



BPM: A Global View

Rashid N. Khan

Founder and CEO
Ultimus

Rkhan@ultimus.com

Latest book:
*Business Process Management:
A Practical Guide*
Meghan-Kiffer Press, 2004

Checkbox BPM

Many years ago I got a really good offer in the mail to buy a sleek, powered speedboat. The company sounded very legitimate. It had a Florida address and a picture of a speedboat manufacturing facility lined with shiny, sleek speedboats. The offer was for a two-seater, portable, speedboat, including the power engine. They were only asking for a \$120 non-refundable shipping fee which sounded very reasonable for shipping a speedboat from Florida to Massachusetts. I had never bought a boat of any type before in my life. But this was a no-brainer, especially since there was a nice lake close to my new home. So I convinced my skeptical wife ("Honey, it is portable. Even you will be able to use it on the lake!"), signed the paperwork and sent in my check. I was expecting a big delivery truck to show up on my driveway in a few weeks with my speedboat that would impress my wife and friends. A couple of weeks later I received a small box via US Mail with a \$2 postage stamp. I opened it up with trepidation with my wife standing next to me, unable to control her laughter. I was hoping against hope that the box was only the advance shipment of some delicate component of my powerboat that needed special delivery. Alas, the box opened up to reveal a cheap plastic inflatable dinghy about 8 feet long. There was a teeny-weeny 2-inch plastic propeller with an electrical motor powered by two AA Batteries! Of course no batteries or pump were included. The package did include two plastic, inflatable life jackets for our safety, thank you. Red-faced and embarrassed, with my wife insisting that I recant all the stories and fun I made of her shopping sprees, I trashed the "speedboat" in the garbage can and did not want to see or discuss it ever again.

Even though this was the work of slick con artists who probably disappeared as soon as they received enough checks from naïve, gullible, first-time buyers like me, I must say that they did an excellent job of matching the "product" with their advertised claims:

- It was a boat. Nowhere is it written that a speedboat cannot be a small inflatable plastic dinghy.
- It was unquestionably portable!
- It was powered. There is no standard that specifies how much power it takes to make a speedboat.
- It was a speedboat, because there is no definition of "speed" associated with speedboats. A speed approximating zero miles per century is still speedy, relative to something!
- They did not promise any accessories. So no batteries and no pump. After all, what do you expect from a free speedboat? Larry Ellison's yacht?
- The promotional flyer never claimed that the speedboat will even remotely similar to the ones in the accompanying photos.

I was the victim of my ignorance, my failure to ask the right questions ("Does it make sense?"), and my susceptibility to hype because I had already bought the vision of speeding around on the lake and impressing my wife and friends. I dared not mention my investment to anyone, because I would have egg on my face. I just took the loss, kept it quiet and did my best to forget about it. Obviously, twenty-plus years later I still have not forgotten it!

It is this buying experience that prompts me to write this column. Every time a customer shops for a product, they either need to do some research to make sure they know what they are getting, or risk disappointment, or worse. In every industry, vendors face the pressure to make broad, powerful claims about their products as they face growing demands from competition, analysts, media and customers with high and sometimes unrealistic expectations.

A couple of years ago I wrote a column titled "The Hying of BPM" that discussed several BPM capabilities where reality and sound reasoning were overshadowed by hype. I continue to see hype increasing in the BPM industry, perhaps more so than in other industries because BPM is relatively new and misunderstood. Many users new to BPM are mesmerized and energized by the hype and the potential they envision to reduce cost and improve throughput by automating and optimizing processes. Often this enthusiasm camouflages the reality that business processes deal with complex, people-centric issues that cannot simply be made to disappear by the magic-wand of BPM software with all its bells and whistles. In this age of visually-appealing graphic and virtual realities, it is easy to craft slick demos for webinars and sales pitches that gloss over the challenges that can only be uncovered during real-world implementations where the rubber meets the road. More and more I see the hype expressing itself in the cloak of what I call "Checkbox BPM."

Let me explain what I mean by Checkbox BPM. Companies and potential users of BPM are inundated with sound bites about the impressive capabilities of BPM from vendors, the press and the analyst community. Convincing case studies are presented that highlight certain cutting edge benefits of BPM in very specific situations, but complete details are rarely provided. Most people have the impression that with software anything is possible. Few have a deep understanding of the actual cost and complexity of developing software or deploying a comprehensive solution. When they get serious about BPM, these users develop a "wish list" of all the things they would like to have in a BPM Suite (BPMS), often in consultation with analysts or incorporating analyst/media reports they have read. These wish lists find their way into RFPs with all the fancy capabilities. Not much thought is given to whether they are truly needed or practical. In many cases, since the customer envisions the purchase of a BPMS for use by a number of different departments with different needs, these RFPs start looking like the specifications of a "do all" BPM solution. The RFPs are invariably distributed to all known vendors of BPM. These vendors look at the RFP, knowing what it might mean if they answer "no" to an item on the list. Their goal is to figure out how to check as many boxes as possible regardless of how unrealistic they may be given the current state of BPM technology. Aggressive and vocal sales people are chomping at the bit, eager to win the deal. There is enormous pressure on product management to say something positive about each and every requirement in the RFP.

Product management can play games with words for only so long without losing credibility. They also know the reality that R&D budgets are always tight, and software features are made to look simple in an RFP. The full implementation of many of these features is often extremely complex and time consuming, not only from the development perspective, but also from the perspective of documentation, support, training, localization, migration and long-term maintenance. Product managers are smart and creative. They know that RFP requirements are generally checkbox items with brief explanations. Analyst rating matrices are not dissimilar. They also know the 80/20 rule: after the sale, 80% of the customers will use 20% of the features. The 20% frequently used features are typically the most practical capabilities that provide proven benefits. Generally they are not the cutting edge and sometimes esoteric features that generate media and analyst hype. To meet as many requirements as possible, creative product managers take a minimalist approach. They invest in the minimum amount of development in a feature which, coupled with the creative use of words, is sufficient to enable them to say "Yes" to popular RFP and analyst checkboxes. Checkbox BPM is born! The more esoteric and misunderstood the requirements, the more Checkbox BPM will thrive in the confusion.

To illustrate my point, I provide two examples of Checkbox BPM in vogue these days.

The first is Round-Trip Optimization (RTO) which is one of my favorites. On paper, RTO is a great concept that is sure to win the hearts of many business users and business analysts. In simple words, RTO is the ability of a BPMS to capture performance metrics from actual process data, and use these actual metrics to optimize the process rapidly. Without RTO, the business analyst has to make guesstimates about performance metrics, such as the task time of each step, the probability of all the paths in the process, the rate at which exceptions occur, and all other variables that define a "scenario" against which the process is optimized. RTO saves time and the need for business analyst guesstimates by automatically creating the scenario from actual real-time data which can then be used to optimize the process. This is such a powerful concept that RTO has become a key capability in the evaluation matrices of many analysts and buyers.

But let's look at the reality by peeling back some of the layers of the RTO puzzle:

- i. Task Time — the actual time it takes to complete a step in the process — is an essential piece of data necessary for optimizing processes. If Task Time is unknown, but all other performance metrics are known, the process cannot be optimized. Think of a production line. If one does not know the time it takes for the worker at each station, the line cannot be optimized. That is why companies employ industrial engineers to do sophisticated time-motion studies of workers at various stations, or employ other means to capture this data. How does BPM software automatically measure the actual Task Time for human centric processes? What technology is there to do that? None! If Task Time cannot be measured from actual data, RTO goes out of the window, and we go back to good old guesstimates or time-motion studies.
- ii. To fully optimize a process, the scenario has to include a number of other performance metrics in addition to Task Time, such as:
 - a. Rule or path probability for each rule and/or path.
 - b. Number of resources used
 - c. Cost of resources used
 - d. Rate of new incoming cases
 - e. Rate of exceptions (late returns, re-submits, etc) because exceptions are one of the key causes of process inefficiency in the real world.
 - f. Relative time at which exceptions such as resubmits and returns occur.

How many BPMS that claim RTO as a feature capture all of the above from actual data? In my observation the answer is very few.

In the real world, changing processes for the purpose of optimization requires far more time and effort than it takes to create a scenario. When a process is changed, it requires retesting the process, moving people around, reassigning tasks, re-training people, changing documentation and getting approval from management and buy-in from the workers. These other things are far more time consuming than the time it takes a business analyst to manually create a scenario. As such, if you can make scenarios by using RTO in a few minutes, but it takes months to make the changes, does that provide a good cost/benefit relationship?

RTO is a good example of Checkbox BPM. Frankly, while every analyst talks about it, and I frequently read about it in the BPM press, I have never come across a customer of any BPMS vendor who is actually using it. Most BPMS vendors comply with the checkbox requirement for RTO by capturing some very basic metrics from real-time data. This RTO speedboat is often an inflatable plastic dinghy, but the buyer does not know that and has already bought the vision.

The second example of Checkbox BPM is SOA. SOA is probably one of the most commonly used, and often misunderstood, term in the software industry today. It has a close relationship

with BPM. Therefore, every BPM vendor, analyst and article includes some reference to SOA. It is becoming a frequent checkbox item in RFPs and analyst evaluation matrices. Many BPMS vendors claim SOA as one of their beneficial features. While many vendors can check the SOA box, the real question isn't "Does the vendor have SOA", but is rather "What degree of SOA". To help answer this question, consider the following:

- i. SOA Consumer - At a high-level there are two aspects of SOA. Is the application a consumer of Web Services, or is it both a consumer and provider? Almost every modern BPMS today can consume Web Services in some way, shape or form. The simple ability to consume Web Services therefore enables the vendor to say they have an SOA. However the ability to consume Web Services does not make an application an SOA application. Combined with some creative verbiage, however, it does facilitate Checkbox BPM by enabling the vendor to say yes to the vague question about SOA in many RFPs.
- ii. Beware of Flat Data Model - A large number of BPMS, even some from leading vendors, use flat database records as their data model. A database record is one dimensional, and the columns in databases are fixed and defined a priori. Web Services, which are the best practice for SOA implementation, use XML for data. XML is three dimensional and supports repeating nodes whose number does not have to be defined a priori. When a BPMS with a one dimensional data model has to consume Web Services that have three dimensional data models, there is no simple way to map the two to each other. Yes one can wave hands and resort to clever programmatic kludges to map and integrate the two. However, as soon as one resorts to programmatic kludges, the BPMS gives up loose coupling and the agility benefits of SOA. A few months ago I made the mistake of using the word "integration" when talking about SOA with the SOA guru of a leading analyst. He emphatically corrected and reminded me that SOA is about "interoperability" and not about "integration". I will never make that mistake again! Interoperability goes out of the window as soon as one has to resort to mapping one-dimensional flat database record models with 3 dimensional variable data models. So if a BPM vendor claims they have an SOA, ask them what is their internal data model. If it is not XML, that speedboat is also an inflatable plastic dinghy as far as SOA is concerned.
- iii. SOA Provider - Lastly, when a product claims that it is has an SOA, the product must be a consumer as well as a provider of SOA i.e. its own architecture must be a SOA. This means that all applications that interact with the BPM sever must as a minimum be (a) loosely coupled (b) use a language independent protocol (SOAP as a best practice) (c) expose the services (WSDL as best practice) and (d) have a dynamic data model (XML as best practice). If a BPMS does not meet these requirements, you should either conclude that it is not an SOA, or rethink your requirement for SOA.

There are numerous other examples of Checkbox BPM; features added to the software to comply with the barebones requirements to enable the vendor to say yes in RFPs without providing all the richness to make the feature truly useful. These include features such as collaboration, standards support, rules engines, and templates. Buyers of BPMS need to understand their requirements and then formulate them carefully and as unambiguously as possible. They also must have realistic expectations about what BPMS can deliver in 2007 and understand that if BPM software is as whiz-bang as it is claimed to be, the penetration of BPM in the industry would not be below 10%. BPMS do provide many benefits, but they do not have the magic that they are sometimes portrayed to possess. The real-world problems that BPMS are trying to solve are far from simple.