# Architecture-driven Enterprise Integration

**Article** *in* International Journal of Management and Enterprise Development · March 2008

DOI: 10.1504/IJMED.2008.017433



**1 author:**

Thomas R. Gulledge
George Mason University
**125** PUBLICATIONS **1,407** CITATIONS

SEE PROFILE

# Architecture-driven Enterprise Integration

Thomas R. Gulledge

Enterprise Integration, Inc.
George Mason University,
Virginia, USA
E-mail: thomas.gulledge@eiisolutions.net

**Abstract:** Business Process Management (BPM) is an ongoing topic of interest for contemporary managers. This interest is documented by a long sequence of methods, techniques and tools emerging and declining in favour over the years. This paper takes a more holistic approach to BPM, moving from efficiency generating techniques, and focusing on BPM as the key element in an approach to achieving total enterprise integration. The approach, Architecture-driven Enterprise Integration, has many dimensions, but BPM is central to aligning information technologies and systems to management objectives and related requirements. Aspects of the approach are demonstrated with documentation from various implementation projects and vendor products. A major contribution of this paper is an understanding that BPM must be examined in a more holistic fashion, and that many BPM methods are severely constrained in their ability to generate radical and significant improvement inside of organisations.

**Keywords:** enterprise integration; business process management; enterprise solution architecture; implementation framework.

**Reference** to this paper should be made as follows: Gulledge, T.R. (2008) 'Architecture-driven Enterprise Integration', *Int. J. Management and Enterprise Development*, Vol. 5, No. 3, pp.265–309.

**Biographical notes:** Thomas R. Gulledge is a Professor of Public Policy and Engineering at George Mason University and the President of Enterprise Integration, Inc. He has designed and implemented many enterprise integration projects, and he specialises in solutions related to logistics and supply chain management. He is also responsible for a series of initiatives relating to supply chain integration, solution design and implementation in Europe and Asia. He is the Author of many works related to these issues, including the Architecture-driven enterprise integration solution framework.

## 1 Introduction: Architecture-driven Enterprise Integration

Architecture-driven Enterprise Integration is a framework and a methodology to aid in the design, implementation, monitoring, managing and controlling of the integrated enterprise. Architecture-driven Enterprise Integration enables the organisational architecting concepts that were described by Sauer and Willcocks (2002). As Sauer and Willcocks point out, the Organisational Architect is

> Someone who is neither all strategist nor all technologist, who guides the translation of a strategic vision to a flexible integrated platform. Organizational architects sustain a dialogue between visionaries and technologists as they define and design the right combination of structures, processes, capabilities and technologies – one that has a greater chance of being responsive to the organization's goals. Thus, true synergy becomes possible. The platform is shaped by the vision and the vision is reshaped by the characteristics of the platform that enable the vision.

Sauer and Willcocks describe the Organisational Architect in general terms, but this paper describes Organisational Architecting in specific terms; designing what is generally called an enterprise 'solution'. The solution is driven by vision and management direction, but eventually results in what Sauer and Willcocks call a platform. Platform, in this context, is a collection of information systems and technologies that enable the management vision.

This paper asserts that the key to translating a management vision into an integrated enterprise solution is Business Process Management (BPM). Business processes define how organisations execute 'work' in order to realise output. Since output drives profitability, then business processes must align with vision and management direction. Systems and other information technologies enable business processes, so there is a natural planning and execution hierarchy that includes these concepts. This planning hierarchy, by the definition used in this paper, forms the conceptual basis for an architecture; a natural business process-oriented architecture that can be used to aid in the design of the modern integrated organisation. This description and associated logic describes the origin of the term: *Architecture-driven Enterprise Integration*.

The term architecture means different things to different people. This paper is not about designing software, networks or other technology-oriented products that are priorities of information technologists. It is about designing organisations and aligning software and other information technologies with the newly designed organisation. BPM is the key concept that defines Architecture-driven Enterprise Integration.

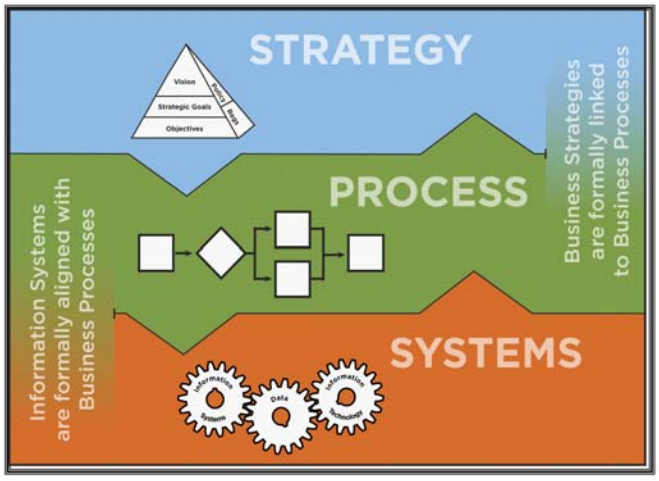
## 2 Business process management

A business process is a sequence of functions that are executed by organisational units, according to appropriate process logic, using any necessary data. This ensures that an overriding task (relating to certain objects) is completely carried out (Kirchmer, 1999). In layman's terms, a business process is a sequence of activities that are executed to perform a particular task. Since sequences are ordered with respect to time, a business process (by definition) is dynamic.

Since managers are interested in how work is performed, business process knowledge is critical for managing, controlling and improving work performance and output. Optimised business processes result in lower cost, and in many cases, targeted competitive advantage. Some business processes are executed manually, but others can be automated. In fact, the discipline that is called Management Information Systems is by and large about the automation of business processes. Information technologies and systems can be used to eliminate manual steps, making processes more efficient, reducing errors and improving cycle times.

Business processes enable enterprise productivity, and technologies and systems enable business processes. These relationships create a potential misalignment problem. Executives define organisational priorities and objectives, so managers are responsible for ensuring that the business processes execute to achieve priorities and objectives. Alignment involves ensuring that management direction is aligned with the business processes, and also that enabling information systems and technologies are properly aligned with the business processes. The alignment of these 'levels' is shown in Figure 1. Since the alignment is from management to technology, this is a type of vertical alignment. This alignment problem has been discussed in detail in earlier work (Gulledge and Sommer 2003).

**Figure 1** Vertical alignment of strategy, process and technology




The alignment shown in Figure 1 is not automatic, and misalignment is the source of many organisational problems. A properly designed enterprise integration framework must include alignment across the three levels.

Enterprise integration is the alignment of strategies, business processes, information systems, technologies and data across organisational boundaries to provide competitive advantage. The process of achieving enterprise integration includes all managerial and technological factors that enable cross-functional business process integration. The result is a customer oriented management structure with information systems that are formally aligned with business processes to establish/retain competitive advantage. This definition of enterprise integration implicitly denotes that business processes flow across organisational boundaries, which adds significant complexity to BPM.

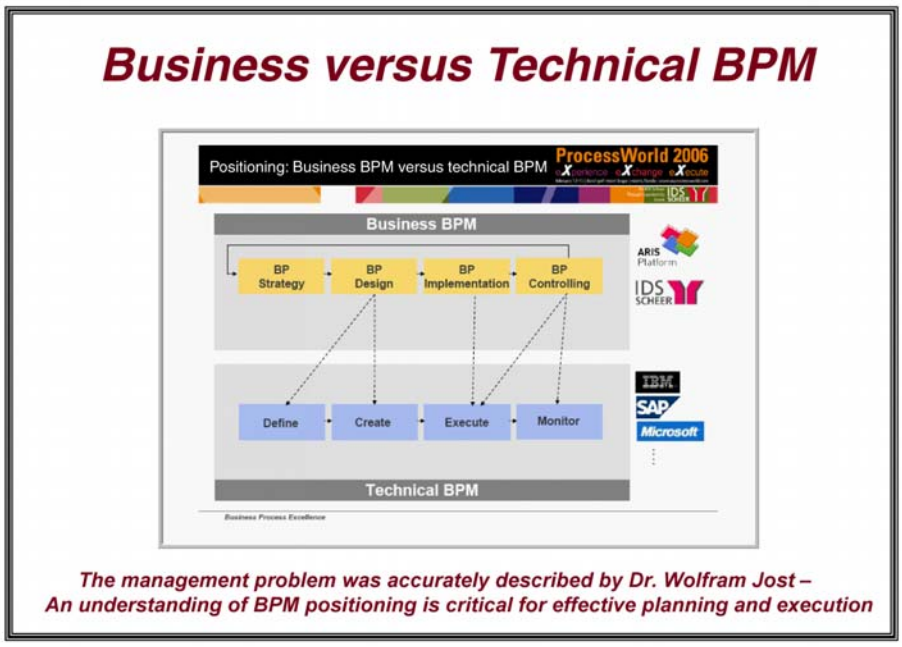
Managers implicitly or explicitly manage business processes. Even if no explicit documentation and management approaches are in place, work is still accomplished, so the processes are implicitly managed. However, explicit management and improvement is desirable, so the discipline of Business Process Improvement (BPI) is core to the management disciplines.

## 3 Business BPM and Technical BPM

Traditional approaches to BPM are useful for managing, controlling and improving business activities. These traditional techniques, reviewed below, fall in the category that is called Business BPM. Business BPM focuses on the high level business processes that describe an organisation's work processes. Management approaches such as continuous process improvement, total quality management, BPI, Lean Six Sigma and others are classified as Business BPM approaches.

Technical BPM is focused on managing the technical resources that enable the higher level processes that are classified as Business BPM. Technical BPM processes are very specific and detailed; following detailed standards, such as the Business Process Execution Language (BPEL), and they enable such low-level techniques as Enterprise Application Integration (EAI), Business Activity Monitoring (BAM), Web Service Orchestration or other process-related techniques that are 'close to the technology'. One differentiator between Business and Technical BPM is granularity, but there are others. For the organisation to be effective Technical BPM and Business BPM must align, or the levels in Figure 1 are not aligned. This alignment is shown in Figure 2, as reproduced from Jost (2006).

**Figure 2** Business BPM versus Technical BPM

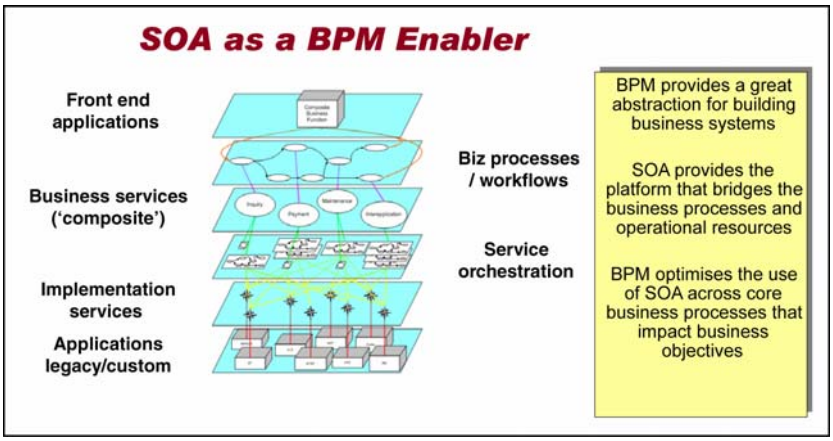



*Source*: Jost (2006).

The alignment of Business BPM with Technical BPM is explicit, and in fact Business BPM defines the requirements for Technical BPM. Figure 3, reproduced with adaptation from Oracle Corporation (2007), depicts how Business BPM is the "requirements definition level" for Technical BPM. The figure is enhanced in Figure 4 to show how Business and Technical BPM fit into the Oracle Fusion structure. When moving from the

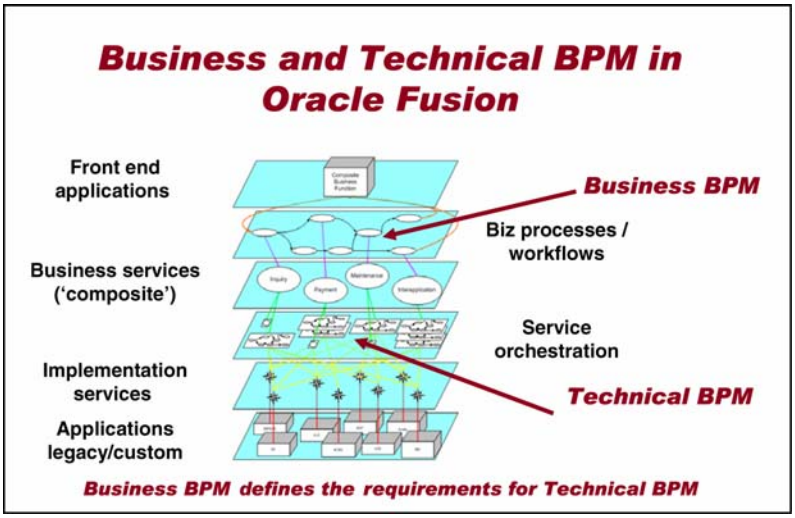
upper to the lower levels in the figure, the transition is made from business requirements to system requirements. If the business process flows across multiple applications, then the resulting business process is known as a composite application.

**Figure 3** SOA as a BPM Enabler in Oracle Fusion




*Source*: Oracle (2007).

**Figure 4** Business BPM as requirements for Technical BPM in Oracle Fusion




*Source*: Oracle (2007).

If the levels are aligned and connected, then a powerful Enterprise Integration objective is realised. Requirements are defined by managers, and these requirements are transferred directly to the technology, creating a powerful monitoring and control framework.

A good enterprise integration framework ensures that Business BPM and Technical BPM are aligned across the enterprise. The discussion below addresses Business and Technical BPM within the context of Enterprise Solution Architectures.

The second main point of this discussion is that Technical BPM is unlikely to be effective if it is implemented independent of Business BPM, and this paper will eventually demonstrate that the converse is also true. The logic of this assertion is straightforward. One can operate efficiently at the technology level, and still be 'doing the wrong thing' if the technical level is not aligned with the business level.

A poor understanding of the differences between Business and Technical BPM has caused much confusion across the management and technology trade literature. Until recently the technology community has by and large ignored Business BPM. Likewise, managers generally have a poor understanding of Technical BPM. However, this dichotomy will gradually disappear as vendor products merge the two concepts using automated tools. For example, Technical and Business BPM are merged in Oracle Fusion Middleware. The direction of SAP is becoming more clear with the release of NetWeaver 7.1, but it is now possible to synchronise Business and Technical BPM within the Solution Manager in NetWeaver, but as far as we know, the direct linkage to Technical BPM is not completed yet.

## 4 Business process architectures

Business and Technical BPM is most useful when applied within the context of a business process oriented solution architecture. Organisations that manage by business process within the context of a business process architecture are described as having a business process orientation, or more simply, a process orientation.

### *4.1 Enterprise architectures*

At the most basic level, an enterprise architecture is a description of an enterprise. For an enterprise architecture to be complete and useful, it must contain information about many aspects of the enterprise. In the architecture literature these aspects are called 'views of the enterprise'. The architecture must contain strategic information about the organisation, including management goals, objectives and strategies. It must contain information about how the organisation executes its strategies in order to achieve its goals and objectives; that is, its business processes and the order in which they are performed. It must contain information about who is executing the business processes, what organisations they belong to and where the work is performed. It must also contain information about how the work is enabled, including the systems and technologies that support or automate the business processes. In this sense, the architecture is indeed a design for an enterprise. And, most importantly, the architecture must make explicit the relationships among all of the above concepts or views. Finally, the architecture contains 'time-phased' information so that the organisation knows what the future should look like; not just the current organisational state. These levels or hierarchies of information, from policy through business process, and down to systems execution must be maintained in an integrated repository. Without this information, the organisation runs the risk of building its systems from the 'photographs'[1] in the minds of its developers and functional proponents. As an analogy, this would be like building a house from a photograph as opposed to a detailed blueprint.

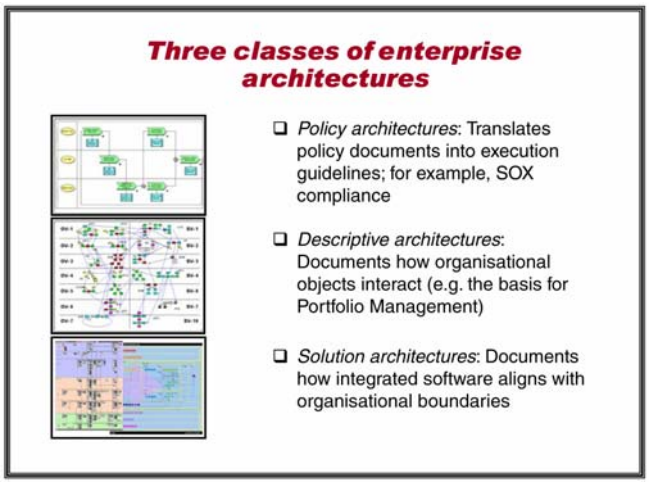
Business processes are a critical component of enterprise solution architectures. As shown in Figure 1, they define a central layer in the enterprise integration model.

Over the years, the concept of an enterprise architecture has been used to denote many types of architectures that do not explicitly follow the above definition. For example, some technologists call a mapping of enterprise IT assets an enterprise architecture. Others call a description of enterprise activities, systems and data an enterprise architecture. With a liberal interpretation of the definition, either could be correct, but we prefer additional preciseness. The focus of this paper is on Enterprise Solution Architectures, which are discussed below.

### *4.2 Enterprise solution architectures*

There are many types of architectures, but at the enterprise level, we have selected a broad classification that includes three classes of architectures. These are shown in Figure 5. Policy architectures translate policy into execution guidelines, making the written policy more precise. Descriptive architectures document how organisational objects interact. These object interactions could be business activities at an operational level, system components at a technical level or logical/physical data objects. Solution architectures show how systems and technologies enable business processes as they flow across organisational boundaries. There is no ranking or preference ordering for these architecture types. The architecture types are just different, and they are used for different purposes.

**Figure 5** Three classes of enterprise architectures

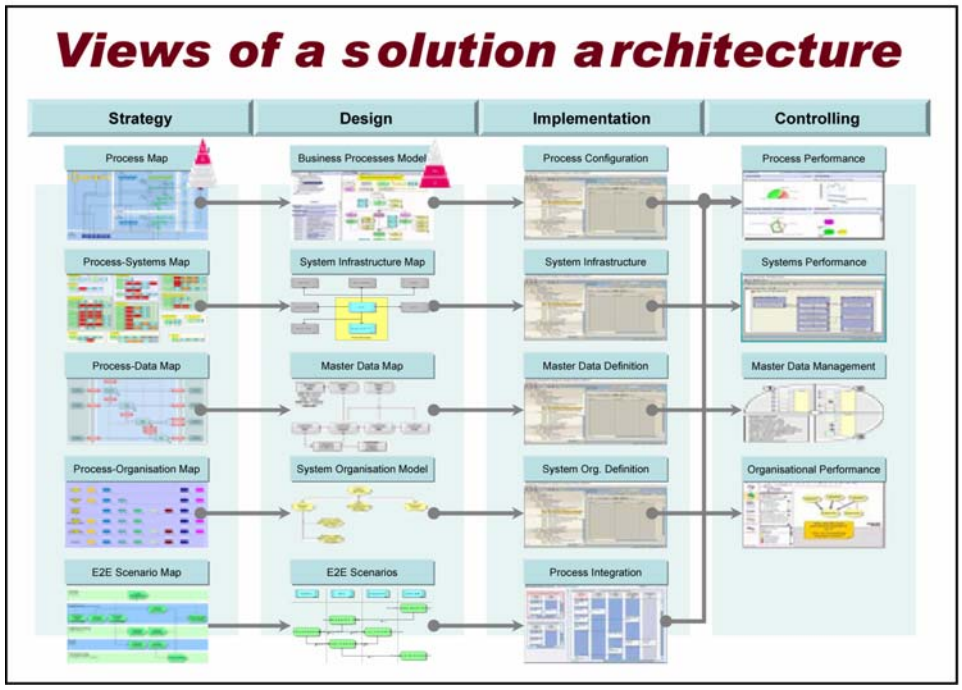



All three architecture types could be presented in one massive architectural model, but to avoid complexity, managers often display the types (and even components within types) separately. As previously discussed, these subcomponents of the enterprise architecture are called architecture 'views'. Figure 6 demonstrates the concept. In this figure, Simon (2004) presents a collection of views that are useful when implementing the SAP Business Suite.

Questions always arise about how many views are needed and what should be considered, and these are difficult questions to answer. The answers require deep knowledge of the products that will be implemented as well as a good understanding of

the implementation environment. However, from a requirements perspective the business process views always dominate, and must be completely understood. Otherwise, systems and technologies will be implemented, but business requirements will not be met.

**Figure 6** Solution architecture view for implementing SAP



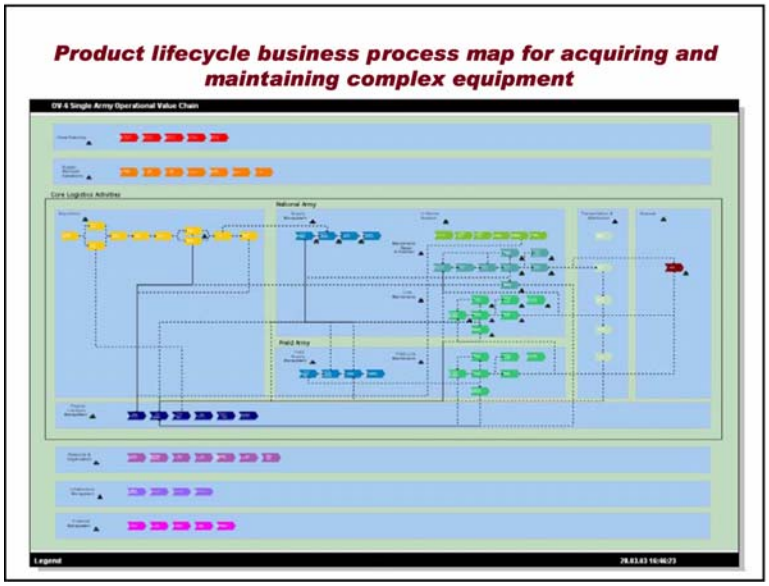

*Source*: Simon (2004).

### *4.3 Dominant business process view*

Technology and system requirements should be based on the needs of managers in supporting business objectives. That is, technologies and systems should enable management effectiveness and efficiency by automating day-to-day work. Business processes describe the day-to-day work of managers, and this work can be documented using business process notation. If systems and technologies do not 'align' with these business processes, the managers' requirements are not met. This misalignment between a technology solution and a business process is called a 'gap', and gaps are the source of most managers' rejections of newly implemented information systems. If the system does not enable the business process, then management requirements are certainly not met. For this reason, we assert that the business process views are critical. If they are not correctly described, then it does not matter if the system and technology views are correct, because the managers' requirements for automating work are not realised.

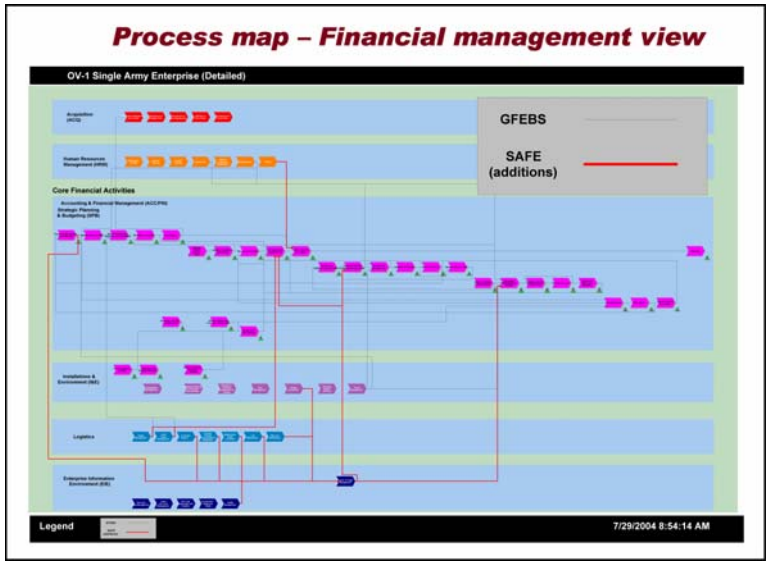
Business process views can be developed and documented at any required level of detail, but the highest level view is the process map. An example process map that is based on Product Lifecycle Management (PLM) orientation is shown in Figure 7.

**Figure 7** Process maps – the dominant business process view



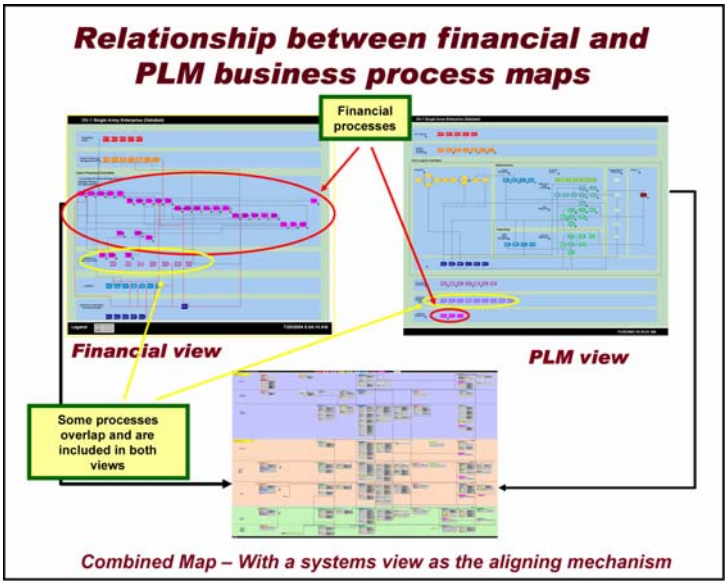

This view is critical for any information system implementation project. It does not matter if the implementation is packaged (i.e. standard) software, proprietarily developed software, instance consolidation or other information system-related projects. This view describes the scope of the implementation as well as a general perspective from a particular viewpoint on a project's boundaries. The viewpoint in Figure 7 is PLM but multiple viewpoints may be required. For example, Figure 8 shows the process map for the corresponding financial management view to the PLM view that is shown in Figure 7.

**Figure 8** A process map with a financial orientation



Of course, the views are related. The PLM business process is intertwined with the financial management process and they can be linked together and aligned with the systems that support both processes. This linkage is shown in Figure 9. Figure 9 is critical for understanding the power of architectural views, especially business process views.

**Figure 9** Project scope aligned with management requirements

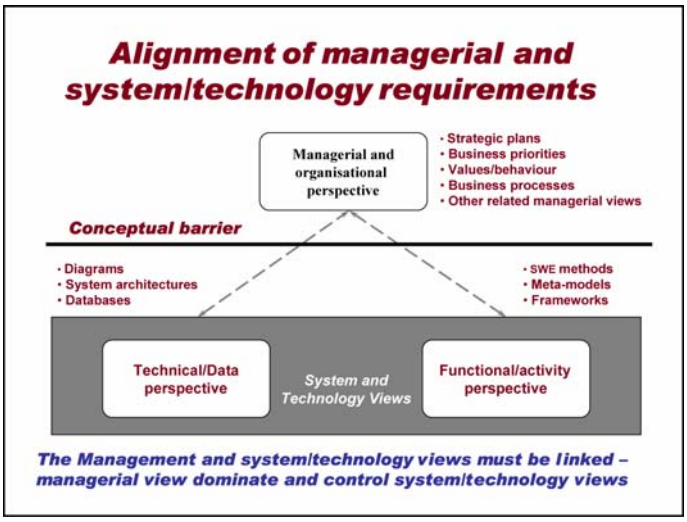



In Figure 9 the Financial and the PLM views define management requirements. The systems view at the bottom of Figure 9 shows how the systems relate to the business processes, providing a powerful mechanism for implementation control and transition planning. Without such views, there is not a sufficient understanding of the project's end-state, nor is there any mechanism for aligning management requirements with system requirements. One can make the argument that this type of planning is not needed for small enterprises or enterprises that run only one integrated system (e.g. Oracle e-Business Suite or the SAP Business Suite). We think this logic is flawed, but the truth is that most organisations do not run one system, so an understanding of the interactions is critical. Also, business requirements must be aligned with software solutions, independent of organisation size.

### *4.4 Relationship to strategic and technology levels*

Of course, planning and execution artifacts are more extensive than simple process maps and related system views. Process maps can be decomposed to any required level of detail, even to the lowest transaction level. While managers may not be interested in these detailed views, they are critical to technologists and system developers. This is the beauty of architectural planning – management and technology requirements are represented in the same plan, with managerial/organisational requirements (business processes) driving and controlling technology requirements. These relationships are shown in Figure 10.

**Figure 10** Planning with integrated managerial and systems/technical views

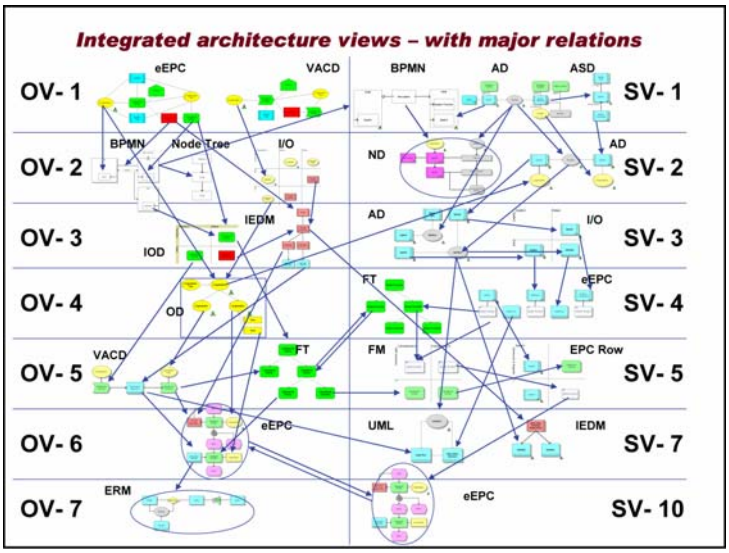



The combining of the views can be complicated, but such work is necessary to

- ensure that managerial requirements are met
- ensure that system/technical requirements are met
- develop an integrated implementation plan
- establish implementation monitoring and controlling mechanisms.

Figure 11 shows a mapping of how Figure 10 can be realised across many managerial and systems/technology views. Of course the views are related as indicated in the figure, and they are integrated; that is, stored in one database as one source of truth.

**Figure 11** Representative views and their relationships[2]



This discussion shows how BPM is linked to system implementation. It clearly demonstrates that business process requirements dominate system and technology requirements. This truism follows directly from the fact that managers generate value for organisations and systems are only enablers. These enterprise integration concepts have been organised into a framework that is called Architecture-driven Enterprise Integration.

## 5 Architecture-driven Enterprise Integration

The repeated assertion in this paper is that business processes define management requirements for enterprise integration projects. The business process requirements are related to system and technology requirements, and the complete architecture[3] is the implementation monitoring and control mechanism for managing enterprise integration implementation projects. The following steps are typically included in an architecture-driven approach to enterprise integration:

1 development of the baseline 'as-is' architecture

2 development of the solution concept 'to-be' (Target) architecture

3 use the architecture to assist with business process gap-fit analysis, software selection or other analytical exercises

4 transition the baseline to a specific solution; that is, implement the solution that is described in the architectural plan

5 extend and align other initiatives with end-to-end business processes as required

6 establish the monitoring and controlling environment

7 manage the implementation project.

## 6 Baseline 'as-is' architecture

This step is a short exercise that has two major purposes:

- it establishes the baseline for transition planning
- it establishes a baseline for measuring realised benefits.

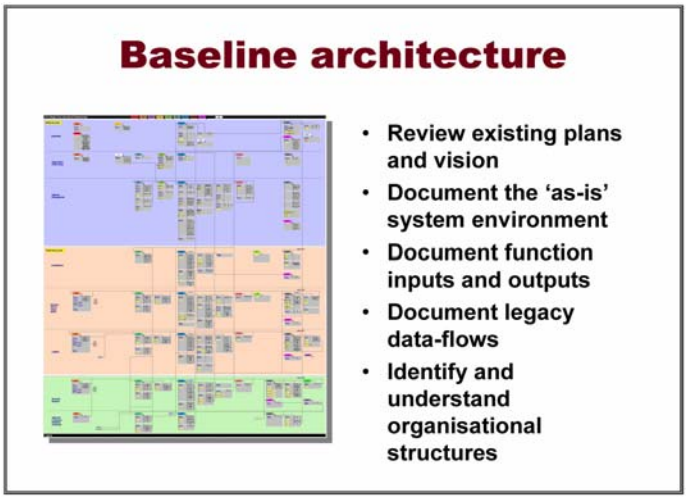
The first item is shown in Figure 12.

The specific details of Figure 12 are not as important as the understanding of the intent. This figure maps business process objects to an organisational hierarchy, which assigns process ownership. It also 'links' legacy systems to business process objects, along with their input-output data-flows. One could link other planning objects to the business process objects as needed; for example, objectives, critical success factors, Key Performance Indicators (KPIs) or any relevant object that could be used for measuring the benefits that are realised from implementing the enterprise integration project.

Figure 12 shows a critical architecture view. Since as-is systems are mapped to to-be business processes, it is possible to develop an orderly legacy system retirement plan. Or, if systems are retained, Figure 12 shows the first step in understanding legacy system

interfaces. The baseline view is not a complete transition plan, but the baseline view contains critical information that is necessary for developing a complete transition plan.

**Figure 12** Baseline architecture for a complex enterprise integration project




## 7 Develop the to-be solution architecture

The necessary architecture views to conceptualise the solution are defined and documented during this step. The concept is shown in Figure 13.

**Figure 13** To-be architecture concept

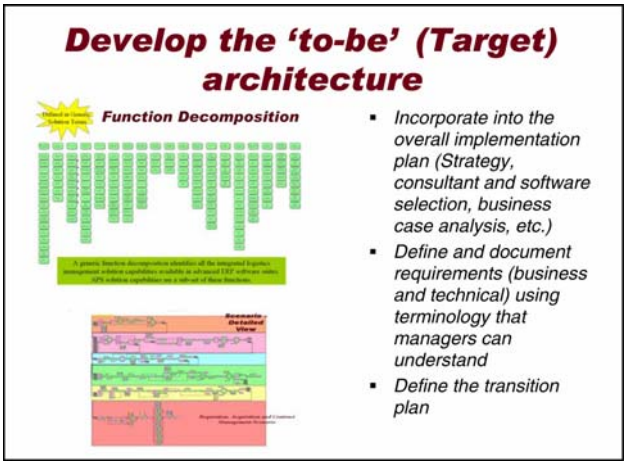
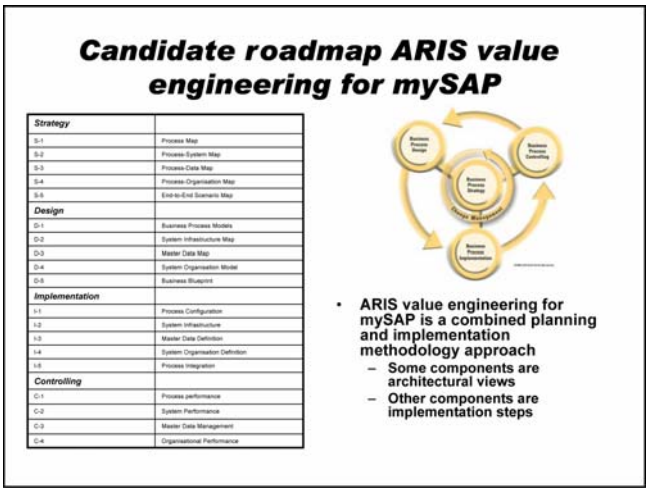



Figure 13 shows two views, but there could be any number of views. In general we follow the ARIS value engineering approach (Simon, 2004), but we modify the concept so that it is not SAP specific. Simon combines solution architecture planning with an implementation methodology to create a 'roadmap' for implementing the SAP software solution. This paper does not advocate following any specific views or steps, but creating

the necessary views and steps required for the implementation project and associated software. Some artifacts will always be used. Others are project and product specific. Simon's roadmap is shown in Figure 14.

**Figure 14** ARIS value engineering roadmap




*Source*: Simon (2004).

The following is noted about the roadmap as shown in Figure 14. All of the strategy views have a business process orientation. The design views are a mixture of process, organisation, system and technology. The implementation views are actually executed in the SAP Solution Manager, which is a NetWeaver component in the SAP Business Suite. The processes (management requirements) drive the design, which drive the system implementation. SAP advocates initialising the implementation project in the Solution Manager. This may lead to a successful implementation, especially if all business processes fall inside the SAP solution. However, for complex landscapes where SAP is one among many solutions, the SAP recommendation can only lead to a localised point solution. In short, with the SAP bounded implementation approach, you are only planning and designing for SAP and its interfaces. You are not planning and designing for how SAP will fit in the larger landscape. This discussion is a natural lead-in for a more detailed discussion of end-to-end business process scenarios, and the subject is addressed in more detail in a later section of this paper.

## 8 Architecture analyses

Once the target solution is defined, analyses may be performed. The possibilities for analyses are endless, ranging from simple business process reviews to discrete next-event simulation. The three most widely executed types of analyses are

- business case analysis
- software selection analysis
- gap-fit analysis.

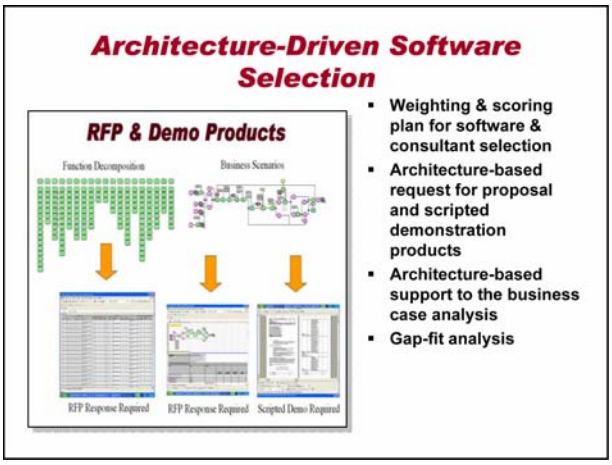
Business case analysis is an absolute requirement. Much has been written on that subject, so we do not dwell on it here. The other two concepts have received less attention, so we focus on those.

## *8.1 Software selection*

The selection of an enterprise software solution is a critical decision, and it is amazing how often companies purchase software without completing a software selection analysis. Commercial software products are not all the same. They have different features and they execute different business processes. Time spent on software selection analysis can result in enhanced benefits realization during the implementation phase. The most important principle in software selection is the recognition that software selection is a business decision, not an IT decision. Acceptance criteria include many complex and interrelated areas, but the decision is driven by business requirements. As noted in this paper, business process alignment is critical, but other related issues are discussed by Elbertsen et al. (2006).

The basic concept is presented in some detail by Baharmast (2005), and is shown in Figure 15.

**Figure 15** Architecture analysis to support software selection




The concept is simple. A generic business process architecture is developed for a representative company. Executives compare the generic architecture to the processes that are identified for enablement in their organisation. This is a critical step, and in order to properly describe company business processes, key employees must document the details of the critical business processes that the vendor's system must absolutely be capable of enabling. As a next step, a new business process architecture is developed, including only the functions that are relevant for the implementing organisation. The company-specific processes will be reused later as part of the monitoring and control phase to keep the implementation on track and focused.

As a third step, the company-specific architecture is compared to a similar architecture for the software product that is under study. This step typically requires significant effort, because all software vendors do not make this information easily available. The experience of the person performing the analysis is critical for this step.

If significant differences are identified between the company-specific architecture and the software-specific architecture, then management has to make a decision. Another product is selected, or the current product is implemented with bolt-ons or interfaces. The differences should be quantified in the business case.

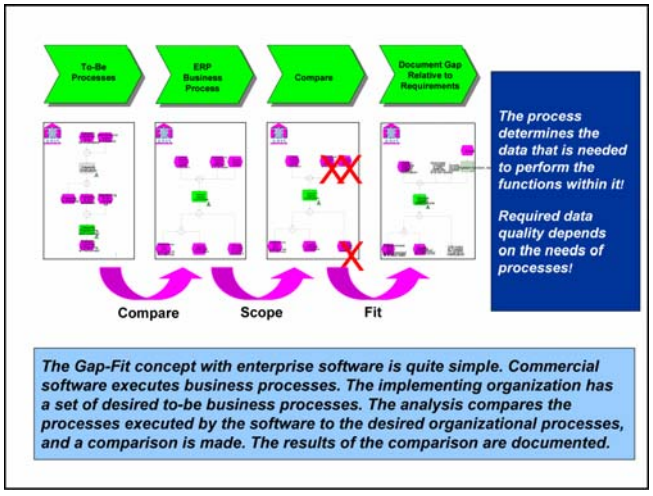
Of course, there is more than architecture analysis in vendor selection, and these non-architecture-related issues must be addressed. For example, reference companies that have already implemented the software must be visited. The visited companies should operate in a similar environment and have been using the software for a period that demonstrates solution stability.

## *8.2 Gap-fit analysis*

Gap-fit analysis is similar to software selection analysis, and the two are often executed sequentially. Once a software product is selected, a more detailed analysis that examines how the product 'fits' the receiving organisation is required. Fit, in this case, means how close the business processes that are supported by the software product agree with the business processes that are in the target architecture (i.e. the requirements). While this step requires effort and time, the Boston Consulting Group has identified this as a critical step for ensuring executive satisfaction with implemented enterprise software products (Booker, 2000; Boston Consulting Group, 2000).

The details of the approach are discussed by Gulledge (2006a), and a conceptual view is shown in Figure 16.

**Figure 16** Gap-fit analysis




Many companies have specific business process requirements that cannot be relaxed. The process may deliver competitive advantage, and if the software product cannot execute the process, then a 'gap' is realised. The trade-off decision with commercial software is the relaxation of business process fit in favour of integration. The business processes supported by the software product seldom precisely meet requirements, but managers are willing to relax the fit in favour of realising the benefits of preengineered integration. For back-office processes that do not add competitive advantage, such a trade-off is acceptable. However, for critical processes that add

competitive advantage, the trade-off may not be acceptable Gap-fit analysis is a methodological approach for identifying the gaps prior to implementation and assessing the trade-offs before the project is underway. The analysis provides a better understanding of the solution while protecting against rework, schedule slippage and the associated increases in cost.

Gap-fit analysis requires two key inputs:

- an understanding of the business processes that are enabled by the software product
- an understanding of the business process requirements from the implementing organisation.

It is the responsibility of the implementation consultants to understand and document the processes supported by the software. It is the responsibility of management, with the help of the consultants, to document and understand the business process requirements of the implementing organisation. Once both are understood, they may be documented and compared using a modern tool as described by Gulledge (2006a).

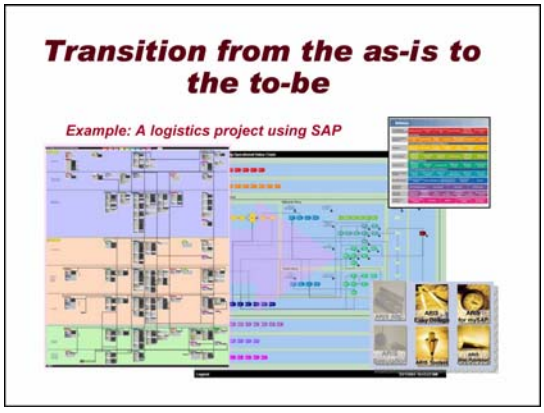
## 9 Transition the baseline to a specific solution

This step is usually the most difficult, but it is critical for success, especially for implementation projects that span complex technology landscapes. The transition plan applies resources to realise an orderly transition from the as-is to the to-be state. This involves a:

- roll-out plan schedule
- interface development plan schedule
- legacy system retirement plan schedule
- software versioning and release schedule
- data migration and data quality plan and schedule
- other components as needed.

The concept is shown in Figure 17.

**Figure 17** Planning for moving from the baseline to the target solution

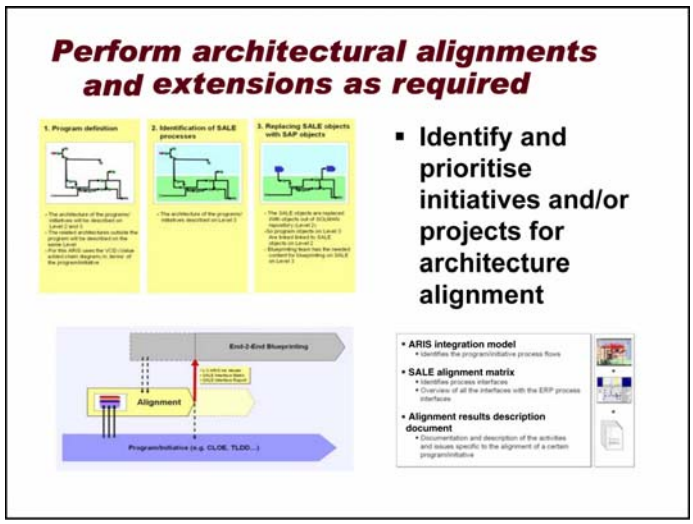


This step requires much skill and domain knowledge. The complexity of even the smallest implementation projects requires that modern tools be used to analyse the transition. A single ad hoc or uninformed decision can result in considerable rework and additional cost. This step must be completed with or without an architecture, but an architecture accelerates and organises the transition planning process.

## 10 Architecture alignment

Organisations seldom enable all of their business processes with a single software product. Including all business processes in one product is sometimes desirable, but 'Big I'[4] is seldom achievable even if it is desirable. The implication is that the organisation must include (or align) legacy and other software products with new enterprise software products. This step cannot be executed independent of transition planning, but it does require special care, treating each aligned system as a separate analysis. The approach is shown in Figure 18, and is discussed in detail by Gulledge et al. (2005).

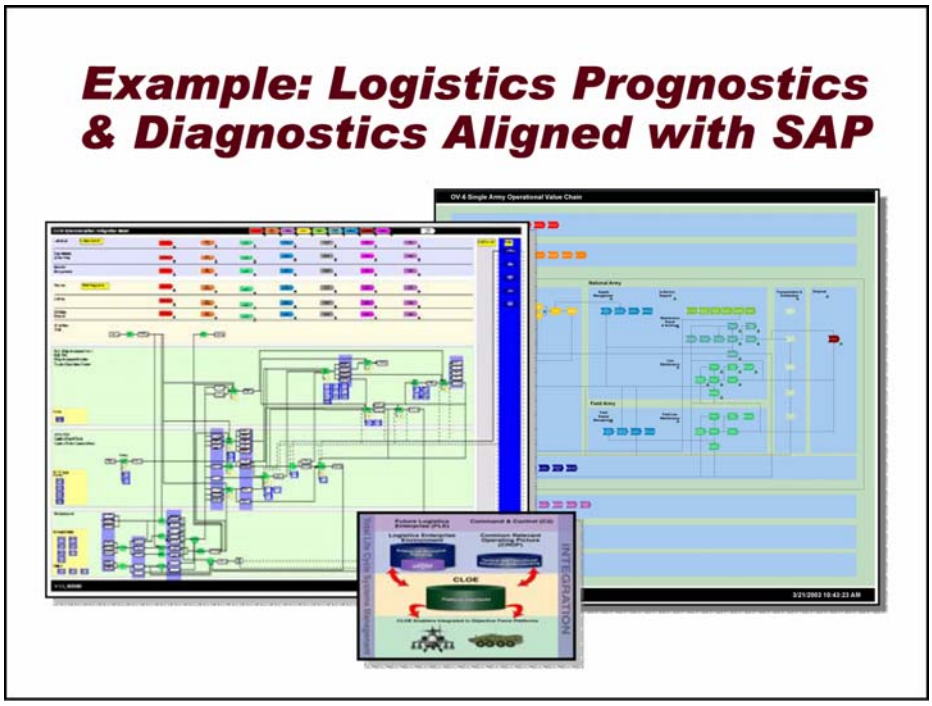
**Figure 18** Architecture alignment




A proper approach to alignment is driven by business processes. Business process integration is ensured by mapping the end-to-end business processes across the initiatives to be aligned. Systems and information flows are linked to business process functions, providing a capability to document high-level interface requirements. The aligned business processes and the associated information flows can be documented in a requirements document and placed under architecture control. This allows better planning and execution by transferring alignment control from the implementation consultants to the business process owners. It also allows better project management and control by the customer by minimising surprises that appear periodically in a poorly planned integration project. Figure 19 shows the business process alignment approach. In this picture a maintenance prognostic and diagnostic system is aligned with business processes that are enabled by the SAP software solution. The maintenance system is

standards-based, but it is proprietary developed and falls outside of the scope of the SAP project. Still, it is an important component of the enterprise solution, so it must be included in the enterprise solution architecture.

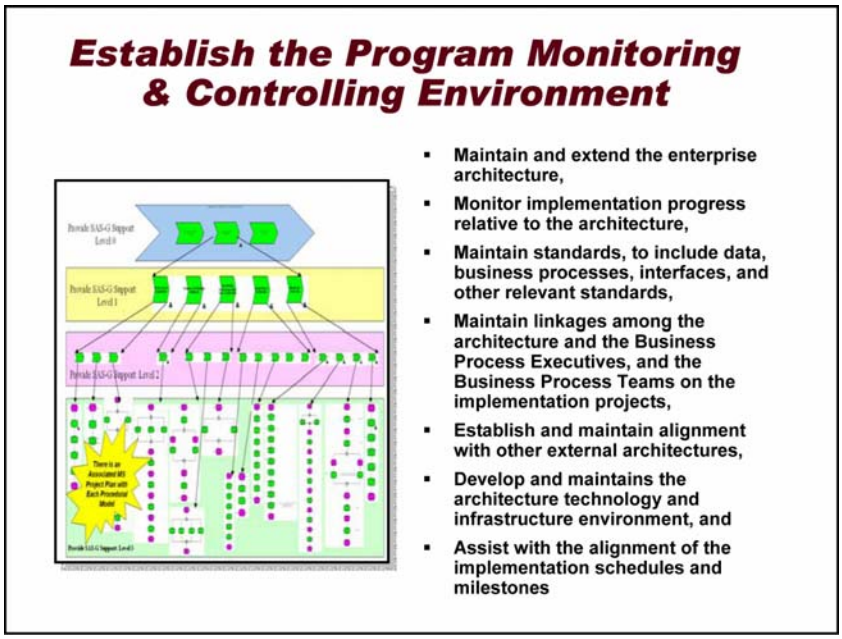
**Figure 19** Platform-level diagnostic and prognostic system aligned with SAP




Eventually the data-flows were mapped across business process functions, and a high level requirements document was produced for the implementation consultants. At the same time, the particular project was included in the overall project plan and scheduled for eventual interfacing. From the SAP project manager's point of view, a better understanding of an interfaced project is provided. From an enterprise point of view, senior managers understand how the combined projects enable a critical business process.

During the implementation phase the complete package is transferred to the implementation team, and the solution architecture becomes the baseline for monitoring the actual work performed. In short, alignment 'pulls' all related projects into the solution architecture, and progress against the plan is evaluated relative to the total business process scope that is documented in the architecture. This monitoring and control environment is documented in the form of a procedural model, and all business processes within the scope of the solution architecture (including all alignment packages) are managed over time. The implementation monitoring and control environment is shown in Figure 20. Technically speaking there is an end date for the project, but in reality, the project never ends. There are continuous maintenance, upgrades and other changes that must be managed over time. One view of a complete technical environment, with an orientation to SAP is presented by Gulledge and Simon (2005).

**Figure 20** Architectures for monitoring and control




## 11 End-to-end (E2E) business process scenarios

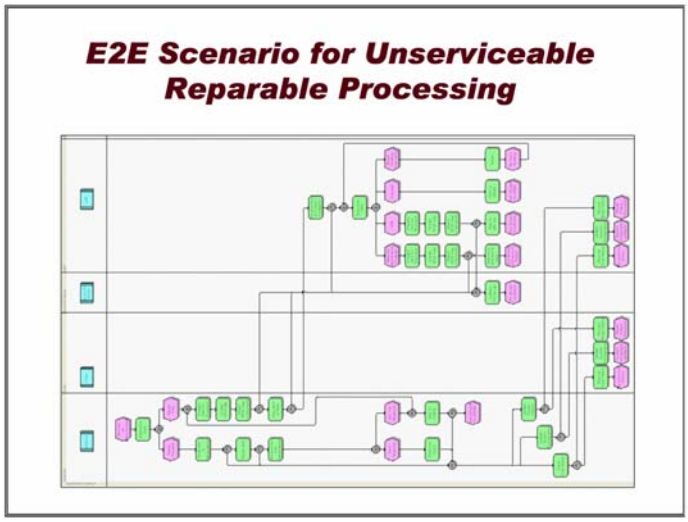
Much of the material in this section is discussed by Bubak et al. (2006), but with a particular orientation to SAP. Other aspects are presented by Frye and Gulledge (2007). In this section the concepts are presented in general terms from an Architecture-driven Enterprise Integration perspective. A true enterprise management approach demands an end-to-end viewpoint that flows unimpeded across the constraints imposed by information systems and organisational boundaries. An example of an E2E scenario is shown in Figure 21. In Figure 21, the business process flow is documented in 'swim lanes'. In this case, the business process "flows" across four systems (the rectangles in the far left column of Figure 21). If the business process is optimised relative to one system while ignoring the others, the E2E scenario is suboptimised. This simple understanding provides tremendous insight into the discussion that follows.

Figure 21 also defines the high-level requirements for a composite application[5] that enables the E2E business process, and it clearly demonstrates that the process cannot be decoupled from the systems. In fact, many additional processes may be enabled by the same system, so it is very difficult to selectively improve any one process without affecting another. This fact is often ignored in enterprise initiatives and portfolio analyses, leading to poorly scoped projects and portfolios that are impossible to implement. It also has significant implications for any stand-alone BPI or business process reengineering initiatives.

The figure also points out one of the fallacies of the Family of Systems approach. Once the systems are built, the processes are "bound" and there is little potential for additional BPI. The main result of this discussion is that collections of E2E business process scenarios define solutions. Software boundaries must be aligned with these

collections of scenarios, and once the systems are built, the processes are bound. That is, the opportunities for BPI are limited since individual systems are usually tied to multiple business processes. This means that even the smallest BPI initiatives must consider collections of scenarios and all systems that enable the collection. It also means that the boundaries for large-scale implementations (e.g. Enterprise Resource Planning (ERP) systems) must be carefully defined.

**Figure 21** An example end-to-end scenario




### *11.1 Business process improvement*

The above discussion provides an insight into a new approach to BPI. Traditionally BPI was viewed in terms of individual processes that are documented, studied and then altered for improvement. Our argument is that the approach is flawed, because the end-to-end business processes are 'bound' by the systems that enable them, and the systems must be simultaneously considered.

## 12 System constraints on BPI

If E2E business processes are modelled in an enterprise solution architecture, all dependencies among business process functions, information systems and associated data-flows are documented and 'object-linked'. We call this representation the Business Process Framework. This means that a change in any business process or system can be analysed relative to its impact on the complete solution. If the right tool is used, this analysis may be as easy as developing a report.

For example, if one system is targeted for shutdown through the analysis of a system portfolio, one can easily identify all process functions that are affected by the change. Likewise, if a process is improved,[6] one can easily understand how the systems must be realigned to accommodate the change. A properly constructed enterprise solution architecture is fully integrated, so the improvement or shut-down analyses consider the fact that BPI possibilities are constrained by the systems. Systems could always be

altered, but at a cost. In many instances, it is just too costly to make the system changes that would be required in order to improve the business process. The constraints on process improvement are real, but there are other constraints on processes that may diminish improvement possibilities.

## 13 Process coupling

Sommer and Gulledge (1999) also argue that legal and regulatory constraints may also bind processes, making it difficult to realise improvement. The indiscriminate use of regulatory measures may, over time, constrain a process to such an extent that it becomes too expensive or even impossible to effectively manage and execute. This over-constraining of organisational processes is termed 'process coupling', because organisational policies and regulations are tightly integrated with individual process functions. The prototypical example of such a process is a public sector acquisition process, which is bound by many regulations and constraints that have evolved over time.

If business processes are so tightly constrained by regulations that they are difficult to execute, managers and employees will find 'work – arounds', or what are often called 'covert' processes. Covert processes are not officially sanctioned by the organisation, and they could create risks. For example, the execution of a covert process might require the violation of a security policy. Furthermore, covert processes are not typically enabled by information systems, which creates an obvious misalignment.

By developing and periodically reviewing the consistency of policies, managerial oversight and performance measurement, an organisation can eliminate potentially dangerous and costly process coupling constraints. Business process documentation and analysis can, in conjunction with the organisational planning process, provide top managers with basic tools to understand where and when processes are overregulated, and senior managers can provide regulatory relief.

### *13.1 Business process reengineering*

In 1990, Hammer published his seminal work on business process reengineering (Hammer, 1990). This work attracted much attention from managers, and by the mid 1990s almost every major corporation in the world had initiated at least one reengineering project. Yet, as a post-mortem it is widely accepted that most of these projects did not achieve their desired results. The key to understanding the post-mortem is directly related to the discussions on business processes being 'bound' by systems.

Hammer advocated radical business process redesign. He used catchy slogans such as – 'obliterate, do not automate'. While such an approach is intuitive, it is not practical, and is certainly very expensive. Designing new processes is actually the easy part of reengineering. Realigning the information systems to agree with the new 'reengineered' processes is the difficult part, and it also extremely expensive. However, if one does not realign the systems to agree with the reengineered processes, the probability of failure increases dramatically. If information flow is still aligned with the as-is processes, then there is tremendous pressure to revert to the as-is. Many companies were caught in what is known as the 'reengineering trap'. They could not afford to realign their information

systems to agree with the to-be processes, but if they did not realign, they reverted to the as-is. Given the high cost of realigning, most firms reverted to the as-is.

Still business process reengineering was a step in the right direction. If for no other reason it helped convince managers and technologists that a business process orientation was important. Some of these issues are presented within the context of business process architectures by Sommer (2004). The basic concept was sound (create new processes and then build new technology to support the processes), but it was not practical. It was too expensive, and it was naïve to believe that organisations would create new systems to align with new processes, when their old systems were still operable. However, a type of reengineering did occur in the 1990s. It was different than envisioned by Hammer, but still it was reengineering, and in this case through the implementation of commercial standard software products.

## *13.2 Business process oriented standard software*

During the 1990s, firms realised that it did not make sense to design, develop and deploy mundane back-office systems. The risk of failure was high, and the systems did not deliver competitive advantage. There was tremendous value from implementing integrated back-office products, but scarce development resources were better utilised by focusing on 'customer touching' systems. It was more cost-effective to purchase back-office systems from companies like Oracle and SAP. Business process reengineering did occur, but in a way that was different than envisioned by Hammer. Recall that Hammer's hypothesis was that firms would redesign their business processes, and then they would design, develop and deploy information systems to enable the reengineered processes.

As previously mentioned, for a number of reasons Hammer's approach to system alignment did not occur. In lieu of this expensive and direct approach, managers purchased their back-office business processes from companies like SAP and Oracle. That is, managers purchased integrated standard software products that executed predefined reference business processes that were coded into the software. This tight integration implicitly solved the BPI problem by completely binding the process to the system. There was some flexibility to 'configure' the business processes, but by and large one accepted business processes that were delivered with the software. So, the reengineering was of the following form. The companies implemented the standard software, and then they realigned their internal business processes to agree with those that are enabled by the software. The approach is covered in detail by Keller and Teufel (1998).

The reengineering occurred, but just the opposite as envisioned by Hammer. Instead of building systems to enable reengineered business processes, firms reengineered their business processes to agree with the processes supported by the commercial enterprise systems. This led to a change in terminology with the focus being on Business Process Engineering as opposed to Business Process Reengineering. The approach is epitomised in seminal work by Scheer (1994). This business process orientation of enterprise software is not well understood by many managers, and in many cases is not even understood by the consultants who implement the software. There is a direct linkage between this critical understanding and a proper understanding of how BPM is enabled through Business Process-Oriented Solution architectures. This linkage is described in the following section.

## 13.3 Business process oriented implementation of standard software

ERP is a popular term that is used to describe enterprise standard software products. While volumes have been written about these products,[7] there are many implementation aspects that do not seem to be well understood. First, the core product is designed to be implemented as a single instance – one source of truth. This is true for even the latest versions of ERP products, so the question must be addressed: What is in the core?

The core varies from product to product, but in general it is the basic back-office business processes that are enabled by the software; for example, financials, purchasing, manufacturing, human resources, etc. The ERP products are designed to execute these critical core business processes as one solution, without complex interfacing and data fragmentation. The data is stored once, and it is instantaneously updated as changes occur in any business process that resides in the core product. The end-to-end business processes that are executed by the products are engineered to operate together with seamless integration. An update in one process is automatically reflected in every process that is impacted by that process.

One does not 'split' this single integration domain, or problems are typically the end result (Gulledge and Sommer, 2004). The software has natural business process boundaries, and if the software is divided (or 'split'), it is extremely difficult to interface it back together. This is because the software is designed to provide tight integration, and the integration is complex data and business process integration that is managed in the application tier of the software. One cannot interface the data without considering how the data interacts with the business process logic.

The confusing aspect of the software that leads to many suboptimal implementations is directly related to how the software is typically designed. The software is modular (or stove-piped) by design. For example core SAP ERP has modules such as Financial Management (FI), Material Management (MM), Production and Planning (PP), etc. This leaves the false impression that the modules can be managed like interrelated families of systems. This is not the case; the data to support all modules are stored in a single data base instance, and when multiple modules are enabled, they actually execute business processes that flow across modules. That is, the cross-module business processes are engineered directly into the product, even though the software can be implemented on a module-by-module basis.

So, to realise the benefits of ERP, one must implement multiple modules, accept the cross-module business processes that are enabled by the software and engineer existing organisational processes to agree with those that are enabled by the software. Most ERP projects that do not meet expectations, or even outright fail, do not adhere to this simple and basic implementation principle that must be followed. The software is tortured, split into fragments and interfaced; and then managers wonder why the software did not operate as advertised. This bad behaviour happens more often than it should. When the business process orientation is ignored, suboptimal implementation projects almost always result, and excessive and costly interfacing and integrity problems almost always occur.

## 13.4 Requirements for service-oriented solutions

The discussion in this section is directly related to Figure 2. Service Oriented Architecture (SOA) is a technical concept that is typically conceived, designed and

implemented by the IT department. The core of a SOA is the technical BPM layer where technical business processes are orchestrated. Business BPM is relegated to management activities that occur as work is being executed, and while discussed, is typically not explicitly considered in a SOA project implementation. The approach advocated in Oracle Fusion is an exception, as it provides a fully integrated solution that melds Business and Technical BPM.

Enterprise information systems are efficient and effective when they precisely provide the relevant information to support critical end-to-end business processes. Of course, the most efficient alignment of systems to processes occurs if all information is in a single system, and the alignment is one-to-one. Unfortunately, this is seldom the case, since end-to-end business processes almost always flow across multiple systems and data sources. This creates a more difficult situation for aligning systems and data with end-to-end business processes. Collections of business processes that flow across multiple systems and data sources are called Composite Applications, and they are typically enabled using service-oriented methods.

Figure 2 implies that the two concepts are related, and indeed they are. As Figure 3 suggests, SOA should be tightly linked to Business BPM. If SOA is an enabling technology, and if Business BPM describes how work is actually executed, then Business BPM actually defines the requirements for SOA. If the technical concept for SOA is designed, developed and implemented independent of the Business BPM layer, requirements will not be met. The concept is enhanced in Figure 22, which is based on a composite schematic from SeeBeyond.

**Figure 22** Business process requirements for composite applications

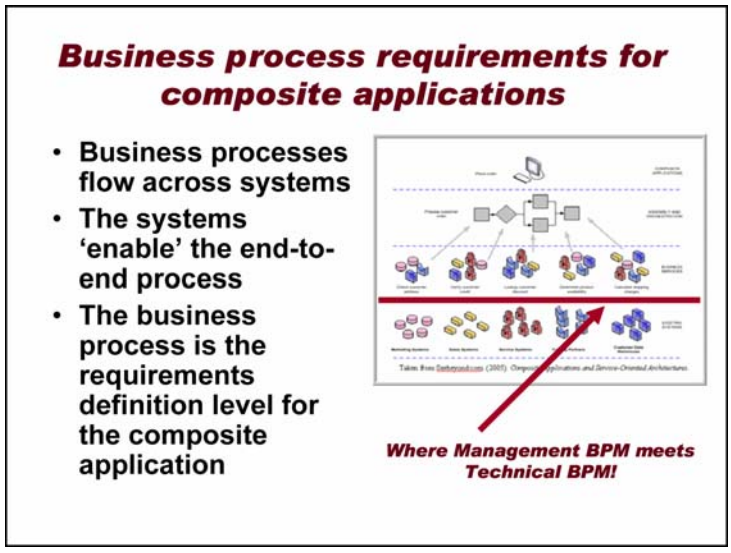


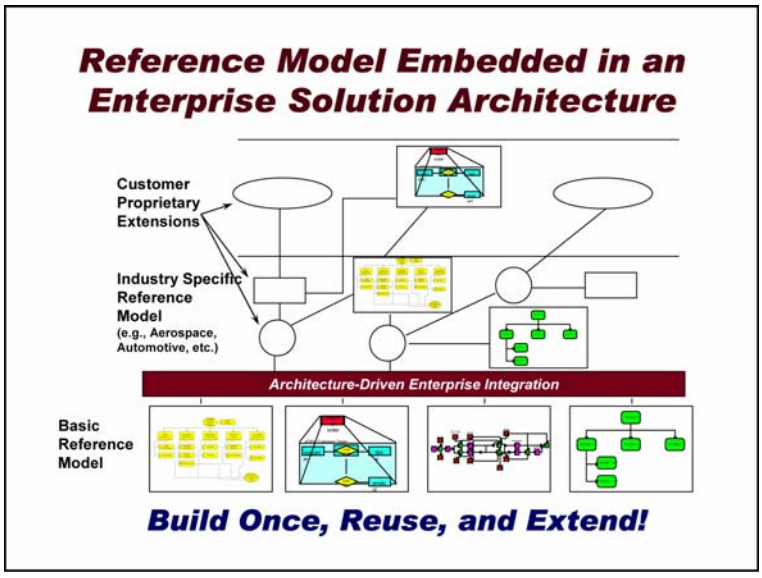

Figure 22 clearly demonstrates why a business process solution architecture is critical for SOA success. If BPM is relegated to the technical level (i.e. Technical BPM), there is no need for a business process architecture. At this level, the BPM is only used for interfacing applications, which may be more efficient (from a technical point of view), but provides no additional capabilities or competitive advantage. Interfacing is interfacing, whether presented as a service or a common application program interface.

However, if Business BPM is considered, then business process integration requires that the full architecture be considered. The broader business process architecture defines the requirements for the system interfaces. It is clear from Figure 2 that Oracle absolutely understands this point. The Fusion architecture is designed to drive Technical BPM requirements from Business BPM, and all is accomplished within a loosely coupled SOA framework. Vertical integration is assured.

## 14 Reference models

A reference model is one or more preengineered and integrated organisational views. For example, one type of reference model might be a business process (one view of an organisation) and a depiction of the data flows (another view of an organisation) that are aligned with the business process. Standard software either implicitly or explicitly executes predefined business processes; hence, by definition is based on the reference model concept. The main benefit of a reference model is that certain tedious views (e.g. the data view as realised in a data model) do not have to be individually developed for every implementation project. The idea is to design and develop once, and then refine and replicate many times. The reference model may have to be tailored for individual organisations, but this effort is significantly less than approaching each implementation as a new software project. Classic references for the design and use of reference models are Hars (1994), Hars and Scheer (1992), Keller and Meinhardt (1994) and the seminal work on integrated software by Scheer (1994). The concept is further explained with the aid of Figure 23.

**Figure 23** The concept of a reference model




The lower level of Figure 23 indicates the core solution components that are available for reuse and tailoring. These reusable components could be business processes, function models, organisational models, data models or any other relevant enterprise view.

Of course, these views could be designed from start to finish, but the whole idea of a reference model is to avoid repetitive work, so the reference components are used as accelerators.

The core reference components cannot address all situations, so it may be necessary to include an industry-specific reference model. That is, it may be necessary to use components that relate to chemical, automotive, aerospace or any available specific reference model that includes reusable components that are specialised for a specific situation. Of course, it is impossible to cover all situations in a reference architecture, so there will always be proprietary extensions. Still, the basic premise of reference models is that implementation acceleration can be realised by accessing and using reusable components.

The most well known reference model in the literature is probably the SAP reference model (Keller and Meinhardt, 1994), but there are newer and evolving reference models that are receiving wide-spread use in industry. Consistent with the theme of this paper, it is noted that all of the reference models discussed below have a business process orientation and can be presented within the context of an enterprise solution architecture.

## *14.1 Supply chain council operational reference model*

The Supply Chain Operations Reference (SCOR) model is the product of the Supply Chain Council (SCC), "an independent, not-for-profit, global corporation with membership open to all companies and organizations interested in applying and advancing the state-of-the-art in supply-chain management systems and practices". Initially developed in 1996, the SCOR model has been widely adopted by industry and government as a common measurement framework for expressing the fundamental operational components of a supply chain. At the macrolevel, SCOR identifies the following Level 1 supply chain processes: plan, source, make, deliver and return. These five Level 1 processes decompose to more than 200 Levels 2 and 3 processes.

SCOR is a framework that provides common language and common processes for depicting and communicating supply chain concepts. The SCOR documentation addresses many of the boundaries of the SCOR model:

- "It does not attempt to describe every business process or activity. Specifically, the Model does not address: sales and marketing (demand generation), product development, research and development and some elements of post-delivery customer support".

- "The Model is silent in the areas of human resources, training, and quality assurance among others".

The SCOR Model was constructed from a manufacturing point of view. A quick examination of the Level 2 processes confirms this orientation. The Level 1 processes are decomposed into second level processes according to a products' method of manufacturing (e.g. M1 Make-to-Stock, M2 Make-to-Order, M3 Engineer-to-Order). The SCOR Model defines processes at a level above that are needed for an executable supply chain design. This is an important point, in that, the model is not tailored to any particular firm. Each implementation of the SCOR model requires that the Level 4 execution environment be modelled as an organisation-specific solution.

The overarching SCOR model serves as a tool for developing an enterprise-specific supply chain measurement framework.

> "The [Supply Chain] Council has focused on general process levels and does not attempt to prescribe how a particular organization should conduct its business or tailor its systems or information flows. Every organization that implements a supply chain benchmarking initiative using the SCOR-model must extend the model, at least to Level 4, using organization-specific processes, systems, and practice."

Hence, the SCOR model is a true reference model.

Some organisations have used the SCOR model as a basis for designing architectures. This can be done if necessary,[8] but the true power of the SCOR model is through its use as a measurement and management model on top of an operating supply chain. The SCC provides process-level KPIs that have been approved though consensus by industry groups comprised of Supply Chain Management (SCM) professionals.

The performance measures associated with the SCOR model may not be perfect, but they are widely accepted by industry, and they have evolved as a type of market-based standard. This is important, because it is extremely difficult for firms to benchmark, if they are not measuring the same KPIs, and if they are not constructing the KPIs the same way. So, the main contribution of the SCOR model is a standardised supply chain performance measurement framework that can be used across organisations. The primary limitation associated with trying to implement the SCOR model is the ability to automate the data collection required to support the framework. Many SCOR model implementations never reach full potential because of the resources required to extract and properly compile data for the construction of the KPIs. Unless the data collection process can be automated, organisations must make a significant effort to keep their SCOR-based measurement projects underway. Some of these problems are discussed by Cavusoglu et al. (2001).

The SCOR model has been instrumental in institutionalising the wide-spread acceptance of business process oriented reference models as a tool to support organisational architecting and measurement. Of course, the SCOR model is limited in that it only applies to supply chains, so that has led to extensions in other areas by other researchers and practitioners.

## *14.2 Design chain operational reference model*

SCOR was never intended to be a full enterprise reference model. The SCC has extended the reference model concept to include two additional business process oriented reference architectures:

- the Design Chain Operational Reference (DCOR) model
- the Customer Chain Operational Reference (CCOR) model.

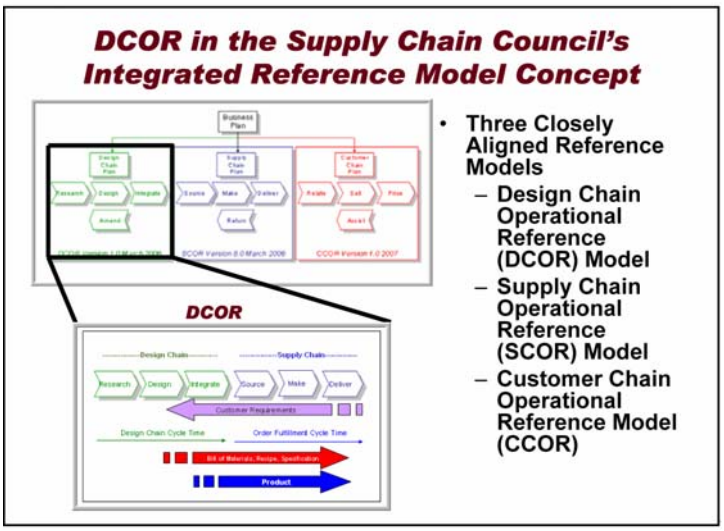
The first version of the DCOR model was released in 2006, and the CCOR model will be released soon. This paper only discusses the DCOR extension, but the concept and use may be extended directly to CCOR, with only the business processes being different. The SCC's concept, highlighting DCOR, is presented in Figure 24.

The DCOR model extends the SCOR model into the core of the enterprise, focusing on those business processes that are known in the literature as PLM processes. The DCOR model is defined by five major business processes:

- *Plan (design chain)*: the development and establishment of courses of action over specified time periods that represent a projected appropriation of design chain resources to meet design chain requirements.
- *Research*: the research management process encompasses the identification and decomposition of research topics, obtaining and synthesising of information and evaluation and publishing or archiving of research findings. This includes the identification of sources of supply, sourcing and validation of materials/products against requirements.
- *Design*: the design management process encompasses the refresh of definition, creation, analysis, testing and release of form, fit and function of an existing product. This includes reviewing and adjusting sourcing, manufacturing, testing, servicing and disposal processes.
- *Integrate*: the integrate management process encompasses releasing refreshed product and new product definitions to Supply Chain for execution and releasing refreshed and new product design documentation to marketing and support organisations.
- *Amend*: the amend management process encompasses the gathering and analysis of product design issues and manufacturability feedback for current products.

As with the SCOR model, the DCOR model could be used as an overarching architecture view, but its strength is through its ability to be operationalised as a measurement and management framework. The same implementation principles apply as they do with the SCOR model. Company-specific implementation details are included at lower levels, with the upper levels providing the common measurement and management reference framework.

**Figure 24** DCOR as a component of the SCC reference model concept



### *14.3 Reference model summary*

Reference models are a standard component of the implementation management toolkit, and Architecture-driven Enterprise Integration leverages reference models to the fullest extent. There are many types of reference models, and the Architecture-driven Enterprise Integration methodology can potentially leverage any type of accelerator that is available. However, since the business process view dominates implementation, business process reference models are the primary interest of this paper. The strength of the architecture-driven methodology is that all components are not rebuilt for each solution, and reuse is a requirement. The point-of-view of the solution architecture, however, is important. A reference model for a collection of financial processes is not the same as for a collection of supply chain processes, but the basic concept is the same, independent of the type of business process. Once the content is developed, it is reused and usually with refinement and extensions.

## 15 Special topics

This section contains a number of special topic areas that rely on BPM for enablement. In all cases, consistency requires that each application area be considered from an Architecture-driven Enterprise Integration point of view. The application areas are not presented in any particular order, and the list is not exhaustive.

### *15.1 Supply chain management*

Enterprise Integration is usually divided into two components:

- *Intra-Enterprise Integration*: Enterprise Integration methodologies and solutions applied inside a particular enterprise.
- *Inter-Enterprise Integration*: Enterprise Integration methodologies and solutions applied across the Extended Enterprise.

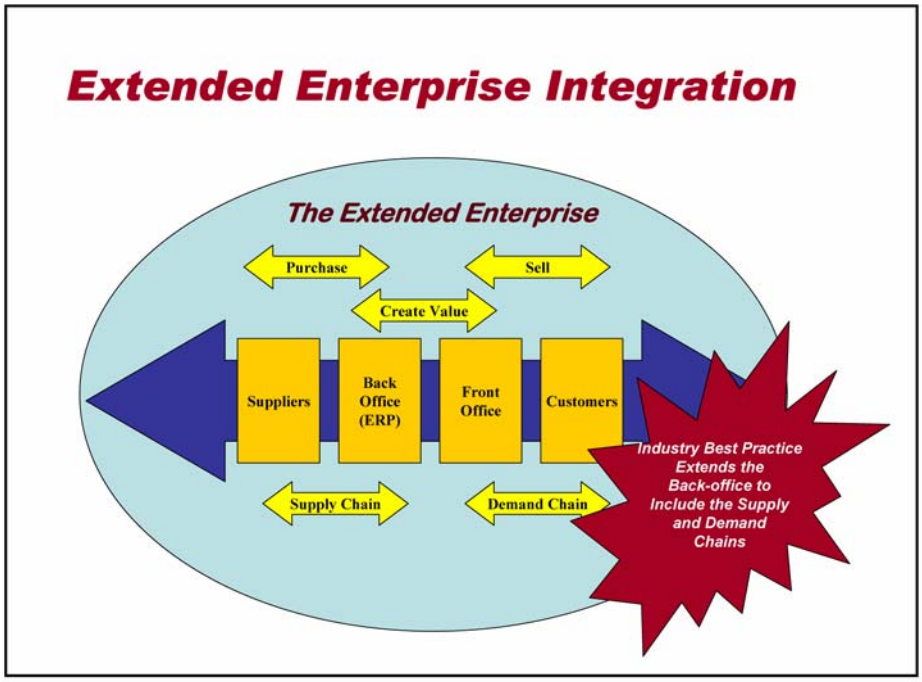
With Extended Enterprise Integration, Intra-enterprise Integration is ‘extended’ to include other entities within the integration domain. These other entities include customers, suppliers, partners and other organisational claimants. The concept is shown in Figure 25. By its very nature the Extended Enterprise is process oriented. Orders originate through customer interaction in the Demand Chain, and the same orders are fulfilled in the back-office and the Supply Chain. This is classical order fulfilment, which is a critical end-to-end business process in many organisations. This business process can be managed, controlled and improved; so the concepts of Business and Technical BPM apply throughout.

This section focuses on one segment of the Extended Enterprise, the Supply Chain. Supply Chain Integration (SCI) is the seamless passing of information (on a need-to know basis) across all supply chain participants. While many advances have been made in recent years, SCI is a non-trivial accomplishment, and remains an elusive goal for many companies.

SCM is a strategy where buyers and sellers collaborate to bring greater value to the customer. The Collaboration includes Supply Chain Planning (SCP) and Supply Chain Execution (SCE) activities. Effective SCM enables business to make informed decisions

along the entire supply chain, from acquiring raw materials to manufacturing products to distributing finished goods to the consumer. At each supply chain segment, businesses make the best choices about what their customers need and how they can meet those requirements at the lowest possible cost. SCI enables SCM. Supply chain software applications can provide benefits such as reduced costs, improved quality and product design and effective inventory management. It is very difficult to separate SCM from SCI, so this paper uses the term SCM to include both concepts.

**Figure 25** The Extended Enterprise Integration model




SCM includes the design, planning, execution, control and monitoring of supply chain activities with the objective of creating net value, building a competitive infrastructure, leveraging global logistics, synchronising supply with demand and measuring performance globally (Jakovljevic, 2004). BPM is important for the planning, execution and control of supply chain activities, but this section focuses on two of the concepts, SCP and SCE. The intent is to demonstrate that both concepts are business process oriented, and both should be managed within the context of an enterprise solution architecture.

## *15.2 Supply chain planning*

SCP is the execution of business processes that plan supply chain activities. In particular, SCP is the determination of a set of policies and procedures that govern the operation of a supply chain. Planning includes the determination of marketing channels, promotions, respective quantities and timing, inventory and replenishment policies and production policies. Planning establishes the boundaries within which the supply chain will operate

(Jakovljevic, 2004). The primary business processes are demand planning (including demand forecasting), capacity planning and scheduling (jobs, workforce and resources). The following analogy can be used to relate SCP to SCE. SCP is the constructing of the blueprint and SCE is the building of the house.

## *15.3 Supply chain execution*

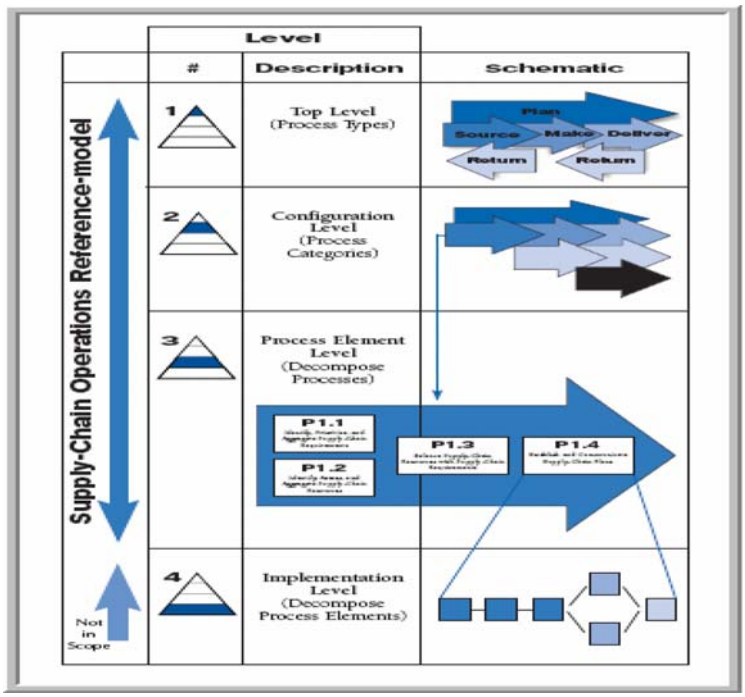
SCE is the realisation of an organisation's supply chain; that is, the automated execution of supply chain functions. For example, the execution of the plan may require creating and sending a purchase order, tracking orders, updating inventory, etc. SCE includes all activities that transform a planned supply chain into a working, live supply chain. SCE software solutions monitor and enable management to the execution of the supply chain. SCE is enabled by

> "execution-oriented software applications for effective procurement and supply of goods and services across a supply chain. It includes manufacturing, warehouse, and transportation management systems; as well systems providing visibility across the supply chain" (Jakovljevic, 2004).

## *15.4 BPM and SCM*

The alignment of BPM with SCM is straightforward. The previously discussed SCOR model delineates the business process orientation of SCM, albeit at a very high level. The process orientation of the SCOR model is shown in Figure 26.

**Figure 26** Process oriented SCOR model




*Source*: Supply chain council

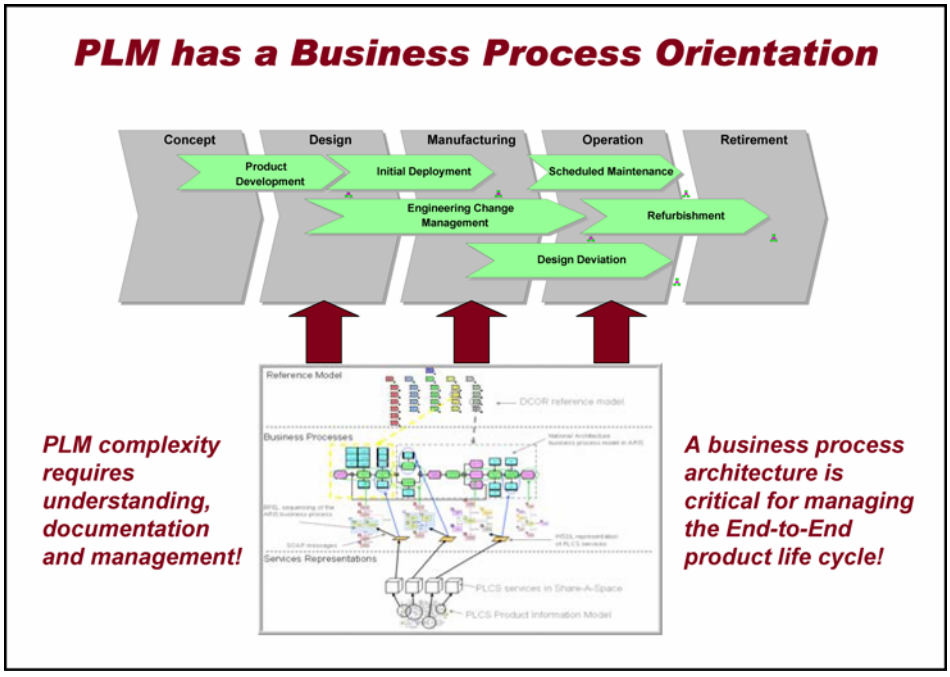
As a further indication of the increasing focus on BPM in supply chains, Smith (2002) notes the emergence of a new implementation area called Supply-Chain Process Management (SCPM). SCPM is enabled by software that uses alerts indicating when events are not occurring as planned, so that the supply chain can be 'resynchronised'. Smith attributes SCPM to a research study by ARC, but in fact the field has grown over the years, and is now known as Supply Chain Event Management (SCEM). The approach is widely implemented in many enterprise products, including those from Oracle and SAP.

## *15.5 Product lifecycle management*

Competitive advantage derives from having the right product at the right place at the right time. This is challenging for any company, since managing the product lifecycle requires working across many organisational units, including design, sourcing, material management, inventory management, environmental and eventually SCM. In some cases the management process may even include outsourced logistics. These enterprise activities are not independent, as they are linked by end-to-end business processes that flow across the organisation and the extended enterprise. PLM is a management methodology and associated business processes, enabled by technology, that help companies manage across the entire product lifecycle.

By definition, PLM has a business process orientation, and the product lifecycle can be described as a pure value-added chain. The process-oriented relationship is shown in Figure 27.

**Figure 27** Product lifecycle management value chain



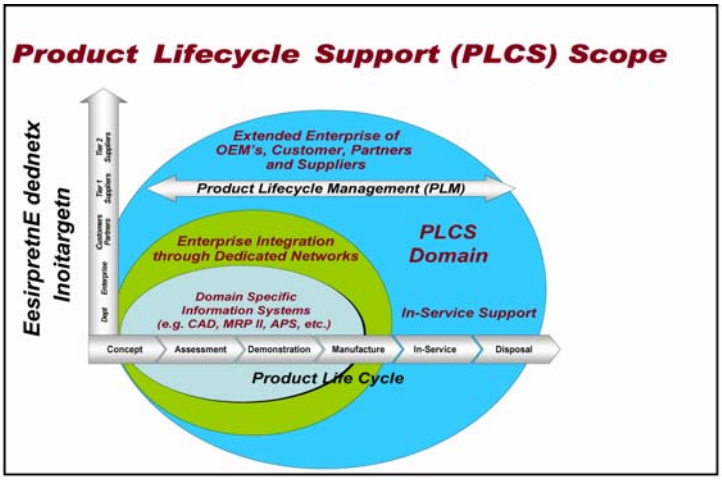
The complexity of the product lifecycle must be noted. From our work, it is clearly one of the most complex business processes within the enterprise, and it requires significant management attention since it is one of the most critical enterprise processes. There are two ways to address the complexity. The PLM process could be enabled with a single vendor point solution, but this often adds additional complexity since interfaces to other enterprise systems must be managed. The other approach is to use a standards-based approach, where international standards are adopted to address PLM interoperability. This latter approach is probably the most promising for the long run, but standards-based solutions are still maturing, and it remains to be seen how they will be widely implemented in vendor products. The Product LifeCycle Support (PLCS) standards-based approach is probably the most promising, and it is presented in the following section.

## 16 Process oriented PLCS

The Standard for the Technical Exchange of Product-model (STEP) AP 239[9] or the Product Lifecycle Support[10] standard was developed specifically to address sustainment for extended lifecycle products. While PLCS defines data models and dictionaries for lifecycle data, it is also a superset of STEP PDM Schema[11] to represent product data including references to other STEP Application Protocols (APs) for different phases of the lifecycle. Therefore, PLCS has the potential to comprehensively represent all lifecycle product data.

In addition to the capabilities of PDM Schema, PLCS supports multiple configurations of products (as-designed, as-maintained, as-built), serialisation of parts, multiple views of product data, systems engineering, extended properties, messaging, technical data packaging and access control. Applications for PLCS include failure analysis, product performance, training, maintenance analysis, repair analysis, distribution and transportation, stock management and requirements management. The scope of PLCS is discussed by Iyer and Gulledge (2005), and is shown in Figure 28.

**Figure 28** The PLCS international standard




*Source*: Iyer (2005).

By definition, PLM is process oriented. The process at the highest level can be presented a value chain, beginning with concept identification and ending with disposal. Figure 7 provides an example of the high-level value chain when documented as a process map. PLM, as Figure 7 suggests, provides a natural case study for Architecture-driven Enterprise Integration.

## 17 Logistics

Logistic*s* is the process of managing the acquisition, movement and storage of materials, parts and finished goods (and the related information flows) through the organisation and its marketing channel. This movement is accomplished in such a way that relevant performance measures are optimised while enabling the fulfilment segment of the end-to-end order management business process. Logistics, by definition, is process oriented, and in many organisations logistics is a core business process.

Many companies provide software solutions to enable logistics business processes, and the key design issue that precedes implementation success is related to the combining of planning and execution into a single solution. While this is not an absolute requirement for providing a solution, it is indeed a requirement for providing an efficient solution. The critical component of a logistics solution is the collection of business processes that are known as Transportation Management. Transportation Management will be discussed below in the context of Third Party Logistics (3PL) providers. The 3PL market is indeed robust, as indicated in the review by McCrea (2007). McCrea reviews over 23 vendors that are providing solutions to this market segment. Many of these solutions are what Kirchmer (1999) calls function-oriented solutions, and admittedly very few are presented and marketed as process-oriented solutions, even though they enable business processes. This deficiency in the literature creates a tremendous opportunity for solution providers who are willing to reposition their products to align with critical logistics business processes. In the next section, we demonstrate this competitive capability using the solution provided by Oracle Corporation.

### *17.1 3PL reference models*

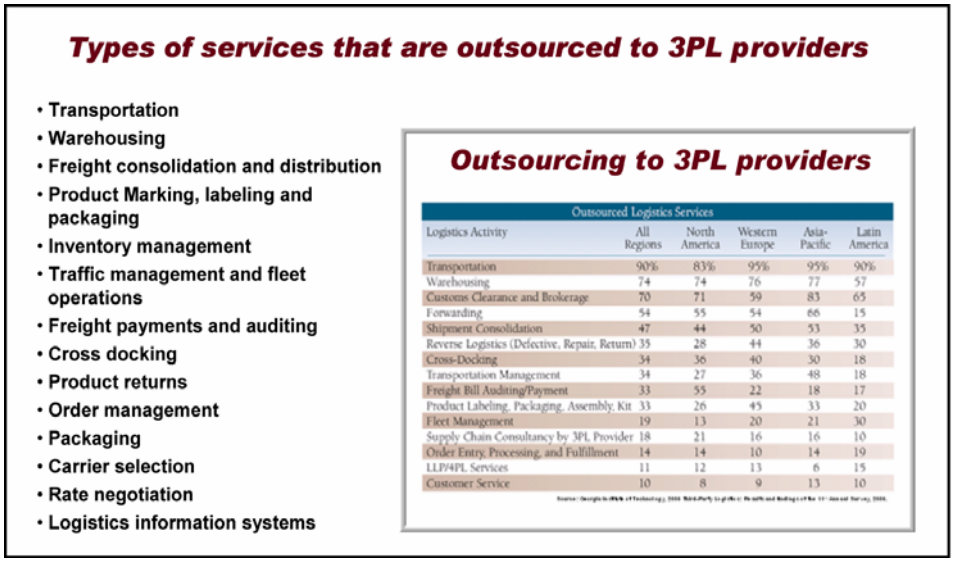
3PL providers are organisations that manage and execute particular logistics business processes, utilising their own assets and resources, on behalf of another company. 3PL providers are transportation, warehousing and other logistics related service providers employed to assume tasks that were previously performed in-house by the client. 3PLs can be either asset-based (owns a fleet and moves shipments) or non-asset-based. In short, 3PL providers provide outsourced logistics business processes to their customers, with a focus on transportation management, and in accordance with appropriate service level agreements.

One also encounters Fourth Party Logistics (4PL) providers. A 4PL provider is a supply chain integrator that assembles and manages the resources, capabilities and technology of its own organisation with those of complementary service providers to deliver a comprehensive logistics solution. In general, a 4PL is light on assets, and is an aggregator and manager of 3PLs. There has been a lot of speculation as to how this concept could really be put into practice in its entirety and whether potential 4PL providers would ever live up to the original definition. A lot of emphasis was placed on

the 4PL provider being a single point of contact for the shipper, while becoming an integrated part of their business to the point of representing their logistics department. Being non-asset based was also regarded as a fundamental feature of a 4PL provider as it would, in-theory, ensure that they would be 'neutral' in selecting the partners for the shipper.

The emphasis of this paper is on 3PL providers, and the types of services that they typically provide are presented in Figure 29. Figure 29 definitely includes more business processes than transportation, but in this field of specialised information systems, a Transportation Management System (TMS) could include all of these business processes. In fact, a good TMS should be able to enable the critical business processes of a 3PL. In the following section we use a reference model approach to emphasise this key point, and use a case study example from Oracle Corporation as part of the presentation. However, to this point, we note that logistics as implemented by a 3PL provider is clearly process oriented, and must be managed as an end-to-end business process scenario within the context of an enterprise solution architecture.

**Figure 29** Third party logistics providers

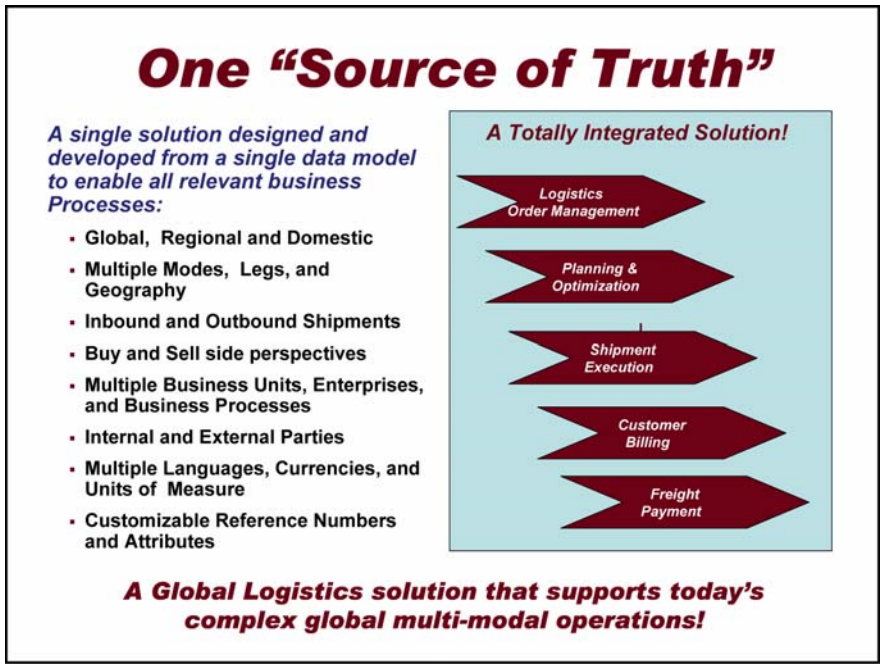



*Source*: Rabinovich et al. (1999) and Sink and Langley (1997).

## *17.2 Transportation management*

A TMS works with one or more transportation modes, including ocean, air, rail, truckload, less-than-truckload and private fleet with a core objective of managing the physical flow of goods. They also manage the flow of transportation related data, documents and money, while providing performance management and collaboration capabilities (McCrea, 2007). Figure 30 depicts the capabilities of the robust solution from Oracle Corporation.

**Figure 30** Oracle transportation management (Formerly G-Log)




As the figure shows, the solution comprises more than the core business process: Manage the Physical Flow of Goods. In fact, the Oracle solution encompasses the critical 3PL business processes, and this is accomplished with an architecture that includes planning and execution in the same data model, and deployed in a single instance of the software product.

The key point of this section is that a TMS is business process oriented and it can be documented in a reference model. This reference model can be used as previously described, with a primary focus on gap-fit analysis and the effective alignment of the TMS software solution with organisational business processes. There is no reason that a TMS should be implemented any differently than any other enterprise software solution. Figure 31 shows a function view of a reference model for the Oracle solution and particular business processes within functions. As previously mentioned these views are powerful accelerators for ensuring that the software is properly implemented.

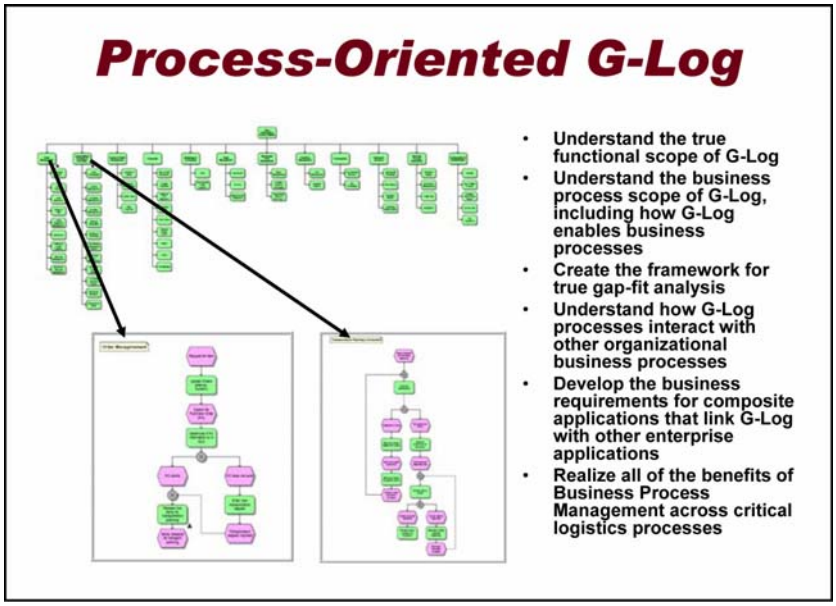
This case study is instructive because it also shows how another class of solutions should be implemented from a business process orientation. While not explicit, it should be obvious to the reader that the full TMS solution could be documented and managed in an enterprise solution architecture, further arguing for an Architecture-driven Enterprise Integration approach to logistics implementation.

## 18 Business performance management

Business Performance Management (BPM) is an umbrella term that describes all of the processes, methodologies, metrics and systems needed to measure and manage the performance of an organisation. It is a top-down approach to optimising business

processes. BPM is sometimes called by the names 'Enterprise Performance Management' or 'Corporate Performance Management'. BPM (Business Performance Management) should not be confused with BPM (Business Process Management), a more bottom-up approach to understanding and improving business processes. Wyse (2006) provides a good review of these concepts.

**Figure 31** A reference model for oracle transportation management (Formerly G-Log)



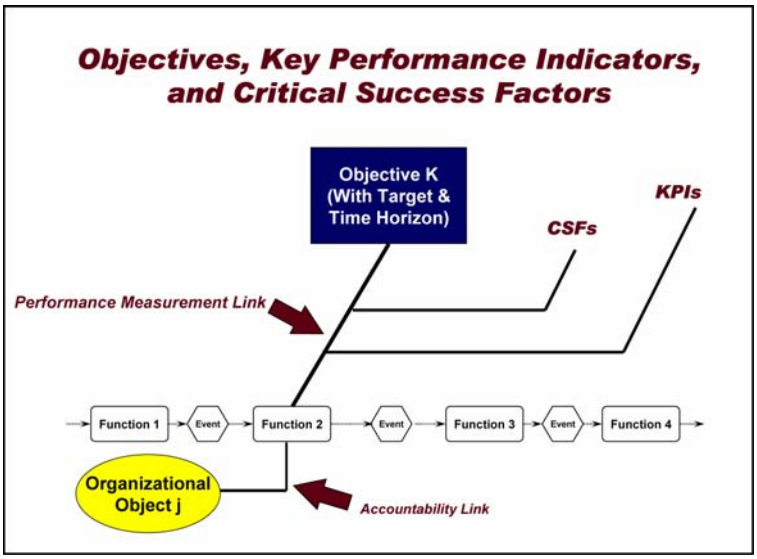

BPM (Business Performance Management) is a type of business intelligence, providing key information to decision makers. Information flows enable business processes, so by definition, BPM has a business process orientation. In fact, a key characteristic of BPM is that data are aggregated by business processes, which is consistent with the fact that managers define their work through the execution of business processes. For example, a common technique that is implemented under the BPM umbrella is the management and monitoring of KPIs. KPIs are linked to the functions that are sequenced in business processes. This is truly the foundational link between BPM (Business Performance Management) and BPM (Business Process Management). The key relationships are shown in Figure 32.

Organisational objectives define the managerial direction that is provided by senior executives. Particular objectives, in this case 'objective *k*', apply to organisational functions. In the figure, objective *k* applies to function 2. If the objective has a target and a predicted date of accomplishment, then it is possible to establish performance measurement. If performance measurement is possible, then KPIs can be defined and monitored, and the same is true of critical success factors. Of course, someone has to be responsible for ensuring that the objective is pursued, and in this example, that unit is 'organisational object *j*'.

All relationships in Figure 32 must be holistically managed, and this is a major source of the linkage between the strategy and process levels in Figure 1. From a practical point of view, if business processes define how managers work, and if KPIs measure

management performance, the KPIs must have a business process orientation. From a foundational point of view, all of the concepts in Figure 32 must be maintained within the context of an enterprise solution architecture, and they are key to the concept of Architecture-driven Enterprise Integration.

**Figure 32** Critical linkages between BPM and BPM




## 19 Lean Six Sigma

Lean Six Sigma is a disciplined approach for driving variation from business processes. The approach is to drive down process variation so that 2–3 defects are realised per one million opportunities. Lean Six Sigma is sometimes confused with Lean Principles that are used to reduce waste. The concepts are related, but Lean Six Sigma is focused on reducing process variability in striving for perfection. By definition, Lean Six Sigma has a business process orientation; that is, the whole concept is about reducing the variability in business processes. Given that fact, Lean Six Sigma should be applied to end-to-end business processes that are embedded in enterprise solution architectures. However, care must be taken with applying Lean Six Sigma to segments of end-to-end processes. If Lean Six Sigma is used to optimise a particular process segment that is embedded in a sequence of process segments, then suboptimisation is most certain to occur. One runs the risk of creating an efficient bottleneck. A good summary of Lean Six Sigma is provided by the Aberdeen Group (2006).

Process mapping is stable in Lean Six Sigma. As discussed above, suboptimality is a significant risk if Lean Six Sigma is not applied to end-to-end business processes. Therefore, Lean Six Sigma should only be applied to those end-to-end processes that are relatively stable in the enterprise solution architecture. This requirement maximises the probability of successfully institutionalising Lean Six Sigma improvements and also ensures that any required process changes are included in emerging information system initiatives.

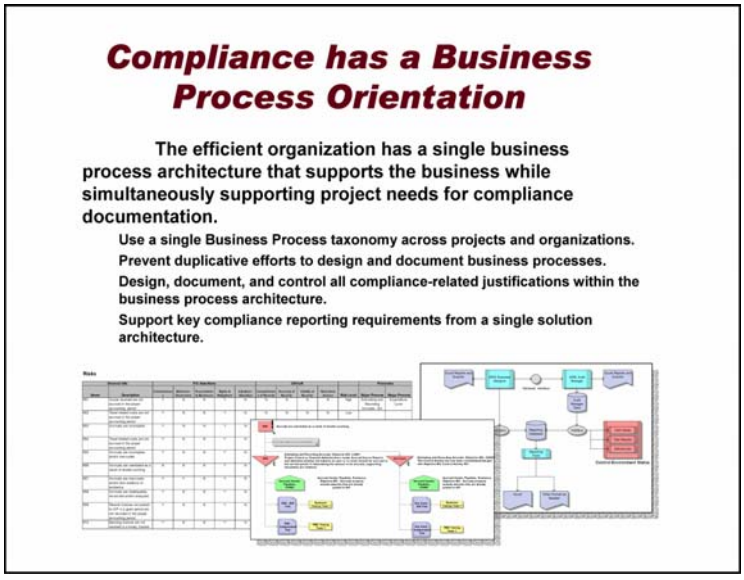
## 20 Compliance

Compliance, in the context that it is used in the information management discipline, is an attempt by the complying organisation to demonstrate that it is conforming with externally imposed laws, regulations or policies. Organisational efficiency can never be improved by imposing additional legal constraints, but effectiveness in terms of social responsibility may be enhanced. That is, compliance may be for the good of employees, stockholders, or other stakeholders who are directly involved with the day-to-day management of the organisation.

Compliance can take many forms. The rules could be written, and management could be legally responsible for complying. However, written rules are a matter of interpretation, and they are never sufficiently precise to define organisational requirements. Sarbanes-Oxley (SOX) financial compliance is a good example of such rules. Organisations struggle to comply, and even at great cost, but the benefits of such compliance activities are still questioned.

Rules are not typically static. In some cases the execution of a rule may trigger the execution of other rules. Rules sequencing may not technically be considered a business processes, but it can be classified as a procedural model, which means that the rules may be modelled using BPM methods. In fact, extended event-driven process chains are often used to develop procedural models. The procedural model helps one to understand the totality of the rules and how they relate to one another, which is an absolute requirement for designing a compliance solution. The basic relationships are shown in Figure 33.

**Figure 33** Business process oriented compliance




As indicated in Figure 33, many companies document compliance rules in tables or spreadsheets. This does provide some order to the rules, but in this format it is extremely difficult to understand the sequencing and the interrelationships among the rules. If the rules are aligned with the business processes that they affect, the sequencing and

interrelationships are automatically documented for management, control and reporting. Furthermore, other modelling methods may be used to document how the rules are implemented in systems and for understanding who is responsible for ensuring that particular rules are enforced.

If the enterprise solution architecture is managed in a single integrated repository, compliance reports can be generated in the same way as other enterprise reports; for example, portfolio management, interface management, etc. While many accounting consultants do not view compliance as an architecture issue, this discussion demonstrates the power of managing compliance from a business process orientation. Compliance is placed under configuration management control, and it is managed as part of the enterprise solution. This removes much of the uncertainty from compliance, while simultaneously providing a technical compliance environment that can be managed over time.

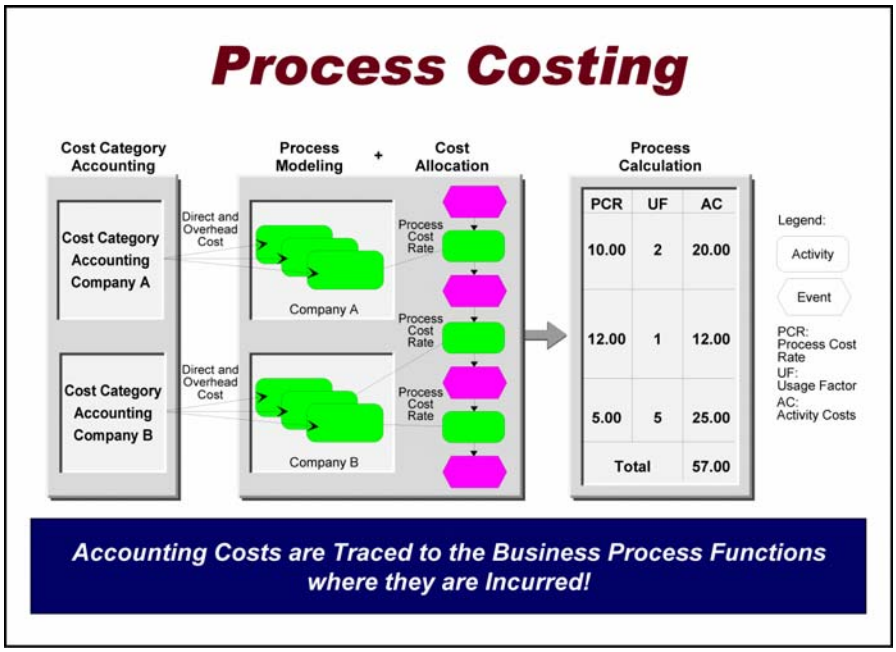
## 21 Process costing

As this paper has repeatedly demonstrated, a business process orientation is effective for enterprise management. For controlling, it is important to understand how BPM concepts relate to accounting, where in many cases the methods are still strongly oriented on traditional accounting structures. In most cases the cost centre structure corresponds to an hierarchical functional-oriented segmentation of an enterprise. If a process-oriented organisational form is implemented, but the traditional cost centre division is maintained, the organisation must administer two different organisational models within the same enterprise. Cost centres are definitely important, but from a management point of view one has to know how the cost centres relate to business processes.

For these reasons managers have historically maintained the traditional cost centre division, while simultaneously trying to reconcile the cost centre structure with the organisation's business processes. One implementation alternative would be to define business processes as cost centres. The relevant accounting costs can then be directly traced to the business process functions where they are incurred. This cost tracing is shown in Figure 34.

The figure is straightforward in concept, and from a managerial accounting point of view is most logical for managing an efficient organisation. These methods for tracing cost to business processes can be attributed to research in the USA and Germany. The relevant literature is related to Activity-Based Costing originating in the USA (Cooper and Kaplan, 1988; Johnson and Kaplan, 1987) and the related German research in process cost accounting (Coenenberg and Fischer, 1991; Horváth and Mayer, 1989). While presented slightly different, both approaches are grounded in the same principles. Since managers perform work through the execution of business processes, the costs associated with business processes are a critical performance measure for controlling purposes. Therefore, the methods presented in this section are directly related to business process oriented solution architectures that align performance measures with business processes to provide the linkage between the strategic and process levels in Figure 1.

**Figure 34** Process costing




*Source*: Gulledge et al. (1997).

## 22 Summary and conclusions

This paper presents a holistic view of BPM, where BPM is presented as the critical component that is central to designing, implementing and monitoring enterprise integration initiatives. While managers and technologists may view organisations differently, this paper argues strongly that the business process view must dominate and that enterprise integration solutions must be implemented from a business process orientation.

To bolster the argument an enterprise integration framework is proposed, which is called Architecture-driven Enterprise Integration. This framework has many views, and it forms the basis for testing many related hypotheses. Its principle characteristics are described as vertical integration as shown in Figure 1 and horizontal integration in the form of end-to-end business process scenarios as shown in Figure 21.

This paper shows how the architectural framework encompasses many common problems that are typically addressed by enterprise architectures, including solution design, software selection, information system alignment and other traditional enterprise architecture implementation areas. The traditional areas are extended, and this paper shows how the framework addresses SOAs and the design and implementation of composite applications.

This paper also argues that the more traditional areas of business process analysis should be reexamined within the context of a solution architecture framework. These include BPI, Business Process Reengineering, Lean Six Sigma, etc. Business processes are not ‘unconstrained’ in organisations, so in many cases they are not candidates for

business process reengineering. Business processes are bound by systems, and while every process is a potential candidate for reengineering, in a practical sense, significant changes may not be possible. Information system realignment to the reengineered processes may be cost prohibitive.

The final sections of this paper are focused on particular applications; demonstrating how they fit within the framework. These application areas include SCM, Global Logistics, Portfolio Management, Lean Six Sigma, Process Costing and others. As noted, the framework can contain reference models, and if desired, can accommodate procedural models for defining implementation methodologies. This last part is critical for a number of reasons. If managers follow these process-oriented implementation methodologies, solution alignment with organisational processes is more likely, and furthermore, an ongoing environment for solution monitoring, controlling, upgrading and improvement is established.

Architecture-driven Enterprise Integration is not a 'silver bullet'. A framework and an implementation methodology are neither necessary nor sufficient for implementation success. However, it is asserted that one is more likely to succeed with a methodology that without.

## References

Aberdeen Group (2006) *The Lean Six Sigma Benchmark Report*, Boston: Aberdeen Group, Inc., September.

Al-Mashari, M., Zairi, M. and Okazawa, K. (2006) 'Enterprise resource planning (ERP) implementation: a useful roadmap', *International Journal of Management and Enterprise Development*, Vol. 3, Nos. 1/2, pp.169–180.

Baharmast, A. (2005) 'An enterprise architecture-driven solution selection methodology', *Proceedings of the Open Group (Europe) IT Architecture Practitioners Conference*, Dublin, Ireland.

Booker, E. (2000) *Enterprise Software Projects Rarely Seem to Satisfy*, InternetWeek, 28 March.

Boston Consulting Group (2000) *Getting Value from Enterprise Initiatives: A Survey of Executives*, Boston Consulting Group, March.

Bubak, O., Farley, R.L., Goebels, C., Gonzalez, P., Gulledge, T.R. and Russell, R. (2006) 'Implementing SAP from end-to-end business process scenarios', *International Journal of Management and Enterprise Development*, Vol. 3, No. 5, pp.419–437.

Cavusoglu, T., Gulledge, T.R. and Kessler, T. (2001) 'Aligning the supply chain operations reference (SCOR) model with enterprise applications: real-time value chain intelligence', in S.I. Gass and A.T. Jones (Eds). *Supply Chain Management Practice and Research: Status and Future*, Gaithersburg: National Institute of Standards and Technology.

Coenenberg, A.G.and Fischer, T. (1991) *Prozeßkostenrechnung: Strategische Neuorientierung in der Kostenrechnung*, DBW, Vol. 51, No. 1 pp.21–38.

Cooper, R. and Kaplan, R.S. (1988) 'Measure costs right: make the right decisions', *Harvard Business Review*, Vol. 66, pp.96–103.

Elbertsen, L., Benders, J. and Nijssen, E. (2006) 'ERP use: exclusive or complemented', *Industrial Management and Data Systems*, Vol. 106, No. 6, pp.811–824.

Eurostep (2007) *ISO:10303-239: Product LifeCycle Support*, Available at: http://www.plcs.org/.

Frye, D.W. and Gulledge, T.R. (2007) 'End-to-end business process scenarios', *Industrial Management and Data Systems* , Vol. 107, No. 6, pp. 749–761.

Gulledge, T.R. (2006a) 'ERP gap-fit analysis from a business process orientation', *International Journal of Services and Standards*, Vol. 2, No. 4, pp.339–347.

Gulledge, T.R. (2006b) 'What is integration?' *Industrial Management and Data Systems*, Vol. 106, No. 1, pp.5–20.

Gulledge, T.R. (2007) 'Enterprise services oriented architectures and end-to-end business process execution', *Journal of the Chinese Institute of Industrial Engineers*, Vol. 24, No 4, pp.268–277..

Gulledge, T.R. and Simon, G. (2005) 'The evolution of SAP implementation environments: a case study from a complex public sector project', *Industrial Management and Data Systems*, Vol. 105, No. 6, pp.714–736.

Gulledge, T.R. and Sommer, R. (2004) 'Splitting the SAP instance: lessons on scope and business processes', *Journal of Computer Information Systems*, Vol. 44, No. 3, pp.109–115.

Gulledge, T.R. and Sommer, R.A. (2003) 'Public sector enterprise resource planning', *Industrial Management and Data Systems*, Vol. 103, No. 7, pp.471–483.

Gulledge, T.R., Hafez, W., Ledwon, M. and Svensson, C. (2005) 'Solution architecture alignment for logistics portfolio analysis', *International Journal of Services and Standards*, Vol. 1, No. 4, pp.401–413.

Gulledge, T.R., Hayes, P., Lotterer, A. and Simon, G. (2004) 'Aligning SAP with the future logistics enterprise', *Enterprise Information Management*, Vol. 17, pp.31–44.

Gulledge, T.R., Hirschmann, P. and Scheer, A-W. (1997) 'Value-based management of inter-organizational business processes', in H. Krallmann (Ed). *Wirtschaftsinformatik '97*, Heidelberg: Physica-Verlag.

Hammer, M. (1990) 'Reengineering work: don't automate, obliterate', *Harvard Business Review*, July–August, pp.104–112.

Hars, A. (1994) *Referenzdatenmodelle: Grundlagen Effizienter Datenmodellierung*, Wiesbaden: Gabler Verlag.

Hars, A. and Scheer, A-W. (1992) 'Reference models for enterprise-wide data engineering', in Petrie, C.J. (Ed). *Enterprise Integration Modeling*, Cambridge: MIT Press.

Horváth, P. and Mayer, R. (1989) 'Prozeßkostenrechnung: Der neue Weg zu mehr Kostentransparenz und wirkungsvolleren Unternehmensstrategien', *Controlling*, Vol. 1, No. 4, pp.214–219.

Iyer, R. (2005) *Standards for Product Lifecycle Management (Presentation), PLM Technologies*, TARDEC.

Iyer, R. and Gulledge, T.R. (2005) 'Product lifecycle management for US army weapon system acquisition', *Proceedings of the International Conference on Product Lifecycle Management PLM 05*, Lyon, France.

Jakovljevic, P.J. (2004) *Glossary of Enterprise Applications Terminology: Part II*, TechnologyEvaluation.com, 29 March.

Johnson, H.T. and Kaplan, R.S. (1987) *Relevance Lost: The Rise and Fall of Management* Accounting, Boston, Harvard Business School Press.

Jost, W. (2006) *Defining BPM with the ARIS Platform, Presentation at ProcessWorld*, Miami, Florida.

Keller, G. and Meinhardt, S. (1994) *SAP R/3 Analyzer: Business Process Reengineering Based on the R/3 Reference Model*, Philadelphia: SAP North America.

Keller, G. and Teufel, T. (1998) *SAP R/3 Process Oriented Implementation*, Harlow, England: Addison-Wesley.

Kirchmer, M. (1999) *Business Process Oriented Implementation of Standard Software*, Berlin: Springer-Verlag.

McCrea, B. (2007) 'TMS steps out of the shadows', *Logistics Management*, Vol. 46, No. 2, pp.53–55.

Moon, Y.B. (2007) 'Enterprise resource planning (ERP): a review of the literature', *International Journal of Management and Enterprise Development*, Vol. 4, No. 2, pp.235–264.

Oracle Corporation (2007) *Service Oriented Archiecture (Presentation)*.

PDES, Inc. (2002) *Usage guide for STEP PDM Schema Version 1.2*, PDES, Inc. Report, Available at: http://www.pdesinc.com/.

Rabinovich, E., Windle, R., Dresner, M. and Corsi, T. (1999) 'Outsourcing of integrated logistics functions', *International Journal of Physical Distribution and Logistics Management*, Vol. 29, No. 6, pp.353–373.

Sauer, C. and Willcocks, L.P. (2002) *The Evolution of the Organizational Architect*, Spring: MIT Sloan Management Review, pp.41–49.

Scheer, A-W. (1994) *Business Process Engineering: Reference Models for Industrial Enterprises*, Berlin: Springer-Verlag.

Simon, G. (2004) *ARIS Value Engineering for SAP (Presentation)*, IDS Scheer International Partner Conference, Malta.

Sink, H.L. and Langley, C.J. (1997) 'A managerial framework for the acquisition of third party logistics services', *Journal of Business Logistics*, Vol. 19, No. 1, pp.121–136.

Smith, T. (2002) 'One supply chain sector stands out', *InternetWeek*, 28 June, Available at: http://www.internetwk.com/story/INW20020628S0001.

Sommer, R.A. (2004) 'Architecting cross-functional business processes: new views on traditional business process reengineering', *International Journal of Management and Enterprise Development*, Vol. 1. No. 4, pp.345–358.

Sommer, R.A. and Gulledge, T.R. (1999) 'Process coupling in business process engineering', *Knowledge and Process Management*, Vol. 6, pp.158–165.

Wyse, L. (2006) 'Business performance management basics: an overview of business performance management and its benefits to the organization', *TEC: Technology Evaluation Centers*, 12 April, Available at: http://www.technologyevaluation.com/Research/ResearchHighlights/BusinessIntelligence/2006/04/research_notes/prn_TU_BI_LW_04_12_06_1.asp#1.

## Notes

[1]Literally, these photographs are often in the form of presentation graphics slides, word processing documents and high-level requirements listed in a spreadsheet. These artifacts are simply not adequate for designing and deploying an integrated solution.

[2]The author thanks Mr. Olaf Geyer for the use of this figure.

[3]A complete architecture includes are relevant architectural views and is maintained in an integrated repository.

[4]Big I is as form of integration where all relevant data for a business process domain is managed in a single database instance, providing a single source of truth. This concept and other types on integration are discussed by Gulledge (2006).

[5]The discussion of E2E scenarios as requirements for composite applications is presented by Gulledge (2007).

[6]Improvement could be achieved by rearranging or deleting process functions. In some cases complete segments of the process could be eliminated.

[7]See Moon (2007) and Al-Mashari et al. (2006) for a review of this literature.

[8]See Gulledge et al. (2004) for an example where the SCOR model was used as the basis for designing a large logistics enterprise.

[9]STEP data this is a comprehensive standard that is presented in a collection of APs.

[10]See http://www.plcs.org/ (Eurostep, 2007).

[11]PDM Schema is part of STEP AP214 Conformance Class (CC) 6. While AP214 was primarily developed as an AP for automotive mechanical design processes, PDM Schema has developed into a broader standard applicable across all industries. PDM Schema is a reference information model for the exchange of a central, common subset of the data being managed within a PDM system (PDES, Inc., 2002).