Warranty Disclaimer

THIS DOCUMENT AND ALL INFORMATION PROVIDED HEREIN (THE "INFORMATION") IS PROVIDED ON AN "AS IS" BASIS AND FOR GENERAL INFORMATION PURPOSES ONLY. ORACLE EXPRESSLY DISCLAIMS ALL WARRANTIES OF ANY KIND, WHETHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT. ORACLE MAKES NO WARRANTY THAT THE INFORMATION IS ERROR-FREE, ACCURATE OR RELIABLE. ORACLE RESERVES THE RIGHT TO MAKE CHANGES OR UPDATES AT ANY TIME WITHOUT NOTICE.

As individual requirements are dependent upon a number of factors and may vary significantly, you should perform your own tests and evaluations when making technology infrastructure decisions. This document is not part of your license agreement nor can it be incorporated into any contractual agreement with Oracle Corporation or its affiliates. If you find any errors, please report them to us in writing.

Third Party Content, Products, and Services Disclaimer

This document may provide information on content, products, and services from third parties. Oracle is not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Limitation of Liability

IN NO EVENT SHALL ORACLE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL OR CONSEQUENTIAL DAMAGES, OR DAMAGES FOR LOSS OF PROFITS, REVENUE, DATA OR USE, INCURRED BY YOU OR ANY THIRD PARTY, WHETHER IN AN ACTION IN CONTRACT OR TORT, ARISING FROM YOUR ACCESS TO, OR USE OF, THIS DOCUMENT OR THE INFORMATION.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Process Classification FrameworkSM (PCF) was developed by APQC and member companies as an open standard to facilitate improvement through process management and benchmarking regardless of industry, size, or geography. The PCF organizes operating and management processes into 12 enterprise-level categories, including process groups and over 1,500 processes and associated activities. The PCF and associated measures and benchmarking surveys are available for download and completion at no charge at www.apqc.org/OSBCdatabase.

ABOUT APQC

APQC is the leading resource for performance analytics, best practices, process improvement, and knowledge management worldwide. APQC’s research studies, benchmarking databases and renowned Knowledge Base provide managers with intelligence that they can use to transform their organizations for better results. A member-based nonprofit founded in 1977, APQC currently serves over 750 of the Fortune Global 1000 companies as well as hundreds of education and government organizations. Learn more by visiting www.apqc.org or calling 800-776-9676 or 1-713-681-4020
# Contents

**Send Us Your Comments** ........................................................................................................................................ vii

**Preface** ................................................................................................................................................................ ix

IT Strategies from Oracle................................................................................................................................. ix
Purpose of This Document............................................................................................................................. x
Audience .......................................................................................................................................................... xi
Document Structure ......................................................................................................................................... xi
Conventions ..................................................................................................................................................... xi

## 1 Key Concepts

1.1 Business Process Management (BPM) ................................................................................................. 1-1
1.2 Definition of a Business Process ....................................................................................................... 1-1
1.3 Business Architecture ....................................................................................................................... 1-1
1.4 EPM versus BPM............................................................................................................................... 1-2
1.5 Process Model States / “Levels” ....................................................................................................... 1-2
1.6 Roles.................................................................................................................................................. 1-3
1.6.1 Business Participants .................................................................................................................... 1-3
1.6.2 IT Participants .............................................................................................................................. 1-5
1.6.3 Supporting Roles ........................................................................................................................... 1-6

## 2 Introduction to Business Process Engineering

2.1 Motivation for Adopting a BPM Methodology .................................................................................. 2-3
2.2 Objectives for a BPM Methodology ................................................................................................. 2-3
2.3 Top-down versus Bottom-up ............................................................................................................ 2-4
2.4 Scope ............................................................................................................................................... 2-4
2.5 Business Process Engineering Lifecycle.......................................................................................... 2-5
2.5.1 Detailed Business Process Engineering States........................................................................... 2-7
2.6 The Oracle Business Process Engineering Framework ...................................................................... 2-8
2.6.1 Strategic Analysis and Business Process Selection ..................................................................... 2-8
2.6.2 Process Discovery and Definition ............................................................................................... 2-9
2.6.3 Business Process Refinement ..................................................................................................... 2-9
2.6.4 Technical Analysis and Business Process Design ..................................................................... 2-9
2.6.5 Business Process Application Composition ................................................................................. 2-9
2.6.6 Integration Testing ....................................................................................................................... 2-10
2.6.7 Deployment Planning ..................................................................................................................... 2-10
3 Business Process Identification and Selection (Analysis Phase)

3.1 Strategic Approach ........................................................................................................................................ 3-2
3.1.1 Functional models .................................................................................................................................. 3-3
3.1.2 Functional Decomposition ..................................................................................................................... 3-8
3.1.3 Business Motivation Models .................................................................................................................. 3-9
3.2 Tactical Approach ......................................................................................................................................... 3-10
3.3 Process Selection .......................................................................................................................................... 3-10
3.3.1 Process Candidates Evaluation Criteria ................................................................................................. 3-11
3.3.1.1 Business Process Information ........................................................................................................... 3-12
3.3.1.2 Benefit Parameters ............................................................................................................................ 3-12
3.3.1.3 Inhibitor Parameters .......................................................................................................................... 3-13
3.3.1.4 Decision Basis Score ......................................................................................................................... 3-14

4 Business Process Design

4.1 Process Definition and Refinement ........................................................................................................... 4-2
4.2 Process Definition ......................................................................................................................................... 4-3
4.2.1 Capture High Level Process Flow ......................................................................................................... 4-4
4.2.1.1 Create Business Process Context Model .......................................................................................... 4-7
4.2.1.2 Key Performance Indicators ............................................................................................................. 4-8
4.2.1.3 Identifying Supporting Applications ................................................................................................. 4-8
4.2.1.4 Business Objects ................................................................................................................................ 4-10
4.2.2 Elaborate Business Process Details......................................................................................................... 4-12
4.2.2.1 Correct Issues and Resolve Inconsistencies ....................................................................................... 4-13
4.3 Refinement .................................................................................................................................................. 4-14
4.3.1 Formulate Process Changes .................................................................................................................... 4-15
4.3.2 Simulate and Optimize ............................................................................................................................. 4-15
4.3.3 Final Business Verification ..................................................................................................................... 4-15
4.3.4 Document User Interactions .................................................................................................................. 4-16
4.4 Technical Analysis and Design .................................................................................................................. 4-16
4.4.1 Capture Technical Specifications........................................................................................................... 4-17
4.4.2 Information Needs Analysis .................................................................................................................... 4-20
4.4.3 Security Needs Analysis .......................................................................................................................... 4-21
4.4.4 Gap Analysis and Technical Feasibility ................................................................................................. 4-22
4.4.5 Define Software Requirements .............................................................................................................. 4-23
4.4.6 Filling the IT gaps ..................................................................................................................................... 4-23
4.5 Business Application Composition ............................................................................................................ 4-24
4.6 Unit Testing .................................................................................................................................................. 4-25

5 Production Readiness

5.1 Testing ......................................................................................................................................................... 5-5
List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1–1</td>
<td>Business Process Engineering Roles</td>
<td>1-3</td>
</tr>
<tr>
<td>2–1</td>
<td>Simplified Business Process Engineering Lifecycle</td>
<td>2-5</td>
</tr>
<tr>
<td>2–2</td>
<td>Business Process Engineering Lifecycle States</td>
<td>2-7</td>
</tr>
<tr>
<td>2–3</td>
<td>Major Activities in the Business Process Engineering Lifecycle</td>
<td>2-8</td>
</tr>
<tr>
<td>2–4</td>
<td>Ancillary Software Engineering Activities</td>
<td>2-11</td>
</tr>
<tr>
<td>3–3</td>
<td>Functional Model Levels</td>
<td>3-4</td>
</tr>
<tr>
<td>3–4</td>
<td>Porter Value Chain</td>
<td>3-5</td>
</tr>
<tr>
<td>3–5</td>
<td>Example Enterprise Process Model (level 0)</td>
<td>3-6</td>
</tr>
<tr>
<td>3–6</td>
<td>Example Process Groups (level 1)</td>
<td>3-7</td>
</tr>
<tr>
<td>3–7</td>
<td>Example Business Function Model</td>
<td>3-8</td>
</tr>
<tr>
<td>3–8</td>
<td>Business Motivation Model</td>
<td>3-9</td>
</tr>
<tr>
<td>3–9</td>
<td>Example Strategy Map</td>
<td>3-10</td>
</tr>
<tr>
<td>3–10</td>
<td>Process Selection</td>
<td>3-11</td>
</tr>
<tr>
<td>3–11</td>
<td>Example Process Candidate Analysis Results</td>
<td>3-15</td>
</tr>
<tr>
<td>4–1</td>
<td>Design Phase Activities</td>
<td>4-2</td>
</tr>
<tr>
<td>4–2</td>
<td>Process Definition and Refinement</td>
<td>4-3</td>
</tr>
<tr>
<td>4–3</td>
<td>Process Definition</td>
<td>4-4</td>
</tr>
<tr>
<td>4–4</td>
<td>Functional Decomposition (level 3)</td>
<td>4-6</td>
</tr>
<tr>
<td>4–5</td>
<td>Functional Decomposition (level 4)</td>
<td>4-6</td>
</tr>
<tr>
<td>4–6</td>
<td>Functional Decomposition (level 5)</td>
<td>4-7</td>
</tr>
<tr>
<td>4–7</td>
<td>Example Business Context Model</td>
<td>4-8</td>
</tr>
<tr>
<td>4–8</td>
<td>Example Application to Function Mapping</td>
<td>4-9</td>
</tr>
<tr>
<td>4–9</td>
<td>Example Applications to Organization Mapping</td>
<td>4-10</td>
</tr>
<tr>
<td>4–10</td>
<td>Example Business Objects in the Message Flow</td>
<td>4-10</td>
</tr>
<tr>
<td>4–11</td>
<td>Example Entities to Application Mapping</td>
<td>4-11</td>
</tr>
<tr>
<td>4–12</td>
<td>Process Refinement</td>
<td>4-14</td>
</tr>
<tr>
<td>4–13</td>
<td>Technical Analysis and Design</td>
<td>4-16</td>
</tr>
<tr>
<td>4–14</td>
<td>The BPM Technical Solution</td>
<td>4-18</td>
</tr>
<tr>
<td>4–15</td>
<td>Logical Data Model</td>
<td>4-21</td>
</tr>
<tr>
<td>4–16</td>
<td>Task Configuration</td>
<td>4-25</td>
</tr>
<tr>
<td>5–1</td>
<td>Impact of Changes on Testing</td>
<td>5-4</td>
</tr>
<tr>
<td>A–3</td>
<td>Weighting Worksheet</td>
<td>A-3</td>
</tr>
</tbody>
</table>
Send Us Your Comments

Business Process Engineering, Release 3.0
E20216-03

Oracle welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

■ Did you find any errors?
■ Is the information clearly presented?
■ Do you need more information? If so, where?
■ Are the examples correct? Do you need more examples?
■ What features did you like most about this document?

If you find any errors or have any other suggestions for improvement, please indicate the title and part number of the documentation and the chapter, section, and page number (if available). You can send comments to us at its_feedback_ww@oracle.com.
The Oracle Business Process Engineering framework introduces a discipline to effectively deliver composite business application projects in a BPM environment. The framework encompasses the traditional analysis, design, implementation, and operational management and monitoring steps of software engineering, but addresses only the requirements specific to Business Process Engineering that are not covered by traditional methods and practices. In this way the Oracle Business Process Engineering framework can be used to simply augment existing practices.

**IT Strategies from Oracle**

IT Strategies from Oracle (ITSO) is a series of documentation and supporting collateral designed to enable organizations to develop an architecture-centric approach to enterprise-class IT initiatives. ITSO presents successful technology strategies and solution designs by defining universally adopted architecture concepts, principles, guidelines, standards, and patterns.

ITSO is made up of three primary elements:

- **Oracle Reference Architecture (ORA)** defines a detailed and consistent architecture for developing and integrating solutions based on Oracle technologies. The reference architecture offers architecture principles and guidance based on recommendations from technical experts across Oracle. It
covers a broad spectrum of concerns pertaining to technology architecture, including middleware, database, hardware, processes, and services.

- **Enterprise Technology Strategies (ETS)** offer valuable guidance on the adoption of horizontal technologies for the enterprise. They explain how to successfully execute on a strategy by addressing concerns pertaining to architecture, technology, engineering, strategy, and governance. An organization can use this material to measure their maturity, develop their strategy, and achieve greater levels of adoption and success. In addition, each ETS extends the Oracle Reference Architecture by adding the unique capabilities and components provided by that particular technology. It offers a horizontal technology-based perspective of ORA.

- **Enterprise Solution Designs (ESD)** are industry specific solution perspectives based on ORA. They define the high level business processes and functions, and the software capabilities in an underlying technology infrastructure that are required to build enterprise-wide industry solutions. ESD’s also map the relevant application and technology products against solutions to illustrate how capabilities in Oracle’s complete integrated stack can best meet the business, technical, and quality of service requirements within a particular industry.

Please consult the ITSO web site for a complete listing of ORA documents as well as other materials in the ITSO series.

A Practitioner’s Guide provides insight and guidance when working with a particular type of technology and addresses the common concerns faced by enterprises and practitioners.

This Practitioner’s Guide provides an approach to Business Process Engineering within a BPM environment. It describes guidelines and best practices that enterprises and practitioners can use to develop a consistent and repeatable approach to Business Process Engineering from initial identification of suitable process candidates through to deployment and operations.

**Purpose of This Document**

This *ORA Business Process Engineering* document describes a practitioner’s approach to establishing a method for business process automation. For the purpose of brevity
this name is often abbreviated to "the Oracle BPM method", or "the BPM method" throughout this document.

This document provides the information necessary to extend an existing software engineering method (such as UP, Agile, etc.) to create an effective engineering approach for the computerization of business processes.

Refer to the ORA Master Glossary document for descriptions of key terms. See the ORA BPM Foundation and ORA BPM Infrastructure documents for architectural background.

Audience

This guide is intended for business process specialists, project and program managers, enterprise architects, process architects, process engineers, and other stakeholders involved in delivering projects in a BPM environment. It provides the background material as the basis to discuss BPM solutions between Oracle consultants and Oracle customers.

Document Structure

This document is organized in chapters spanning introduction, background, the core business process engineering method, and appendices. The method organization is loosely aligned with the classic engineering phases of analysis, design, etc. The chapters include the following topics:

Chapter 1 provides a number of key concepts that form the basis for the rest of the guide.

Chapter 2 is an introduction and overview of the Oracle Business Process Engineering Framework which outlines the methodology for Business Process Engineering.

Chapter 3 presents a detailed description of the analysis phase of Oracle Business Process Engineering Methodology covering strategic analysis, business process identification and selection.

Chapter 4 provides a detailed description for Business Process Delivery covering process definition, design, and implementation.

Chapter 5 provides a high-level approach to production readiness in Business Process engineering.

Chapter 6 covers the engineering considerations for monitoring in BPM.

Appendix A outlines the Oracle Business Process Selection framework.

Appendix B is a list of recommended supplementary reading.

Appendix C provide lists of external references found throughout this document.

Glossary provides BPM specific definitions of terms that found throughout this document.

Conventions

The following typeface conventions are used in this document:
"SOA Service" - In order to distinguish the "service" of Service-Oriented Architecture from the wide variety of "services" within the industry, the term "SOA Service" (although somewhat redundant) will be used throughout this document to make an explicit distinction for services that were created as part of an SOA initiative; thus distinguishing SOA Services from other types of services such as Web Services, Java Messaging Service, Telephone service, etc.

"Automation" - the word automation has been used to describe many different types of mechanization and computerization of, primarily human, activities and processes. Under the heading of BPM, the term automation was adopted to refer to the computerization of human work functions. In today’s BPM the verb "to automate" is used to refer to the act of making a business process model executable by placing it under the control of a computer system (specifically a BPMS). The "executable model" part is a relatively new concept and while it is true that earlier workflow systems automated technical (IT) system processes, the new approach for automation of business processes must be distinguished from this and the BPM subset of human workflow computerization.

Architectural terminology - When appearing in this document, the terms view, viewpoint, stakeholder, and concern are used according to the definitions appearing in the IEEE 1471 "Recommended Practice for Architectural Description of Software-Intensive Systems". The term perspective however, does not appear in IEEE 1471 and is defined instead, within the context of ORA, to refer to a particular viewpoint on to the ORA core reference architecture (in this case the technological viewpoint of BPM).

"Methods and methodology" - this document describes a framework approach for business process automation which is intended to be adapted to the needs of any given organization and to integrate with existing engineering methods. As a framework for the development of specifically tailored methods this document is considered to be a methodology i.e. a system for the creation of instances of methods.

The terms engineering and architecture are most commonly applied to the technical activities in IT (providing a distinction from other activities such as management). However, for the purpose of this method definition, these terms are used in a broader context encompassing the business activities and artifacts need to support effective Business Process Engineering.

Due to the collision in the abbreviations for the separate, but connected, subjects of Business Process Management and Business Performance Management, this document distinguishes the latter as Business/Enterprise Performance Management or simply by the abbreviation "EPM".
This chapter contains important concepts, definitions, and background material that support the business process engineering method, but are not directly part of the method itself.

1.1 Business Process Management (BPM)

Oracle defines BPM as a strategy for managing and improving the performance of a business through continuous optimization of business processes in a closed-loop cycle of modeling, execution, and measurement.

A useful extension to this description appears in The Third Wave stating, BPM encompasses the discovery, design, and deployment of business Processes and, in addition, BPM includes the executive, administrative, and supervisory control of those Processes.

1.2 Definition of a Business Process

A business process is a set of linked activities performed by people and systems that deliver some kind of value through a product or process to internal or external customers. A business process describes a sequence or flow of activities in an organization with the objective of carrying out work. The flow of a process is controlled by decisions, triggers, events, and exceptions, any of which may be influenced by rules or policies.

A process also has performance indicators which may be compared to actual performance during execution. The activities in a process are performed by system or human actors (or groups conforming to a role) which may be internal to an organization or external (business partners).

See ORA BPM Foundation document for a more detailed description of a Process and its scope.

1.3 Business Architecture

Business architecture reflects the design decisions for process, resources, rules, roles, regulations, and responsibilities that will maximize business value and minimize overhead. As such the business architecture is employed as the blueprint for operating and transforming the enterprise. Key elements of the business architecture are:

---

- A motivation model which encapsulates an organization's business goals, objectives, initiatives, and metrics that define success factors
- Business Functional models that define what value is provided by one organization to another - including both internal and external functions.
- Organizational model which depicts roles, responsibilities, and collaborations, defining how and by whom functions are to be provided and used.
- Business process models that define the activities, steps, and information flows between processes to carry out business functions.
- Associated business rules and policies which are utilized by a Governance framework to ensure that stakeholders adhere to and enforce these policies.
- Information and the business semantic vocabulary
- A roadmap to define the steps to achieve business transformation objectives.

Whether business architecture is part of an overall enterprise architecture discipline or it is simply an input into enterprise architecture seems to be a matter of opinion and varies between organizations. Either way, to truly reap the potential rewards that BPM offers, organizations need to adopt a more comprehensive view of business processes, as well as taking the overall enterprise architecture into account, rather than automating and managing individual business processes.

### 1.4 EPM versus BPM

Like most domain terms BPM has more than one meaning: to us (the readers of this document) it should mean “Business Process Management”; however, a related field of work uses BPM to mean “Business Performance Management” (a field that is hopefully supported by our BPM). In order to avoid ambiguity in this document we have adopted Oracle’s term “Enterprise Performance Management” (EPM) in place of the expression “Business Performance Management” and its colliding abbreviation.

### 1.5 Process Model States / “Levels”

Many “levels” and categorizations are used throughout the Oracle BPM Methodology and it is important to apply them in their appropriate context and avoid confusion. One such set of levels used in this document refers to the degree of detail in a business process model as it transitions from an early analysis communication vehicle to its ultimate executable state. A hierarchy of three levels is typically described as follows:

1. Documentation of the "happy path", used to describe how the BP flows to achieve the desired result. Level 1 is intended for use by the business stakeholders, it should be constrained to a limited palette of notational elements suitable for untrained users.

2. Bridges the gap between business and IT by filling in the details of the exception handling, answers difficult questions like compensating transaction, etc. Level 2 makes extensive use of the modeling notation and is intended for Business Analysts and Process Architects.

3. Completes the technical activity of associating (wiring) "services" together, defining message flows and data mapping & transformation, etc. in order to make it executable. Level 3 is intended for use by IT, generally the process developer and architect.
This outline is similar to the three levels of modeling originally described by Bruce Silver and adopted by the OMG. Silver names his three stages business process model evolution, “descriptive modeling”, “analytic modeling”, and “executable modeling”.

Other kinds of levels appear throughout this document, such as the levels in a functional model, and it is important not to confuse them.

1.6 Roles

The following section outlines the core organizational participants required for a successful BPM project. While role names and granularity vary between organizations it is important to ensure that the responsibilities identified here are mapped appropriately to any given organization structure.

There are a number of business and technical participants in the Oracle BPM Methodology and it is important to ensure coordination between these two major groups. The BPM Analyst and Process Architect, in particular, represent an important point of convergence between business and IT.

Figure 1-1 Business Process Engineering Roles

1.6.1 Business Participants

**Business Leadership** - Drives requirements by setting business goals, objectives, and priorities. The business leadership provides initial inputs, such as, business motivation definitions including high-level vision and mission statements. The business leadership provides funding for the BPM initiative and typically expects measurable return on investment.

Business leadership might include the following more specific role names:

- Executive (C-level) Management
- Line-of-Business (LOB) Management
Roles

- Business Process Manager
- Unit / Division / Departmental / Branch Manager
- Business Service Owner
- Quality Manager
- Compliance Manager

The business leadership uses BPM analytics in a broader Business/Enterprise Performance Management (EPM) context and determines the need for major business process change.

A significantly large BPM initiative may identify a new executive role, perhaps called "Chief Process Officer", who will be responsible for fostering a process-centric business culture, influencing the development of skills, systems, and behaviors, and championing an enterprise “process architecture”.

**Process Owner** - Managing process flow, task assignments, policies, rules, objectives, and performance measurement, the Process Owner is responsible and accountable for the performance of end-to-end business processes. The Process Owner owns a specific process or processes, and is the primary authority for making decisions necessary to resolve conflict or overlap between processes.

Process Owner is the subject matter expert (SME) for the business process(es). He is the business primary contact for the Business Process Analyst. People in this role however, specialize in a small number of processes and don’t have the broader perspective of other the business leadership across the enterprise.

Establishing the Process Owner role has emerged as a best practice for BPM since the key business processes cross organizational boundaries and typically have no single point of responsibility. Commonly, gaps (known as the white spaces) exist between the departmental silos of business process activity where ownership is undetermined and knowledge is sparse. The creation of the Process Owner role solves these problems by identifying individuals to be the experts responsible for the end-to-end flow of key business processes. While authority for departmental or divisional business process activities remains with respective managers, the Process Owner is able to take a holistic view and make better recommendations for enterprise-wide process improvement.

Business process administration during runtime also requires considerable business knowledge and the Process Owner should be expected to support this key operational activity.

**Process Participants** - The participants (aka “process performers”) in a business process perform the human aspects of the business process task. Commonly referred to as “end users” in software engineering, these participants provide task execution details to the Process Analyst and may be involved in acceptance testing. During analysis for process automation it is important to interview the end users and watch them work to see what they actually do (human activities are commonly more diverse than expected).

Some more specific role names for users of the business process might be:

- Administrator
- Supervisor
- Worker
- Call Center Agent / Customer Service Representative (CSR)
- Clerks (Back-office processing)
- Business Partner
- Customer

In addition to the end user performers of the business process work, the automated process commonly requires the support of additional participants categorized here as administrators and supervisors. While supervisors are responsible for overseeing the work performed by the end user participants at the process instance level, administrators are responsible for more general oversight including like role assignments, view management, etc. In addition, supervisors will be in charge of making sure the work is processed in a timely manner, managing exceptions, delegation, and reassignment of work as needed.

**Process Analyst** - Also commonly known as Business Analyst (or Business Process Analyst), the Process Analyst is primarily responsible for capturing and managing the graphical business process models. The Process Analyst also captures related business process requirements, drives process optimization, recommends changes, and evaluates change requests from the business. The Process Analyst has business and modeling skills and liaises with the Process Architect for technical coordination.

The Process Analyst has primary responsibility for making incremental process improvements (as opposed to major change from the business leadership). The Process Analyst also

- Identifies and codifies business rules
- Uses business objectives as input to determine KPI
- Provides business specification for new capabilities
- Directs UAT

### 1.6.2 IT Participants

**Process Architect** - Performs analysis and design of technical aspects of the process, taking the process specification from the Process Analyst for technical analysis. The Process Architect also specifies additional technical software requirements, such as application integration, UI development, etc. and works with the Process Analyst to design technical specifications for new functional requirements.

The Process Architect may also be responsible for

- Defining technical integration strategies
- Technical specification for new IT capabilities
- Directing system and integration testing

In an environment where SOA is fully implemented, discovery of services for functional requirements in the process model can be performed effectively by non-technical participants (typically the Process Analyst). In cases of less developed integration architectures however, the Process Architect would be required to identify the most suitable sources of application functionality and business entities.

This is a specialized architecture role similar to "solution architect" in traditional software engineering, but with an emphasis on understanding business process modeling and the details of process implementation in addition to more general software architecture skills.

**Process Developer** - May also be called Process Designer, enhances the process model to make it executable in the IT environment by configuring data mapping and transformation of activity inputs and outputs, and defines external data required by the process, etc.
Roles

1.6.3 Supporting Roles

Supporting roles provide security, data, enterprise architecture assistance along with traditional software and service development needs identified by the BPM project. The activities performed by these participants commonly follow their own methods and should be handled separately from the core BPM activities.

These supporting roles include:

**QA Manager** - Directs acceptance testing and validation.

**Operations Manager** - Handles deployments and technical OA&M (Operations, Administration, and Management)

**Software / Service / UI Developers** - Develops any required new or extended functional capabilities specified by the Process Architect. Traditional software development requirements may emerge from the BPM project including integration, SOA Services, or enhancement to existing application functionality. In the case of human workflow a task-list user interface (UI) or existing application UI may need to be developed to support the human interface to business process.

**Enterprise Architect** - Provides oversight to ensure IT strategies and standards are applied. The Enterprise Architect works with the Business Leadership to

- Identify business architecture inputs to BPM (e.g. objectives for program, business motivation, strategy maps, etc.)
- Helps determine the need for major business process change

**Security Architect** - Ensures corporate security policies are followed and provides input to the process design for authentication, authorization, and access between potentially siloed applications and departments.

**Data Architect** - Provides support for information access in the process design and may be responsible for accepting process analytics feeds from the BPM system to support broader BAM/BI/EPM initiatives.

The list above is intended to identify roles specific to BPM (indeed a number of these are entirely new roles identified by best practices emerging from practical experience in the field) so some traditional IT software engineering roles have been omitted for brevity. This is not to suggest that roles, such as, project manager are not needed. In fact, all BPM projects should follow the highly successful software engineering pattern for the project leadership team, which encompasses stakeholder liaison (Project Manager), technical liaison (Process Architect), and project sponsorship (Process Owner).

The Process Analyst and Process Architect are critical roles in the automation of a business process (and are therefore drawn in the center of the diagram). These roles are categorized separately under business and IT because one has a greater business focus while the other is more technically oriented. In other respects the skills of the Process Analyst and Process Architect are very similar, and so, in smaller projects, they may be performed by a single person. This is just one example of how the role-to-people relationship is applied in accordance with the size of the project. The value of a complete role list is in ensuring that all the skills and responsibilities needed for successful business process engineering are accommodated by the project, regardless of the way in which they are mapped individual performers.
Traditionally business processes were described using loosely defined graphical artifacts with uses limited to supporting business analysis discussions and requirements definition for software projects. Today a more rigorous business process modeling language can be applied and the graphical model extended to include IT functions to make it executable by an IT system. Furthermore, the operation of the business process can be monitored and analyzed by the Business Process Management (BPM) System to help the business to identify opportunities to continuously improve business process performance. This can all be achieved with minimal involvement from IT.

The act of placing a business process under the control of a BPM system is called automation (not to be confused with automation of the 1990's which merely involved replacement of human tasks with computerized tasks). This document describes the Oracle BPM Methodology for achieving automation and realizing the full potential of today's BPM.

Automation has long been a goal of BPM, but until now the tools for making a business process executable have been fraught with IT constraints and technical intervention. Only now is it possible to take a business process model, make associations with human workflow and IT applications, and place the flow of these activities under the control of an automated system without the need for specialized coding effort to tie it all together. In combination with a Business Rules Management System (BRMS) and Service Oriented Architecture (SOA) modern enterprises can achieve the extremes of agility demanded by today's business environment.

This "zero code" approach to business process automation must be guided by an appropriate method designed specifically for the key BPM stakeholders (primarily business specialists) that also aligns with current IT environments and practices. The Oracle BPM Methodology is an extension to well established engineering practices, designed to guide the implementation of business process automation and continuous improvement while supporting Enterprise Performance Management (EPM).

The Oracle BPM Methodology is an agile strategy that defines an iterative approach to Business Process Engineering highly suited to the rapidly changing business environment (driven by market demands, mergers and acquisition, etc.) and continuous incremental business process improvement. It is an adaptable strategy that complements existing software engineering practices and is compatible with common integration approaches.

So far we have talked in broad terms about the opportunities available from a modern BPM solution, but the need for a guiding method may not be so obvious. The following list outlines the benefit of using the Oracle BPM Methodology as a
foundation to create a standard practice for business process implementation for given organization.

1. Predictability: by applying a consistent, proven approach to BPM engineering the results and the cost can be determined in advance with a high degree of confidence.

2. Business driven: by identifying a range of well known business motivation artifacts as drivers for business process identification and providing analytical feedback based on business defined KPI’s, the Oracle BPM method effectively ensures continuous alignment with business needs.

3. Closing the business - IT gap: the Oracle BPM method identifies stakeholder roles and their information exchanges at every step of the engineering process, ensuring effective business and IT collaboration.

4. Process identification: the BPM method provides guidelines and tools for process computerization candidate analysis leading to clear justification and measurable objectives for the BPM engineering project. This approach is also applied to business rule and KPI identification.

5. Traceability: the method captures the key decisions and associated business motivation artifacts to support impact analysis and enable traceability throughout the business process lifecycle.

6. Measurable: as with computerized business process itself, the Oracle BPM engineering process is self-monitoring, providing a feedback loop enabling continuous improvement.

7. Ease of integration with existing methods (adaptability): the Oracle BPM method describes the incremental activities that are needed to extend existing, established methods in both business and IT domains. The Oracle BPM method integrates with existing methods, avoiding the disruption otherwise caused by replacement of established methods and practices.

8. Timely evaluation of IT assets enables effective planning: by predicting downstream integration requirements, gaps in the IT application landscape can be assessed and required enhancements can be specified and coordinated as part of an overall BPM program.

9. Supports complete BPM lifecycle: the full lifecycle of continuous improvement is described by the method providing a starting point for initial process computerization that merges seamlessly with a refinement procedure for continuous improvement.

10. Clear definition of duties: the Oracle BPM method defines roles that can be easily mapped to any customer’s organization chart (potentially exposing gaps in the BPM team composition).

It is important to note that the foregoing list enumerates the benefits of employing a suitable method and supporting practices for the computerization and continuous improvement of business processes; it is not intended to describe the benefits of BPM or associated BPM systems. The benefits of BPM can be found in the ORA BPM Foundation document.

By using the pragmatic and incremental approach described by the Oracle BPM Methodology customers can expect to accelerate their business process automation while reducing the risks associated with the development of any business application. Using the Oracle BPM Methodology ensures the automated business processes meet the needs of the business and eliminates unforeseen outcomes.
Even in circumstances where customers outsource BPM work, the Oracle BPM Methodology still provides these benefits when applied by a suitable governing body.

Finally, with all these benefits available from a suitable BPM methodology, the *ORA BPM Maturity Model* mandates a correctly administered method as part of any maturity assessment and as a requirement for roadmap planning.

### 2.1 Motivation for Adopting a BPM Methodology

Why do we need an engineering method for the business driven activity of business process automation? IT organizations have become very adept at delivering reliable software engineering projects, although historically this was not the case. This change is due in part to the emergence and refinement of the project delivery methodologies that exist today. With an increased need for business and IT alignment and the enterprise scope of most BPM programs the lessons learned in traditional software engineering must be applied to BPM.

The plethora of software engineering methods in common use within IT, tend to be inadequate for BPM because they:

- are code-centric (BPM strives to for "zero-code" implementations)
- are rarely *model-driven*
- lack the steps necessary to effectively capture business process
- overlook business Key Performance Indicators (instrumentation is typically an afterthought)
- do not support continuous improvement (beyond the planned iterations of the project)
- lack vision beyond the scope of a single project

An effective method for BPM must address these inadequacies and support discovery of existing processes while encouraging cataloguing and linking of related assets; separation of business concerns from technical concerns, while fostering business-IT cooperation. Part of that involves establishing a common language and understanding through techniques such as functional decomposition and process leveling. Business automation cannot be specified in a single requirements hand-off (or “thrown over the wall to IT”). Effective cooperation between business and IT is an absolute requirement and an ongoing cycle of improvement, as the process evolves, mandates a continuous dialogue. All these needs must be addressed by a BPM method.

Ultimately BPM will put more control in the hands of business leadership and move IT further out of the equation as we move closer to the idea of the third wave. To do that requires a method, along with effective use of modern BPM tooling, which unambiguously assigns responsibilities and ensures a consistent, repeatable, and effective approach to business process automation.

### 2.2 Objectives for a BPM Methodology

The following lists the key objectives for this methodology:

- Define a consistent and connected framework for delivering BPM projects
- Address the unique aspects and concerns of processes engineering
- Align with business goals and objectives
- Emphasize business community collaboration
2.3 Top-down versus Bottom-up

Historically software projects were either driven by high-level business requirements (top-down) or by opportunities discovered in systems infrastructure and technological advancements (bottom-up). Today’s software engineering methods often recommend the benefits of a compromise between “top-down” and “bottom-up”, commonly referred to as “meet-in-the-middle” approach (e.g. as found in the Oracle SOA Enterprise Technology Strategy). BPM however, must be business-driven and a corresponding strategic (top-down) approach is necessary for the success of any BPM project.

A bottom-up approach, while still valuable in the supporting infrastructure projects (such as SOA Service engineering), is unable to anticipate the needs of the business or even adequately identify enterprise-wide business processes. Taking a bottom-up approach is likely to yield, at best, only technical orchestrations or, at worst, distorted, unrealistic business processes designed to meet the needs of the application systems, not the business.

Business users should be free to express business processes in the way that will most benefit the business and its customers, unencumbered by the constraints of technology. This mandates a top-down approach for business process engineering.

2.4 Scope

The Oracle BPM Methodology is a framework approach for the automation and ongoing improvement of business processes spanning original strategic analysis and process selection through traditional engineering lifecycle phases, technical design to deployment, and looping back via monitoring and process analysis.

The initial phases of the methodology are particularly distinct from traditional software engineering, leaving the majority of this document focusing on business architecture asset identification, strategic analysis, business process selection, and business process identification and discovery.

A number of related disciplines are referenced by this methodology, but are not described in detail here, since they are well established and documented elsewhere: these include business strategy planning and business architecture; business process modeling; SOA Service and other software engineering. Various business architecture and strategic planning artefacts (e.g. strategy maps, value chains, etc.) are identified as potential inputs throughout this document, but the approach for creating them is beyond the scope of this document (and most likely also those responsible for implementing BPM within an organization). SOA Service Engineering can be found in the ITSO document *Software Engineering in an SOA Environment*.

While this methodology is intended to enable business process improvement it is not intended to be a guide to business process improvement itself. Specifically it may be an enabler for *Business Process Reengineering* (BPR), Business Transformation, or
Business Process Change Management; however, these practices are not covered by this guide. Case Management is a relatively new branch of BPM that is not yet formally defined; in fact, it does not even have a universally agreed definition within the BPM community. However, case management is largely a function of modeling and implementation style, with a notable dependency on architecture capabilities and tooling support, and is not significantly influenced by the method. This alternative style of BPM, although it is not called-out for specific treatment in this document, is therefore supported by this methodology.

There are many facets to a successful BPM project. Other texts on the subject typically stress the importance of addressing three the broad categories of “process, people, and technology”. While coverage for each of these key areas of concern can be found in the BPM Maturity Model and associated assessment and overall approach documents, the focus of this practitioner guide is the business process. The technology aspect of BPM is covered in detail in the ORA BPM Foundation and ORA BPM Infrastructure documents. The broad subject of people change management, organizational culture, and human behavior however, are beyond the scope of this document. Guidance for such organizational concerns should be handled instead within the realm of project and program management.

2.5 Business Process Engineering Lifecycle

There are many software engineering methodologies available today, most of which follow the same sequence of analysis, design, implementation, and deployment. In the case of a business process lifecycle, monitoring, while always present in the deployment/operational stage, is critical to BPM. Deployment also plays a smaller role, although the traditional elements of testing, packaging, etc. must not be neglected entirely. For these reasons the business process lifecycle is commonly represented in the form shown in Figure 2–1 below.

Figure 2–1  Simplified Business Process Engineering Lifecycle

The purpose of this diagram is to emphasize the closed-loop that is so important to continuous improvement in BPM. The business process lifecycle tracks the process
from inception through retirement, and includes it’s evolution through new versions created through each cycle of process improvement.

- **Phase: Analysis**

  This phase focuses on the business analysts and end users analyzing an “as-is” process and capturing the issues and challenges facing this process, i.e. delays, disconnects, etc. A business process can be decomposed into several sub-processes, which have their own attributes, but also contribute to achieving the goal of the super-process. The analysis of business processes typically includes the mapping of processes and sub-processes down to activity level. (See functional decomposition later in this document). Once the issues and challenges have been uncovered, an improved “to-be” process is designed to resolve these issues.

  As part of the closed-loop aspect of these phases the second (and subsequent) iteration provides the opportunity to perform “what if” analysis to determine whether or not the process could be enhanced by making changes in the underlying process design and execution phases.

- **Phase: Design**

  This phase deals with taking the documented “to-be” process and translating it into a form that can be understood/implemented by systems. Traditionally, this has involved manual coding of each step of the process, but these manual efforts can now be minimized. BPM solutions have the ability to improve this phase by automating, to varying degrees, the input of process definitions, the use of tools which produce a graphical representation of the desired behavior, and their translation into executable code.

- **Phase: Implementation**

  The new/updated process and associated implementation infrastructure is brought together as a composite business application to be deployed, tested, and promoted to production systems along with training for end-users and IT.

- **Phase: Monitoring**

  This phase is concerned with providing information for both the daily administration of business processes and long term evaluation of process performance and effectiveness. Monitoring is based on Key Performance Indicators established in the early stages of business process engineering. A dashboard presents a graphical representation of the status of executing processes allowing business users to see detailed information about running process instances, enabling them to respond quicker to process bottlenecks that may be limiting overall business operation. Analytics presents performance information for both immediate response and for use in simulations and potentially business performance management systems.

The phases described here, while useful at high-level, are a simplification for the purposes of a complete business process engineering methodology. Analysis and design, in particular, are required in both the business and the technical phases. This detail is elaborated in the following sections starting with a description of the business process state changes occurring in the engineering lifecycle. An introduction to the phases of the Oracle Business Process Engineering lifecycle appears in the subsequent section.

This simplification lends itself however, to a practical organization of major phases for a business process engineering project. For this reason this document has been organized along the basic lines of the simple project lifecycle model, having chapters (loosely) corresponding to analysis, design, implementation (production readiness), and monitoring.
2.5.1 Detailed Business Process Engineering States

The Oracle BPM Methodology defines a number of states that the process must transition through in order to become an executable business process.

*Figure 2–2 Business Process Engineering Lifecycle States*

Identified - The process has been selected for modeling and automated process management.

Defined/Refined - The business aspects of the model have been captured, modeled, and refined to a sufficient level to support this iteration of process optimization. On the first iteration through this cycle the identified process selected for automation undergo process discovery and definition; on subsequent iterations the process, having been already defined, is merely refined.

Designed - The technical aspects of the process model have been addressed including integrations, message handling, interface specification, exception handling, security, transactions, etc.

Composed - All external integrations, message wiring, and security roles have been configured.

Tested - Any software engineering needed to support the process has completed and has been tested along with the process.

Deployed - The executable process model, and all external dependencies, have been deployed into a production environment.

Approved - Final QA approval. All testing and user acceptance has completed.

Operational - The model has been commissioned and BPM system manages execution of the business process.

Monitored - Process metrics being are collected and provide actionable business intelligence.
2.6 The Oracle Business Process Engineering Framework

The Oracle Business Process Engineering Framework describes the Oracle approach to identify business processes, select those suitable for automation, and make them executable. The activities necessary to transition between all the lifecycle states, outlined in the previous section, are described within this framework.

The diagram in Figure 2–3 below shows the major activities in the business process engineering lifecycle. This diagram uses only short names for the engineering activities for brevity in the diagram, but these names are disambiguated with their full names in the following brief descriptions for each activity.

Figure 2–3 Major Activities in the Business Process Engineering Lifecycle

The following sections offer a brief outline of the method lifecycle while subsequent chapters of this document cover each stage in more detail.

2.6.1 Strategic Analysis and Business Process Selection

Successful business process engineering ideally starts by taking business drivers from various potential sources of business motivation descriptions, including business plans, strategy maps, and associated value chains, etc. This approach enables us to identify core business processes and determine their business value alignment in the selection procedure. A tactical approach will also be shown later in the method detail sections that allows selection to proceed when access to these sensitive business documents is not available or in cases where business processes have been identified by some other means.

This activity applies a comprehensive set of evaluation criteria to select the business processes most suitable for automation, identifies the processes we are going to automate, and initiates the business process automation project.
2.6.2 Process Discovery and Definition

In this activity the selected business processes are subjected to discovery, in which many participants collaborate to agree upon a model describing the next level of detail for the business process as it is in its current state (i.e. the "as-is" state). The Process Analyst applies further detail to the business process model, elaborating the model and resolving inconsistencies.

This step also defines the project that supports the engineering activity for the business process (or processes). Project definition includes initial sizing of the effort involved in the automation project.

In this activity the business process project transitions from identified to defined.

In the defined state the major elements of the process have been captured. This includes the flow, its start and end events, business exception paths, the process participants and their interactions, the exchange of data between the process activities, and the interaction with external information sources and application functions.

2.6.3 Business Process Refinement

Refinement refers to the improved version of the business process, also known as the "to-be" state and this is where the process enters the cycle of continuous business process improvement. On entering the loop for the first time (from the definition activity) changes should be kept to a level to strike a balance between the risk of over-ambitious change and demonstration of value through early wins.

Subsequent iterations through the continuous improvement cycle bring greater intelligence about the business process and augmented simulation data from monitoring and analysis. New or changed business drivers may also be introduced in this activity to accommodate changes in the business environment.

In this activity the model is enhanced with details of business rules, business requirements, and underlying application and information services that are expected to support the business process model.

2.6.4 Technical Analysis and Business Process Design

This next major activity in the engineering life cycle includes a feasibility analysis to ensure that the IT applications and systems are capable of supporting the automation as it is currently defined. A gap analysis is performed to specify requirements for extensions to existing IT capabilities needed to support the “to-be” business process model. Approved IT changes initiate software and service engineering projects separate from the business process engineering project.

The business process design describes the details of interfaces and the messages that flow between them, transactions and transaction boundaries, security constraints, and exception paths for system error events.

On completion of this IT focused activity the business process is considered to be in the designed state.

2.6.5 Business Process Application Composition

The Business Process Application Composition activity completes the technical steps necessary to make the business process model executable.

Implementation of the technical aspects of the business process include configuring business rules, Human Task definitions, and wiring Service Component Architecture
(SCA) components to support integration and service orchestration requirements; User Interface (UI) applications may also be constructed.

In this activity the business process advances its state from designed to composed.

2.6.6 Integration Testing

This activity takes a white-box testing approach to ensure the technical integration between the process engine and the underlying application functions, information services, and various other external data and systems, performs appropriately.

After a successful iteration of composition and integration testing the business process enters the state of implemented.

2.6.7 Deployment Planning

Deployment planning packages the composite business application and all its supporting software components and describes the procedures necessary to transfer it out of development. The package also includes operational procedures and end user documentation and training. The first target for the deployment package, prior to production deployment, is user acceptance and potentially performance testing environments.

Once a complete deployment package has been transferred out of development it is said to be deployed, or at least, deployment ready.

2.6.8 Approval

Next we entered the approval process. Here the QA team plans and manages the end user and performance testing. This step is commonly called user acceptance testing (UAT).

2.6.9 Commissioning

Once approved by QA the composite business process application moves into commissioning. This is where the business process transitions into full production operation and hence, the state changes to operational. The process is no longer undergoing testing, but is live in production. This live production state may, at first, be limited to small set of end users (i.e. a pilot or limited production release) during which, aspects of the business process other than its technical implementation may be examined.

2.6.10 Monitoring and Analysis

Monitoring and Analysis support both Operational Administration and Management (OA&M) of the business process in the production environment and business process improvement. The analysis of data gathered from predefined Key Performance Indicators (KPI's) provides critical information to the business analyst to support continuous improvement and continue the engineering lifecycle by re-entering the refinement step.

The final state of the composite business process application, before re-starting the continuous improvement cycle is monitored.
2.7 Incremental and Iterative

One of the primary reasons for putting a business process under the control of a BPM system is to enable ongoing, continuous improvement of the business process. The BPM Oracle BPM Methodology focuses on getting this lifecycle of continuous improvement started by injecting a degree of business architecture to support the identification and selection of business processes best suited to computerized BPM treatment. Once the business process has been automated the engineering cycle is iterated to produce incremental improvements throughout the lifecycle of the business process.

By employing the Oracle BPM Methodology in every iteration of continuous, incremental improvement, successful implementation is ensured. This is achieved, for example, by early identification of IT application changes, simulation with the support of actual runtime data, effective testing, recording change justification for traceability, etc.

2.8 Related Software Engineering Activities

The business process engineering project (abbreviated to “BPM project” here) is defined (or identified) by the selection of a process (or processes) suitable for automation.

During the analysis phase of the BPM project deficiencies in the supporting infrastructure may be identified. These gaps spin-off other engineering activities that are subject to other practices, such as, service engineering, traditional software integration, etc.

Figure 2–4 Ancillary Software Engineering Activities
Development work can be performed external to the process method, as we will discuss later. It may require the instantiation of a new project. These flow lines are shown in dashed green.

In addition to addressing human interaction, represented by the UI elements in the center of the diagram, there are potentially ancillary software engineering activities, as we mentioned earlier, coming out of the technical analysis and design step. This can include most of the traditional IT “integration” functions, such as, service creation, application integration, data access, etc. As we have said already it is important to separate these responsibilities and ensure they are subjected to appropriate software engineering practices (and not left to the discretion of a maverick process engineer). Later we will show an example of integrating these related efforts with the “BPM - SOA composite” where we identify the touch-points and approach to integrating the Service Oracle BPM Methodology (from the SOA ETS).

Another important point to note here is that these ancillary efforts and human workflow developments must be a defined and quantified as early as possible in the engineering life cycle. In order to support effective planning, this information needs to be available during the definition phase.
The success of a business process engineering project often hinges on picking the right process(es) to automate. This is called "business process selection" and the application of a practical, quantitative analysis of process candidates can effectively distinguish those processes best suited to automation and prioritize them for process engineering.

The ideal starting point for business process selection is, of course, a good list of candidate processes. This is known as "business process identification" in which, ideally, candidate business processes are revealed in a strategic analysis of the business. Strategic analysis involves various forms of business motivation documents (business plans, strategy maps, etc.) used in conjunction with functional models to identify processes at the right level of granularity and establish their strategic value.

Unfortunately, the high-level, often sensitive information required by strategic analysis is not always available in a business process engineering project. In such cases a compromise is needed, but still should not be taken at the expense of process selection. The following sections describe the "strategic approach" and an alternative "tactical approach" for business process candidate identification, followed by a procedure to achieve effective process selection with or without strategic inputs.

The ITSO Service Engineering document describes a procedure for “identification and discovery” (of SOA Services), but in BPM these are separate activities spanning analysis and design, so the BPM and SOA terminologies are not interchangeable. Also, some simplistic approaches to process engineering claim that discovery is the first step (see Chapter 4 for more on process discovery), but this results in skipping the crucial “process selection” step. In this engineering method “identification and selection” are clearly delineated from “discovery and definition” to maximize effectiveness in a process engineering projects regardless of scope and scale.

The full process for identification and selection is summarized in the diagram below:
As shown in the diagram in Figure 3–1, process selection can be based on strategic or tactical objectives. These two approaches commonly arise from either strategy-driven (with executive sponsorship in a process-centric organization) or operational (arising from a specific business problem) initiatives; or else they may be distinguished by their scope, e.g. enterprise versus departmental. Either approach feeds into a process of scoring and prioritization, which ultimately leads to process selection.

The strategic approach focuses on business process identification and selection using a variety of tools and techniques elaborated in the following sections. The tactical approach, on the other hand, assumes the business process automation candidates have been selected by some other means and need only to be justified and prioritized for business process automation projects.

The following sections focus on the strategic approach, which includes an introduction to the Business Functional Model and Business Motivation Model.

### 3.1 Strategic Approach

Strategic analysis provides the greatest benefits to selection by identifying processes that are most in alignment with business goals and objectives. Strategic process identification may be the result of adjacent efforts focused on business strategy planning, business architecture planning, or business process re-engineering, but in any form these assets enable a direct linkage to business value.

The strategic analysis approach uses business function models in order to achieve the right level of granularity and to identify functional and process overlap and duplication.
The diagram below highlights the strategic analysis portion of the business process identification and selection procedure using both the business motivation and functional models as inputs.

**Figure 3–2 Business Value Alignment**

Both functional models and motivational models are invaluable to identifying process candidates.

### 3.1.1 Functional models

Functional modeling has played many roles throughout the history of IT and software engineering approaches. It has been applied at the project level as a tool to organize requirements against functional concerns and assist the assembly of viable project architectures. Functional modeling has also been applied at the management level to assist enterprises in understanding current IT capabilities, where they are used to plan and justify additional IT projects. Finally, functional modeling has played a role in organizing concerns for facilitating a business process centric approach to project identification and analysis.

Across these various uses of functional modeling, all share the same hierarchical representation, which makes them fairly widely recognizable and therefore useful to a wide range of stakeholders. The broad adoption of this concept also fosters common approach (although not a standard) between tools and there are many different tools available that support non-cyclic tree structures.

A functional model is a means of representing business functions and processes in a hierarchy. Figure 3–3 below shows the Oracle ITSO levels and their names in this functional hierarchy.
Figure 3–3  Functional Model Levels

Level 0: This level defines the top level business functions of the enterprise, which might include topics such as: Manage IT, Manage Financial Services, and Manage Customer Service, etc. Typically very little detail is required, other than high-level descriptions for the function performed by each child node defined in this level.

Level 1: This level decomposes the parent high-level business functions into groups defining similar processes performed by the high-level business function. For instance, Manage Enterprise Information, Manage IT Knowledge, and Develop and Maintain IT Solutions are all examples of children of the Manage IT from Level 0. Again, descriptive information defining the family of processes should be sufficient at this level for detailing nodes.

Level 2: This level defines individual core business processes within the parent business process group. It includes defining the roles and high-level functions performed by the process. Understand that it is not always necessary to have more than one child for each parent. At this level, the introduction of high-level use cases can be introduced to begin detailing the processes identified in this level. Otherwise, if a business process modeling approach will be used, then high-level business process models (or process maps) can be established at this level.

Level 3: This level breaks down a business process into corresponding Business Activities. Business Activities are the activities performed as part of a business process that may yet be broken down across several tasks. (Think transaction, rather than method here, where the transaction may be executed, as a unit, over several actors). Again, use case analysis is a great way to represent nodes in this level when taking a purely functional approach.

Level 4: This level breaks down Business Activities, into finer grained Business Tasks. Tasks are typically performed by a single actor (system or manual), but may involve many steps. Again, use cases are a valid way to represent nodes of this level, although,
Strategic Approach

Business Process Identification and Selection (Analysis Phase)

at this level, they are more likely to also involve other descriptive documents and diagrams to supplement them.

**Level 5:** This level breaks down a business task, into finer grained steps. This is the lowest level of detail necessary for functional modeling. Typically detailed requirements are all that is necessary at the step level.

The strategic analysis phase of business process engineering focuses only on the functional hierarchy from level 0 to level 2 based on the ITSO definitions above. The practitioner can use the descriptions above to align function and process descriptions and, if necessary, decompose to the next level until a set of core business processes has been identified.

**Handling Requirements:** When getting down to the requirements analysis, groups of requirements can be attached at level 2 and below. Requirements encountered at a parent node cascade through the hierarchy and implicitly apply at the child levels (and grand child, etc.). This is an especially useful method for scoping or establishing a hierarchy of requirements ensuring complete coverage all the way down to the detail level. It is also a useful way to broadcast non-functional requirements rather than repeating them at every node in every child level. For the purposes of strategic analysis this approach can be used to document more granular requirements prior to breaking them down in to finer grained requirements.

Business Functions at level 0, also known as an enterprise process model, initially assist in framing discussions and to get an overview of the business functions across the enterprise or within the domain scoped for the BPM program.

A well established approach to developing and communicating this high level enterprise process model was developed by Michael Porter, known as the Porter value chain. Porter's model is intended to represent a series of functions that create value for the organization, referred to as the “money pipe”, accompanied by a number of supporting functions. See the diagram in Figure 3–4 below for the typical graphical representation of Porter’s model.

*Figure 3–4  Porter Value Chain*
A more recent refinement of the Porter value chain is described by the American Productivity & Quality Center (APQC). Figure 3–5 is a representation of the highest level, generic business functions using the APQC approach.

**Figure 3–5  Example Enterprise Process Model (level 0)**

```
<table>
<thead>
<tr>
<th>1.0 Develop Vision &amp; Strategy</th>
<th>2.0 Design &amp; Develop Products &amp; Services</th>
<th>3.0 Market &amp; Sell Products &amp; Services</th>
<th>4.0 Deliver Products &amp; Services</th>
<th>5.0 Manage Customer Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0 Develop &amp; Manage Human Capital</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7.0 Manage IT</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.0 Manage Financial Resources</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.0 Acquire, Construct, and Manage Property</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.0 Manage Environmental Health &amp; Safety</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11.0 Manage External Relationships</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12.0 Manage Knowledge, Improvement, and Change</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

The APQC model defines a Process Classification Framework (PCF) with 12 highest level business functions that are common to all businesses (industry specific versions of the PCF are also available). These 12 functions correspond to our Level 0 categorizing the key business activities. Five “core” business processes are shown across the top while “supporting” processes are listed horizontally below them.

The APQC is a Member-based non-profit research and education organization whose mission is to work with organizations worldwide to improve productivity and quality.

The APQC believes that a common language to describe organizational processes is necessary to perform measurement and improvement of business processes. In the absence of existing standards the APQC created the PCF claiming:

"Today, the PCF is the world’s most widely used process framework and allows organizations to speak a common language about functions, processes, and activities independent of structure. Updated annually the PCF is organized into 12 distinct categories, including five categories of operating areas and seven of support areas. Each category contains groups of processes and activities that, when considered as a whole, represent the operations of an organization. The frameworks are available in cross-industry and select industry-specific formats."

The purpose of the PCF is slightly different from the Porter value chain in that it classifies business processes in a first step toward "benchmarking", or measurement of their effectiveness, using a database of objective metrics. While the objectives of the PCF and the Porter Value Chain are slightly different they both present the "level 0" business functions in a similar fashion (despite the reversal of horizontal and vertical aspects in the presentation).

The PCF uses the Open Standards Benchmarking Collaborative (OSBC) database of measures for business process benchmarking.
The OSBC is a global initiative to develop a common, standard framework for process definitions, measures, and benchmarks that are available to participating organizations worldwide to improve performance.

The OSBC Research Database has over 1,000 individual metrics and performance drivers covering approximately 65 processes and functions.

The APQC PCF “decomposes” the 12 business functions to level 1 process groups in our function hierarchy. An organization can begin functional modeling by developing its own model down to this level or simply by using the PCF (or one of its industry specific versions) as a starting point.

Once the Business Function(s) have been selected, the next step is to narrow down the focus to the Process Group(s), which have been categorized as a Level 1 in the example levels of the enterprise group. See Figure 3–6 for some sample process groups.

**Figure 3–6 Example Process Groups (level 1)**

Once the Business Process Group(s) have been identified the next step is to decompose these into Core Business Process(es). Figure 3–7 below shows a useful representation for this step and the following section outlines the functional decomposition approach.
3.1.2 Functional Decomposition

Functional decomposition is a technique for developing an understanding of business processes and their level of granularity in an enterprise. The focus for business process selection should be Core Business Processes (level 2), for example “Expense Reimbursement” is a core business process in the example in Figure 3–7 above.

It is important to note that this is NOT an organizational chart. Simply mapping the activities that people perform in their organizational roles will not lead to an effective functional model since many functions and processes have variations occurring in other parts of the organization. This approach requires a top-down decomposition without regard for organizational boundaries in order to get a clear picture of the core processes that span the enterprise.

A true business process model (as opposed to technical orchestration) can now be recognized as representing a Core Business Process. These business process models, also known as process maps at this level, identify the primary Business Activities at the highest level of the business process flow. Further business and technical analysis, in later phases of the engineering process, continue this decomposition into the lower levels of the functional model.

The levels in functional model follow a broadly used approach, however, there is no standard for leveling or naming, but any variation on this theme should help facilitate consistent communications, as long as it is widely adopted across the enterprise.

The functional modeling is commonly used in business and IT for a number of different purposes; some example uses are listed here.

- At the management level to assist enterprises in capturing, understanding, and communicating business functionality
- At the process level as a tool to organize process specifications against functional concerns
Strategic Approach

- At the application level as a tool to decompose and categorize functional capabilities
- In SOA Service engineering as an aid to service identification and discovery

3.1.3 Business Motivation Models

The Strategy Map is a common tool used to articulate and visualize an organization’s business strategy. It is based on the Balanced Scorecard, which promotes the idea of looking at strategy in a balanced way, e.g. across multiple perspectives. In a strategy map, the financial goals influence customer goals, which influence business processes, which in turn influence company assets. The goals across perspectives are linked and support each other.

Figure 3–8 Business Motivation Model

Figure 3–8 shows the categories used as a starting point for the development of a strategy map. Kaplan and Norton developed the approach of categorizing lagging versus leading indicators of business health in their seminal work known as the Balanced Scorecard approach to measurement. These indicators can be seen in the horizontal categories (with lagging indicators at the top to leading indicators at bottom) in the diagram. Strategy Maps are used in conjunction with Balanced Scorecards to represent business strategy categories and their relationships to represent the effect of change.

An example of a strategy map is shown in Figure 3–9 below.
By identifying linkages between candidate business processes and key business strategies in the strategy map we can evaluate the significance of the processes to the business. This technique is referred to as Value Alignment and a representation is shown in Appendix A.

Development of Strategy Maps and other motivation models is beyond the scope of business process engineering, but they are described here as a guide to help the BPM practitioner identify important sources of information as input to the process.

3.2 Tactical Approach

In the absence of strategic business inputs, such as top-level functional models and business motivation, it is still possible to document process selection criteria and to evaluate candidates without the assessment of Value Alignment.

A tactical approach is often taken when getting started with BPM to raise visibility and to gain senior management commitment for future project work. In the early stages of a BPM program it is fairly straightforward to identify core business processes and “low-hanging fruit” for process improvement. In such cases we focus on existing processes that are causing the organization the most pain, e.g. high cost, inefficient, taking too long to complete, inconsistent, low satisfaction, etc.

Business processes identified in this way can now proceed as process candidates to be used in the next step of process selection.

3.3 Process Selection

The Business Process Selection Framework is used to select which potential business process should be realized through automation. Candidate process scoring is not used to decide if new process or functionality should be created, rather it is simply used to decide if the business process should be realized as an automated process. If a process
candidate fails selection for automation it does not reflect on the value of the business process, instead it merely indicates that the business process is not an appropriate candidate for employing BPM technologies.

Now that a number of processes have been identified they can be prioritize based on a scoring mechanism.

**Figure 3–10 Process Selection**

The right-hand side of the diagram in Figure 3–10 is the core of the selection process used to determine which business processes are most appropriate for automation. Business processes can be evaluated regardless of whether they were identified taking the tactical approach, driven simply by requirements, or the strategic approach using top-level functional decomposition and business motivation inputs.

### 3.3.1 Process Candidates Evaluation Criteria

The ITSO Business Process Selection Framework is a spreadsheet tool that can be used evaluate the suitability of business processes for automation, that is, having greatest suitability for execution in a BPM system. The tool also prioritizes business processes as engineering projects. The tool evaluates the business value alignment (in the case of strategic analysis) along with various other benefits expected from automation. Inhibitors to automation are also considered and in some cases a process may be determined unsuitable: for example if a process is too complex (high risk), or is well optimized and doesn’t change, there may be no value in automating it. The following section describes the parameters used for evaluation of business process candidates.
3.3.1.1 Business Process Information

For tracking purposes the following information should be captured to identify business processes throughout the engineering lifecycle:

Name: A unique name identifying the business process candidate.

Owner: This information is used to indicate owning organization or project. The name and title of specific individual(s) responsible for the overall goals, objectives, and decision making should be captured.

Status: The status represents the current state of the process candidate within its lifecycle. The set of available statuses can be customized for a particular business, but the defaults include:

- Proposed: Represents a candidate that has been proposed but it has not yet been justified for decomposition.
- Justified: The candidate has been justified for decomposition
- Not Justified: The candidate failed justification and will remain inactive.
- Assigned: The candidate has been assigned to an owner, but process analysis work has not yet commenced. A process candidate becomes assigned when justification has determined that the process candidate should be decomposed into service candidates and assigned to an analysis team.
- In Progress: The process candidate has been assigned and decomposition work has commenced.

Version: The version field comes into play when an extension to an implemented or in-progress process candidate is required. In this scenario a new version of the process candidate is created. A new version of a process candidate has its own lifecycle, and when created starts out with a status of “Proposed”, which must be justified before it will become scheduled for computerization.

3.3.1.2 Benefit Parameters

The following lists the benefits parameters for the business process candidate evaluation. Classifications for evaluation are provided where scoring is not simply applied on a range of high to low. Scores are assigned to these parameters and weightings can applied as needed.

Value Alignment Score: The value alignment score weighs the alignment of the business process to the enterprise's business and BPM goals. This score is evaluated in the Process Value Alignment worksheet using inputs from strategic analysis.

Process Category Score: The category score weighs the process based on how the process contributes to the business. Evaluation of this score has the following options (high values indicate positively towards justification):

- Core Process: Indicates that the process represents the main business activities of the enterprise. They represent the core competencies that can differentiate the organization in its market. Examples of core processes include:
  - Order Management
  - Account Management
  - Claim Processing
- Strategic Process: This signifies that the process represents the enterprise’s strategic activities by ensuring that the enterprise's meets its specified objectives.
Supporting: This signifies that the process represents the non-core activities that support the core and strategic activities. Support processes guide, control, plan, enable or provide resources for the core processes of an organization. Just because a process is a support process does not mean it should be worked on last. Core process problems are sometimes caused by ineffective support processes. Examples of support processes are:

- HR on-boarding
- Maintenance
- Procurement
- Scheduling

**Executive Interest Score:** The executive interest score weighs executive interest and support. A higher score value indicates positively towards justification. Proposed values include:

- Senior Executives have a strong interest in the results and output from this process and are willing to sponsor and support this initiative.
- An appropriate executive has a strong interest in the results and output from this process and are willing to sponsor and support this initiative.
- Senior Executives have a strong interest in the results and output from this process but no clear indication of support or sponsorship is known.
- An appropriate executive has a strong interest in the results and output from this process but no clear indication of support or sponsorship is known.
- No clear interest in the results and output from this process from any executive.

**BPM Project Score:** The BPM project impact score weighs to what extent a project has been identified, justified, and scheduled for development. A higher score value indicates positively towards justification.

- BPM Project justified and scheduled for development
- BPM Project justified
- BPM Project identified
- No BPM Project identified

**Business Impact Score:** The business impact score weighs the impact automating the process would have on the bottom line of the business by either reducing costs or increasing revenue.

**Improvement Opportunity Score:** The improve opportunity score weighs the opportunity for improving the process once the process is managed and monitored.

**Response to Change Score:** The response to change score weighs the likelihood that the process will require modifications to meet changing business needs.

### 3.3.1.3 Inhibitor Parameters

The following list describes the inhibitors that have a negative impact towards process selection.

**Lack of Structure Score:** The lack of structure score weighs the lack of structure in the process. Sample categories are for evaluation of this score are:

- The process is ad hoc or highly unstructured collaborative process.
The core process is structured, but frequently cases arise that do not follow the structure and instead require ad hoc processing.

The process is mostly a structured process with infrequent exception cases requiring ad hoc processing.

**Complexity Score:** The complexity score weighs the scope and complexity of the process. The process should be small enough in scope and simple enough in complexity to be appropriate for this effort.

**Organization Impact Score:** The organization impact score weighs the extent of reorganization and/or changed organizational policies required for this process. The authority to make these changes must be taken into consideration.

**Resources Involved Score:** The resources involved score weighs to what extent are resources such as employees involved in the process.

**Integration Complexity Score:** The integration complexity score weighs the complexity of integrating the applications necessary to support the process.

**Knowledge Gap Score:** The knowledge score weighs the availability of knowledge about this process.

- There is little or no knowledge about the processes available.
- There is minimal access to knowledge about the process resulting in significant gaps in knowledge about the process.
- There are some gaps in knowledge about the process but the gaps are manageable.
- There is plenty of information available about the process.

### 3.3.1.4 Decision Basis Score

The Oracle Business Process Selection Framework measures alignment with goals, objectives, and initiatives along with benefits and inhibitors to automation. The framework generates a numeric score (Decision Basis Score) that can be used to select and prioritize business processes for automation. Further details of tool can be found in Appendix A.

The collection of benefits and inhibitors make up 13 parameters that are weighted according to their importance in any given customer situation. More parameters can be added or removed as need and weightings can be adapted to suite any specific company’s needs.

The Business Process Selection Framework generates a graph from the Decision Basis Score. An example of this graph is shown in Figure 3–11.
Figure 3–11  Example Process Candidate Analysis Results
Unlike traditional software engineering, design and implementation are not clearly delineated in BPM due to the model-based, or “zero code”, approach. In traditional software engineering, modeling is a design activity while implementation is about producing code that will run within the current infrastructure. In process engineering the business process model evolves through a series of steps and stakeholder input until it is finally executable. This last step of making the model executable is called “business application composition” and it is the closest parallel to implementation in software engineering.

The design portion of this phase is comprised of a number of distinct steps. First, discovery and definition complete a one-time activity to define the business process model needed to introduce the selected business process into the continuous improvement cycle. For subsequent iterations of the continuous improvement cycle, a refinement step is all that is needed to make the business level adjustments to the model that are justified by modeling and analysis (as described in Chapter 3).

A technical analysis and design step follows either the one-time discovery and definition or subsequent refinement steps, to determine feasibility of executing the business process model within the existing capabilities of the IT application infrastructure. Specifically, this step checks that the supporting application functions and data meet the requirements of the process model step and are suitably exposed so they can be accessed by the process engine. In this case a complete portfolio of SOA Services is a significant asset, providing a service contract to describe the service capabilities and interfaces that enable rapid integration into the model in the final implementation step.

In many cases however, gaps will be identified in the business specifications of an existing service or in the technical implementation of application functionality. Technical analysis and design are needed here, not just to specify how to fill these gaps, but also to determine whether it is feasible to implement the business process in this way. In some cases a model will be returned to the previous step to be redesigned to operate with the current constraints of IT application infrastructure; in other cases a service (or software) engineering project is initiated to change or create new services.
The diagram in Figure 4–1 above highlights the activities involved in the design phase of the process engineering lifecycle. These activities are described in detail in the following sections.

4.1 Process Definition and Refinement

The diagram in Figure 4–2 below shows both definition and refinement together in order to identify their separate entry points and explain the interplay between the two steps.
Definition and refinement represent the business elaboration and optimization steps of the engineering process. Definition happens the first time through, and refinement happens at each new version or revision. In definition the “as-is” details of the business process context, process details, and the process model are captured. This initial capture of “what the business does today” needs only to be performed once prior to first automation. In the refinement procedure the "to-be" business process model includes incremental improvements to be carried into the first and all subsequent iterations of the continuous improvement cycle. In subsequent cycles, process activity metrics from real-world operation of the process should be available to help with refinement. At the completion of either definition and/or refinement technical concerns such as technical analysis, test case development, and UI Design are addressed in the next major activity.

The entry points shown in the diagram are from

- process selection into definition where the high-level process model is elaborated (and documented) until a sufficiently detailed model (including exception flows, detailed message flows, etc.) is ready for input to the main engineering cycle
- entering from process monitoring we are already in the continuous improvement cycle and simply need to formulate changes as the cycle re-enters the refinement activities. Inside refinement, simulation and verification activities support assessment of changes to user interaction and associated test cases.

### 4.2 Process Definition

Process definition is the activity in which the details of the core business processes, identified previously in the analysis phase, are expanded in a business process model which is later to become the executable model.

The diagram in Figure 4–3 below separates definition in order to look in more detail at each of the steps along with inputs and outputs.
4.2.1 Capture High Level Process Flow

In the first step of process definition the Process Analyst meets with Process Owner and process participants (the end users who interact with the process and perform process steps) to understand the process from a high level. Typically not all participants are available, or needed, for this step. The analyst captures “the big picture” and as much detail as audience and time permits.

The first thing to be captured is the business process context as follows:

Performed by: Process Analyst

Actions:

- Meet with Process Owner and business leadership, e.g. those familiar with the business context (goals, objectives, rules, etc.) and have an understanding of the high level end-to-end flow of activities.
- Capture process context and high level flow.
- Not concerned with the intricate details of how something is done or why it is done that way - that requires subject matter experts (SME’s), who may not be participating at this level.

Deliverables:

- Business Process Context
- Initial capture of the process model
The model should be captured in an iterative process as a continuation of the functional decomposition that was started in the earlier business analysis phase in support of business process candidate selection. This hierarchical model that is ideally represented by a hierarchical notation (as is the case with BPMN). The benefits of this hierarchical approach are:

■ Entire business process is captured in a single model
■ Model begins with top-level view that depicts the entire end-to-end process on a single page
■ Each activity represents a sub-process which can be expanded to reveal more detail
■ Lowest level includes technical details required for run-time execution

This approach also corresponds to the “three levels of modeling” outlined in Section 1.5

The modeling starts by capturing the highest level process flow that represents the core business process identified in our earlier analysis.

■ Capture happy path & most common flows
  – Focus on manageable portions
  – Use only basic symbols (activities, decisions, split/join, start, end)
  – Work through the flow in iterations, start shallow, then fill in more detail later
  – Use post-its and/or whiteboard; capture results afterward into a modeling tool

■ Identify:
  – What is commonly done, and what decisions are made
  – Who performs the activities, or are they automated
  – Where are they done, and when (business calendar)
  – Who makes decisions, and what business rules / policies / regulations are involved
  – What are the goals, objectives, measurements, and KPI’s
  – Initiating events, business objects, end state

Use a limited pallet of BPMN symbols to develop this business process flow (start, end, activities, decision, and flow connectors). This core business process starting point corresponds to “level 2” in our functional decomposition, but in terms of detail it is best described by “Bruce Silver’s level 1” model.

Tools used here can as simple as a whiteboard or flip-chart in a workshop scenario: forcing people to learn to use a tool can bog-down the analysis. Tools that are too complex are likely to create unnecessary distractions or put-off the business users.

Capture the human and system triggers that unambiguously define the start and end events of the process flow.
The high-level core business process model can be linked to our function model. Now we can proceed with functional decomposition to the "level 3" business activities.

Drilling down a level deeper into our expense processing example we see the consultant completes a expense request, a coordinator reviews the request, and a manager approves it. The expense report is implicitly transferred between these participants and the activities, but the detail of the message flows is not yet developed at this stage in the modeling.

Back to our functional model we check that our business tasks for this activity are showing up in the process model.

As we move through this process of mapping our increasingly more detailed models to the function model we can ensure that process models are maintained at the right level of granularity.

Finally we start to see application functions / services and human tasks (the things that people actually do) appearing in our model along with exception handling and external data relationships.

This is the final stage of decomposition and corresponding to the "level 2" model which is now ready for technical preparations before finally becoming an executable model.
Again we check our function decomposition to make sure we have captured all the necessary business steps and that our model is showing appropriately leveled activities.

Here we see a numbering scheme indicating that our process elements have been decomposed to the last level in the hierarchy “step” (so the diagram encapsulates a “task”). It is also important not to decompose any further (this becomes apparent when a model describes software programming steps rather than steps in a business process).

Activity diagrams of this type can be used to apply filter criteria (e.g. manual steps) to help us pick appropriate operations. It should also be used in conjunction with functional models for grouping (this will support service boundary analysis in the later Delivery activities).

### 4.2.1.1 Create Business Process Context Model

The business process context identifies all the things associated with our business process separate from the concerns of the flow of the process. 

Figure 4–7 shows a simple graphical representation of an example process context model.
The process context model identifies the things we hope to capture as we elaborate the process details in the next step of the definition activity.

### 4.2.1.2 Key Performance Indicators

In the business process context model the goals have been transferred from the strategic analysis, but here they appear as "objectives" which are tangible statement of things that must be achieved to meet the goals. These will later support selection of KPI’s for measurement of process performance.

Capturing KPI’s at this early stage of analysis is not commonly seen in a standard software development project. However, in BPM early definition of these metrics is vital for successful process performance measurement and ultimately feedback into the continuous improvement loop.

More detail in the KPI definition can be developed later in the process, but it is important to capture these high-level statements from the business at this stage.

### 4.2.1.3 Identifying Supporting Applications

The mapping of business functions to applications can be surprisingly complex. The diagram in Figure 4–8 below shows example of a graphical representation of applications to a functional model mapping. This example uses the APQC PCF approach described earlier. This form of diagram will make it easy to determine the application(s) involved in a process as well as where multiple applications happen to perform the same function.
Another example in the diagram below maps applications to the organizations that use them. This model might help ensure that applications that are used in the process are not forgotten during process modeling. Using this model, applications can be checked against the organization unit that uses them and determine if they play a role in the process.
4.2.1.4 Business Objects

Business objects are the pieces of information that flow between the activities in the business process model. In later technical design phase these will become the message flows in the technical representation of the model.

In the earlier example of the expense approval the initial focus was identification of the activities in the business process flow. The next step is to identify the business objects that are necessary to perform each activity and are passed between them as inputs and outputs. These business objects are shown in the lower portion of the diagram requiring details to be completed before moving to the next stage.
An important part of identifying business objects is a mapping of entities to applications. This helps to determine which systems are responsible for the information needed in our business process flow.

**Figure 4–11  Example Entities to Application Mapping**

Figure 4–11 is an example of a “CRUD” (Create, Read, Update, Delete) matrix showing, not only which applications hold the information we need, but also some insight into their level of responsibility for the data, e.g., whether they update or merely read it.

Some business objects are easy to identify, having perhaps just one “system of record”. When it is unique and stored in one place it’s easy to produce a canonical data model. Unfortunately most core entities, such as customer, product, etc., are not so easily defined. Stored across a wide variety of applications, these business objects are more difficult to define. For this reason it is important to start to collect this information early in the business analysis phase, asking the business user directly which systems he/she uses in each activity in the process flow. In this example we see multiple applications read and update product information.

- Some business objects are fairly unique and clearly identifiable, typified by…
  - Single source
  - Clearly defined gold source
  - Master Data Management
  - Canonical Data Model
- Other business objects can be a source of great confusion and frustration…
  - Many unrelated sources
For each business object, attempt to determine:
- What is the primary/gold/master source of the object?
- Who is the authority/steward/owner of that source?
- Are there other sources, and how do they relate to each other?
- How can this object be semantically defined to be unique from other similar objects, and what semantic community does it belong to?

4.2.2 Elaborate Business Process Details

The next step in process definition collects a lot more detail about how the process actually works. This involves interviewing the people that actually perform the process.

Performed by: Process Analyst

Actions:
- Meet with each person or representative of a group that performs a business task or makes a decision as part of the process. Capture the details.
- Watch for issues such as unnecessary risks, delays, challenges, etc.

Deliverables:
- Process (task-level) details
- Enhancements to process model

A practical approach to capture the “as-is” business process is to hold a series of collaborative workshops where key stakeholders detail the business process.

The information we collect can be logically divided into three categories: general information, execution of the task itself, and the involvement of IT resources in carrying out the task. This information is gathered for each task.

General task information
- Description
  - Task name
  - Structured / Unstructured
  - Manual / Automated
- Background
  - Why the task is performed
  - Why it is performed this way
  - Known issues
  - Suggestions for improvement
- Roles
  - Roles, groups, or persons that perform this activity
4.2.2.1 Correct Issues and Resolve Inconsistencies

The final step in process definition is to identify and rectify problems and inconsistencies that arise in the detail definition of the business process. Examples of problems arising include the following:

- Differences between the process as stated by LOB owners and what the workers involved were actually doing
- Inconsistencies between business goals, objectives, business rules, regulations, and how the process is being performed
Contradictory details captured from different participants in the process (e.g. person A said use system X, but person B said use system Y)

Unnecessary steps, churn, risks, delays, duplicate data entries, etc.

Inconsistencies in the way the overall process is carried out, e.g.:
  - One group takes time to capture data that no one uses
  - Two groups are doing the same work
  - Decisions are not being made consistently throughout the process
  - Known issues and suggestions (from task detail capture)

If the issues highlighted are minor, they can be corrected and the model re-published. If there are major issues raised however, it may be necessary to re-convene another workshop session to correct them.

Once the documented business process has been agreed, a decision needs to be made on whether another level of detail is required. If this is the case then another round of interviews/workshops will need to be scheduled.

4.3 Refinement

The original "as-is" process capture was performed in the definition portion of the engineering cycle. This included high level capture, elaboration (task detail capture), and rectification of inconsistencies. The next step is refinement, in which we consider improvements and the "to-be" business process that we will eventually pass into technical design.

The diagram below shows the activities in the refinement process.

*Figure 4–12 Process Refinement*

The activities in the refinement flow are elaborated in the following sections.
4.3.1 Formulate Process Changes

This step determines the improvements to be made in the "to-be" process in the initial automation of the business process and in subsequent iterations of the continuous improvement cycle.

Process model changes may be influenced by many factors including:

- New or changing business goals & objectives
- Known issues and suggestions (from task detail capture)
- Internal policies and business regulations
- Mergers and acquisitions
- Business process re-engineering effort
- Analysis of runtime process metrics / optimization (BAM)
- Changes triggered by business that affect the IT environment, e.g., new applications, cloud / SaaS, automation, etc.

4.3.2 Simulate and Optimize

Simulation models are supported by the BPM engineering tool and maintained alongside the model itself, potentially taking metrics from the runtime system to improve the realism of the model. Initially parameters are defined to attempt to represent the real-world environment for the process. For human tasks this includes worker availability, everything from the business calendar to thinking time for process activities. Similar parameters can be defined for systems. With this information assembled to define the simulation model, the tool executes various scenarios to calculate limits, such as maximum throughput, identifies bottlenecks, and estimates exception rates. All of this can be used to improve the flow and predict the process performance and necessary resource capacities in the runtime BPM environment.

With these details in the simulation model, obvious process improvements can be identified, such as:

- Activities that can be performed in parallel
- Activities that no longer need to be performed
- Duplication
- "Dead end" activities
- Re-ordering the process to improve efficiency
- Activities and decisions that can be automated
- Ways to improve efficiency by creating a better user experience

4.3.3 Final Business Verification

Here the business users review the business process model (ideally using a view within the same modeling tool as the architects and analysts who constructed the model) and its simulations to confirm the model accurately represents their needs.

The review of the process model with all stakeholders should highlight the following:

- Applicable business goals, objectives, rules, and regulations
- Flows, activities, decisions, sub-processes, exceptions, etc.
- Triggering events, inputs, outputs, end state
4.3.4 Document User Interactions

Storyboards can be used for quick and simple illustrations; otherwise, for more detailed definitions screen mock-ups are used. This is the domain of more traditional software development, so we won’t spend much time here, but the interface to the process participants is vital to the success of the BPM implementation.

Ideally these user interactions won’t require entirely new interfaces, but this requires a deep integration with the existing applications that are providing the application services.

4.4 Technical Analysis and Design

Technical analysis involves capturing the technical specifications for the process, determining what functionality needs to be built, and creating specifications for additional IT application functions.

It is up to the organization to determine how the missing pieces will be developed. Options include service engineering, standard software engineering, or ad hoc development.

The process for technical analysis and design is shown in the following diagram.

*Figure 4–13  Technical Analysis and Design*
Technical analysis and design is the first step towards making the model executable (what Bruce Silver refers to as "level 3")

We start with the process definition from process refinement (the "to-be" model) and collect additional technical details to support the analysis in this phase. The gap analysis shown here is very different from any gap analysis that may have been performed as part of the business analysis: here we are assessing the ability of the IT applications and systems to support our business process. Supporting software engineering activities follow to ensure the composition in the next phase has all the necessary IT assets needed to make the business process model executable.

### 4.4.1 Capture Technical Specifications

**Performed by:** Process Architect, with support as needed from Enterprise Architect, Data Architects, and Security Architect

**Actions:**
- Map business process activities and decisions to IT assets
- Capture detailed technical mappings and requirements
- Enterprise Architect can help ensure proper alignment with technology principles, standards, and application mapping
- Data Architects can help define data models, mappings, and translations
- Security Architect can provide insight on applicable security policies, risks, and architecture specifications

**Deliverables:**
- Process model technology level view
- Technical specifications
- Logical data models
- Security analysis findings
To capture the technical specifications, we'll look at different aspects of the process itself, including:

- Triggering events
- Manual tasks
- Automated Tasks
- Process completion
- Process management
- Information
- Security

*Figure 4–14 shows a technical view that can be used to represent the technical considerations for the process. The process model itself appears in the center of the diagram while all the external systems and human actors it touches are highlighted and linked to the process model.*

A business process flow must always be started by a start event (shown far left of the diagram). This can include human initiators as well as any number of system interactions, from receiving a fax to a complex event analysis system detecting some significant condition. The BPM system itself (bottom right) might also start a process based on timer for example.

- What triggers the process to start? Person, application, process, service, etc.
- What transport mechanisms and protocols are used? SOAP, TCP-IP, FTP, XML, JMS, etc.
- What forms of response need to be supported? Synchronous reply, asynchronous reply, no reply, exceptions
- What information is supplied? Data model, content type, schema, entities,…
- What validation needs to be performed on the data?
- What access control restrictions apply? Roles, groups, devices, locations, times, etc.
- How is authentication and authorization performed? LDAP, IdM system, Identity propagation, etc.
- How is identity provided? User name and password, certificate, WS-Security token,…
- What are the confidentiality and integrity needs? Encryption and digital signature requirements?

At the top of the diagram we have the people the process is interacting with. Three types of interaction are represented here:

1. The user on the far left is interacting with a traditional application which is in turn integrated with the BPM system, saving the additional steps (and potentially not changing the way he interacts with the system).
2. A user is prompted to interact with a traditional application, but most likely through a task management interface.
3. The user is involved in an entirely manual interaction with the BPM system, such as approving a request.

Regardless of the type of interaction, there are many details to capture such as:
- Who are the human actors and how are they notified to begin a task? Roles and groups; task list portlet, E-mail, mobile notification, etc.
- What systems will the process interface with to support manual tasks, and what is the sequence of interactions?
- What systems will the user interact with to perform manual tasks?
- What information needs to be presented to the user, and in what form?
- What time-out behavior must be implemented?
- What metrics need to be collected for this task?
- What security requirements apply and how will they be implemented? User authentication, authorization, audit logging, LDAP, IdM, …
- What security requirements apply to the information?
- Encryption, data masking, content protection,…

The backend systems (represented in the lower left of the diagram) are integrated with service tasks ideally using SOA techniques, but potentially also using legacy integration architectures (EAI/MoM, point-to-point, or custom integration). These backend systems commonly have their own, departmental level, workflows and technical processes. In many cases it is not necessary (or technically feasible) to decompose these business activities and take control of the flows in the core business process model. This will be a area for discussion of feasibility and requirements between the business and IT.

- What functions need to be performed?
- What information needs to be processed?
- What are the RASP requirements?
- What are the QoS and transactional integrity requirements?
- What SOA Services are available to satisfy these requirements?
The BPM system itself is also capable of injecting events into the flow, such as alarms or detection of certain conditions. It also collects performance information about the running processes and may even perform real-time analysis triggering certain events or notifications.

- What key performance indicators need to be captured and reported from the process itself?
- How will they be propagated?
- What measures need to be taken to ensure process integrity? Data quality, transactions, rollback, audit, etc.
- What business calendar will the process follow?
- What task or process time-outs need to be configured?
- How are exceptions handled and reported?
- What security considerations apply to the process itself? Data encryption, administrator auditing, etc.

4.4.2 Information Needs Analysis

Information needs analysis takes the following steps:

For each interaction, determine:

- What data (entities, attributes, content) needs to be supplied to the actor in order for the activity to be completed or decision to be made?
- Where will the data come from?
- In what form will it be presented? XML, Object, HTML, .DOC, etc.
- What logical data model(s) should the process adhere to?
- What aggregations, mappings, and transformations are required to produce the desired presentation format?
- What data is returned?
- What state change(s) have occurred?
- What systems need to be updated / notified / synchronized?
- Are there any transactional requirements?
- What mechanism will be used to move data?

Review each of the above with Data Architects and Data Stewards.
Figure 4–15 Logical Data Model

Figure 4–15 shows an example of the logical data models that may be associated at the various different functional levels of the enterprise. Each data source inherently has its own logical model. In some cases a higher level model has been defined, perhaps via MDM or data integration efforts. If such a model exists, then processes within the scope of that model should use it. If no higher level models exist, then a logical model should be defined for the process. This is preferred over adopting the source model in that it avoids propagating the source model beyond its intended scope.

In the diagram each of the logical models shown belongs to an application. In a business process that typically spans applications it is important to be aware of data dependencies that span applications since this demands specialized data services that may not already exist.

Some organizations have an enterprise-wide canonical data model (shown at the top of the model). The enterprise data model is rare, but if available it should be used as much as possible in defining data needs for the process.

In the case that data models are not already defined it is often better to start with an industry standard model (e.g. HR-XML for HR systems, HL7 for health care, etc.) rather than an application specific data model.

Oracle’s Application Integration Architecture (AIA) for example defines canonical models, independent of the applications, but with translations and transformations for interaction with the applications.

Whatever level(s) the logical model exists, our technical solution may need data transformation services to exchange data with the applications.

4.4.3 Security Needs Analysis

One of the problems encountered in business process automation lies in bestowing the authority to initiate the application functions on the BPM system. This authority must be maintained and propagated securely to the applications where credentials are commonly not uniform between systems; worse still the authenticating mechanisms may be different.

Use these steps to determine the scope of the security issues and formulate appropriate solutions.

Identity and Access Control:
- Determine mechanisms used to authenticate users that interact with the process
- Determine mechanisms used to authenticate system interactions
- Map identified roles / groups / people to identity management system entities
- Establish access control policies for process triggers (start and resume points)
- Determine authentication and/or identity propagation strategy

Integrity and Confidentiality:
- Review security classifications of data and/or content used by the process
- Determine security risks to information based on network exposure, location, etc.
- Identify encryption requirements to ensure required level of confidentiality
- Identify data masking requirements based on data classifications and user access levels
- Identify process and data integrity requirements that might necessitate digital signatures and/or non-repudiation
- Determine key management strategy
- Identify and address risks introduced via process execution such as persisted state, cached data objects, audit logging, and cluster network traffic

Finally, review the security specifications with Security Architect.

Originally the “swivel-chair business process” involved a human participant login-in to multiple backend application systems. To provide equivalent access to the BPM system a number of solutions are available. The simplest solution is a centralized “single sign-on” system that authenticates the BPM system to all backend applications. However, it is sometimes necessary to provide access based on the status of the human initiators and participants and this may even influence the flow of the process. In these cases it is necessary to propagate the participants credentials or security tokens.

Confidentiality can often be delegated to a content management system (CMS), but again it is necessary to appropriately identify the user involved in the activity to enable the CMS to provide the right amount of information.

### 4.4.4 Gap Analysis and Technical Feasibility

The gap analysis is also known as a “technical feasibility” study since the supporting services, information needs, or application integration may not be available and the cost or delay to develop them may be sufficient to return the business process back to the previous phase to be reworked (this exception flow is not shown in this diagram).

Performed by: Process Architect

Actions:
1. Locate existing SOA Services, applications, interfaces, etc. that match functional and data needs
2. Compare non-functional requirements versus needs
3. Determine:
   - Which existing assets are an exact match for the process
   - Which assets are a partial match (functional or non-functional)
   - What capabilities are not currently provided
4. Contact SOA Service / application owners to discuss usage of existing assets and feasibility of revisions where a partial matches are found

Deliverables:
- Discovery and gap analysis findings

This is where we are figuring out what functions need to be built or accessed to support each task in the business process model versus what currently exists.

If you are implementing BPM over a SOA foundation it should be relatively simple to line-up the functional models used for service engineering along with service contract descriptions to determine where services meet the requirements of the business process tasks.

SOA Service discovery methods include repositor search, examination of Functional Model, and examination of service contracts. Other forms of application functions can be found in Application Functionality Matrix and a Data Entity Matrix.

What does the process require that needs to be developed?
- Partial Match: Revisions to existing SOA Services, application integrations, etc.
- No Match: New functionality to perform automated tasks
- Software to support user interactions and devices
- Means to integrate the process with existing systems, other processes, applications, etc.

For each gap identified in the technical application and information support we define software requirements and perform the necessary software engineering activities to fill these gaps. This method does NOT define how these other engineering activities should be performed: they may be fulfilled through SOA Service engineering, for example, or by any other established method and integration architecture.

4.4.5 Define Software Requirements

The process architect defines the requirements for the missing functionality / integration for each identified gap. These requirements might take the form of use cases, “stories”, etc. according to the software methodologies employed by the IT organization.

Performed by: Process Architect

Actions:
- For each "gap", create specifications needed to support development of missing functionality
- Include functional and non-functional criteria
- Form of documentation can be whatever the development team can best work with, e.g. use cases, requirements list, storyboard, etc.

Deliverables:
- Software/service requirements documentation

4.4.6 Filling the IT gaps

The preferred approach to “filling the IT gaps” is SOA Service engineering because it lends itself to optimal integration with the BPM method. SOA Service engineering
Capabilities even appear in the maturity model used in the assessment of a BPM maturity.

In reality SOA may not be used, or is not sufficiently matured in an organization, so other legacy strategies, such as EAI, may need to be used.

The last resort is ad hoc development. This never results in a long-term sustainable solution and should be avoided.

**Service Engineering Option**: All needed functionality is fed into a service engineering process where proper service justification, classification, development, testing, and provisioning are performed.

**Software Engineering Option**: Formal software development process, used either when service-enablement is not justified, or when the service engineering option does not exist.

**Ad Hoc Development Option**: Software is developed by the process team strictly to satisfy requirements of the process.

The resulting software may include:

- New automated functions not performed by existing applications
- Enhancements to existing applications to support the process
- Custom code to integrate the process with existing applications

### 4.5 Business Application Composition

The goal of BPM in any IT environment should be to get to the point where business processes can be composed (and re-composed) by the Process Developer without the need for the intermediate step of software engineering. SOA will certainly help in realizing this goal, but the current reality is that many business process model tasks will initially require some additional IT intervention, or may even pass basic feasibility tests.

Ultimately this activity should be sufficiently non-technical that we, at least move toward the utopian state described in the Third Wave, and start to eliminate IT involvement in business process engineering altogether.

In any case task configuration is the final technical step needed to make the business process executable.

Task configuration is a graphical and declarative activity (i.e. no procedural programming) for wiring services to tasks, specifying the attribute level details of the message flows and data mapping between tasks, KPI’s, etc.

Performed by: Process Developer

**Actions**:

Configure the process in accordance with the technical specifications defined by the Process Architect

Configuration may include:

- Importing interfaces to SOA Services and other external points of integration
- Importing or configuring data sources and logical data models
- Message mapping, transformation, and validation
- Technical component wiring and exception handling
- Defining transaction boundaries
- Configuring process and task time-outs
- Setting up alerts and notifications
- Declarative encoding of business rules
- KPI instrumentation
- Configuring process security
- User Interface / worklist configuration
- Any remaining task / role assignments and business rules configuration, etc.
- Unit testing

Deliverable:
- Configured process, ready for testing

These activities are represented in the process model below.

**Figure 4–16 Task Configuration**

The resulting assembly of the business processes with application services integration, business rules, KPI’s, worklist applications, etc. is now a composite business application ready for testing and other preparations prior to live operation.

### 4.6 Unit Testing

Unit testing in software engineering refers to testing (usually by the developer) of individual software modules, typically the class level in object-oriented programming. In BPM this equates to testing at the process step level, or a collection of steps making-up an activity, by the Process Developer. This level of testing simply ensures that the individual process steps (or activities) execute, within the constraints of the development environment, as specified by the requirements. It is unlikely that anything but the simplest of business processes can be tested from end-to-end at this stage in the engineering process. Testing of the complete process can only be effective after all points of integration have been tested (see integration testing in the next
chapter) and the complete composite application is deployed to a near-as-possible real-world environment.

Unit test are usually written by the process developer using full knowledge of the working of the code and its environment (this is known as white-box testing). These tests are constructed to ensure the process step acts appropriately on the possible range of inputs it can receive and produces an appropriate result. This activity focuses heavily on testing the handling of exceptions to ensure corner cases (that is, inputs and other stimuli outside the anticipated boundaries of normal operation) and all available branching paths are executed.

Use of testing framework, such as JUnit, can provide significant benefits, including repeatable tests leading to more effective regression testing, and systematic development of a catalog of tests providing opportunities for later system testing and even load simulation for performance and scalability testing. Test Driven Development (TDD) may also be applied in which automated tests are derived from the business and technical requirements (and may even become part of the requirements) prior to any development of the process steps.
Unlike the earlier phases of BPM engineering, the final stages of testing through commissioning do not deviate significantly from traditional software engineering; therefore, this chapter provides only a brief description of production readiness procedures while highlighting areas of particular interest to BPM.

The procedure for taking an application from development to a live production environment is commonly summarized under the heading of deployment. Deployment however, refers only to the part of process in which the application is prepared for transfer from the development environment, in a complete, integrated package, and made ready for use in a run-time environment. Initial uses for the deployment package and associated run-time environments, include various levels of testing and approval prior to the ultimate production go-live. Numerous other activities are required along the way and the complete procedure for production readiness of a composite business application should include the following steps:

1. Integration and system testing
2. Deployment preparation
3. Acceptance testing and approval
4. Commissioning

There are many variations on these basic software engineering themes, but the sequence is particularly important to maximize the effectiveness of testing and minimize rework when problems are discovered. For example, integration testing and system testing (also known as requirements testing) should not require a complete end-to-end working composite application and can, therefore, be completed before deployment preparation, eliminating the need to repeat deployment when problems are found in these testing steps. On the other hand, acceptance testing, which may include performance and load testing, requires a complete deployment to a near-as-possible production environment.

Testing clearly does not occur exclusively in the "testing phase", but this is where the majority of testing effort is focused and typically a QA team comes together to design and execute these tests. More detailed descriptions of these testing steps appear under the relevant phase sections later in this chapter, but a summary of common test types relevant to BPM is listed here by phase:

<table>
<thead>
<tr>
<th>Table 5-1 Test Activity by Phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase</td>
</tr>
<tr>
<td>Refinement</td>
</tr>
</tbody>
</table>
Rather confusingly "integration testing" is often called "system integration testing (SIT)" and "system testing" is often referred to as "requirements testing"; these may also be combined under the heading of "system and integration testing".

It is also important to note that the cost of testing, in terms of equipment, personnel, and coordination increases as we move through this list and, as with any software engineering project, it should not be necessary to apply every test in its entirety to every change. BPM by its nature is also particularly prone to change due to its strategy of continuous improvement and core objective of business agility so we need to determine the scope of a release, categorize changes, and assess a risk level for every type of change in order to apply testing efficiently.

Some of the key issues for business process engineering arise from the broader involvement of business participants and the ease with which major changes can be made using current BPM tools. Changes to business process flows can now, theoretically, be made by business Process Owners and Process Analysts without support from IT. However, IT must still ensure changes are not allowed to flow into production in an uncontrolled manner. This is largely a concern for BPM governance (see A Framework for BPM Governance for more information). Governance first requires a complete and consistent method for BPM engineering upon which to define its policies and procedures. So, although governance is beyond the scope for this discussion, this document is an important part of the foundation for BPM governance.

Business process engineering raises a number of new governance questions that require us to assess the potential impact of changes, specify which roles should be permitted to make changes, and what testing and acceptance criteria should be applied.

One example of a BPM specific concern is testing of business rules: should we test and approve every business rule change or should we limit the scope of business rules to minimize the potential impact to a running system? Is it simply sufficient to test business rule changes in a simulation environment? In extreme cases there are modifications (refinements) that will trigger a software change: this is likely to trigger both traditional software testing and separately business process testing and approval.

The first step in an approach to production readiness is to identify all the different types of changes that can arise in a BPM system. The question of roles and responsibilities can be deferred to the governance framework. The following list is a

Table 5–1 (Cont.) Test Activity by Phase

<table>
<thead>
<tr>
<th>Phase</th>
<th>Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Composition</td>
<td>Unit Testing</td>
</tr>
<tr>
<td></td>
<td>Regression Testing</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Testing</td>
<td>Integration Testing</td>
</tr>
<tr>
<td></td>
<td>System (requirements) Testing</td>
</tr>
<tr>
<td></td>
<td>Process Collaboration Testing</td>
</tr>
<tr>
<td></td>
<td>User Interaction and Usability Testing</td>
</tr>
<tr>
<td>Approval</td>
<td>User Acceptance Testing (UAT)</td>
</tr>
<tr>
<td></td>
<td>Performance and Load/Stress Testing</td>
</tr>
<tr>
<td>Commissioning</td>
<td>Operational Testing</td>
</tr>
<tr>
<td></td>
<td>Availability and Disaster Recovery (Business Continuity) Testing</td>
</tr>
</tbody>
</table>
suggested starting point for categorization with some examples of changes with varying potential impact to a production system.

- **Roles**
  - Addition of end user participants
  - Addition of new swim-lanes with new flow path
- **Key Performance Indicators**
  - Add new, modify, remove indicators
- **Business Rules**
  - Add new rules to existing rule chain
  - Adding input or output arguments
  - Changing threshold values
- **Model and business process flow**
  - Further automation of existing business processes
  - Creation of new business process
- **Activities**
  - Addition of service task using well established (SOA) service
  - Call-out to new or non-SOA service
  - Human task workflow
- **Events**
  - Addition of timers
  - Business exceptions
    * Addition of exception paths and activities
  - System errors
- **System interfaces and integration techniques**
  - Changes to underlying architecture (e.g. introduction of SOA)
- **Underlying application code**
  - Additional custom code
  - Updates to packaged application
  - Packaged application replacement
- **Other system considerations**
  - Infrastructure upgrade
- **Change in business conditions**
  - Anticipated increase in sales volume
  - Introduction of new business partners
  - Mergers and acquisitions

Considering the diversity of this by-no-means-complete list it is immediately clear that we not only need to form complex rules about how changes flow into production, but we also need a comprehensive impact analysis mechanism. Changes to business
process could impact the operation of IT systems while changes in the IT applications and infrastructure could have significant up-stream effects on business processes (if they are not sufficiently decoupled).

Some of the items in this list seem fairly innocuous, for example, “addition of end user participants” is unlikely to have much effect on the business process flow; however, increasing the number of participants (e.g. rolling-out a pilot automation across the enterprise) is likely to increase the load on the BPM infrastructure and the underlying systems.

KPI’s, on the other hand, have no bearing on the infrastructure or even the business processes itself. Instead changes to KPI’s affect the measurement of the business process and so the impact to the analytics, particularly trend analysis and long-term performance measurements, must be considered carefully. KPI’s may fall outside the scope of testing considered here, but the potential impact to existing analysis and reporting activities must be traceable.

By using a categorization scheme of this type, a matrix can be developed specifying the level of testing required by every change type within a release. The following table shows a simple example of such a matrix using the testing types and some sample change candidates from the categories we have discussed here.

![Figure 5–1 Impact of Changes on Testing](image)

Another consideration unique to BPM is the introduction of long-running business processes to the IT environment. Some business processes span days or even weeks, but in traditional non-BPM environments IT maintenance schedules work around business user needs and these users manage the flow of the business process between systems outages. In a BPM system, business process states must be carefully maintained between outages, upgrades, and most interestingly between business
process changes. In these cases a policy must be established to ensure consistent handling: should all long running processes be quiessed to an end state before changes are applied, should prior versions of a business process be allowed to run to completion after a change, or should an “upgrade” process flow be necessary to transition in-flight processes from an earlier version? The answers to these questions will ultimately be unique to the organization, so the important thing is to consider them carefully in advance and develop well-communicated policies to ensure consistent handling of all eventualities.

The following sections explore these major phases of the business process engineering lifecycle that contribute to production readiness in more detail.

5.1 Testing

This phase encapsulates the most rigorous testing activities and is commonly the responsibility of a separate Quality Assurance (QA) team in collaboration with the developers, architects, and analysts. Here a test plan is created along with a set of test cases for functional and non-functional testing. The various testing types are outlined in the following sections.

5.1.1 Integration Testing

Since a BPM system is fundamentally concerned with managing process flow between business process activities, both system and human, it necessarily involves significant integration with the systems and applications that provide the underlying functions. These functional assets may be geographically dispersed and technologically disparate. Similarly the architectural strategies used to perform the integration may also involve multiple approaches. All the integration points arising from various combinations of applications, technologies, and geographies must be tested in the context of our composite business application before proceeding with the next steps of deployment packaging, acceptance testing, etc.

Integration testing in the context of BPM is required to test all the interfaces between the business process flow and all its touch-points. These touch-points include:

- Business application functions
- Data and content
- User interface applications involving human participants
- Other external events

Each of these touch-points may use various integration technologies and architectural strategies, such as SOA, MoM, EAI, or proprietary mechanisms. These combinations must all be tested to ensure consistent operation in the final composite business application.

A best practice in integration testing is to create a separate "test plan" for each integration. This includes interaction diagrams and test cases that covered the interactions. This approach isolates changes to backend systems, compartmentalizes the integration testing, and ultimately makes regression testing easier.

5.1.2 System Testing

Also known as "requirements testing" this activity validates the functional capabilities of the BPM composite application against the business and technical requirements specifications.
5.1.3 Process Collaboration Testing

Process Collaboration Testing is an extension of system testing that is unique to BPM. Process Collaboration refers to the interaction, via message flows, between processes both within "pools" and choreographies between "pools". That is to say interaction between business processes that execute within the same environment as well as those communicating between separate environments.

The focus of collaboration testing is ensuring messages that flow between business processes are acceptable to all participants in all cases.

5.1.4 User Interaction and Usability Testing

User interaction is primarily concerned with the interaction between the BPM system and the end user i.e. human process participant. Other user interactions that may be tested at this stage include process monitoring and administration interfaces as well as KPI analysis when custom solutions have been developed.

User interactions may be handled through a worklist manager or a simple web interface integrated with underlying applications.

5.2 Deployment Planning

Before proceeding to the various end-to-end tests and final go-live, the composite application must be prepared to be transferred efficiently and completely to real-world environments. The environments we refer to are ultimately live production platforms, but prior to go-live there are various testing environments needed to facilitate stakeholder approvals.

Deployment is comprised of the following activities:

- Release planning
  
  Software release planning is usually represented by a document describing a policy for organizing the production of regular software releases. It describes an approach for iterative software releases including a versioning scheme, milestones, release frequency, supporting practices, and tools to be used.

  In the case of BPM the regularity of business process releases is unlikely to be as predictable as traditional software, however, some aspects still apply. Versioning, in particular, along with its associated practices and tooling, is critical to the BPM release process.

- Version tracking

  A versioning scheme identifies what is running in the production and testing environments at any given time and supports problem analysis, migration, deprecation, and retirement.

- Document dependencies

  A full description of the dependencies for the composite application being deployed is necessary to ensure the suitability of the target environment. For example, specific versions of underlying applications and a named list of associated services should be included. A full dependency description not only defines the operating environment, but also enables impact analysis when dependent systems are being considered for update, replacement, or retirement.

- Preparation for use by end users

  - User documentation
- Training

- Packaging

Packaging is a central concern for deployment. It identifies the constituent parts of a deployable composite along with the mechanisms for building, updating, and unpacking the package.

- Development and documentation of procedures for
  - Installation and activation of the package.
  - Installation reversal (in case of problems).

- Upgrade procedures

- Deactivation and decommissioning

### 5.2.1 Factors Influencing Process Deployment

A key concern for BPM composite application deployment is cohesiveness of the packaging. Some BPM systems may have the technical capability to promote certain types of changes to a running environment (including production) while the system is fully operational without interruption to in-flight processes. These changes may include roles, business rules, KPI’s, and even the flow of a business process. Other systems may simply require the deployment package to be broken into separate parts (when deploying to disparate systems for example). Either case must be carefully managed.

### 5.2.2 Process Migration

When a new version of a business process is being provisioned migration procedures must be developed to specify how the transition should be handled. In the simplest case all in-flight process instances would be allowed to run to completion while all new instances run the new version: perhaps a time limit is determined for the retirement of the previous version and a process administrator is required to intervene to complete or re-run outstanding instances. The ability to concurrently run multiple versions in this way is, however, dependent upon the capabilities of the BPM system.

Another consideration for process migration would be user training in cases where a change to a business process impacts the flow of human tasks or the system’s interaction with human participants.

### 5.3 Approval

Approval simply means to get agreement to put the composite business application into a live production environment, which implicitly involves various forms of acceptance testing to verify its readiness.

Various stakeholders have different concerns that result in different types of testing. Tests are categorized into functional and non-functional testing and the test types that commonly apply in business process engineering are as follows:

- Non-functional testing
  - Security testing
  - Usability testing
  - Performance and scalability testing
- Regression testing
User Acceptance Testing (UAT)

The performance and scalability testing is also called load testing or stress testing, however, it is important to note that this non-functional test at the approval stage is very different from any form of load testing that may have been performed in a design-time simulation tool. Design-time workload simulation typically attempts to predict the workload of non-system assets e.g. human workers, delivery trucks, milling machines, etc. At this final approval stage, with the benefit of a complete functional system, performance testing focuses on the ability of the system assets (i.e. processors, applications, rules engines, etc.) to operate at the workload levels predicted by the business requirements.

5.3.1 Best Practices

A SOA strategy underpinning the BPM architecture provides an isolation layer separating the BPM platform from the source systems that provide application data and functionality. This “loose coupling” can be exploited during testing to simulate application functions and avoid the need for a complete implementation of all enterprise applications for the purpose of functional testing.

Non-functional testing, in particular performance testing, does require as-near-as-possible replication of all production platforms involved in the business process. Some simplification can be achieved by establishing the performance characteristics and load limitations of individual applications and systems and using the data to augment the business process simulation model.

The practice of running test "streams" within a production environment incurs substantial risks and should be avoided.

5.4 Commissioning

In software, as with any engineering practice (e.g. building, industrial plant, etc.), commissioning is defined as the process of assuring that all systems and components are installed, tested, operated, and maintained according to the operational requirements from the business owner. Once again testing and assurance appear outside the scope of the core testing owned by the QA team. In this case testing is required to ensure that the operational procedures perform according to requirements.

This phase may also involve installation of additional hardware and infrastructure software based on sizing specifications from the architecture team.

Operational procedures take the form of documents (“the ops manual”), scripts, and an operations management framework (tool suite) capable of incorporating new dashboards and operational information. The standard operational procedures relevant to a BPM system include the following:

- Start / Stop
  Procedures for cleanly starting and stopping process instances, queues, engines, and hardware platforms for maintenance, upgrades, etc.

- Escalation
  Describes how to respond to problems that may arise, such as stalled queues, systems failures, etc.

- Backup and restore
  Backup systems must ensure transitional consistency for business processes during data or system recovery.
- Business continuity plans

  Specifies how to recover business services spanning all forms of service disruption from minor hardware failures to full-scale disaster recovery.

  The acceptable duration and other parameters (loss of data, business opportunities, etc.) for maintenance cycles, problem resolution, and disaster recovery should all be specified by Service Level Agreements (SLA’s).

  In this phase the operations team usually takes responsibility for the deployable software package and works with the process engineering team to execute installation and upgrade procedures. Typically these procedures are tested during the commissioning of the test systems which requires careful planning since the test systems are a prerequisite for the testing phase.

  Commissioning of the composite business application may also involve setting up support, such as, training for both end users and helpdesk personnel.

  Ultimately the purpose of commissioning is to create a secure and effective environment to enable the OA&M (Operations, Administration, and Management) team to maintain continuous operation of the business applications. The Administration and Management of OA&M is covered in the next chapter of this document.
6

Business Process Monitoring

Keeping in mind this document is concerned with "methodology", this chapter is not intended to be a recipe for monitoring business processes. Rather its focus is what we should expect from business process monitoring and what we need to do to achieve those objectives.

Although monitoring finds itself at the end of our cycle (and the end of this document), it is important to recognize that there would be nothing to monitor without the earlier steps of establishing requirements for Key Performance Indicators (Definition Phase) and attaching them to the model (Technical Analysis and Design). Since activities of selecting and attaching KPI’s have been covered previously they are not revisited here and this chapter focuses on what little remains for an engineering team to do to ensure that monitoring (and analysis) provides the fullest benefits to the business community.

6.1 Benefits and Opportunities

The mantra of today’s BPM is "monitor, manage, and optimize" and key differences between BPM and traditional software engineering include the empowerment of business users and associated enablement of continuous improvement of the business processes. These benefits are achieved largely through monitoring providing both real-time visibility and data to support effective analysis.

The purpose and broad benefits of monitoring in BPM are summarized in the following list:

- Conformance: The ability to operate within predefined values
- Performance: The ability to measure performance and gather the information necessary to improve
- Better understanding of Business Process Effectiveness and an enabler for Root Cause Analysis
- Competitive Advantage: The right information collected for the business allowing management to make better business, IT, and operational decisions
- Understanding business value and providing justification for strategic changes
- Basis for automated responses to anticipated operational issues

The intrinsic role of monitoring in BPM presents three major opportunities to improve support to the business which must be carefully considered when defining the method for business process engineering. These three opportunities are described in the following sections.
6.1.1 Business Process Administration

Business Process Administration is a category of operational management related to IT's OA&M, however, it requires specialist business knowledge and should be the responsibility under the business owner. For this reason in particular, the information should not be confused with system information (such as server resource utilization) and must be presented, along with the necessary controls, in an easy-to-use dashboard.

The key objective of process monitoring for administration is to monitor health of business processes and respond to alerts and exception conditions. The category of exceptions we are primarily concerned with here is unhandled business process exceptions in which the process flow has not accommodated particular business scenarios: this may be by design, leaving rare, complex circumstances to drop-out for human intervention, perhaps with the intention of providing further analysis of these situations before designing a computerized flow to handle them. The other category of exceptions, which must be separated from business exceptions, is system errors and exceptions. Examples of system exceptions include server resource exhaustion and software errors: these conditions should be intercepted by IT administration or else immediately escalated by the business process administrator.

Business processes can be better managed during runtime when monitoring supports effective response to alert conditions. Common scenarios in this category are:

- Unhandled exceptions
- Stuck or stalled processes
- Excessive queue length or processing times

Business process exception monitoring enables the business users to proactively detect exceptions and initiate procedures to resolve issues. Administrators can also identify common exceptions, such as, a repeated ordering mistake with particular supplier, and may respond by pausing the affected processes until resolution is reached.

In addition to defining approach for responding to alerts, the business process engineering method should also identify the need for escalation procedures. The role associated with process administration, in addition to handling a variety of situations, must also be responsible for triage of issues to other support organizations, such as, unhandled business process conditions to customer support or system exceptions to the IT helpdesk.

Another opportunity for monitoring within business process administration should enable the BPM system to dynamically change the behavior of a business process instance, thereby providing an automated response under carefully predefined circumstances. An automated response requires the system to modify the execution of a process instance based on real-time, higher-level analysis, potentially combining other Business Intelligence (BI) gleaned through Business Activity Monitoring (BAM) and/or Complex Event Processing (CEP). This approach may be the ultimate resolution to the rare, complex cases where the process flow drops-out to the administrator via an unhandled business exception because the information necessary to complete the flow is not directly available to the BPM system or the process model designers. An example automated response to an unhandled exception might be an alternative handling of credit card transaction risk assessment when the in-house credit card validation system is unavailable.

Modifying process flow at runtime provides greater agility in the business process by enabling the business users to dynamically change process behavior without redeploying the process. Also, business rules are used to decouple decisions from...
process flow while BAM allows users to modify rules based on real-time analysis of broader business events and influences.

In the case of business process administration the method must make provisions for both detecting and reporting these conditions as well as specifying the administrative role that responds to them. The associated administrative duties require access and authority to make appropriate runtime changes to individual process instances.

6.1.2 Continuous Business Process Improvement

This second category aggregates and analyzes data from KPI collectors to monitor the performance of business processes (rather than individual process instances) to identify optimization opportunities. Data collected through monitoring and its off-line analysis supports the following activities:

- Enhancement of business process simulation models to ensure more realistic analysis
- Discovery of bottlenecks to improve business process efficiency
- Quantifying the benefits of incremental improvements

This category of activities supported by monitoring is concerned with analysis of as-is business processes running under BPM and associated incremental improvements. This is distinct from major process change (see next category) and commonly the responsibility of the Process Owner. Again, the method must provide for extraction of the necessary data and the chain of activities that deliver it in a usable form to the Process Owner, while also empowering that user to make process improvements.

6.1.3 Supporting information for Business/Enterprise Performance Management

EPM is a topic that is beyond the scope of business process engineering (although some may group it with Business Process Management); however, it is important that a business process engineering method recognizes and accommodates this valuable category of data collection and its preliminary analysis, required to support major process change. This case requires executive level authority and design, commonly involving Business Activity Monitoring and other Business Intelligence techniques, in which analytical data from BPM is just one source input.

The collection and analysis of business process performance information may be much the same as in Section 6.1.2 above, but its target users and their analysis (and associated tooling) are quite different.

The process engineering method in this case, having little to do with the use of the data, must merely ensure that the appropriate data to support this activity is collected and transmitted accordingly. This is fundamentally a case of collecting requirements from executive level management and ensuring that corresponding indicators are applied to the executable process model.

6.2 Business Process Engineering Activities

As we have already stated, the primary enabler for business process monitoring, establishing KPI’s, has already been covered in earlier steps in this method. Also, much of what is done with the intelligence provided by the monitoring system is beyond the scope of an engineering method, while the challenge of providing it is largely an architectural concern. All that remains for a business process engineering effort at this stage is to identify the roles and their responsibilities to ensure effective
implementation of monitoring technologies and ongoing analysis and application of the data it produces.

The participant roles along with their concerns and associated activities are summarized in the following list:

Executive business leadership

- Identifies business KPI’s
- Real-time business insight and Business Activity Monitoring in combination with other Business Intelligence
- Ultimately supporting a broader Enterprise / Business Architecture enabling Business/Enterprise Performance Management

Process Owners

- Identifies process (instance level) KPI’s
- Incremental process improvement, problem resolution, and escalation
- Not requiring a broader Enterprise Architecture (although still desirable) this level of monitoring may be implemented standalone or within a departmental BPM initiative

IT Operations

- Augments technical support, service level monitoring, and Root Cause Analysis
- Run-time monitoring of processes provides data that can be correlated with system resource utilization to support real-time insight into business load characteristics and enable long-term capacity planning
- This approach requires a coordinated IT architecture strategy to establish the link between business metrics and system resources.
A

Business Process Selection Framework

The following is an overview of the Oracle BPM Process Selection Framework.

A.1 Value Alignment

The Value Alignment worksheet is used to measure how well candidate business processes align with the larger goals, objectives, and initiatives of the organization. The Value Alignment worksheet is shown in Figure A–1 below.

Figure A–1 Value Alignment Worksheet

The goals, objectives, and key initiatives of the organization are recorded at the top of the worksheet. Each business process candidate is then measured against the goals, objectives, and initiatives by entering a value from 0 to 5 (5 being the highest) based on how well the business process candidate aligns with the goal, objective, or initiative. The worksheet then applies quartiles to the resulting scores to assign a Value Alignment Score. The Value Alignment Score is propagated to the Process Candidates worksheet (see below).
A.2 Candidate Evaluation Parameters

The Parameters worksheet is used to define the criteria used to score the process candidates. Both the realization benefit criteria and the realization inhibitor criteria are defined in this worksheet. The Parameters worksheet is shown in Figure A–2 below.

Figure A–2 Parameters Worksheet

<table>
<thead>
<tr>
<th>Parameters Worksheet</th>
<th>Benefit Criteria</th>
<th>Inhibitor Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Realization Benefits Scores</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Realization Inhibitor Scores</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Scores</strong></td>
<td><strong>Innovativeness</strong></td>
<td><strong>Realization Impact Scores</strong></td>
</tr>
<tr>
<td><strong>Alignment Score</strong></td>
<td>The alignment of the process to the business processes of the enterprise’s business unit and the enterprise</td>
<td>The alignment of the process to the business processes of the enterprise’s business unit and the enterprise</td>
</tr>
<tr>
<td><strong>Business Impact Score</strong></td>
<td>The business impact score is a measure of the expected impact of the process on the business. The score is based on the business process and the business unit</td>
<td>The business impact score is a measure of the expected impact of the process on the business. The score is based on the business process and the business unit</td>
</tr>
<tr>
<td><strong>Integration Complexity Score</strong></td>
<td>The integration complexity score is a measure of the complexity of integrating the applications necessary to support the process.</td>
<td>The integration complexity score is a measure of the complexity of integrating the applications necessary to support the process.</td>
</tr>
<tr>
<td><strong>Knowledge Gap Score</strong></td>
<td>The knowledge gap score is a measure of the gap in knowledge regarding the process.</td>
<td>The knowledge gap score is a measure of the gap in knowledge regarding the process.</td>
</tr>
</tbody>
</table>

For each criterion, a text score and the numeric score are defined. Providing a separate text score makes the Process Candidates worksheet more readable and allows process candidates to be evaluated without selecting actual numeric values. After the process candidates have been evaluated by selecting the appropriate text score, the numeric values can be adjusted (if necessary) without requiring revisiting the evaluations.

A.3 Weighting

The weighting worksheet allows the individual criterion to be weighted based on the relative importance of each. The Weighting worksheet is shown in Figure A–3 below.
The weights for both the realization benefits and realization inhibitor scores must total 100% for the spreadsheet to score the process candidates correctly.

### A.4 Process Candidates

The Process Candidates worksheet is used to evaluate process candidates by selecting a score for each of the criterion. The Process Candidates worksheet is shown in Figure A–4.

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Process Candidate 1</th>
<th>Process Candidate 2</th>
<th>Process Candidate 3</th>
<th>Process Candidate 4</th>
<th>Process Candidate 5</th>
<th>Process Candidate 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Realization Benefits Score</td>
<td>4.80</td>
<td>4.43</td>
<td>4.06</td>
<td>1.43</td>
<td>4.29</td>
<td>7.43</td>
</tr>
<tr>
<td>Category Score</td>
<td>Core</td>
<td>Strategic</td>
<td>Supporting</td>
<td>Supporting</td>
<td>Strategic</td>
<td>Core</td>
</tr>
<tr>
<td>Executive Interest Score</td>
<td>High</td>
<td>High</td>
<td>Medium</td>
<td>Low</td>
<td>Medium</td>
<td>High</td>
</tr>
<tr>
<td>BPM Project Score</td>
<td>None</td>
<td>Identified</td>
<td>Justified</td>
<td>None</td>
<td>Justified</td>
<td>Justified</td>
</tr>
<tr>
<td>Alignment Score</td>
<td>Low</td>
<td>Medium</td>
<td>Medium</td>
<td>None</td>
<td>Low</td>
<td>Medium</td>
</tr>
<tr>
<td>Business Impact Score</td>
<td>Medium</td>
<td>Low</td>
<td>Medium</td>
<td>Low</td>
<td>Medium</td>
<td>Medium</td>
</tr>
<tr>
<td>Improvement Opportunity Score</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>Response to Change Score</td>
<td>Medium</td>
<td>Low</td>
<td>Low</td>
<td>Medium</td>
<td>Medium</td>
<td>Medium</td>
</tr>
<tr>
<td>Lack of Structure Score</td>
<td>Medium</td>
<td>Low</td>
<td>Medium</td>
<td>Medium</td>
<td>Low</td>
<td>Medium</td>
</tr>
<tr>
<td>Complexity Score</td>
<td>Medium</td>
<td>Medium</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>Organization Impact Score</td>
<td>Low</td>
<td>Medium</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>Resources Involved Score</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>Integration Complexity Score</td>
<td>Low</td>
<td>Medium</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>Knowledge Gap Score</td>
<td>Medium</td>
<td>Low</td>
<td>Medium</td>
<td>Low</td>
<td>Low</td>
<td>Medium</td>
</tr>
<tr>
<td>Decision Basis Score</td>
<td>2.00</td>
<td>0.93</td>
<td>1.00</td>
<td>1.24</td>
<td>1.20</td>
<td>5.26</td>
</tr>
</tbody>
</table>

The selections available for each criterion are limited to the text scores provided in the Parameters worksheet. The numeric scores are calculated using the numeric scores in the Parameters worksheet and the weights from the Weighting worksheet.
The *IT Strategies From Oracle* series contains a number of documents that offer insight and guidance on many aspects of technology. In particular, the following *Oracle Reference Architecture* (ORA) documents may be of interest:

**ORA SOA Foundation** - This document is suggested pre-reading for those wishing to get a deeper background to the SOA aspect of this document. It presents important basic concepts of SOA that are instrumental to building applications for a SOA environment. It covers topics including the components of a service, service layering, service types, the service model, composite applications, invocation patterns, and standards that apply to SOA.

**ORA SOA Infrastructure** - Infrastructure plays a key role in a successful enterprise SOA environment. The SOA Infrastructure document describes the role of infrastructure and the capabilities it provides. It offers an array of views to define infrastructure for SOA, including logical and physical views, as well as technology and product mapping.

**ORA BPM Foundation** - This document presents important basic concepts and an architectural foundation for BPM in IT.

**ORA BPM Infrastructure** - The document identifies the architectural capabilities associated with the BPM conceptual architecture; it then extends this conceptual architecture to present a logical architectural view and identifies the components necessary to realize the capabilities required by BPM; the Oracle products associated with BPM are then mapped to this logical architecture. The document concludes with an outline of BPM deployment considerations.
Re-engineering the Corporation, a Manifesto for Business Revolution by Michael Hammer and James Champy (Harper Collins, 1993)


IEEE 1471 Recommended Practice for Architectural Description of Software-Intensive Systems

Bruce Silver "three levels of Business Process Modeling”
http://www.brsilver.com/wordpress/2008/12/03/bpmns-three-levels-reconsidered

American Productivity & Quality Center (APQC) http://www.apqc.org
The following BPM specific terms and abbreviations are included here for easy reference. Please see the ORA Master Glossary for other terms used in the various ORA documents.

**Business Performance Management (BPM)**
see Performance Management.

**Business Process Eengineering (BPR)**
Business Process Re-engineering (BRE) was business process improvement, that is, making people (workers) and their activities operate more efficiently towards the goals of the business. See Michael Hammer and James Champy (1993) "Re-engineering the Corporation: a Manifesto for Business Revolution". BRE quickly became a new challenge for IT to address new demands from the business and the resulting disparity between the business and IT soon became known as the "business-IT gap".

**Business Rules Management System (BRMS)**
A system extending originally specialized rules engines technologies to provide an accessible, centralized repository for business rules with standardized interfaces and rules languages.

**Case Management**
Case management is a relatively new branch of BPM which revolves around the development of a "case file" (e.g. law enforcement case, legal case, various research activities). In case management the process is must be more flexible than in traditional BPM to accommodate unforeseen paths, activities, or participants, and to avoid stiffling innovation.

**Enterprise Performance Management (EPM)**
see Performance Management.

**Model-driven (MDD/MDA)**
Model-driven approaches derive machine executable artifacts from graphical models. Well known model-driven strategies include Model Driven Development (MDD) and Model Driven Architecture (MDA) in which executable code is generated from graphical expressions of an application system.

**Modeling language**
A system of graphical notation used to express various aspects of a computer application, such as, its process flow, message exchange, component interaction, etc.
The use of the notation is constrained by a set of rules to ensure consistent interpretation of their meaning.

**Performance Management (EPM)**
Performance Management is the name given to the activity of measuring and executing enterprise-level business improvement strategies. Historically, Performance Management referred to the HR activity of human capital management, however, this is now just one component of the enterprise-wide performance management of an organization as a whole. Performance Management also known as Business Performance Management (BPM), Enterprise Performance Management (EPM), and Corporate Performance Management (CPM).