*Business*Advantage™

A White Paper Describing Our Business Oriented Approach to Process Reengineering and Improvement, Process Modeling and Software Requirements Planning

TradewindsGroup

Tradewinds Group, Incorporated
Consultants to Management
Post Office Box 3601
Oak Brook, Illinois 60522-3601
www.tradewindsgroupinc.com

Copyright Notice and License Agreement

Table of Contents

Introduction

This white paper provides an overview of the business process reengineering or improvement methodology that we rely upon at the TradewindsGroup. This paper is therefore a discussion of the practices, tools and techniques that we rely upon when we work with clients to identify, model, simulate and plan for process and systemic improvement within an organization. We have formalized our knowledge, expertise and practices into a formal methodology, which is now known as BusinessAdvantage™.

BusinessAdvantage (commonly abbreviated to and known as BA) is actually divided into several evolving bodies of knowledge, including business process change, software package selection, software package implementation and customized business software development. The information contained in this document represents the current body of knowledge regarding the business process reengineering and improvement techniques found within BusinessAdvantage, as a whole.

This paper places particular emphasis on process reengineering when software packages are involved, or on what is often called technology enabled business process change. In such cases the software package is considered the centerpiece of an organization's business process improvement or reengineering objectives. However, the business process improvement and reengineering knowledge in this paper applies across the entire spectrum of project types that BusinessAdvantage can be applied to.

Work Packages and Project Planning

BusinessAdvantage also serves as a framework for the development of a project plan, that breaks any change-related project into stages or phases of work that logically divide the complexities of the project into plannable units of work. These stages work together to ensure the change effort is planned, controlled and monitored. Within each stage of this approach are specific activities, or tasks, and there are deliverables, or outcomes associated with each such task or activity.

A project stage or phase generally represents a major milestone in the project life cycle. However, further detail is needed for effective planning and execution of the project. For each stage or phase of the project a work breakdown structure is created. From the work breakdown structure work packages are developed. The work package is *a 'lower-level'* or *detailed* work plan. The work package represents *plannable units of work*. The project plan documents the details of the tasks, resources and demand placed on those resources for each assigned task.

BusinessAdvantage includes project plan templates process that can be used as the basis for identifying the stages or milestones in a project along with the specific tasks in conjunction with the project. However, our experience tells us that no two process improvement, software package implementations or software development projects are ever identical, nor is there such a thing as a 'typical' project. Therefore, tailoring a project plan specifically to your business requirements is always necessary.

Change Management and Document Control

The easiest way to loose control of a project is through unforeseen changes in scope and the mismanagement of project issues. BusinessAdvantage includes a *formal* change and issues management process that is used throughout any project and for all deliverables. A change management process thus controls both project scope and the expectations of end users.

Deliverables

BusinessAdvantage includes model documents or templates for many tasks related to project plan development and task execution. These documents and templates can be used as is, or modified to accommodate the specific needs of a given organization. It is generally recommended that 'finalized'

documents are published in Portable Document Format (PDF) and archived on permanent media, such as a CD-ROM. PDF files are viewed using the Adobe Acrobat Viewer, which is available without charge from Adobe Corporation.

Business Process Improvement: An Overview

Finding the root causes of operational bottlenecks and problems or of business issues is the essence or core of what business process improvement is all about. Although modern day authors such as James Champy, Thomas Davenport and Michael Hammer are credited with kindling the current interest in business process reengineering, the ideas are largely rooted in industrial engineering or "scientific management" concepts first devised around the turn of the last century by Frederick W. Taylor. There are several common names given to business process improvement, including business process change; business process reengineering and business process transformation. All of these terms can be used interchangeably.

In general, reengineering presumes improvement will take place by critically reviewing and reconstructing processes, generally from "the ground up". There are several approaches that can be taken with respect to executing business process improvement and both will be described in this white paper.

The first business process reengineering approach is referred to as the "clean sheet" approach while the second is referred to as the "technology enabled" approach. One of these approaches in particular – technology enabled business process improvement – is more germane to business process change or improvement using software packages.

The starting point for business process improvement is usually what we refer to as "the process audit", which presumes a "process orientation" both of which will be discussed in this white paper.

The Importance of a Process Orientation

One of the important contributions of the business process reengineering movement has been the reemergence of a process focus, something industrial engineers have been practicing on the plant floor for decades. Over time, somehow the process focus gave way to organizational structure. Unfortunately, organizational structure often times placed unnatural barriers or constraints on process flows and hence a source of waste or non-value adding activity.

Fortunately, authors such as James Champy, Thomas Davenport and Michael Hammer helped to place "process" at the center point of business activities once again, through the so called "business process reengineering revolution". When processes are reengineered, instead of processes flowing through or "aligned" to organizational structures, the organizational structures are "aligned" to the processes – or at least the organizational structure should not impede the process flow in any way. It should come as no surprise to learn that most all software packages employ a "process orientation". In fact many of these packages now incorporate business rule driven workflows that further drive forward a process orientation such that it can route or move information through a "traditional" organizational structure.

The "Clean Sheet" Approach

In the clean sheet" approach to business process reengineering, the presumption is you're "starting over, in effect with a clean sheet of paper" when designing or reengineering a given business process. The clean sheet approach is often times called "radical reengineering" as well. This approach assumes that there are "no barriers" to the amount of business process change that can be proposed or undertaken. This was the often-controversial type of business process reengineering espoused in the early 1990's by Michael Hammer and James Champy in particular.

The clean sheet approach largely ignores the constraints of available technologies and what capabilities or enablers of change may exist. A clean sheet or radical reengineering project can result in recommendations to replace some or all of an organization's existing business application software. Thus, it is important to understand the role of software packages in the clean sheet approach. If a software package can enable the changes envisioned by the clean sheet approach, so be it, otherwise the reengineered solution is largely a custom software undertaking. Thus this type of reengineering project is executed *exclusive of software package selection*.

Unfortunately, the results in the corporate world from many "radical" reengineering projects have been mixed, at best. In addition, while the clean sheet approach toward business change or improvement may suit the large multinational corporation with "deep pockets" quite well, it has not "played well" on Main Street with smaller and middle market business organizations – where the second approach, technology enabled business process reengineering has proved more popular. Smaller and middle market business organizations that have real budget realities to deal with have learned that the best way to reengineer their businesses.

The "Technology Enabled" Approach

The second approach to business process change is referred to as "technology enabled" business process change. In this latter case, the impetus driving business process change is new technology and for purposes of this discussion, its' a business application software package. Existing business processes are abandoned, changed or replaced by new processes that exist through or as a result of the technology adopted. Think of the commercial, off the shelf software package as an "agent of business change". When an organization buys a business application software package – virtually any package – that organization will undergo some degree of business process change or transformation. (Even if the organization doesn't really want any process changes!)

Although one might feel that the differences between these two approaches is largely one of semantics that is not entirely true. A clean sheet approach does not presuppose what the end solution for automating the process will be; it could just as easily require a completely custom solution or it could make use of a commercial off the shelf software package. However, under the technology-enabled approach, it is squarely the adoption of specific hardware or software technology that is driving business process change.

"Technology enabled" change versus "radical" change

The technology enabled business process reengineering approach is most often associated with software package implementations while the clean slate approach is generally more appropriate when a custom-developed systemic solution is desirable or feasible. Given the increasing popularity of software packages, the emphasis in this paper is on the technology enabled business process reengineering approach. However, the tools and techniques discussed herein will apply to either approach.

We contend that why bother going to the time, effort and expense of acquiring and implementing any software package unless the intent is to accomplish at least some technology enabled business process improvement using that software. A new software package should not simply represent "a new way to do exactly what was done before". No process benefits will accrue if such an attitude prevails. Many software packages purchased and installed to address Y2K issues were done so with this attitude. Of course maybe that's why studies and surveys show such wide dissatisfaction with the results achieved – especially from ERP systems.

It must be readily accepted that the adoption of any new technologies into the workplace will, to a certain degree, change existing business practices or processes. However, such changes may themselves not represent radical change from what the organization's present day conditions or processes. This is especially true when legacy systems exist that may already automate part or all of an existing business process, but perhaps these legacy systems don't provide integration to the level desired; for instance to achieve dramatic improvements in supply chain management.

Nonetheless, both a rethinking and retrofitting of all affected business processes is necessary in order to move forward with the implementation of new technology and hence that is why it is referred to as technology-enabled business process change. It should be well understood that when business process change is technology-driven, for instance by a commercial off the shelf package, it means *that the organization has made a conscious decision to adjust its' "old" ways of doing business*. It is simply not good enough to replace an old system with a new system and not attempt to gain at least some order of magnitude business improvements at the same time. If such is not the case, then why bother with a new system at all?

My way or no way!

In the case of any commercial off the shelf software package, included with the package is an "out of the box" way of doing business. But what's important to understand here is that the software package vendor's view on the "way of doing business" may be different, even radically so from yours. In fact, it may be so different a perspective that it is counter to the "best interests" of your business operations. This is also why it is crucial to *model existing business processes before selecting a software package*. The availability of such business process models can be used as the basis for building software demonstration scenarios that will allow an organization to compare and contrast the vendor's "way of doing business" versus your existing process.

Orchestrating Business Process Improvement

We base our process improvement work on what we believe to be an extremely simple, straightforward framework or strategy for business process reengineering work. This strategy is referred to as " The USA Principle" and it is the brainchild of Karl M. Kapp. This is not a new model by any means as Kapp first described this model in several articles in the mid-90s, including an August 1996 article entitled "The USA Principle" in Manufacturing Systems (now MSI) magazine.

Kapp advocates use of a three-stage strategy in business process reengineering. These three stages are:

  − Understand Processes
  − Simplify Processes
  − Automate Processes

Kapp contends – and we strongly agree – that *processes should not be automated until they are first understood and simplified.* We have chosen to build our approach for conducting business process reengineering work around this strategy. What's more, we have found that in practice this strategy is quite effective. This strategy applies equally to either clean sheet business process reengineering or technology enabled business process reengineering.

The Process Audit: Finding the Waste

The process audit is the method or approach used to gain a "deep understanding" of business processes and particularly seeks to identify any process inefficiency or "waste". The process audit leads to recommendations or a "process vision" for redesigned, streamlined or simplified processes. The process vision "makes the call" regarding whether the process redesign is a candidate for automation. In turn, it is this *process vision that represents the " detail" that will drive both the selection and implementation of any software package* in the technology enabled business process reengineering approach or in building custom system specifications using the clean sheet business process reengineering approach.

With that said, it is important to understand that typically a business process improvement project is not just about reengineering a particular business process or function, it is generally about *reengineering or overhauling the entire business* – it is a corporate-wide reengineering project. Certainly an enterprise wide

system undertaking such as an ERP system selection and implementation will demand this type of extensive review. Thus in the case of a process audit that is conducted in concert with a technology enabled business process improvement, a collective or enterprise-wide process vision should be the objective. The enterprise wide process vision will in turn lead to the automation of business processes and practices through the ERP system.

The process audit would follow any given business transaction on a "cradle to grave" basis throughout its' life cycle, across the entire organization. The lifeblood of any for-profit or not-for-profit is arguably the same – the provision of some good or service that is desired, needed or wanted by other entities or individuals. If the deliverables – that is the products or services the organization provides to customers can be identified, then so too can the processing threads (although perhaps with different names) that acquire or build components or end products and provision them to customers. The major processing threads in a typical manufacturing organization are defined as:

- − The Lead to Cash (Also called the Quote to Cash or Order to Cash) Process
- − The Procure to Pay Process
- − The Design to Build Process
- − The Plan to Make Process
- − The Enterprise Asset Management (or the Manage the Enterprise) Process

Although the process audit may be described from a singular perspective it is important to understand that process auditing is an iterative process, concluding when all of business processes have been understood and reconstructed or simplified in this manner.

Taking the Process View of the Organization

We advocate taking a process view of the organization when conducting the process audit. The process view of the manufacturing organization – or any organization – can also be viewed as a "systemic process", consisting of a series of "linked" acquisition, conversion and delivery processes, requiring both inputs and outputs through interaction between internal processes and the customers and suppliers to the organization.

Business processes are themselves sub systems within an overall systemic view of the organization. This systemic view of first the organization and second of each its business processes is the basis for a specific process modeling technique that we employ in our work, referred to as the SIPOC method (pronounced "sye-pahk"). The particulars of the SIPOC process modeling technique we employ will be described in later sections of this white paper.

Process Reference Models

Another approach to conducting a process audit is to use a "process reference model". Process reference models are usually industry related and are generally sponsored by trade groups or associations. These groups are usually comprised of industry suppliers and customers, educators and solution providers, including software vendors and consultants.

The Supply Chain Council, an example of one such industry group, has published a popular reference model. It is referred to as the Supply Chain Operations Reference (SCOR) model. This model has only four orientations, including:

- − Plan
- − Source
- − Make
- − Deliver

The process reference model has two important roles in business process improvement and reengineering projects. First, industry sponsored process reference models, such as the SCOR model, are usually followed

by the major software vendors. One drawback in most process reference models however, is that the reference model does usually not specify the exact implementation of a process. Thus the interpretation or implementation of a reference model may vary widely between software packages.

Second, these models are themselves considered composites of best practices. This is where the value lies during the process audit; the reference model is used as benchmark for business process improvement. The final process adopted or recommended should embrace the best practices of the reference model.

Understanding Processes

Before attempting to radically alter a business process, or overlay a current process onto a new technology, it is imperative to gain a "deep understanding" of the current process or what steps are done and why each step is completed. The deep understanding needed is gained primarily through an interviewing and write-up technique that was previously referred to as the SIPOC method. The SIPOC method is part of an overall workflow that is followed in our work.

This workflow and technique is one that works for us and has been developed based upon its practical application and refinement in the field. We have found the workflow and techniques presented here to be efficient when time is short and the reengineering effort, in general, is focused around systemic possibilities.

When considering consulting firms to assist in an organization's improvement efforts, it should be understood that every consultancy will have varying techniques for fulfilling the requirements of gaining the deep understanding of current and future processes that is necessary to fulfill the analysis and redesign of any given business process.

Don't overlook the front line staff's role in process improvement

It can be said that an organization's own staff will know more about their day-to-day workings than anyone else will. In fact, ask any one of these individuals and they likely already have a "mental list" of a half dozen or so ways they can improve their job or a given process. Thus it can be said that the line worker's role as a "subject matter expert" is every bit as important as the manager or knowledge worker when undertaking process improvement.

The organization must minimally encourage the articulation, if not the implementation of employee suggestions for work improvements. Therefore, the process auditing team should certainly seek to learn from the front line staff about any of "their ideas" regarding ways to improve processes. Being able to make one's work more productive or meaningful to the organization is can be viewed as another form of job enrichment. In addition, "buy in" to process change will be stronger when the front line staff has been involved and are able to "see" their own ideas acted upon.

Process Mapping: The Starting Point

The starting point for process understanding is a "roadmap" of the business process formed through what are known as process maps. Such process maps can take on two dimensions – a high-level organizational view and a more detailed process view. We treat organizational process mapping and business process mapping as separate entities. When an organizational or enterprise-wide perspective is necessary in a project, organizational process maps are required, when the view is more narrowly focused on a specific process, they are optional.

Process Mapping is more or less an inventory technique. The inventory of business processes at the organizational level is constructed through a series of interviews about *what is done by the organization, by whom and for whom*. For the organizational view, C-level executives should typically serve as the subject matter experts.

At the process level, the inventory of business processes is also constructed through a series of interviews about *what is done, by whom, for whom, how often and why.* Such interviews are usually conducted with the subject matter experts for a given business function – the process owners. It can be said that the business process view identifies the "what" and "when" of the business process or business function. This is an iterative step and must be repeated for each function or processing thread considered "in scope" for the purpose of the reengineering effort.

Organizational Process Maps

The Organizational Process Map (OPM) is an organizational view of the business processes that addresses at a high level, what does this organization do—what business events drive the execution of business processes within the organization.

The artifacts important at the organizational level are the inputs, such as customer orders and outputs, such as customer product shipment. Major processing threads of the organization can be identified through and should link to corporate mission statements. The organizational structure provides the fuel for asking questions as to the role, if any, that a given business unit plays within a business processing thread. The goal is to construct, for each identified business processing thread that an organization undertakes, a map that illustrates the linkages or collaborations between customers and suppliers (internal and external) engaged in the exchange of inputs and outputs (tangible, such as products and intangible, such as business documents).

The organizational process map format consists of a series of columns, or so called "swim lanes". The columns represent business units that engage in a given business processing thread, such as "Sell A Product" or "Provide Post-Sale Product Support". Thus, any business unit within the organization that touches the process in some way (even a small way) represents a column. The entries made into the columns in an Organizational Process Map are the names of the inputs and outputs to the business process.

Business Process Maps

Using the as Business Process Map, the goal is to identify all processes conducted or performed within each business process on a routine and non-routine basis. When they exist and if they are current, departmental policy and procedure manuals are sometimes helpful in constructing these maps.

While in an Organizational Process Map the swim lanes represented the suppliers and customers, in a Business Process Map they represent dimensions of time or frequency of process execution. While entries into Organizational Process Map columns were names of business process inputs and outputs, in a Business Process map they instead represent the names of specific, lower-level parts of the business processes that are executed at the respective frequency of occurrence.

The process map format consists of a series of up to six columns, or so called "swim lanes". The columns are labeled: Daily, Weekly, Monthly, Quarterly, Annual and As-Needed processes. Note that As-Needed processes can be classified as Daily Processes since technically speaking they can occur on any given business day.

SIPOC Business Models: The Core Component

The next step is modeling each of the specific business processes that had been noted on the process map. As was done with the Business Process Map, this is also an iterative process that is completed once all of the processes on the process map have been reviewed and modeled. *The technique used to document or model each business process is referred to as the SIPOC technique, which is best viewed as an interviewing, write-up and modeling technique all in one.*

From the process map, a SIPOC Process Model is produced for each of the identified processes. It can be said that this document describes or identifies the "who", "what", "where", "how" and "why" of the business process. Recall the earlier suggestion that the organization of a business organization and its business processes can be represented or viewed systematically. In fact this is the underlying assumption behind the SIPOC modeling technique. The SIPOC technique identifies the suppliers, inputs, processes, outputs and customers of every candidate process, to form what is called an "AS IS" or "current state" process model. The SIPOC "AS IS" model thus becomes the important deliverable in this stage of a business process improvement or reengineering initiative.

When practical, we attempt to complete the Business Process Map and the SIPOC Business Models related to that map in one or two working or facilitated meetings or "interviews" with the "process owners" each lasting several hours in duration. Once the Business Process Map is completed, work can begin on the SIPOC Process Model. Invariably, during the SIPOC modeling previously unmentioned business processes will emerge and must be added to the Business Process Map and obviously are then candidates for a subsequent SIPOC Process Model.

Sometimes additional meetings are needed if not all of the functional business areas, or so-called "process co-owners", were represented in the interviews but have been identified as "participants" in a given process flow. Their role in the process must be fully understood, as must any specific requirements they have for "inbound" outputs from the process being reviewed.

The SIPOC Method: How it works

The SIPOC method relies on a "cradle to grave" review of the business process under review. Usually this is a structured interview session built around the SIPOC Process Modeling document format with those who are intimately involved with the process. The process works as follows: First, the group is asked to identify the outputs (and customers) from the process, then about the inputs (and suppliers) needed by the process. Finally, the discussion shifts to the conversion process, focusing on how the inputs are transformed through processes and procedures into outputs. Sample documents are gathered or requested at this time to construct a simple "data model" corresponding to the process.

When we model business processes, our focus is on addressing a series of process related questions. For this purpose, we use a process interview checklist. This tool serves as both an "agenda" and as an instrument for quality assurance to assure our overall understanding of a client's given business process.

It is best to record working session notes on a flip chart, then attaching each flip chart sheet to the wall as the meeting progresses. It's a good way to confirm (and add to) discussion items made on a continuing basis throughout the session. An added advantage: the facilitator won't need a note taker in attendance as these flip chart pages can then be used to construct the formal SIPOC Process Model write-up document, which is illustrated in Figure 6.

Note that 3M Corporation makes a portable easel pad with "Post-It" note style paper, available through most office supply merchants. This paper provides an excellent medium for note taking during these process interview sessions. We attach each page to the wall as it is filled so that all that is said can be reviewed throughout the session. We then use these notes one again when preparing our formal documentation.

As a general rule, it is best that the sessions are conducted with two facilitators present, therefore the additional analyst or consultant can ask relevant business questions that the first person might not have thought of. This is especially true when the first person is working outside their usual area of expertise.

From Narrative to Visual Models

Many individuals prefer to model business processes through visuals, by preparing diagrams. Our preference favors a narrative format, but that is not to say that we don't also produce process-related diagrams. The fact is that some individuals understand a process more so visually than in a written format, so such visual process models are a necessary deliverable from the overall business modeling process. Also, such process charts represent an excellent way to guide a review meeting and thereby "confirm" the completed or documented process model with the original working session attendees.

While our clients appreciate visuals, they aren't as interested in paying for them. Thus, our challenge as a consulting firm is to add value to the project – through process visuals – without inflating the project's cost. We have found a product – allCLEAR – that helps us meet this challenge.

We prepare process diagrams (actually flowchart style process diagrams) using allCLEAR. This is a Windows based software product that is available through us directly, or from our business partner affiliate, PROQUIS. (www.proquis.com). We have been using this product ourselves for well over a decade now and should note that it has stood the "test of time and user interest" by surviving at least three separate ownership changes during this time period.

We like this product because it means that we spend very little time preparing the visual models themselves. The process works as follows:

- Visuals are created by "copying" the narrative processing steps from the SIPOC document that was previously prepared.

- These "narratives are "pasted" as a list of process steps into the allCLEAR program. The result? AllCLEAR instantly translates the narrative into a process-oriented flow chart, using only words and common punctuation to so.

We find this technique save lots of time, especially for the dozens of SIPOC Process Models that are typically needed in an enterprise-wide project.

Use Cases and the UML: An alternative method

An alternative to using the SIPOC method presented previously for documenting and modeling a business process is to prepare a "use case". Use cases and the closely related visual modeling language known as the unified modeling language (UML) are popular techniques for describing software system specifications.

The UML has been billed as a standard or universal method for describing and translating user requirements into software design specifications, particularly for today's object oriented software systems. However, there is nothing to say that use cases and the UML cannot be used to document and model business processes for purposes other than software development. The major difference between the use case based method and the SIPOC method is that use cases can best be described as "dialog based" and are therefore particularly adept at detailing the dialogue between "a user" and "the system".

As for UML diagrams, they are typically prepared using a special visual tool or workbench that is geared toward the software engineer. The UML defines or specifies a series of standard design diagrams to be produced by software engineers. With a little bit of creativity and energy, several of the UML diagrams are particularly applicable to process reengineering work.

However, a UML "workbench" product is generally needed to produce these diagrams. While the UML notation is included as a template set in some flowcharting software packages, UML diagrams can't be as easily created as they can through a UML workbench. In addition, these workbench products have other features that are germane only to the software engineer. Thus, when a clean slate reengineering approach is employed and it is a good chance that such work will likely result in the need to develop custom software, the UML and use cases provides an efficient alternative process modeling technique.

As a general rule, we take the approach that use cases begin as narratives that are later advanced to a visual presentation by using one of the UML-based visual modeling tools. In practice, we find that use cases and their corresponding UML diagrams are not as easily understood as are the SIPOC narratives and corresponding process oriented flowcharts produced by allCLEAR. To support any use case modeling work that we may be called upon to complete, we have devised a series of templates that help us to prepare narrative use cases that can then be used as input into any UML modeling tool. (However, as for a UML workbench we are tool neutral, since this is really the domain of software development and is therefore largely programming language driven).

System Problems and Business Issues

Often times we will hear that "the system" is to blame for a given business problem or issue; usually it's prompted by a phone call we make and we're hearing excuse "Your order was late due to a system problem". In fact, we've heard this far too often and perhaps you have as well. Most of the time this stock answer is a "mask" for other shortcomings an organization may have. Simply put, it is often not the system "at fault" at all. In our experience, process failure and human error are by and large much more significant factors when measuring an organization or any given system's shortcomings.

Systems are by and large far too anal and binary to fail; they require a lot of help – usually the human kind – in order to fail. This is not to say that systems are defect free. But by and large, most system failures are the result of weak designs (a human error no less). They are so poorly designed from a technical or infrastructure perspective that the system isn't functioning or workable in its' delivered state – usually though these are performance or volume issues (again human errors estimating the bandwidth or capacity needed) more so than anything else. With that said the process auditor must be certain "the system" is really the root cause of an ineffective process and of course must understand why it is so.

Reconstructing Processes

Once the "AS IS" models of the business processes are completed, the process of analyzing these processes and making specific change or simplifications recommendations can be completed. This analysis process will result in a series of "TO-BE" or "future state" business process models. In the case of a "clean sheet" change approach, one would recommend ways to streamline processes and where applicable, suggest best practices, benchmarks and standards to be adopted and ultimately would produce a business system requirements document (BSRD) to automate the redesigned process.

Recall that the clean sheet approach considers the future state modeling without any consideration or regard for the practical implications of technological support of the new process. The presumption is that the organization will "do what it takes" to make the process changes happen. We submit this is often a naive, if not dangerous position. It is also why, in our estimation, that some attempts at business process reengineering fail outright or produce little if any of the desired benefits. One can't simply assume that technology enablement is practical or even feasible.

Developing a Process Vision

In the case of technology enabled business process improvement, instead of a set way of doing the process, a "process vision" should emerge. This process vision is then used as a way of benchmarking "off the shelf technology" that will be used to implement business process improvements or changes. This requires understanding the technologies that are available and how they specifically facilitate automation of business processes. Strive for objectivity in the process design, not bias toward any one system. Don't let current or past systems experience limit the vision. Identify and specifically note any industry specific processes or requirements where possible.

Of course one the problems an organization will have is envisioning or imagining the future state when they may have little or no exposure to the software technologies they wish to exploit. In such instances,

most reputable software package consultants can help an organization develop this process vision. Understand that most consultants will have some biases toward a particular software package or software packages. As a general rule, consultants can't possibly know all the intricate details, designs and inner workings of the literally hundreds and hundreds of packages that are available. Thus, consultants tend to specialize, usually based upon their experiences with specific software packages. Consultants who have experience with multiple packages are indeed best. Their multiple package experiences should help them produce a more generalized process vision.

The process vision should be constructed in terms of a "cradle to grave" vision or business transaction scenario for the major or business critical transactions or "processing threads" of the organization. These process visions form the nucleus of what we refer to as "business process scenarios", or "demonstration scripts" that become the centerpiece in software package evaluation and selection or "test cases" when developing a custom systemic solution to automate a business process.

We find that use cases are often an ideal format for presenting such process visions because of their user/system interaction perspective. In reality, a product demonstration scenario or test script is nothing more than this process vision supplemented by sample business data.

Documentation Leverage: Cheat to Win Through Reuse and Adaptation

It is our conviction that any deliverable produced during business process reengineering should be useful and relevant throughout aspects of the software package life cycle or the software development life cycle. For instance if use cases are used to document a process vision, by merely adding sample business data they are "transformed" into product demonstration scenarios or test cases.

As another example, AS IS or current state SIPOC Process Models represents the "starting point" in creating a SIPOC model for the TO BE or future state. This is done by "reworking" or "transitioning" the model, and hence the documents, from one state to another during the brainstorming and innovation that occurs when revamping processes. Obviously the Business Process Maps are then reworked in this same manner to reflect how processing threads will change on an overall basis as well.

Yet another example of document leverage occurs when Business Process Maps and SIPOC Process Models are used to transition into Business Process Proposal (BPP) documents during system implementation. Business Process Proposal documents translate the organization's business processes into the specific software package features and functions used to automate the business process. Although a Business Process Proposal will rely heavily on the SIPOC process model, an intimate understanding of the software package is also a prerequisite.

The Business Process Proposal documents are then further leveraged in terms of constructing training materials. For instance, training exercise guides that "formalize" transaction scenarios into step-by-step exercises conducted on the system may be desired. These exercise guides are usually necessary for larger implementations, but are necessarily needed for smaller implementations where training might be conducted on a one-on-one basis.

The Business Process Proposal documents are also instrumental when building Quick Reference Cards, which are summaries, usually a one or two page laminated card that highlight the system menu paths corresponding to job related or departmental processing (i.e. Accounts Payable, Receiving, Storeroom) functions. These are something that we would put together for any size implementation. They usually serve as good memory joggers especially for infrequently or little used system functions. There is still another important document that will ultimately emerge out of the process auditing initiative

At TradewindsGroup, one thing we try to do is to avoid rewriting the software package documentation that is provided by the software vendor. One of the reasons for buying a software package is to outsource software development, including the development of any detailed documentation. We generally propose holding customized documentation to a minimum. The focus of any customized documentation should be

limited to what your organization does with the software (i.e. exercises, examples) and when and what features apply to your business process.

A second reason why an organization should not spend enormous sums on documentation is that in our experience after training, most of the documentation becomes "shelf-ware" – that is it becomes reference material that will be used only infrequently, at best.

Generally speaking, the training exercises and quick references represent about all that is needed to supplement the vendor's materials. Sometimes vendors will provide exercise guides that are usable as is, so all that all that may be needed in those cases are data sheets that are germane to your business organization.

There are times when more elaborate documentation is needed, particularly for complex functionality or when the vendor's documentation is of poor quality (usually it's simply unclear or is not logically presented) for any given process.

To ensure that your organizations is not buying a into significant writing task, it's a good idea to review documentation for each of the major business processing threads as a part of the overall software evaluation process. Be particularly careful with software that is developed in the non-English speaking world, (which is happening more and more) and is then adapted to English as the documentation is often of poor quality and may in fact require lots of rewriting to be particularly useful in the future.

Eliminate Redundancies in Process Documentation

Sometimes there may some redundancies between processes. For instance any number of departments might identify a processing step of "check inventory balances". At some point, preferably before preparing business process proposals during the software implementation process, it will be desirable to "inventory" all business process steps to identify common steps so they're only documented once in terms of the how they should be completed within the software package.

Pilot the Process Changes

Once agreement or consensus is reached on what the business process will "look like", as portrayed through the TO BE or future state model, its time to "step through" the process change in terms of a "conference room pilot". The purpose of this step is to validate the appropriateness and feasibility, from a practical standpoint, of the proposed process change. This is done by "process simulation".

Some of the business process modeling tools can "simulate" processes, but only in a very narrow sense. Their simulation is usually cost or throughput (time) based and as such this action alone does not "visually" prototype or simulate the "complete" business transaction or process as this step implies. However, in no way do we want to imply that cost and throughput simulation is not a valuable exercise; it is simply not sufficient. What we propose must be done is a far more extensive simulation.

The prototype simulation should "step through" the business transaction from "beginning to end". "Recreating" a prior, verifiable business transaction, using the proposed processing steps completes the simulation. This recreation of a prior transaction uses the business data from that prior business transaction, in the proposed process or system, as if it were now real – hence allowing for the comparison and verification of results between the real and simulated transaction. Each different transaction that is to be demonstrated or prototyped requires a scripted, business transaction scenario.

In the case of a technology enabled change approach, this would involve actually setting up the process in the software and completing the relevant software based transaction(s). Usually this is done by a vendor or reseller in terms of a software package product demonstration and is limited to the more critical business processes as represented by one or more business transaction scenarios contained in the request for proposal (RFP) document sent to vendors.

Although it would be desirable to see every aspect of a software package before it is acquired, that is usually not practical. So, although process auditing may yield dozens of business process models, only the most significant transactions, usually a half dozen or fewer – the core business transactions – would actually be reviewed in terms of the product demonstration. The remaining business process models would be held for prototyping during the actual software package implementation. Many times, these additional transaction scenarios are variants of the core transaction scenarios. Usually more complex transaction scenarios are chosen for demonstration purposes.

Prototyping of simplified or revamped business processes sans the availability of a software package that actually implements the desired functionality is a different and is albeit, a somewhat more abstract process. One technique calls for construction of prototype software models, perhaps partially or even fully functioning, that would represent what would be built. Prototypes offer a vision of what is to come, but often end users expect that if it's on the computer now, it's a functional, deliverable software solution. But, prototypes are often significantly less capable than a user wants. Thus, we believe system base prototypes for most business applications are counterproductive and wasteful. Thus, the approach we like and prefer to use is "storyboarding", which is discussed in the next section of this white paper.

User Interface (UI) Modeling

Sometimes visual representations, in the form of "screen mock-ups" for a complex business process to be undertaken by a software system is essential to reaching consensus and a complete understanding on the overall processing requirements. We call this User Interface or UI Modeling.

This is usually a mandatory step when designing and developing a custom business application software system or a web site, perhaps when engaging in a clean slate business-reengineering project, however, it is usually an unnecessary (or at least uncommon) step when dealing with software package requirements.

Where the technique does have relevance to software package selection and implementation is in two areas. First, we find this a particularly useful technique for illustrating how a complicated business process, such as a product configuration feature, might be expected to work in terms of a software package. Second, this is a helpful technique for visually prototyping any software customizing requirements that will be needed to "fill in" any gaps between your requirements and the software package.

Our user interface modeling technique relies on "sketching out" or "white boarding" the process vision is in terms of the "user interface view" of the software enabled business process or transaction flow in a facilitated workshop setting. Once we have a User Interface design, we use a "storyboard" approach to "confirm" the design with the user community.

We have typically used Microsoft PowerPoint to present user interface models. Each PowerPoint slide animates a different step or feature in the software enabled transaction or process flow. This technique is also an excellent way to "visually prototype" a reengineered or reconstructed business process when taking the clean sheet business process reengineering approach and serves as a springboard to defining the underlying process automation requirements.

Process Benchmarking: Calculating the "Worth" of Business Performance Improvement

Business performance is closely associated with and is in fact is measured by the success of process outputs in meeting or exceeding the specific performance requirements or product characteristics of customers. Assuming the SIPOC models of processes are complete, the next step is linking processes or drivers to metrics.

The first step is to identify specific "output requirements or characteristics" – criteria that each process or system output must achieve. Do this for each output listed in the SIPOC model. The second step is to determine how often these requirements are met on the first "attempt" or "try" – without any further work

or rework required. Such output requirements or characteristics are examples of the kinds of business metrics affected by the drivers or underlying processes and processing actions and of course by any process improvements.

The third and final step is to calculate the cost of failure – the cost to rework the output when the first try or attempt fails. Each successive percentage point improvement should demonstrate savings equal to the marginal cost of the rework that has been avoided and thus represents a cost savings to the organization that can be monetarized for cost and benefit analysis purposes.

Process Simulation

The best technique for benchmarking of new processes is through simulation and sensitivity analysis. For instance, an advanced version of allCLEAR, the process modeling tool we like, has the ability to define process-oriented metrics, such as costs or yield per step in a spreadsheet-like format for each processing step.

The simulation of process performance occurs by the "execution" of the process (at least on paper, or in a simulated manner) with "standard values" for any "user-defined" metrics that in effect, simulate the process performance "under load". By engaging in sensitivity analysis – that is by changing standards, one value at a time, or perhaps simply by changing the number of times a process is executed – greater understanding of the proposed processes' performance "under load" can be demonstrated.

Simulation is most often associated with large scale reengineering efforts under the "clean sheet" scenario. It is a method of reducing or at least of understanding the risk and underlying costs and benefits associated with process improvement.

Automating Processes

Only after the business process has been rationalized, or as Kapp suggests, "simplified", does the process become a candidate for automation. The logic here is simple enough; if you shouldn't be doing something in the first place, why bother automating it? To do so only perpetuates process waste. Not only does the automation of business processes provide labor savings potential; it also provides a "scalable" solution. Thus, as the business grows, the scalable process is one that won't quickly become a major bottleneck in the overall business operation. For smaller and middle market business organizations, scalability to accommodate anticipated, future growth is often a major reason to adopt automated processes.

In a technology enabled business process reengineering effort, software packages are usually involved as the "engine of process automation". However, most software packages have some very real limitations. As in the case of an ERP system, many organizations will find that it is unlikely a single ERP system from among the hundreds that are currently in existence will automate all of their business processes in the exact manner desired. Therefore all organizations are further advised to establish a "process automation agenda" that represents a set of priorities for their immediate task automation initiatives.

If a package can't meet the most critical or most essential of your process automation requirements – usually deemed to be 80% or better – the software package is likely not a "good fit" for your organization and will not allow it to accomplish its' business reengineering goals. To knowingly accept an inferior software package solution that requires extensive and costly customization or modification is a risky business proposition. If your organization does not have a "good track record" on meeting its' business system requirements today through its' existing software development infrastructure, the software package as an additional variable won't make the situation better and perhaps will worsen it.

Buying Features for the Future

While automation is clearly the *best long-term alternative* for most business processes, also realize that *not all processes are immediate candidates for automation*. Therefore it is important to "buy for the future" in terms of any software package. The typical software package has, on average, a life span of between 5 and 15 years, so they are truly long-term decisions. Make sure any package can support automation of as many processes as possible, even if they won't be features of the software package that are deployed immediately. Good "future feature need" examples related to ERP might include multi-plant processing and foreign currency processing.

Business Process Reengineering: When Should It Be Done?

TradewindsGroup believes an organization is best served by completing the business process understanding and reengineering process *prior to software package selection*. The reasoning here revolves around Kapp's USA principle – that business processes should not be automated until processes are understood and simplified and particularly our contention that selecting the wrong software package in terms of an organization's business model will prove to be a bad if not a terrible business decision.

How then does process understanding and modeling help with software selection? The reason is simply that from the modeling process will emerge the business processing scenarios that can be used as a software package demonstration and evaluation script. In addition, the business process models can be used to build a list of required software features or functions necessary to fulfill the organization's specific processing requirements. Business process understanding and modeling is critical to successful business process reengineering and to software package implementation.

The Case for Just in Time Business Process Reengineering

Many practitioners contend that reengineering – in advance of software implementation – is an unnecessary or perhaps even a wasteful exercise. This view presumes that software packages are considered as a "special case" of reengineering. Simply put, the "new" business process model is, practically speaking, based squarely on what the selected software package will allow.

Another argument against any pre-implementation reengineering is that software packages are largely commodity buys and as a general rule, are already full of best business practices so why bother with a reengineering of the business if the software will "natively" do so using these best practices. Therefore, it is simply a matter of transitioning the organization to "the new ways" of the software package.

In practice, we have also found resistance, if not reluctance to engage in any formal business process reengineering. So often we find that reengineering is a bad word, perhaps due to past reengineering failures or blunders, but this should not cause a wholesale mistrust of reengineering. Though, we don't necessarily recommend or condone either the "no" or the "deferred reengineering" view, from a practical standpoint, we understand the cost and time constraints that often dictate otherwise and that "a consolidation of effort" is often the only practical solution we can offer a client.

In such situations we improvise an "alternative" workflow strategy, referred to as "Just in Time Business Process Reengineering". We define Just in Time Business Process Reengineering is the "implicit reengineering" that must occur with the adoption of a software package. Other terms we considered using to describe this implicit reengineering process was "Organic" or "Native" Business Process Reengineering. Although both seemed a better a fit, neither took into account *exactly when* we usually needed to perform the reengineering effort, which was *during* software implementation and hence where the idea of using "just in time" sprang forth.

The way this alternative workflow works is as follows: The AS IS modeling is completed *during implementation,* using the standard techniques proposed in this chapter. A future state or TO BE modeling process (the reengineering step) is not done. Instead, using the current state or AS IS model, future state Business Process Maps and Business Process Proposals (by definition these are forward looking) are then

prepared by "retrofitting" the organization's business processes and practices to what the software package has to offer.

Thus, the organization's business processes are "natively" or "naturally reengineered" to the software package's "organic" business process model.

As a final note on this matter, this type of alternate reengineering arrangement works best for a relatively small, unsophisticated single plant or single facility based organization desiring a rapid or so called "fast track" implementation – where few bells and whistles are wanted or needed and customization is not on the table as an option.

Business Process Improvement: An Epilogue

This white paper presented a perspective on how to conduct business process reengineering or business process improvement. The paper placed particular emphasis on process reengineering when software packages are involved, or on what is often called technology enabled business process change, in which case the software package is considered the centerpiece of an organization's business process improvement objectives.

In the case of software packages, TradewindsGroup contends that the first two steps of business process reengineering – understanding and simplification – should precede software selection and that improvements to processes, as measured through business process related metrics or benchmarks, can be monetarized for the purpose of providing the requisite financial justification needed for reengineering business processes through automated techniques, particularly using software packages and complementary technologies.

As for clean sheet business process reengineering – it is self evident that understanding and simplification must precede any process automation. We feel that clean sheet business process reengineering is particularly useful when radical change is needed – when old business models are failing and threaten the continuing operations or competitiveness of an organization.

If you're undertaking process change to meet the objectives of implementing a software package you are selecting or that has already been purchased you'll find that taking a clean sheet business process reengineering approach will likely prove to be a costly distraction. Why? Changing software packages to conform to business objectives is not only counterproductive, but often defeats the purpose in buying a packaged solution. Rather, it is best to find candidate software packages that conform to your business o model and satisfy large numbers of business objectives (not less than 80% as a general rule) before attempting to implement such.

About TradewindsGroup

TradewindsGroup provides technology-focused business process consulting and educational services for manufacturers, distributors, contractors and builders, financial service and other professional service firms in the Chicagoland area. We focus our efforts toward meaningful and tangible change – the making of business process and business system improvements in customer relationship management, order fulfillment, project management, supply chain execution, manufacturing and field service operations and financial management.