

Love Affair With Web Services Waning?

Howard Smith and Peter Fingar

The Component Based Development Forum, an analyst firm and think-tank covering business software creation, reuse and management, recently wrote, “SERVICE ORIENTATION - SO WHAT?” They went on to explain, “First we had Web Services. Then we realized that Web Services were simply a technology, and we needed architecture to manage the loose coupling of pretty much everything. For a time SOA was top of the toy box. Then we started to see Service Oriented Programming and Service Oriented Design. All very sensible, even if it is a bottom-up process of discovery. Of course the logical conclusion to this is that we are progressively establishing a comprehensive approach to architecture, modeling, design, programming, deployment, interoperability etc that is service oriented.”

Really?

We see something else. The industry is progressively establishing a comprehensive approach to *business process* discovery, design, deployment, execution, operations, analysis and optimization. Web Services technology-inspired terminology is clouding the fact. Take one prominent example:

There’s a wonderful new book by Dr. Ravi Kalakota and Marcia Robinson entitled “Services Blueprint: Roadmap for Execution.” Despite the title and a chapter entitled “Services is the Mega-Trend,” the majority of the book is concerned with Business Process Management (BPM). Indeed, the book is replete with the word “process.” Much of the book concerns itself with discussions of process configuration, process flexibility, process trends, process digitization, the composite process layer, the need for business process management, translating services into processes, linking processes via integration layers, from a technology focus to a process focus, better process automation, enabling composite processes, mapping processing into applications, process outsourcing, creating a process strategy with six sigma, moving from strategy to process design, picking a process improvement method and customer-centric process transformation. The list goes on and on. And all this is taken directly from the table of contents alone!

If we venture inside the book we find that it too is largely concerned with processes. Explicit statements are made as to the significance of processes to the so-called “Services Blueprint.” In Table 1.1, entitled “Historical Perspective: Changing Process Priorities,” the year 2003 is tagged as “Services-Centric,” but is then defined as “Digitization of Cross Enterprise Processes; End to End Supply Chain Enablement, Business Process Outsourcing and Business Process Management.” Numerous examples of end user companies that have executed on “process improvement” are given throughout the text and the authors correctly point to the logical next step in Six Sigma, “process digitization.”¹ Kalakota and Robinson point to the reality that “there is a clear pattern that can be gleaned from the turmoil: non-stop digitization of business

¹ See “Digital Six Sigma,” Smith and Fingar, November 2003, BPTrends.com

processes ... Digitization is the outcome of the non-stop business need to be more customer-driven and process-centric ... A successful digitization effort in one in which the company treats technology not as a sole solution, but as an enabler for innovating, improving and integrating business processes.”

The dichotomy between Kalakota and Robinson’s use of the title “Services” in a book about “Processes” is difficult to fathom when one considers other bold statements made in the text, such as, “Service platforms are the emerging foundation for integrating and digitizing end-to-end processes,” “The emerging service platform is the quintessential process environment,” “Translating customer’s need into business objectives, business objectives into processes, and processes into interactions is the role of service platforms,” “As a result, we need a much better understanding of how to create, deploy, maintain, and enhance cross-enterprise processes,” and “For most companies, the next step in the digitization journey is setting these composite process in motion.”

Kalakota and Robinson even point out the weakness of a services-based approach, stating that, “It is not enough to have multiple Web services scattered all over the place. You need a framework that can pull everything together and make everything talk to each other.” This sounds like a process to us. They go on to state that, “Services are built from composite processes,” and “In our opinion, the design, management, and integration of business processes are increasingly what separates the good companies from the rest of the pack.” They give examples of emerging services platforms, describing them as “an approach for designing and developing cross-application business processes,” and that this entails “A great deal of sub-process design,” and “The services perspective helps to eliminate functional stovepipes and replace them with processes that focus on creating value for customers.” Finally, they provide a message for business people and technologists when they correctly point out that, “People tend to trivialize the need for process management. This is where initiatives often fall into problems. Actually, process management is becoming a field unto itself.”

Should the book have been called “Processes Blueprint: Roadmap For Execution”?

Can’t we just call a business process a business process?

Where are the “Services” in Kalakota and Robinson’s book? What’s in a name anyway? As they say on page 244, “The different operational lines of business speak the process languages of Lean Enterprise or Six Sigma.” So why confuse readers by introducing terminology (Services) inspired from Web Services?

The notion of ‘service’ in Kalakota and Robinson’s book is really a synonym for three things.

First, they equate the need for services (a.k.a. processes) with the notion of a “focal point”, another new term they introduce. They define “focal points” as business drivers such as “easy to do business with”, “single voice of the customer,” “low daily prices,” “fast and responsive service” etc. But why introduce the term *focal point*. We think business people already understand the concept of a *business driver*.

Second, they equate service with process outcomes.

Third, they use the term “Service platform” to refer to a new class of mission critical enterprise software. But while some vendors who are developing infrastructure software may use the term “Services platform” to refer to their software, we don’t think what they are developing is what Kalakota and Robinson had in mind when they used the term “Services platform.” From diagrams in the book, we think Kalakota and Robinson are referring to Business Process Management Systems (BPMSs).

While vendors are free to describe their products using their own terminology, many, including those that used to call themselves EAI or workflow, are today using the term BPMS. Some have even adopted the standard symbol for a BPMS, a cube. *The BPMS is a business application that allows the management of business processes without dipping into the technical plumbing*, including Web services plumbing. A BPMS frees business people from infrastructure, integration and other complexities, just as the relational database (RDBMS) freed ERP users from equivalent file system and data plumbing. In this sense, BPMS products are the combination of a service-oriented architecture (SOA) and a process-oriented architecture (POA). The former is a technical infrastructure, the other a breakthrough business application. SOA and POA enjoy a symbiotic relationship. This pattern of symbiotic relationship between a standards-based commodity platform and a new innovation has been repeated time and again in the IT industry.

One common example of an innovation and a paradigm shift built on a standards-based commodity is the simple, yet eloquent, spreadsheet. The convenience and low cost of the breakthrough was so striking that it led to the PC revolution in business. The spreadsheet could not have been successful had it not been for the fact that personal computers—a standards-based commodity—were spreading like wildfire elsewhere in society. To the business, the PC loaded with a spreadsheet meant a radical simplification of routine calculations, transferring to the everyday businessperson a function that had once required special programming skills. Quite literally, business people could not understand the value of a PC until they saw someone working with a spreadsheet.

Unix and the RDBMS is another example. No one in business knew what to do with mid-range Unix computers until they saw the value of departmental business data management using new-fangled mid-range relational databases. Today, few in business know what to do with Web Services, until they see a team of business and technical architects working with a BPMS. While incumbent platform vendors continue to entice end users to invest in IT on a promise of Web Services interoperability and reuse, CIOs are becoming increasingly aware that the ROI in Web Services lies not in the infrastructure, but in the BPMS. The value of Web Services lies in the technical infrastructure, the standards-based environment in which the BPMS, and other business applications, can thrive.

There are striking correspondences between diagrams in Kalakota and Robinson’s book and the architecture of BPMS. But they continue to stress the word *service*. Towards the end of the book they claim that “Clearly, we are in between eras. Everywhere, strategists, senior and mid-level managers are caught between process-centric models (current state) and service-centric models (future state),” and they urge

businesses to bring “service thinkers” into process teams. But they then go on to list “Seven points to ponder.” The list includes the statements, “Organization outputs are produced through processes” and “A process improvement methodology is critical.” What all this seems to boil down to is an assertion by Kalakota and Robinson that “services” and their corresponding “focal points” are useful ways to determine requirements for new end-to-end processes. Perhaps they are. If so, let’s celebrate. But one cannot help feeling that this book, published as it was in June 2003, was actually written during the peak of the IT industry’s Web services hype curve ... which was precisely the same time that the Web services technical community was waking up to the significance of business processes, culminating with the submission to OASIS by IBM and Microsoft of BPEL4WS (Business Process Execution Language for Web Services).

Was the book caught in the services-processes time warp of the last year and, as a result, unclear as to the terminology it should use? Did the editors wonder how to reconcile the difference in terminology between the Web Services and BPM communities?

Kalakota and Robinson’s enumeration of numerous business processes in each chapter is another indicator of the book’s focus on processes, rather than services. Indeed, we hope that business people won’t be put off the book based on its title, because the process content is quite relevant to them. Kalakota and Robinson give these processes catchy titles such as Order To Cash, Engage To Close, Transact to Fulfil, Build To Order, Plan To Produce, Resume To Work, Goal To Reward and many others. But in the Figures where they place those end-to-end processes they are labelled as the “Services Layer,” even though they sound, and look uncannily like, business processes. Why not call a business process a business process? If we do that, business people will immediately understand what we are talking about, and that’s a very good thing.

But perhaps the most obvious element missing in Kalakota and Robinson’s otherwise excellent book is the process lifecycle itself. For while attributing to services (and service thinking) many wonderful attributes, the book omits to say how services, or for that matter processes, are actually improved in line with business objectives. There is an assumption at the heart of the book that the business objective, the focal point, is somehow inviolate and never changing. Kalakota and Robinson’s focal points are high-level aspirations, such as “easy to do business with.” What they do not show is how a business changes from not being easy to do business with, toward being easy to do business with, and how they maintain a constant improvement in being easier and easier to do business with. After all, a business is not going to wake up one morning, become “easy to do business with,” and then forget about that topic thereafter.

So whether one calls things “services” or “processes,” we ask how are they taken through the lifecycle of process improvement, from discovery, to design, to deployment, to execution, to operations, to analysis and optimization? In short, where is *business process improvement* in Kalakota and Robinson’s service model? For lifecycle management is itself a business process—and how does *that* process manifest itself in a technology-inspired service stack? We wonder how many business people have looked at technical architecture stack diagrams and wondered about this? It’s not enough to create an end-to-end process at a moment in time, for it will surely

change the minute it is created—that’s just the way of business. Time, change, and improvement are the watchwords of BPM, not Web services.

Where Kalakota and Robinson explicitly refer to “BPM” they mostly assign it to a mid-tier of “integration software.” They then align that to EAI technology, about which they say, “Managers must understand process integration issues before they invest millions of dollars. Many find themselves frustrated when they discover that multi-million dollar investment they made in Enterprise Application Integration (EAI) software is not delivering the ROI the salesperson promised.” They then go on to say, “To lower the cost of integration, the newer-generation XML-based EAI is being engineering under the banner of business process management (BPM) tool kits.” Such a perspective misses the essence of BPM, management of the complete lifecycle of change within the business process.

Our conclusion: Despite having the wrong title, it’s a great book. But if business people have to bow to IT industry gorillas for the terminology they use, perhaps there is a compromise to be made? Could G2000 firms persuade IBM and Microsoft to rename Web Services, Web Processes? Could BPEL4WS just be called BPEL and can it lose its 4 Web Services tail? Could vendors stop attributing wonders to Web services technology (SOAP, UDDI and WSDL) that is more rightly attributed to the ability of a BPMS to manage the lifecycle of business process improvement?

To illustrate the difference in perspective between *services* and *business processes*, we have included here Figure 6.2 “Supply Chain Blueprint” from Kalakota and Robinson’s book, and re-rendered it as conceived from a BPM viewpoint.

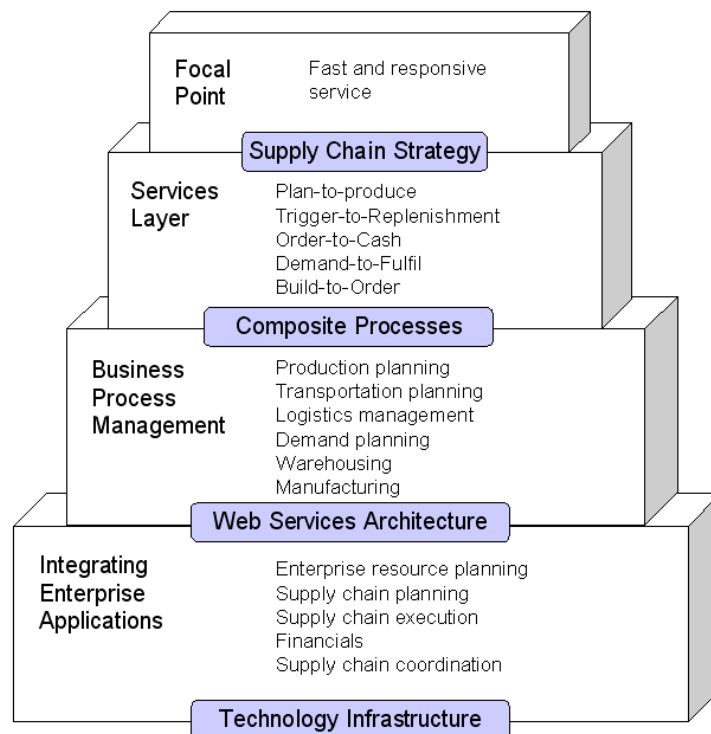


Figure 1 - Kalakota and Robinson’s Supply Chain Blueprint

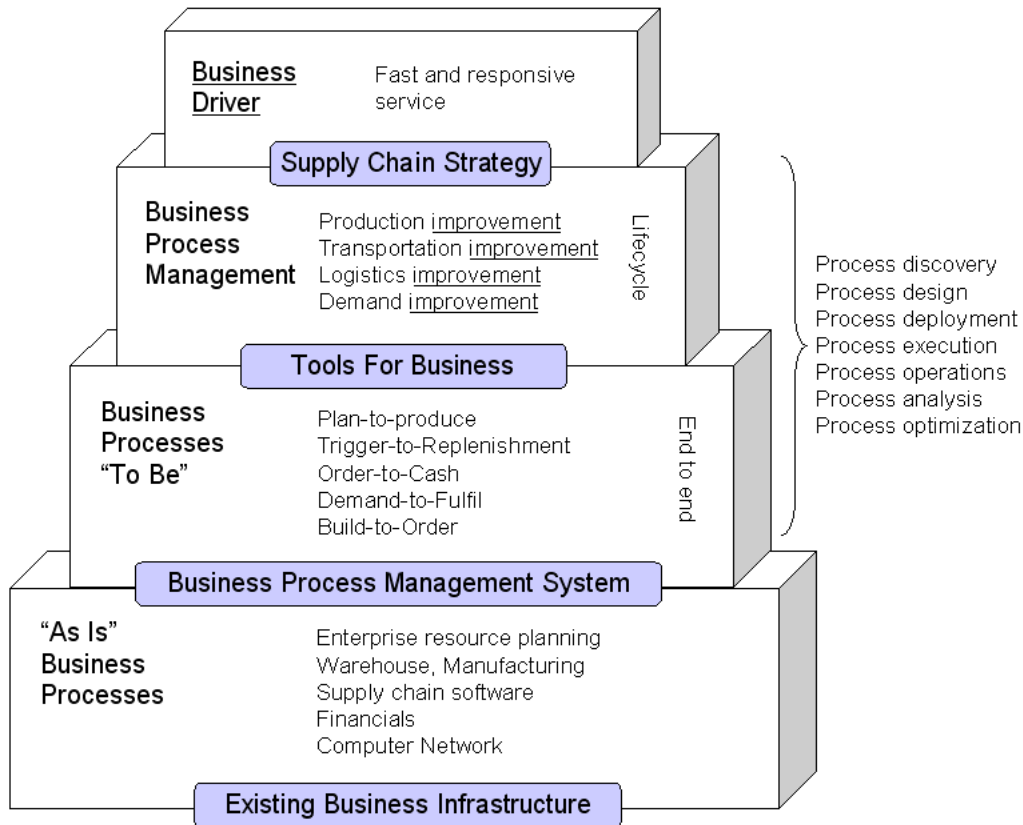


Figure 2 - Smith and Fingar's Supply Chain BPM

Can't we just speak the language of process?

Over the first fifty years of commercial IT, the level of abstraction used in developing business information systems has continually moved higher and higher, from wiring plug boards, to assembly languages, to COBOL, to objects, to components and finally to a business paradigm, the business process. Each new paradigm required its own terminology, and now it's imperative that we address the terminology business people use.

After fifty years of IT automating the business, the BPMS is the business analyst's CAD/CAM system that allows the business to automate IT. The significance of the BPMS is that BPM can now be of the business process, for the business process and by the business process. This is the message of BPM and the message obfuscated by Kalakota and Robinson's book and an IT industry enamoured with Web services technology and terminology. While the *Harvard Business Review* is telling business leaders that "IT doesn't matter," the IT industry continues to baffle those same business leaders with technology jargon. It's time to call a business process a business process. BPM isn't some technical integration layer buried in a Web services technology stack; it's the king of the hill, for what businesses really want is direct control of business process lifecycle improvement, not by IT-proxy.

For the business people that control IT investments and spending, Web services terminology is yet another foreign language they don't care to master—to them *Web*

services don't matter. They do indeed understand business processes, more than many IT people care to admit, so perhaps it's time for IT to master their language, the language of process. To do so will require re-factoring technical architectures with a process-oriented architecture.

A future article will describe the process-oriented architecture of BPMSs in more detail. The article will explain how the BPMS leverages past investments in IT and the way in which it interacts with its operating system, just as RDBMS was built on Unix. The operating system for the BPMS is IT's new commodity, the standards-based, networked operating system of the Internet era—Web services.

+++