We will look at each phrase in the definition, but not in the order they appear. Surprisingly, even though the first reference is to work tasks, they are the least important aspect of a process. Far more important are the result, the customer, and the event. Let’s take Stephen Covey’s sage advice, and “begin with the end in mind.”

...that achieves a specific result...

The only reason a business process exists is to deliver a specific result. That result might be goods, such as the products requested on an order, or services, such as information in response to a query. The all-important guideline is: The result must be individually identifiable and countable. The processes Develop New Product, Resolve Service Problem, Fulfill Order, and Hire Employee all conform to this guideline. You can identify the specific new products that are developed, and count them. In other words, it is possible to count how many times the process Develop New Product was completed. Similarly, it would be possible to identify and count the service problems that were resolved, the orders that were fulfilled, and the employees that were hired. However, you cannot count how many Research and Developments, Help Desks, Telemarketings, or Human Resources were completed because those are departments or functions, but not processes. A good process name clearly indicates the result or end state of the process—new product is developed, service problem is resolved, and so on.

...for the customer of the process...

This is critical: A customer is the recipient or beneficiary of the result produced by the business process. This customer may be a person, an organization, or even a broad marketplace, but the customer can be identified and can pass judgment on how satisfactory the result and the process are. The customer might be internal to the organization, such as the employee whose service problem was resolved, or the department that receives the newly hired employee.

...initiated in response to a specific event...

You must be able to trace a process back to the earliest event that triggers or initiates it. Think of the process as a machine that is inactive until the on switch is flipped. In other words, the event is what makes it go. Effectively, the event is a specific request for the result the process produces. Develop New Product begins in response to the event “market opportunity is confirmed,” which is a request for a new product that will satisfy the market opportunity. Resolve Service Problem begins in response to the event “customer reports service problem,” which is a request for the service problem to be resolved. Identifying the earliest event is not always easy. Does Fulfill Customer Order begin in response to the organization’s receipt of an order, or customers initiating an order, or customers realizing they have a need? Sometimes, there are multiple events that can initiate a process. In any case, the effort in determining the event(s) is worth it—once you have an event and a result, it is far easier to trace the sequence of tasks that transforms the former into the latter.

...work tasks...

The business process is a collection of actions, activities, steps, tasks, or whatever you want to call them. Generally, we will use the term “step,” but they are all identifiable pieces of work, typically done at a point in time (in one sitting) by a single actor, or multiple cooperating actors. An actor may be a person playing a specific role, a job function, a department, or even an
automated system. A step in the initial workflow model will likely divide into more and finer steps during development of more detailed models. Just think of a work task as any activity an actor needs to accomplish in the course of a process. A more precise definition is not necessary at this point.

...a collection of interrelated...

The steps in the process must interrelate—they are not just an arbitrary collection of work. For instance, we do not want to end up analyzing Joe’s job or the human resources (HR) department. Joe does a variety of tasks, from taking orders to handling customers’ problems. The HR department does a variety of things from hiring and firing to benefits and reimbursement of training costs. In both cases, the only relationship among the steps is that the same performer does them all. That may be interesting, but it is not a process. Joe and the HR department probably participate in many processes. In a process, the steps are interrelated through sequence and flow—in simple terms, the completion of one step leads to (flows into) the initiation of the next step. Just as important, the steps are interrelated by dealing with the same work item, such as a specific service order, or benefit enrollment, or whatever the process deals with. Further, all of the steps are interrelated by being traceable back to the same initiating event. For example, when Joe finishes taking one customer’s order, he may return to resolving another customer’s service problem, but in process terms the two are unrelated—they deal with completely different work items and are part of the response to completely different events.

...a business process is...

Throughout this book, the terms “process” and “business process” refer to this definition, which is consistent with the basic principles that made BPX such a phenomenon—a process delivers a result to a customer.

So what?

It might be occurring to you that this definition is self-evident. That occurs to us on a regular basis, too. Most companies already know they need to focus on their business processes, and they know that these processes transcend specialties and organizational structures. But it is surprising that with all the exposure process orientation has received, many, if not most, businesses still cannot state what their major processes are, much less describe how they should, or even do, operate. How can this be? The answer, it turns out, is that the concepts of organizational structure, specialization, and, more than anything else, function, are so embedded in the organizational psyche that it is amazingly difficult to break out of functional orientation.

Business process versus function

Perhaps the most common error in defining business processes is to mistake a function for a business process. In this discussion, we are defining function as an occupation or department that concentrates skills. A function is a kind of work, or a field, which typically involves similar skills and tools, and has its own language. Customer service, research, engineering, manufacturing, logistics, marketing, sales, human resources, finance, shipping, billing, collections, and accounting are all functions, not business processes.

Since the Age of the Factory, organizations have been traditionally organized around functions. Education, job titles, and careers are all based on function, so functions take on such an importance that it is easy to mistake a function for a business process. “Hey, processes are important, and my function sure is important, so I guess my function must be a process.” Well, functions are important—in fact, we will go against the prevailing wisdom and say that being process-oriented doesn’t mean dismantling your functions. But differentiating your functions and your processes is paramount. Let’s look at the differences.

In the case of an order for a customized widget, multiple functions are involved. You must take an order (Direct Sales), orchestrate the flow of materials (Logistics), make the widget (Manufacturing), pack and ship the widget (Shipping), invoice it (Billing), collect the money (Collections), reflect the transactions on the company’s books (Accounting), and deal with various issues along the way (Customer Service). If your business process is defined as just one of these functions, you are likely to cause or perpetuate problems. Work methods will be defined for the benefit of the individual function, not to optimize the manner in which work flows through the functions. The idea that functions are self-contained units that
impede the flow of work has given rise to the prevailing terminology—“functional silos,” “functional stovepipes,” or “functional castles”: strong walls that keep the outside out and the inside in. Functions are described as “vertical,” like a silo, and processes as “horizontal,” cutting across functions.

Recently there has been a different sort of problem: defining cross-functional work (which is good!) that doesn’t yield a business process (which is bad!). These structures are “areas” or “horizontal silos.” For instance, Customer Relationship Management is a popular term (along with Supply Chain Management and Demand Chain Management), and is often referred to as a process, as in “our CRM process.” It is cross-functional, the way a business process should be, but it is not countable, so it’s like a function. You cannot say how many Customer Relationship Managements you did yesterday. And if you try to map, assess, and improve Customer Relationship Management, you might find it to be a very frustrating endeavor following all the paths. The underlying problem is that the CRM area includes multiple cross-functional business processes such as Secure New Customer, Resolve Customer Inquiry, and Complete Customer Communication, each of which can be studied as a separate process.

Other business process characteristics

So, we have established that a business process begins in response to an event, proceeds through a sequence of tasks, and ultimately yields a result for the customer of the process and the other stakeholders. Let’s add a few more characteristics to the definition:

Measurable

We must be able to measure the business process in whatever way is important to the stakeholders. Customers may care about the effort they have to invest and the total time until they receive the result. The organization’s performers may care more about training time or the impact on their own productivity statistics. The owner or manager will want to track cost, overall customer satisfaction, and other variables. A well-defined and well-designed business process must satisfy the demands of all stakeholders.

Automation

In looking at individual tasks within a process, automation may or may not play a role. A task could be totally manual, for example, “interview client.” However, nowadays almost all processes are at least partially automated, and with the emergence of straight-through processing (STP), many processes, such as executing a trade order on a stock exchange, may be completely automated. In that example, the initiating event could be the detection of a particular condition, such as a preset stock price limit being reached, and the result could be the deposit of funds in the client’s account. Essentially, this means that automation is a nonissue when deciding whether or not a step belongs within a process—a step could be totally manual, partially manual with automated support, or fully automated. We mention this because it emerges as an issue. Unbelievably, within the past year, we were told by one group of reengineering consultants that process models should only include steps that involved people, and by another “guru” that process models should only include automated steps. Obviously, neither approach would lead to a full understanding of the process.

Levels

Processes can be described at various levels. Initially, simply identify the process, as in Resolve Service Problem. Next, identify the milestones within the process, and then map the process workflow to increasing levels of detail. Any number from two (the business process and its milestones) up to five or so is reasonable, depending on the situation. There are no widely used terms for the various levels of the processes, or the steps within them, although we will define three levels of swimlane diagrams in Chapter 9. We can continue breaking the process down until we are describing individual interactions between people and systems, which means we have gone too far—that is what use cases are for.

Customers: internal and external

Every process has a customer, the person or organization expecting the primary result that the process delivers. Obviously, customers must be identified so we can obtain their assessment, and ensure that the process design meets their expectations. Another reason to focus on the customer is that in many processes, there is no overall responsibility—no one in the organization makes sure the process is completed. So we must focus on the customer because the customer is the glue that holds the process together, and
must retrigger the process to move it along. A recent example is that one of us had a major appliance fail, and had to “walk” the warranty claim from the dealer to the manufacturer to the local service organization to the appliance repair agency to which they subcontracted the job. The moral: Sometimes only the customer sees the entire process from beginning to end, so identifying the customer is essential to understanding process behavior.

We sometimes distinguish processes depending on whether the customer of the process is internal or external to the organization. Figure 4.1 shows an example of each type of process.

Processes that serve external customers are typically why the business exists, so they are often referred to as core processes. Most businesses have only about seven to ten core process clusters or process areas in total. These are the horizontal silos. Examples are Market Research, Customer Acquision, Product Development, Supply Chain Management, Order Fulfillment (which includes manufacturing in a build-to-order environment), Customer Relationship Management, and Customer Service. These areas are useful for high-level presentations and to get people “into the ballpark,” but not specific enough to analyze and improve. When it gets down to actual workflow process modeling, you have to get down to real processes.

A process focused on an external customer is ideal for a project, because it’s so much easier to demonstrate that the bottom-line performance of the company is being improved. But for some business processes, there is only an internal customer. HR-oriented processes such as Enroll in Benefit Program or Resolve Contract Issue are definitely internal. Some experts say focus only on the external customer—we disagree. How you treat your internal resources inevitably translates into how your customers are treated. If you treat your employees poorly, you may soon lose them all and go out of business. Logistics win wars—you need to support the fighters if you expect them to win. Just be sure you don’t arbitrarily redefine the process to use an internal customer when there really is an external customer.

Processes that serve internal customers are sometimes referred to as supporting processes, and within this category we have seen them further subdivided into technical supporting and social supporting. Technical supporting processes provide or enhance infrastructure, and they almost always serve other business processes. Examples include Provide Facility, Deploy Application, and Develop Business Process. Social supporting processes provide or enhance people, for example, Recruit Employee, Hire Employee, Train Employee, and Provide Employee Benefits. Your scope will generally be smaller than the major business processes we have used as examples here, but the pattern of core and supporting will scale down and still be useful.

All processes are important, so don’t use this as a way to decide which processes are more or less important, as some experts do. This classification scheme is useful because it helps avoid drifting out of a process’s natural boundaries. For instance, if you are mapping a process within Product Development (core), and you find yourself including training activities (social supporting), you may be mixing two different processes, which can get very confusing.

Closing advice

Keep two key points about the nature of business processes in mind when defining process boundaries:

1. Processes are hidden within your organization.
2. There is a tendency to define business processes too small.

Processes are hidden

Business processes are usually not evident, because they are hidden by organizational structure, jobs, and systems. Don’t be surprised or
disappointed if your organization has trouble identifying its processes and gaining consensus. In some cases, it is easier for customers to define the business process because they are the only ones who see it all. That was the case at a telecommunications company. They were quite sure that there were four processes, corresponding nicely to the organizational boundaries illustrated in Figure 4.2.

1. Capture Service Order;
2. Assign Facilities;
3. Install Service;

The customers, of course, did not care about the organizational structure of the phone company—they just wanted their telephone service activated. The real process that was eventually identified was Activate Telephone Service. Remember, even if the organization doesn’t recognize it, the process is there—somehow, the initiating event eventually makes its way to a result. It’s hard for companies to identify their processes because of years of conditioning in looking at things organizationally.

**Bigger is (usually) better**

Business processes coordinate their elements: people, resources, systems, and work. Without business processes, everything would be done on an ad hoc (and probably uncoordinated) basis. In a well-designed process all of these elements are well coordinated, including the individual work steps.

You should walk the process backward from any point in the process until you find the customer (internal or external) that generated the earliest event that kicked off this work, backward and forward from the cog to find the other wheels, until you reach a customer on each end.

If you take a single “natural” process, like our Activate Telephone Service example, the odds are it will perform better if designed as a single larger process than if the same result was achieved through five smaller processes as illustrated in Figure 4.3. But why should this be so, especially if each of the small processes is well designed? Let’s go back to the notion of coordination to answer that.

In an ideal process, the inputs and outputs of the individual steps are well coordinated. That is, the output from one step flows smoothly and uninterrupted to become the input of the next step in the process. That is the whole notion of workflow. Process designers often accomplish this within the boundaries of a process, but are not usually as successful at the process boundaries. That is because there is a strong tendency, when trying to achieve optimization within their process, for designers to put constraints on work entering and leaving their process. Rekeying of data, batching of inputs, transport to a separate location, and fixed processing schedules are some obvious examples. Processes often require their customer—whoever precedes them in the flow—to reenter data into a format that is optimized for the receiving process. This makes the receiving process internally optimal, but the overall (natural) process is slowed by

---

Figure 4.2 Process and organizational structure are often confused.

![Diagram](image)

Figure 4.3 With processes, bigger is usually better.
the additional work. Worse yet, errors will be introduced during the rekeying of data that will ultimately cause even longer delays. Batching introduces similar problems. The idea behind batching is that a process will handle individual work items most efficiently when they are grouped into batches of like items. The process performers can then get set up to handle that type of item most efficiently. It appears that the process is very efficient, but only within that process for those performers—the natural process is less effective. Individual work items might spend a lot of time waiting for their batch to get big enough, or for their turn to come.

This phenomenon was described by Eliyahu Goldratt, and it leads to this conclusion: Processes should be defined as large as possible, because multiple, small processes each tend to strive for internal efficiency; local optimization leads to overall suboptimization. This really is not surprising, because it is essentially the same problem as functional organizations putting constraints on the entry and exit of work.

Well-meaning process improvement teams can unwittingly make a large process less effective by locally optimizing a subprocess. This leads to an important corollary: If you know that your project scope is less than a complete business process, be sure to focus attention on expediting the flow of work into and out of your process.

Putting the definition to work

Everything we have covered in this chapter boils down to defining a process by discovering the earliest event and the delivery of the final result to a customer that signals the complete handling of that event. It still isn’t easy, though, as evidenced by the difficulty organizations have in discovering their true business processes. In the next chapter we will look at some techniques for putting the material in this chapter to work defining the boundaries of your processes.

References


[2] WordNet 1.6, copyright © 1997 by Princeton University. All rights reserved.


1. See The Goal by Eliyahu Goldratt, which describes this and other important process improvement concepts in the context of an entertaining novel.