

Executive Summary:

BPM products will usher in a major shift in the efficiency with which business managers are able to control business processes. Software applications will increasingly be organized and managed by BPM tools that business managers can control. Vendors are rushing to create tools that can be used to create BPMS applications.

Contents:

- The Forerunners of Today's BPMS Tools
- BPM Tools
- Other Features That BPM Tools Might Have
- Developing Software With a BPM Tool
- BPM Tools Versus a BPM Suite
- BPM Tools, BPM Suites, and Application Servers
- BPM Tool Features Recap
- What Will BPM Tools Be Used To Accomplish
- BPM Tool Vendors
- Sources of BPM Tools
- Summary



Paul Harmon
Executive Editor
pharmon@bptrends.com

www.bptrends.com

BPM Tools

This month the BPTrends Newsletter will focus on Business Process Management tools. We will define BPM tools to reflect the usage we think enjoys the broadest support from a wide variety of users and analysts, and then use it consistently in future newsletters.[1]

In essence, a BPM tool is used to create a BPM application (or a Business Process Management System.). A BPM application uses a business process description to coordinate the execution of a set of implementation elements. The implementation elements can be either employees, who perform manual tasks, or software components that perform automated tasks. The primary importance of a BPM application resides in the fact that it helps managers organize resources in terms of processes, and subordinates software applications to the achievement of business process goals. Thus, a BPM system is more than another software automation initiative. On the other hand, BPM applications depend on software tools, and, in most companies, the creation of BPM systems represents the IT initiative with the greatest potential to improve business processes.

BPM tools represent an evolutionary development with major roots in business process modeling, CASE, workflow, rule-based systems, EAI, and packaged applications. In the past three years, vendors who would formerly have positioned their products in one of these categories have repositioned their products and now refer to them as BPM Tools.

Gartner estimates the revenue from BPM product sales reached between \$520 and \$543 million in 2003. Dataquest estimates that the worldwide BPM tools market grew 20% in 2003 to \$543 million and expects that it will continue at a double-digit pace through 2004. Keep in mind that most of these sales are sales that would have been recorded as workflow or EAI sales two years earlier. Nevertheless, the fact remains that many companies are interested in developing BPM applications and are buying BPM tools to explore the possibilities. Delphi, in its 2003 worldwide survey of 500 companies, found that 48% of the company managers they contacted said they were in an early stage of evaluating BPMS. 23% said they had no plans to develop a BPM application, and 29% said they were beginning to deploy BPMS applications.

The shift to BPM tools, for many companies, may only represent a change in how they refer to their software tools, but the interest in developing BPM applications is real. In this issue of BPTrends we will provide an overview of the reasons for the interest in BPM tools, and an introduction to the nature of BPM tools. In future issues, we will explore the problems of developing and managing BPMS applications, consider detailed case studies, and compare and contrast the leading vendors in the BPM tools marketplace.



The Forerunners of Today's BPM Tools

Historically, software applications have been developed to automate specific activities within functional departments. Thus, early applications were created to automate bookkeeping in the accounting department, or to capture and maintain the names of employees for payroll. In the Nineties, business process gurus like Tom Davenport and Michael Hammer argued that companies could achieve major breakthroughs in efficiency if they would redesign their business processes to take advantage of the latest IT techniques. BPR led to some successes, but didn't achieve anything close to what it might have, simply because the IT techniques available in the early Nineties were not able to support the kinds of changes desired by most of the companies that attempted BPR. Two major IT technologies, however, became popular as a result of early BPR efforts—workflow systems and packaged applications.

Early workflow systems sought to automate document processing. In essence, paper documents were scanned and represented as data. Then, instead of passing paper documents from one clerk to another, electronic data was transferred by a workflow system from one clerk's computer to another. To design a workflow system, one began by creating a process workflow diagram that showed how the data or documents were to flow through the organization. If one wanted to subsequently change how a document was processed, one simply changed the diagram, and the workflow system automatically responded to routing the document in the manner specified by the redesigned diagram. Workflow systems had an interface that allowed analysts to specify the flow of documents, store the actual documents as data in a database, and rely on a "workflow engine" to implement the diagram, as the process is actually executed, by moving actual data to specified computer terminals in the order indicated in the diagram.

Packaged applications consisted of dozens of software applications, or modules, that could be strung together to support different company processes. Ideally, by choosing the right modules and stringing them together in the right order, a company could create a software system that processed data to support its native processes. Packaged application systems have, more recently, been described with terms like ERP systems, enterprise systems, or CRM systems, depending on their emphasis. Each consists of anywhere from ten to hundreds of software modules. Workflow-like engines are usually used to manage the flow of control from one application module to another. The vendors have competed by adding modules, based on the

idea that if a company offered enough different modules, a user could surely find the right subset to support whatever process the company wanted to implement. In fact, most companies have installed large numbers of ERP modules, and still find that they have islands of automation that don't integrate into comprehensive solutions. Worse, since the underlying technology initially used by the packaged application vendors was developed in the late Eighties, it isn't very flexible.

The Internet, the Web, and associated open standards have provided companies with the flexible technology that was missing in the early Nineties. Using open Internet standards and utilities, companies can now create much more flexible systems than they could in the early Nineties. If software modules are encapsulated as components, and the Internet data language, XML, is used to move messages between the components, very flexible systems are possible. The Internet runs on phone lines that already exist almost everywhere. User can access Web applications with browsers that rely on interfaces that are much easier to create and maintain the alternative user interfaces. All this means that companies, today, are in a position to build much more comprehensive and agile systems than they were in the Nineties.

In the past 3-4 years, companies have begun to move beyond the idea of portals and distributed applications, and have embraced the idea that a Service Oriented Architecture (SOA) can be created that will support virtual processes. The idea is that your company doesn't need to do everything itself. It can create processes that include subprocesses created and managed by other companies. Thus, many companies now routinely rely on a company like UPS or FedEx to handle the final warehousing and delivery of their products to their customers. At the same time, however, the company doesn't simply pass its products to UPS or FedEx and forget them. It relies on information passed back via the Internet to track the deliveries. Thus, a customer can come to a company's website and learn exactly where the package is in the delivery chain. In effect, from the customer's perspective, the UPS or FedEx process is just a subprocess within a company's overall value chain. This kind of integration is made possible by the fact that both the manufacturer and the delivery company are using the Internet, that both company's software applications are encapsulated as components, and that they are all communicating via XML.

The Web has also made it possible for customers to go online and connect with potential sellers. This has led to a

revolution in the way companies think about applications and products. Historically, companies were likely to think of products in terms of features that might be useful to targeted audiences. Today, companies are much more likely to think in terms of processes. Companies now realize, as a result of watching Web interactions, that customers increasingly think of products in terms of the series of steps they need to go through to get what they desire. What series of steps does the customer need to go through to get information from your company? What does it take to arrange to buy a product? How much trouble is it to find out when the item will ship? What steps does the customer need to go through to return a defective product? These kinds of concerns have always worried process designers, but now they also worry CEOs and product line managers.

Thus, almost without realizing it, companies have arrived at a new era in software design. Instead of designing applications that do specific things, companies are increasingly thinking of creating process specifications that then manage whatever applications are needed to implement the process. One could say that workflow has grown up, and has extended itself to support the Web. Or, one could say that we have entered the age of Business Process Management Systems.

BPM Tools

In essence, a Business Process Management tool is a software package that allows a business manager or business analyst to describe process, and, later, as needed,

to modify the process. From a software architecture perspective, one could describe BPM products as a new layer of software that sits above applications and uses business process specifications to coordinate applications.

A BPM software tool includes a process-diagramming interface for the manager to use to define the process and a workflow or BPM engine that starts applications when they are needed and terminates them when the process no longer requires their services.

There's quite a bit more to it than our definition suggests, but let's start with a simplistic overview. In Figure 1 we picture the two core BPM elements. One is the description of the process. The other is a BPM engine that follows the script implicit in the process description and manages the invocation of applications when the process is actually executed. In effect, a business analyst describes what is to be done, and the BPM engine then "reads" the description, invoking each implementation component in order.

Let's be sure we understand a BPM tool's primary value claim. BPM tools make it possible for business analysts to change how processes work without having to ask IT to reprogram. Some claim any business manager would be able to do this, but that's unlikely, except in the case where the business manager feels really comfortable with software systems and process diagrams. (Recall that surveys have shown that most of today's business managers do NOT define processes with diagrams. Instead they use text outlines.)

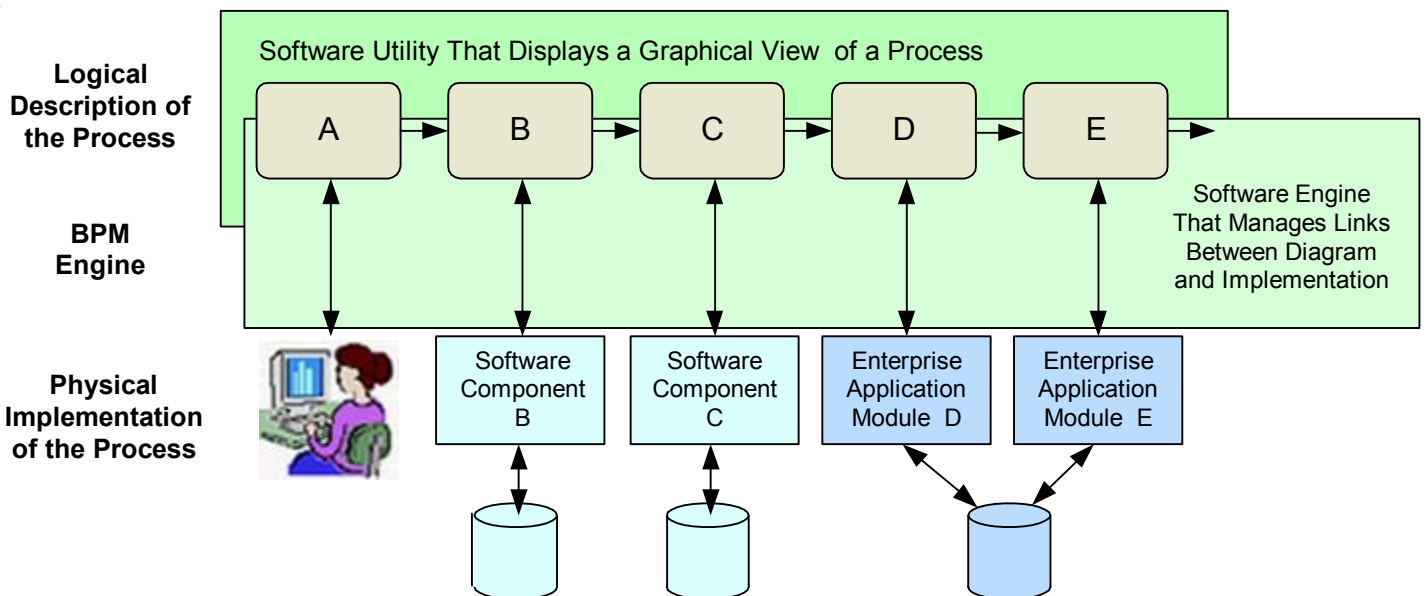


Figure 1. The two core elements of a BPM tool.

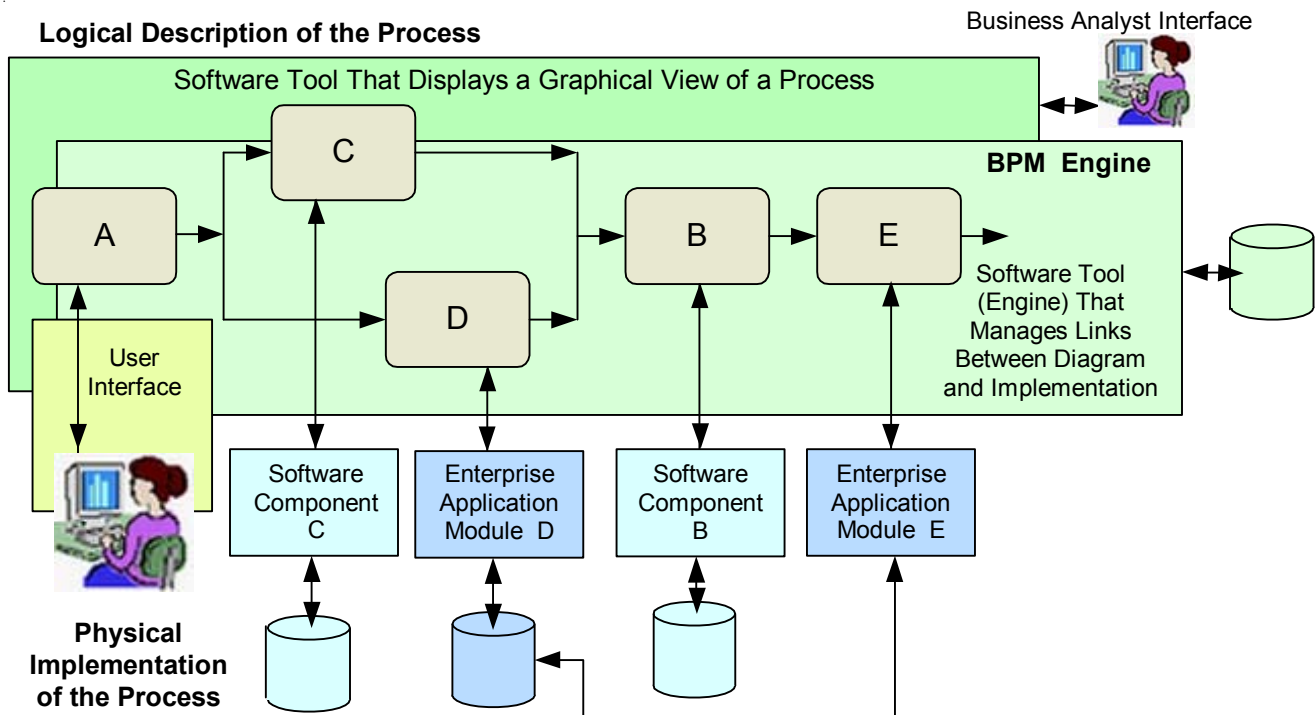


Figure 2. A BPM tool has been used to reorganize how a process is implemented.

Figure 2 suggests how a business analyst might have used a BPM tool to change a process diagram and thereby automatically change the underlying flow of the implementation software. We assume that the same underlying implementation components are still in place and that they function as they did in Figure 1. Now, however, the order in which they are invoked has changed. Moreover, the changes have been accomplished without the intervention of IT developers.

The ability of a BPM tool to reestablish links to underlying software components without the support of IT requires a rather flexible BPM engine. We will discuss the implications of this flexibility a bit later. Meantime, we want to underline what the BPM tool did NOT do. The BPM tool, as we have defined it, did NOT create any new components. It simply allowed the business analyst to rearrange the order in which existing components were used. Some BPMS advocates have suggested that BPM tools will “automatically” generate the code needed to provide new implementation functionality. We don’t believe that will be a key part of most BPMS products. We’ll also consider that claim later.

Before that, however, let’s consider the elements required by the BPM tool we have pictured in Figures 1 and 2 that we have not discussed yet.

A Process Design Tool and a Database

Most BPMS tools will have a process design utility that a business manager or analyst can use to define and revise a process. If the BPM tool vendor really hopes that business managers will be able to use the tool, the design utility had better be really easy to use.

At the moment, each process modeling, workflow, and EAI tool on the market offers a slightly different notation. In the past half-year, there have been two major efforts aimed at creating a standard notation to be used when drawing a process-diagram. One, by the OMG, has resulted in a new version of UML, 2.0, which includes a specification for an improved activity diagram. The other effort, by a broad group of vendors working under the sponsorship of the BPMI organization, has created a notation especially tailored for BPM tools.

The BPMI specification, called the Business Process Management Notation (BPMN), supports two levels of detail. Using only the simplest set of symbols, a manager can describe a process. By adding more detailed symbols, a business analyst can refine the process description so that it is so precise that a BPMS engine can execute it. Hopefully, these two approaches can be merged, and a standard notation will emerge. It would be a lot easier to promote the

use of BPM products by managers if managers could learn a single notation and then be able to understand any process diagram they look at. This movement will be accelerated if companies considering acquisition of BPM products indicate that they will only acquire BPM products that support a standard, open business process notation.[2]

Just as every BPM tool will support a business process design capability, every BPM tool will have some data to store. It will need to store the process description itself, the BPM engine, and data that is created when the process is actually executed. Similarly, the BPM tool will probably have a directory and security information about which employees are to be sent information and who can change the process flow.

In Figure 2, we pictured an analyst interface to the process design tool and a database associated with the BPM engine.

Business Rules

Very simple processes can be described by diagrams, but more complex processes require that the business analyst incorporate business rules in the process description to handle decisions that need to be made during the execution of the process. The decisions made during some activities may only require one or a few rules. Other activities, in which complex analysis, configuration, or design decisions are made, may require tens or hundreds of rules. Put a different way, some process descriptions model processes that are always executed in exactly the same way. Most processes, especially those involving customers or suppliers, vary a little each time they are performed. In cases where slight changes in the context result in different decisions or paths through the process, rules are required to handle each of the different situations that might occur when the process is executed.

There are different ways of incorporating business rules. The easiest, from the business analyst's perspective, is to incorporate the rules as properties of activities. Thus, an analyst might click on an activity box in a process diagram and open a window that allows the analyst to enter rules. As long as one can specify clearly what the outcome of an activity will be, the rules can be stored in a database and processed by traditional means. If the activity is complex and involves situations and decisions that are hard to anticipate in advance, one needs to rely on rule-based systems that use an inference engine and rules that can deal with uncertain outcomes. An inference-based rule system might conclude, for example, that an application fulfills 90% of the credit requirements you have established,

and then recommend to an employee that the application be approved. Or, an inference-based system might be used to determine how to pack a box with a set of purchased items that are unique to each customer.[3]

We believe that all BPM tools will include rules and rule-processing capabilities. We don't believe that all will necessarily support inference-based rule systems, although the more sophisticated BPM tools probably will. By the same token, some vendors will create BPM engines that handle inferencing rules, and others will incorporate inferencing systems from third parties as special utilities.

So, assume that some of the activities shown in Figure 2 will contain business rules to help determine how to accomplish those activities. Similarly, there may be rules associated with the arrows to determine the flow of data between activities.

Support for Manual and Automated Activities

The actual implementation elements are not part of the BPM tool as we have defined it.

In the example shown in Figure 1, we picture three different kinds of implementation elements. First, we have an employee manually implementing activity A. Presumably, the BPM tool sends information to the employee. Perhaps it's a scanned application form. Perhaps it's a request for the employee to approve a change in the customer's status, or to approve a purchase. We assume that the employee takes the appropriate action and enters it into his or her terminal and that completes activity A. Once activity A is complete, the BPM engine invokes activity B.

There will probably be products that claim to be BPM tools that lack the ability to support employees, and will be confined to managing the execution of software components. We don't think this limitation will prove widely acceptable. In the long run, we expect that every serious BPM tool will support some type of user interface, probably a browser-based interface that will allow the product to incorporate manual activities into the overall process description.

Just as the process designer utility allows a business analyst to open an activity box and insert rules, it will allow an analyst to open an activity box and insert information on manual tasks.

Manual tasks are usually described in terms of "work items" or "work lists" and are stored in the BPM tool database. Presumably, a business analyst, using the process designer,

would be able to create or modify the work lists associated with specific activities just as he or she could modify rules associated with specific activities.

We assume that activity B and C are implemented by software components developed by the company that created the process the BPM tool is to manage. Perhaps one is a legacy application written in COBOL that runs on a mainframe. Perhaps the other is an application written in C++ and running on a Unix workstation.

On the right side of Figure 1, we pictured modules from an enterprise suite – say, an ERP application from SAP or PeopleSoft. If our BPM tool is actually going to link and coordinate these different applications, it is going to have to have at least some of the capabilities of an Enterprise Application Integration (EAI) engine, or it is going to have to rely on a powerful middleware system, like the OMG's CORBA, perhaps as implemented by IONA or by IBM's WebSphere. This has led some analysts to suggest that BPM tools are really EAI tools. They are more, but they incorporate EAI technologies.

Other Features that BPM Tools Might Have

Now let's go beyond the assumptions we made in creating Figure 2 and consider other capabilities that might be included in a BPM tool.

Business Activity Monitoring

Many have suggested that all BPM tools will also include a Business Activity Monitor (BAM) component that will provide managers with information about how the system is performing. Some BPM tools certainly will include BAM. We doubt that all BPM tools well..

For BAM to work effectively, a lot of work needs to be put into the user interface. Imagine a complex process. In most cases, it would be overwhelming to show the entire process to a senior manager with numbers constantly changing as items moved through the process. BAM works best if it provides senior managers with good overviews that only emphasize the most important details. If the process is low-level, then a senior manager is not going to be interested in the details, only the outcomes. If the process is a high-level process, then it will be complex, and someone is going to have to work to tailor an interface so that it only shows the most important information.

In fact, most serious BAM products incorporate intelligent data mining and simulation capabilities to analyze the data,

make predictions, and, in some cases, to make recommendations. At the moment, at least, these are not tools that managers or business analysts can use unaided.

It would be nice to think that one could build the capability into the average BPM tool to automatically generate a good, high-level overview of the process, with appropriate data graphics to sum up the key events. It's more likely that business managers will focus on using BPM tools to manage and modify processes, and leave the design of good BAM systems to specialists. This isn't to suggest that BPM tool vendors might not sell tools that were designed to integrate easily with BAM tools. This will undoubtedly happen, but we expect the BAM will be an added product, programmed separately, and not an integral part of most BPM tools.

Separately from BAM, most BPM tools will provide ways for users to monitor the processes that the BPM tool is managing. This kind of runtime information may help a supervisor or a business analyst, but its rather different from the kind of high-level overviews that most managers currently associate with BAM products.

Simulation

Some writers have suggested that simulation will be a key element of BPM tools. It would certainly be a nice capability to have, if one was skilled in the design of simulation systems. Most business managers and analysts could not use simulation. It's complex to set up and quickly becomes overwhelming if one doesn't really understand what one is doing. In most cases, simulation is only done well by experts. Most of the leading business process analysis tools incorporate simulation capabilities, and the vendors report that they are hardly ever used. We don't expect that simulation will be an important part of most BPM tools.[4]

BPMS and Service Oriented Architectures (SOA)

For many, BPMS is synonymous with the coordination of Web services. In other words, BPM tools, from this perspective, should be designed to manage processes that are distributed on many different servers. In effect, each service can be conceptualized as a subprocess of the virtual process that the BPM tool manages. Some companies will use BPM tools for exactly this purpose. Others, however, will use BPM tools to manage internal packaged applications or local Sarbanes-Oxley systems. It's too early to speculate on all of the uses companies will find for BPMS technology.

At this point, suffice to say that some BPM tools will be designed to handle processes that are distributed across the Web while other tools will be tailored to emphasize

processes that reside within the bounds of a single organization, or run within a single location. In the long run, most BPM tools will probably need to support SOA, but most probably won't provide that capability for awhile yet.

BPEL and Other XML BP Languages

Closely related to the Web and SOA, is the use of XML business process languages. Those who are planning to implement SOA systems have been working for the past several years to create an infrastructure for such systems. We are all familiar with the various protocols that have been proposed, including SOAP, UDDI, WSDL, and, XML. At its simplest, an XML business process language can be a set of definitions designed for a specific industry to simplify passing a particular type of business information. There are over 100 different BP XML languages for industries. They range from languages to help banks pass information to languages that allow printers to pass data about manuscripts. Some companies will create BPMS tools that support one or more of these languages.[5]

More ambitious efforts have focused on creating generic XML business languages that incorporate a BPM engine as a part of the language. A few years ago, IBM was working on WSFL, Microsoft was working on XLANG, and a vendor consortium, BPML, was working on BPML. In 2003 IBM, Microsoft and BEA combined WSFL and XLANG and proposed BPEL4WS (Business Process Execution Language for Web Services, which has since been abbreviated as BPEL). The developers have since submitted BPEL to OASIS, a Web standards body, and it is being revised and extended. Meanwhile, before it is even an official standard, BPEL has become the *de facto* standard for managing Web-based processes.[6]

A programmer could, using BPEL, design and manage a Web based process. Some will. More attention, however, has been focused on the idea that BPEL will be incorporated into BPM tools (1) to provide a standard BPM engine, and (2) to provide an open description of processes thereby making it easier for companies to exchange information about processes.

It's too early to determine exactly what the final version of BPEL will look like. At this point it looks like most BPM tool vendors will incorporate some version of BPEL, but it's also likely that other ways of creating BPM engines will be explored.

Pi Calculus

There has been quite a bit of disagreement about the need for a theoretical foundation for BPMS products. Some have argued that BPMS products must be based on new theoretical paradigms, like Pi Calculus, or at least on new languages, like Business Process Execution Language (BPEL).[7] In our opinion, this isn't a question that needs to be answered at this time.

Some BPM tools will rely on a theoretical approach, like Pi Calculus, but most won't. (BPEL isn't a complete implementation of Pi Calculus and probably never will be.) Ultimately, users will focus on the results produced by the various BPM tools and not on the internal code used to achieve the functionality. Time will tell if it is more effective to extend existing models or create new models based on new paradigms. Some BPM tools will be developed using BPML or BPEL and will support a more or less complete version of Pi Calculus. Others will be developed that only provide some of the functionality provided by Pi Calculus. Still others will be developed to support paradigms that are only now being developed. The key will be the performance that the tools offer. Practical results always trump theory.

That said, lots of vendors will embrace BPEL. BPEL is as much about open standards, XML, SOA, and the Web, as it is about Pi Calculus. Lots of companies will want BPM tools that can communicate with other BPMS products and that can support open standards. BPEL will provide that. Some products will be entirely written in BPEL or some other language that implements Pi Calculus. Most will mix some conventional code with some BPEL code to achieve the functionality they think their customers will demand.

Developing Software with a BPMS Tool

Some authors extend the idea of a BPM tool to include code generation. Consider the example shown in Figure 3. In this case, our BPM tool is managing a process in which employees perform one activity. Suppose the business manager responsible for this process decided that it would be better if the tasks performed for activity A were automated and done by a software component. In this situation, the question arises as to whether the BPM tool should manage the automation function, or if the activity should be automated by buying or building a software component and then simply integrating the component into the existing process.

To clarify this, ask yourself what would be required to incorporate component development into the BPM tool? One possibility would be to allow the business analyst to click on the Activity A box and diagram a subprocess in such



detail that the diagram could then be used to generate code. This is precisely the path that some BPM tool vendors have taken. BPEL and BPMN were developed, at least in part, to support this kind of approach. If one thinks about this, however, one realizes that the average business manager or analyst probably won't be able to specify the subprocess with enough precision to generate code. In effect, one is stretching the BPM tool and converting it into a kind of CASE product. Obviously, some activities are very simple, and there will undoubtedly be some cases in which an activity could be easily defined in enough detail to support code generation. In our opinion, however, those cases will be the exceptions.

It's more likely that the tasks performed by the employees who currently perform activity A will require more analysis and some programming to establish how data is to be handled and to assure that the activity is performed quickly. This is a debate that Howard Smith and David Frankel touched on in articles written for BPTrends, and we refer readers to those white papers for a more detailed discussion of the issues involved.[8]

BPEL is designed so that it can be used to define a process with enough precision that a BPM engine could interpret the BPEL description and, in effect, generate the code for a component. We expect that BPEL will be used in this

manner when dealing with simple subprocesses, and that it won't be used in this way when more complex subprocesses are involved. Thus, in most cases, we suspect that BPM tools will support limited software development and will also support links to software development tools.

We expect that the most sophisticated BPM tools will support using the process tool to define a process that can then be passed to a software development tool for more detailed analysis and code generation. Thus, the business analyst might define activity A in some detail, and then pass the resulting process description to IT analysts who would use the OMG's Model Driven Architecture (MDA) to shift from a process model to a class or sequence model and proceed to use a UML-MDA-based development product like IBM's Rational Rose to define a software component with supporting database and interface elements. This is the approach that has been proposed by David Frankel and others, who see BPMS and MDA as very complementary approaches.[8]

For most companies, a clean interface that facilitates an easy transfer of information from the BPM tool to MDA/UML-based software development tools will probably be the desired BPM tool functionality.

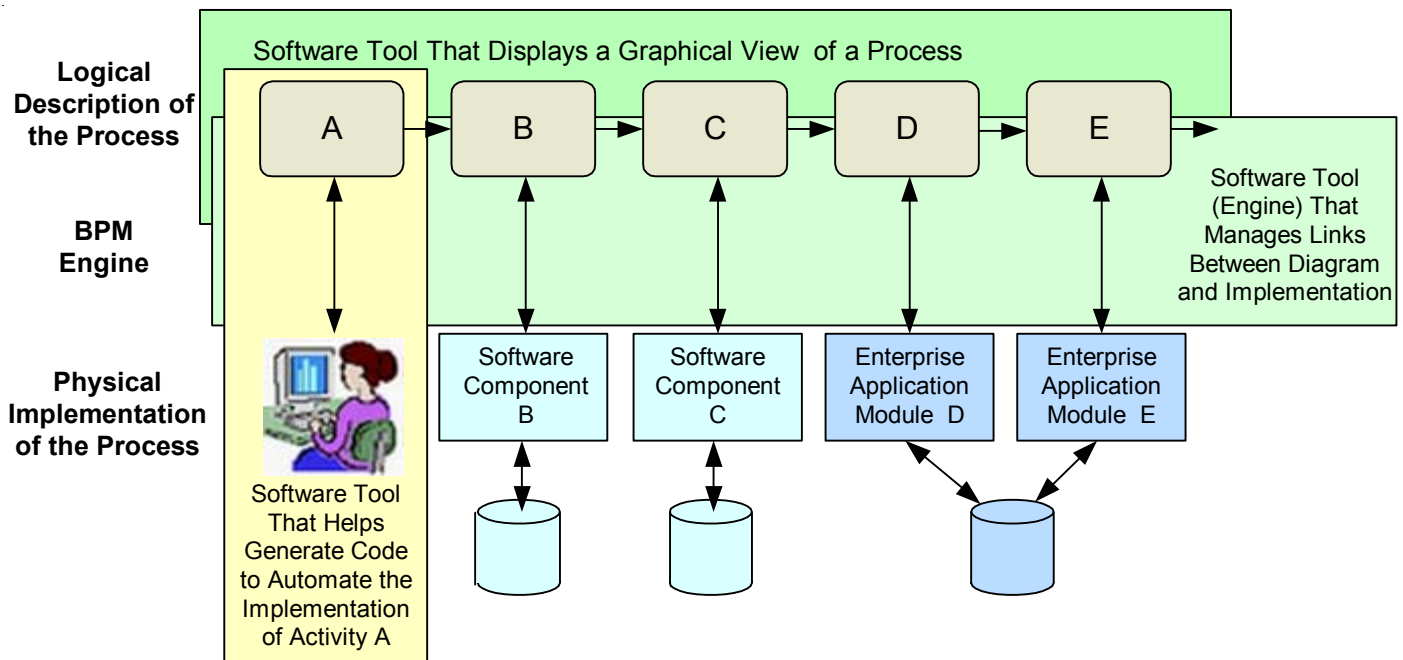


Figure 3. A BPM tool with a software development capability..

BPM Tools Versus BPM Suites

So far, we've described BPM tools and emphasized that a BPM tool does not include the components or application modules that it is managing. A BPM tool simply controls the message flow and the invocation of components and modules.

In fact, we expect to see companies offering products that combine a BPM layer with a specific set of components or modules. All of the major packaged application vendors are interested in BPMS. The logic, for these vendors, is to incorporate a BPM capability into their existing product offerings. In effect, you will be able to buy a SAP BPM product that is tailored to manage SAP application modules. It may or may not be good at invoking and managing non-SAP application modules. For companies that are already committed to using a given set of application modules, buying a BPM product tailored to work with the modules may make excellent sense.

For other companies, that are more interested in integrating and managing a more diverse suite of components or modules, this will be the wrong way to go. These companies will want an independent, generic BPM tool that can manage any set of components or modules they decide to use.

Anticipating this development, we will use the term **BPM tool** to refer to a package that includes a modeling element and a BPM engine and associated database, but no application components or modules.

We will use the term **BPM suite** to refer to a package that combines a BPM tool capability with a pre-established set of components or modules.

To summarize, we are suggesting that a BPM application or BPM system can be created either by programming in a BP XML language, like BPEL, by using a BPM tool, or by using in BPM suite. In some cases, depending on how comprehensive the BPM suite is, there won't be much difference between a BPM suite and a BPM application. In other cases, one will start with a BPM suite and add additional components to create a more extensive business process management application.

Gartner has begun to give its own names to BPM-based applications. They have referred to some SOA-based applications as **composite** applications. At the same time they have proposed the term **business process fusion**

applications to describe applications resulting from extended package application suites that incorporate BPM capabilities.

Our use of the term BPM suite is similar to what Gartner has termed a Business Process Fusion platform. Our use of the terms BPM application or BPM system is broader and covers both BP Fusion and Composite applications.

Chordiant's BPMS Suite

Figure 4, on the next page, illustrates the Chordiant BPM Suite, which Chordiant markets primarily to financial, telecom and retail organizations.

Chordiant provides a Process Design Tool in which any process can be examined and modified. The business analyst can add steps to processes and specify rules to control processes. The analyst can also define Work Lists that will guide employees in the execution of specific processes. The BPM Engine it referred to as Chordiant's *Straight Through Service Processing*. It controls the execution of the processes and the invocation of components. The Chordiant BPM suite has three tailored employee interfaces that provide employees with access to data and to the various processes clustered with each interface. Thus, when a call center employee contacts a customer, a portion of the interface shows all the data on the customer. Chordiant is designed to allow the business analyst to easily link in non-Chordiant databases and legacy applications into a larger process. Another part of the screen shows the employee what processes are available. Typical processes that the employee might invoke include Change of Address, Lost Credit Card, Change of Credit Card Limit, and Closing an Account. Once the employee selects a process, the interface presents the employee with a Work List, and guides the employee through the steps involved in the process. Thus, Chordiant provides a nice example of a BPM suite that is tailored to support employee activities rather than the activation of entirely automated components.[7]

BPM Tools, BPM Suites, and Application Servers

Java started life as a programming language. In a reasonably short time, complete Java development packages, specifically J2EE, were implemented in conjunction with application servers that were tailored to support J2EE. Thus, when one thinks of Java today, one often thinks of J2EE application servers, like IBM's WebSphere or BEA's WebLogic server.



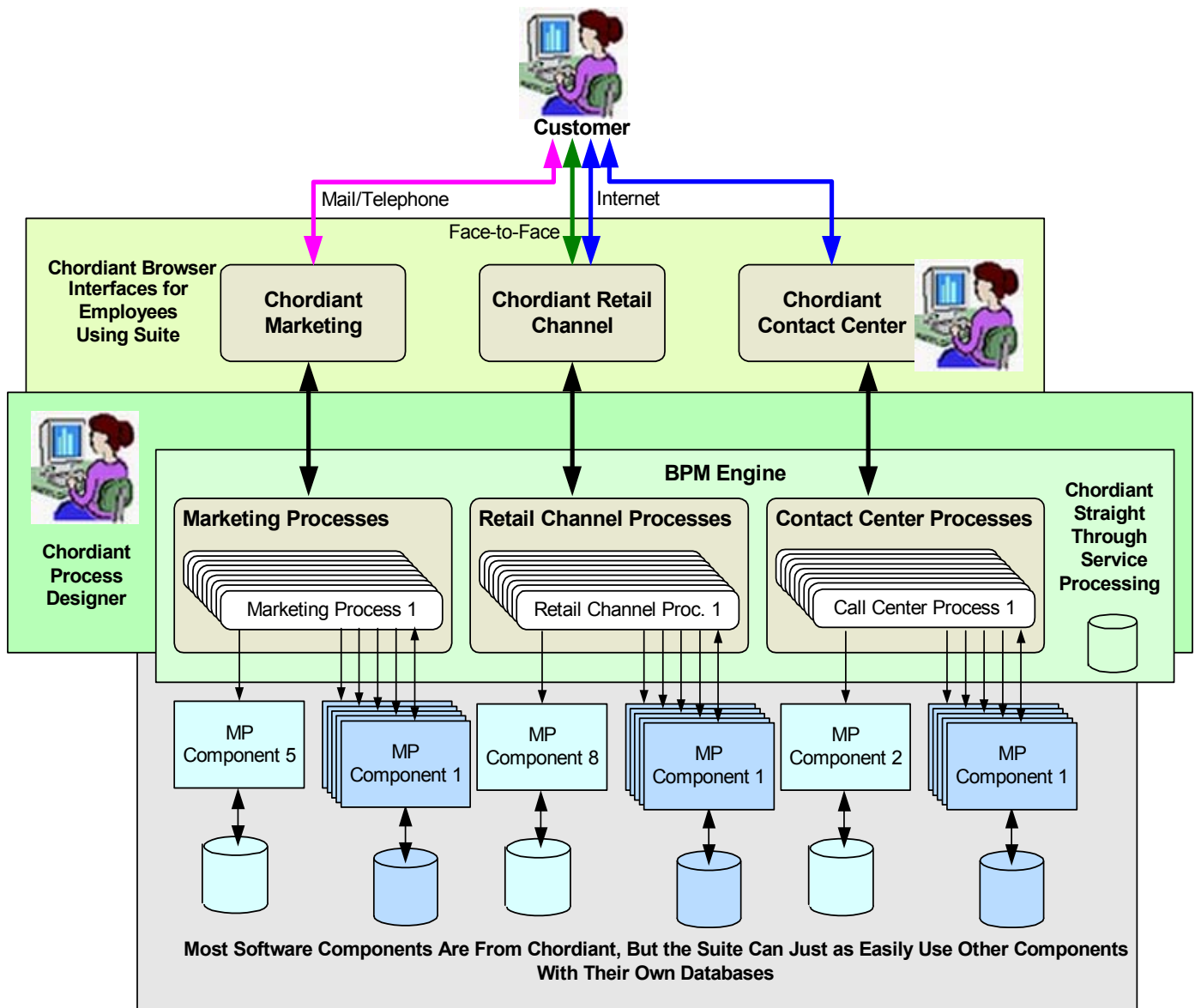


Figure 5. Chordiant's BPM suite.

It is likely that major infrastructure vendors like IBM, BEA, and Microsoft will move toward BPM server-based tools. Similarly, packaged application vendors like SAP and Oracle who incorporate BPM capabilities into BPM Suites may support them on their own application servers.

Most BPM tools and suites will probably be designed to run on multiple server platforms. Thus, being an application server product will not be a feature of all BPM tools or suites, but a special feature of some versions offered by specific vendors.

A BPM Tools Recap

Table 1 summarizes the various BPM features we have discussed. We expect lots of variety in the next couple of years as the market sorts out what is valuable about generic BPM tools and what is only needed for more specialized or industry specific applications. After 3-4 years, predictably, only a few leading BPM tool vendors will predominate and the rest that survive will shift and focus on special niche markets. In the next 2-3 years, however, expect to see lots of vendors enter the BPM tools market.

What Will BPM Tools Be Used To Accomplish?

As we have already suggested, it's too early to try to define all the ways that BPM tools and suites will be used.

Some small company might conceivably use a BPM product to model all its business processes. The vast majority of companies, however, won't attempt anything like that. Instead, they will reserve BPM tools for new applications, or especially critical applications.

For many, the major use of BPM tools will be to gain better control of the company's existing packaged applications. Thus, BPM tools will be used to model the processes that are currently implemented by EAI or CRM modules from companies like Oracle, SAP, PeopleSoft, Siebel, and others.

For other companies, BPM tools will be used to create Service Oriented applications that integrate processes from multiple sources. In this case, flexibility and support for Web standards will be at a premium.

Leading BPM Tool Vendors

Forrester recently evaluated the BPM market and suggested that the top ten pure-play BPM tool vendors were FileNet's Business Process Manager 2.01, Fuego's FugeoBPM, HandySoft's BizFlow 8.7, Intalio's Intalio|n3, Lombardi Software's Teamworks 4, Metastorm's e-Work Version 6, Pegasystems' PegaRULES Process Commander 4.0, Savvion's BusinessManager 5.0, Staffware's Process Suite v10, and Ultimus' BPM Suite 6.0.

Going further, Forrester analyst Sharyn Leaver argued that the three strongest players at the moment were Savvion's BusinessManager 5.0, Intalio's Intalio|n3, and Fuego's FugeoBPM and that Savvion's product was the strongest across the board.

It's important to note that Forrester's analysis is confined to "pure-play" product vendors, companies that have created new products to compete in the BPM space. Vendors from other areas, like Workflow and EAI, are going to be strong contenders as well, and, even more important, systems vendors like BEA, IBM, and Microsoft will undoubtedly offer superior products integrated with their

middleware and operating system tools. Similarly, packaged application vendors will offer BPM suites that are integrated with their existing packaged applications.

Once the systems and packaged application vendors enter the market, most of the pure play vendors will fall by the wayside or become niche specialists. In other words, it is way too early to determine which vendors will ultimately dominate the BPM market.

Sources of BPM Tools

As the interest in BPM tools continues to grow, we expect that many vendors who are currently positioned in categories listed below will seek to reposition themselves as BPM tool or suite vendors. We fully expect that in a year or two, analysts will begin to divide BPM tools into subcategories, to emphasize that some tools are better for one type of

| A BPM Tool | |
|---------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Must Have | <ul style="list-style-type: none"> - An Environment that allows the display and modification of a graphical description of a business process and the specification of business rules to define decisions required during the execution of the process. - A BPM engine that uses process description to control the invocation of manual or automated software components. - A BPM repository in which to store process and session information. - Ability to capture manual worklists and display them for employees. |
| Most Will Have | <ul style="list-style-type: none"> - An open standard XML language like BPEL that will facilitate Web Service processes and passing modeling information between products - A limited BAM interface that provides managers with information about the specific process being managed. |
| Some Will Have | <ul style="list-style-type: none"> - Ability to support simulation of a business process. - Ability to support the development of new implementation components by diagramming the component activity and then generating code - Ability to support the developing of new implementation components by passing requirements to software developers via a mechanism like MDA - The ability to generate senior management BAM dashboards - Be integrated with an application server environment |
| A BPM Suite | |
| To the list above, a BPM Suite adds specific software applications. | |

Table 1. Features of BPM Tools and Suites.

application while other BPM tools will be specialized in still another type of application. We've listed some of the companies that have produced or will produce BPM products and suggest some of their strengths and areas where we think they will be challenged.

Business Process Modeling and Analysis Tools

There are a wide variety of software tools that can be used to analyze and model business processes. Popkin's System Architect, Proforma's ProVision, Casewise, and IDS Scheer's ARIS, for example, all come to mind immediately. All of these tools support diagramming processes, most can capture business rules, and most store information in a repository. Some support limited code generation. None of these tools, at the moment, include a BPM or workflow engine that can coordinate changes in the process diagram with the execution of implementation components, but several have announced their intention to support BPEL.

Some companies may seek to create BPMS applications simply by generating BPEL code. We expect that some business process modeling vendors will support BPEL generation. Popkin has already released a version of its product that supports the BPMN process notation, and will probably add BPEL generation later. Other vendors will undoubtedly follow this same path. We don't, however, expect most BP Modeling Tools vendors to convert their products into BPM tools. Instead, most will remain analysis and redesign tools and will simply create ways to hand off redesigned processes to BPM tools for runtime management.

Workflow Tools

We're using workflow broadly to include both content and document management tools. Workflow tools already have most of the capabilities we expect to find in BPM tools. They already provide process diagramming utilities and, practically speaking, in most cases there is no real difference between a workflow and a BPM engine. Workflow tools already support the creation of process diagrams and the use of business rules. They incorporate a repository. The best can integrate manual and automated tasks. The workflow vendors will need to add some intelligence to assure they can handle process changes without programming, but this should be easy for them to accomplish. The leaders in the workflow market will incorporate BPEL, re-brand themselves as BPMS products, and establish early leadership in the BPMS market.

Simulation and Real-Time Tools

Most simulation and many real-time products are organized around a process model. The simpler of these products are simply business process modeling tools with simulation added as a feature. The more powerful simulation tools, like CACI's Simprocess and real-time tools like Gensym's G2, can be used to model a process and then implement and execute the process. These powerful tools are already very much like BPM tools; in some ways, the best are more powerful, except that most are narrowly focused to help automate specific types of real time processes. One or two might decide to become more generic and try to position themselves as a BPM tool. Others may integrate with application suites to create BPM suites. Most, however, like the BP modeling and analysis tools, will probably continue to focus on what they already do well.

Business Rule Tools

There are a number of Business Rule tools on the market. Most are either repository-based rule vendors, or are inference-based rule products that have evolved from expert system tools popular in the Eighties. The best have already formed partnerships with workflow and EAI vendors that need to offer support for rules in their products. Rules are an important feature of BPM tools, and inferencing is rather close to a BPM engine, but process diagramming capabilities are more important, and most rule tools are not presently designed to handle processes, as such. It will probably prove easier for the rule vendors to partner with vendors that have already created process tools, than for the rule vendors to add process diagramming capabilities to their products, though one or two will undoubtedly attempt it.

EAI Tools

EAI tools are good at coordinating the execution of diverse implementation components. Most, however, are not organized so as to be driven by a business process diagram, and changing most of these tools to enable them to operate in the manner of BPM tools would probably be hard. We don't expect many EAI tools to make it into the BPM category, though some will, and we wouldn't be surprised to see some partnering with other vendors to provide some of the EAI capabilities needed to support a BPM tool.

MDA Tools

Several vendors who have been working closely with the OMG have created tools that implement the OMG's Model Driven Architecture (MDA) standard. In effect, these tools are software development tools. The MDA standard, however, supports both UML activity diagrams and, via its Business Process Metamodel, could support other process diagrams.

Thus, it's conceivable that one of these vendors, say Adaptive, which is strong on both MDA and business process, might extend its MDA tool to support the BPM paradigm. More likely, however, these tools will be used by others that want to link their process diagramming capabilities to underlying software development tools.

New BPM Tool Vendors

A few companies have been established to create BPM tools from scratch. Most of these vendors are focused on implementing an XML BP language, like BPEL, and on offering users the ability to manage processes and to generate new code from the process paradigm. For reasons already stated, we believe process-based code generation will only play a limited role in the evolving BPM market. Most of these new companies will find themselves competing with BPM tools that have evolved from workflow, EAI, and other categories named above and will lack the capital or practical capabilities those other tools have developed over the past decade. One or two may ride the interest in SOA and BPEL to establish themselves. Others will probably prosper if they partner with other companies and, in effect, specialize in offering a powerful BPM engine that others can embed.

Major System Vendors

For lack of a better term, we intend this category to include IBM and Microsoft and a few others like BEA and SUN. IBM has BP modeling products (Holosofx), Software modeling products (Rational), Workflow products (MQSeries Workflow), EAI products, and MDA products, and they are already merging them all in their WebSphere suite. In addition, IBM, like Microsoft, is heavily committed to SOA and is one of the creators of BPEL, which it has turned over to OASIS for standardization. It's impossible to imagine that IBM won't produce a high quality BPM product, probably offered in conjunction with its WebSphere application platform. IBM does not sell packaged applications, but often partners with both packaged and best of breed application vendors who provide implementation components. IBM will probably offer both a stand-alone BPEL product and a server-based product designed to function with a wide variety of underlying application components supplied by its partners.

Microsoft will undoubtedly offer a BPEL-based product. At the same time, Microsoft is beginning to roll out its own CRM and EDS applications and may well offer a complete BPM suite. The other major infrastructure vendors will probably also offer BPM products.

Packaged Application Vendors.

Major Packaged Application Vendors like SAP, PeopleSoft, Oracle, and Siebel all have extensive sets of implementation modules. Their modules are supported by their own workflow engines. Although most of these vendors have process diagrams to describe how their modules might be assembled to support specific processes, the process diagrams are not central to the applications and can't be used to control the applications. A business manager can't use a SAP process diagram to automatically reconfigure a set of SAP applications. Most of the leading packaged application vendors created their products in the client-server era, before components or EAI. Most are struggling to shift their modules into components and to incorporate EAI elements that will allow them to link with other components more efficiently. This has proven an uphill struggle, but these vendors have lots of capital, and most will undoubtedly offer a BPM capability, probably by partnering with other vendors who are more closely aligned with processes.

Newer packaged application vendors, like Chordiant, for example, have already transformed their products into BPM suites. These vendors, with component-based products written in J2EE, have found it easier to create process tools and to link those tools to their workflow engine. Thus, although the older, more established packaged application vendors will have to struggle to achieve BPM, many newer packaged application vendors will rapidly and successfully reposition themselves as BPM vendors.

Summary

Interest in Business Process Management tools is intense at the moment. In many cases, the potential of this new technology has been over-hyped. Few BPM applications will ever offer real companies all the benefits that the most fevered promoters have promised. Worse, some companies will undoubtedly try to realize some of the more fantastic visions that have been proposed and will spend lots of money and achieve dramatic failures.

That said, we believe that the shift to BPM applications represents a major watershed in the way companies think about processes and software applications. Every major company should explore this technology. The key thing about BPMS, however, is very simple. BPMS suggests that businesses should begin by defining their processes, and then implement software components or applications to support those processes. If that idea takes hold, it will change the way IT operates, and that will be very important. It will be important because it will make companies more process-

oriented, and, therefore, more customer-oriented. And it will begin the reorientation of IT from an application-creation focus to a focus that emphasizes supporting business processes.

Secondarily, for many companies, BPM tools and suites will facilitate greater control by business managers and business analysts, and make it possible for them to quickly make important changes in applications without their needing to consult with IT.

Most of the claims about BPM tools eliminating software development are unlikely to be realized any time soon.

BPM tools are the natural evolution of workflow, EAI, and packaged applications. It is too early to tell what it will become, or what form it will actually take when it is mature. BPMS will evolve right along with Web Services and the interest in managing companies to achieve process goals. Without predicting the future, however, it is clear that most companies will use the BPMS applications that they create in the next few years to gain more control over their business processes.

Notes

[1] We've argued in the past that BPM should be used to refer to a variety of different approaches to Business Process Management and not tied narrowly to software-based approaches. We would prefer to use the term Business Process Management Systems (**BPMS**) to refer to the approach we describe in this issue of BPTrends Newsletter. In researching this newsletter, however, we noticed that all analysts and most popular articles use the term **BPM** to refer to the approach we are describing here. We don't want to try to confuse the market by insisting on a less common term, therefore, we bow to what has become the popular convention. In passing, however, we note that many managers and business process practitioners focus on improving process management practices without relying on software systems. Thus, we will continue to use the phrase business process management, broadly, and use BPM more narrowly to refer to software-oriented approaches.

[2] For more information on BPMN and UML Activity Diagrams, see the two white papers on the BPTrends site by Owen and Raj, *BPMN and Business Process Management*, and by White, *Process Modeling Notations and Workflow Patterns*. Go to www.bptrends.com and use Search, or check under Publications/White Papers.

[3] For more information on business rules and rule-based systems, see our July 2003 newsletter: *Business Rules: An Introduction*. Go to www.bptrends.com and use search.

[4] For a study of the use of simulation in the development of BAM systems, see the white paper by DeFee and Harmon, *Business Activity Monitoring and Simulation* (February, 2004) on the BPTrends website.

[5] For more information on XML BP languages, see our February Advisor on *XML BP Languages: An Update*. Go to www.bptrends.com, then Publications/Advisors.

[6] For more information on BPEL, visit BPTrends and check the Technical Brief by Brown and Szeffler, *BPEL for Programmers and Architects*. Go to www.bptrends.com, then Publications/Technical Briefs.

[7] For an argument about the role of Pi Calculus in BPMS, go to BPTrends and check three papers: Smith and Fingar's *Workflow is just a Pi process*, van der Aalst's *Why Workflow is NOT just a Pi Process*, and Pyke's *Does Better Math Lead to Better Business Processes?* All are under Publications/Articles.

[8] To review the technical arguments for the exclusive use of BPEL for code generation or for the use of MDA, visit the BPTrends site and check the papers by Dave Frankel, *BPM and MDA: The Rise of Model-Driven Enterprise Systems* (June 03) and Howard Smith, *BPM and MDA: Competitors, Alternatives or Complementary* (July '03). Use Search or look under Publications/White Papers.

[9] For more information on Chordiant's BPMS suite, check www.chordiant.com

Business Process Trends Newsletter is copyrighted and may not be reproduced for commercial purposes, either in whole or in part, without the express written permission of Business Process Trends. Business Process Trends grants individuals the right to download, print, and distribute this newsletter for non-commercial purposes only, and only in its entirety and in the format in which it appears on this site.