

Getting the Process of BPMS Right: The Need for an Implementation Methodology

Salman Akhtar and Haleem Vaince

Introduction

So yes, BPMS is important. Maybe even revolutionary. But delivering BPMS solutions remains a challenge as practitioners in the field know only too well. In many ways, BPMS platforms have outpaced BPMS practice. What that means is that while we do have the cutting edge technology platforms to build BPMS solutions, the process of doing so still uses the old ways of IT.

So how do we do BPMS right? In this paper we discuss:

- Why is a BPMS specific implementation methodology necessary?
- What are the elements of such a methodology?
- What kinds of benefits can be driven by the use of an implementation methodology?
- How would this fit with other methodologies?

NOTE: We use BPMS (Business Process Management Systems or Software) to refer to the use of software products that can capture models, facilitate changes, manage the runtime execution of business processes, and provide business managers with information about the execution of their processes. We use BPM (Business Process Management) to refer to all aspects of process within the enterprise, including process architecture, business process redesign, management and measurement, and continuous process improvement efforts. There are several BPM methodologies available, but there is a great need for a BPMS implementation methodology designed to help companies use BPMS products to identify and solve appropriate BPM problems.

The Promise of BPMS

As a technology, BPMS makes three fundamental promises:

- **Agility:** the ability to deliver business outcomes driven by processes faster than previously possible
- **Flexibility:** the reduction in cost and time of making changes in business processes
- **Visibility:** control over and validation of how processes are being executed

So what's really revolutionary about BPMS is its "close-to-the-business" nature coupled with the ability of leading BPMS products to enable solution delivery 2-3 times faster than other software technology options like J2EE, .NET etc.

The Challenge of Delivering BPMS Solutions

Delivering BPMS solutions is not simple. The difficulty arises from several factors.

The Complexity of BPMS

First, BPMS solutions tend to be complex. Many processes where BPMS solutions are implemented are also cross-functional. This leads to a situation in which multiple stakeholders associated with the process may have conflicting ownership claims, turf to protect, and goals and priorities for improvement. Even in organizations where the enterprise has designated horizontal "process owners", this can still be a problem. And since the impacts generated by changing cross-functional processes typically ripple out over a larger number of people than single-function processes, the process (re)design has to be undertaken with great care.

The second reason for BPMS complexity is the large number of applications typically involved in supporting cross-functional processes. Departments across an organization have legacy systems and applications using many different technology platforms and integrating all of these, and in the right order, can be a challenge. There is then, a platform complexity.

Finally, the BPMS solution delivery team must have the ability to master multiple skills and vocabularies, including Business Analysis, Process Design, technical mastery over BPMS platforms, integration skills, IT infrastructure optimization, etc., and reconcile them into a project design that maximizes business value creation yet can be executed within the cultural framework of an IT department. IT organizations frequently overlook the interdisciplinary nature of BPMS and think of the problem of executing BPMS as being a simple matter of acquiring coding skills on a new platform. In this all too common scenario, the probability of problems and the risks of outright project failure rise significantly, as data from the BPMS Institute shows. So a third dimension of BPMS complexity has to do with the requirement for a new mix of skills and some cultural change.

Thus, the first challenge in delivering BPMS solutions comes from facing complexity arising from all three major sources: Process, Platforms and People.

The difficulty of managing BPMS projects

Managing the delivery of BPMS projects is very difficult. This arises most directly from the very short 45-90 day development lifecycles of BPMS solutions. The tools that have traditionally been used to manage IT projects have evolved over time with the goal of managing risk over large-scale long-duration efforts. Most standard project management methodologies such as PMI or Rational Unified Process (RUP) cannot cope with the iterative, bursty pace of BPMS. This situation is further exacerbated by the widespread use of distributed and offshore development teams by many large enterprises. Most offshore delivery processes, based on traditional waterfall techniques, are simply not designed to react and respond in these timelines. Finally, measuring progress is a challenge. The typical weekly status report will not give sufficient sense-and-respond time when the entire delivery lifecycle might only last seven or eight weeks.

A second cause of the difficulty lies in the fact that there are no clear roadmaps on how to capture requirements in a BPMS project. The traditional methods of Use Case Analysis or Functional Analysis don't directly apply to BPMS solutions. In our experience, most teams default to simply using the Process Model (captured using a modeling tool specific to the BPMS platform). The process model is certainly the first step in capturing the project requirements. However, from a development standpoint, it is not complete. Some of the additional elements required to deliver a functional BPMS solution include process variables, the user interface for various forms, integration points, business rules, validation rules etc.

Lacking a single document where these elements can be captured, most BPMS development teams end up with the process requirements split out over a number of tools and spreadsheets etc. This lack of a single canonical set of requirements makes solution delivery a real challenge for the development team, especially if it's geographically distributed.

This lack of a complete set of requirements poses an even greater challenge to the QA team, since they cannot easily develop a comprehensive set of test cases to ensure solution quality. The QA team is also faced with the short timelines which make traditional QA processes and templates almost unusable.

The perils of applying a standard project methodology

Here are some other key issues faced by development teams that try to use traditional IT project delivery methodologies:

- Estimation Models: How do we size BPMS projects to come up with cost and time commitments?

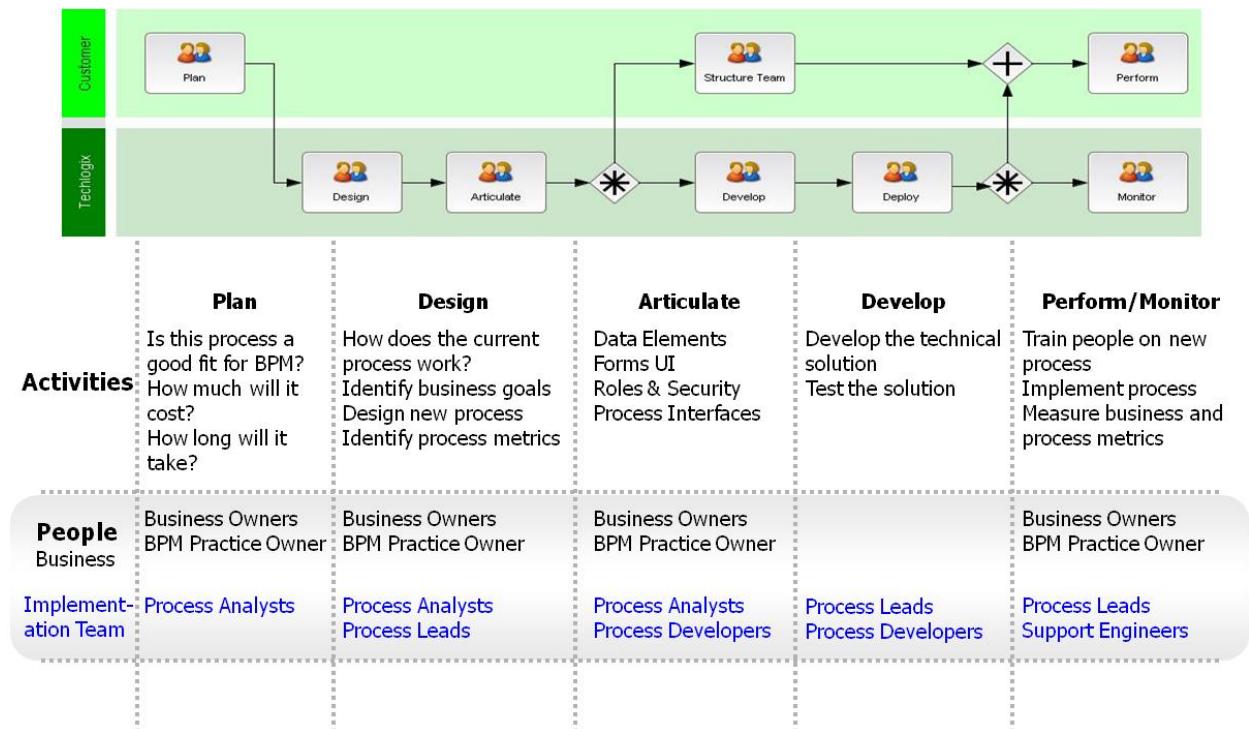
- Missing Templates: What do we use in place of standard IT project templates for Functional Requirements, Architecture, and Technical Design etc?
- Change Management: How do we control change in projects that are both cross-functional (leading to a high potential for conflict and delay in decision-making) and must be executed across very short timelines?
- Project Metrics: How do we know if our project is on track? What metrics allow us to benchmark our performance both internally and externally?

The elements of a BPMS Implementation Methodology

A successful implementation methodology should cover the full lifecycle of a BPMS solution. This would start from Fitness Analysis (should we even use BPMS?) through integration within the IT landscape, estimation, resourcing, the Analysis and Design phases, progress tracking through development, testing and finally integrating analytics and monitoring with process. And all of this should be done in a way that lays the foundation for continuous improvement by the Business Ownership after deployment.

Sample Methodology Overview

The figure below shows an overview of a sample Methodology which is described in some detail below.



This is the first phase of the project and it precedes the actual project kickoff. This phase focuses on assessing the suitability of the process for BPMS. This is followed by a cost and time analysis for budgetary numbers.

Phase 2: Process Discovery and Design

Description

This is the first phase of the launched project and it commences as soon as the project is kicked off. This phase is completed with a finalized, "To Be" Process Model and Project Plan documentation. In this phase, the Process Analyst along with a set of Process Leads interact with

the customer's business and technical users to clearly identify and document the new process design.

Process Definition: Primary Tasks

Typical tasks carried out in this period include:

Functional and Technical Requirements

- Identify specific relationships between proposed or existing process
- Design the proposed process flow
- Identify important process events (activities performed, trigger events, start and end conditions)
- Identify all integration points within the process
- Discuss the requirements document with stakeholders and identify any changes or additional requirements

Project Planning

- Preparation of a detailed Project Plan
- Establishment of project scope and Critical to Quality (CTQ) deliverables

Conceptual Architecture (if green field implementation)

- Study and Review existing systems and architecture
- Determine the requirements for Security, Scalability and Availability

Process Design: Deliverables

The deliverables of the Gather Requirements phase are:

- Process Map Diagrams
- As-Is Process Benchmarks
- Project Plan

Phase 3: Process Articulation

Description

In the second phase the process design is taken as a starting point and the solution design is completed. The activities include identification of various factors such as data fields, validation types, detailing of integration points, rules and commands, alerts and notifications, security and any interactions amongst other processes.

Process Articulation: Tasks

Typical tasks carried out in this period include:

- Prepare the data dictionary and rules governing data (validation, visibility etc)
- Design the integration process events — email notifications, automated data population, intelligent workflow routing, etc.
- Design the specific integration points required for the process
- Document process related policies if any

Process Articulation: Deliverables

The key deliverables in the Design phase is the Process Specification Sheet which details:

- Workflow Steps: Details all Workflow steps including step type, name, process status, results and timeouts
- Workflow Step-Field Map: Maps all data fields to each workflow step and documents field behavior for each step; also details roles associated with that workflow step
- Field Validation: Lookup data; specification of dynamic lookups if any; custom field validations if any
- Transition Map: Details all possible transitions from each workflow step
- Notifications: Documents format and details of all notifications

Phase 4: Process Development

Overview

The Process development phase begins with the completion of the technical design and finishes with the solution being completed and run through system testing. The development phase involves the development of process components as specified in the Process Model and the Process Specification Sheet. It also involves the building of common frameworks, building of integration logic to handle data transformations, validations and connectivity to external systems as required.

Process Development: Tasks

Typical tasks carried out in this period include:

- Develop the solution components according to design including Forms and UI
- Code Review
- Preparation of Test Plans
- Unit Testing
- Perform integration testing using the available QA environment
- System testing: Perform end to end tests of the solutions with verification of all applicable user access test roles

Process Development: Deliverables

Deliverables in the Integration Development phase include:

- Tested application build
- Test Plans
- Test Data Strategy

Phase 5: Perform/Monitor

Overview

This phase begins with user acceptance testing and rollout of the process in a production system and ends with the deployment of standardized Process metrics to monitor the new process.

Integration and System Test: Tasks

Typical tasks carried out in this period include:

- Carry out User Acceptance Testing
- Train users on new process
- Monitor implemented process performance using standardized Process metrics

Integration and System Test: Deliverables

Deliverables in the Perform/Monitor phase include:

- In production process

Sample Roles and Responsibilities

The table below describes one possible team structure that can be used for BPMS implementation. The Process Definition and Requirements phase is mainly performed by the first two roles while the Implementation is performed by the roles that follow.

Role	Responsibility
Process Analysts (Typically Onsite) (Process Design Lead)	Typically person with 7-10 years experience. Tasked with understanding the current process and conceptualizing and designing a new process. Interfaces with key business process owners as well as internal IT staff. Leads process design/redesign effort producing the process model and as appropriate uses process simulation to optimize the design.
Per-Process Leads (Typically Onsite) (Process Articulation Lead)	Process Articulation, i.e. Process model developed by the Process Analyst is further detailed by the Per-Process Leads. Tasks include definition of data elements (data slots), defining process interfaces (connecting points with all external systems in question), defining the performers (roles and responsibilities of the actors in the system), defining the forms/UI. The role of a Per-Process Lead is a challenging task based on fact that BPMS type solutions are susceptible to extremely rapid changes, and therefore need to be managed very carefully with the business owners (to get business sign-off in short time frame). Capturing detail pertaining to above and handing over to offsite implies that things are documented and transmitted in a manner that is understood by the developers who would be implementing the solution.
Process Developers (Can be Offsite)	Once the work of the Per-Process Leads is completed, the Process Developers use the Process Flow and the Design Sheet to implement the specific workflow.
Integration Architect (Typically Onsite)	If significant integration work is required an Integration Architect can provide direction for this effort.
BPMS Platform Architect (Typically Onsite)	To ensure that the platform is being utilized in an optimal manner.
QA Lead (Can be Offsite)	The quality engineer studies, understands, and helps in the improvement of the requirements of the project. The quality engineer develops the test plan. The quality engineer develops and executes test scripts. This person records test results during testing activities and documents test faults in the defect log.

Methodology Templates and Artifacts

A key part of a BPMS Implementation methodology should be a set of templates that are used across all aspects of the solution delivery lifecycle. Some of the specific areas these templates should cover include:

- Fitness Analysis
- Estimation Model
- Resourcing Model
- Process Specification Sheet
- Progress & Risk Tracking
- Process QA
- Process Metrics

When is a BPMS Implementation methodology useful?

At what point should you start using a BPMS Implementation methodology within your enterprise? Many enterprises go through a cycle in which they select a BPMS platform, get staff trained on process modeling (and perhaps some of the Business Process Level methodologies) and then launch pilot projects. After the initial pilots are successful (often with the help of Professional Services from the BPMS platform vendor), future BPMS projects get handed over to the existing set of IT resources both internal and external.

This is the point where implementation failure happens most frequently.

It is exactly these resources who need to be trained in a BPMS Implementation Methodology so they can approach projects using a defined process and leverage best practices on both delivering and managing the implementation of BPMS projects.

Value for Stakeholders

What does a BPMS implementation methodology deliver for key stakeholders?

The Business Owner

For the business owner, a BPMS Implementation Methodology should:

- Improve speed of delivery of BPMS solutions
- Ensure benefits are realized through controls and metrics
- Ensure business process and IT solution are aligned
- Achieve strategic goals through continuous process improvement

The BPMS Center of Excellence

For the BPMS Group or BPMS Center of Excellence, the BPMS Implementation Methodology should:

- Standardize project execution for repeatable outcomes
- Codify best practices in BPMS across the enterprise

The IT Team

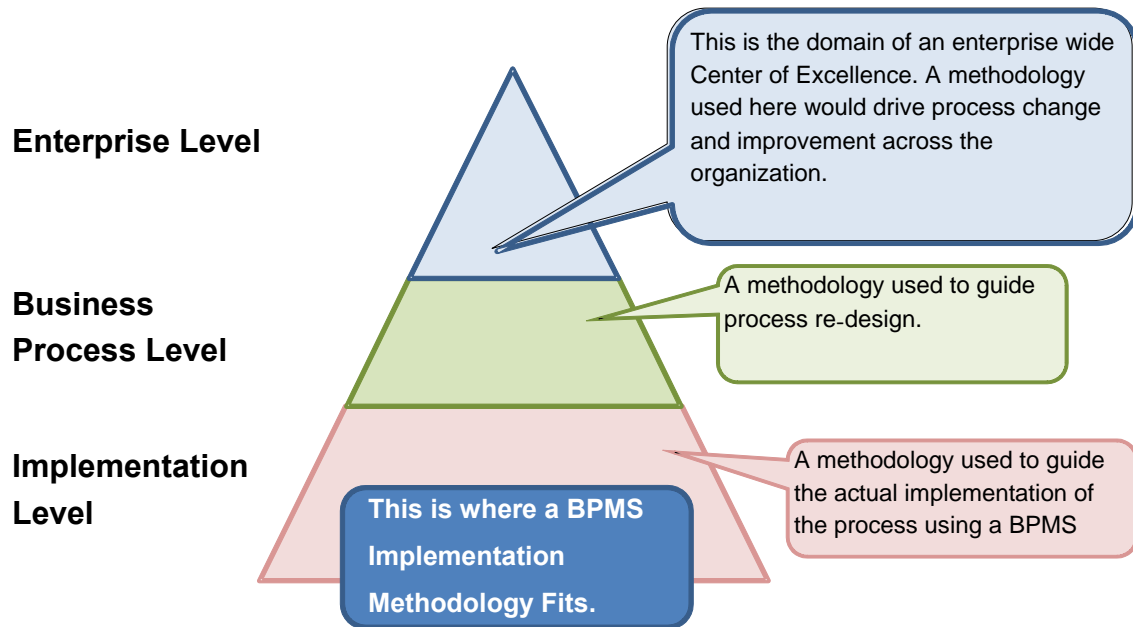
For the IT Team, the BPMS Implementation Methodology should:

- Facilitate project delivery with defined interaction points and roles with IT
- Work with IT landscape by specifically designing for integration points

How does a BPMS Implementation Methodology fit?

There are a number of BPMS methodologies available. Where does a BPMS Implementation Methodology fit within these and how best can you use it?

One way of answering this question is to step back and take a broader view of what goals various methodologies are trying to achieve. Paul Harmon, Executive Editor of BPTrends defines three separate levels where methodologies operate which are shown below:



Conclusion

Implementing BPMS is a process. The likelihood of success is a function of the fitness of the process and how well it is executed. As we have seen, standard IT project methodologies are not a good fit for BPMS projects. A BPMS Implementation Methodology specifically designed to address these gaps is essential for repeated success.