



Driving Business Applications with Real-Time Data

BP Logix
September 2015



Process Driven, Business Focused

www.bplogix.com

www.bplogix.com | (877) 627-5871

Introduction: Building Business Applications

We've arrived at a seminal moment in the evolution of business technology. For decades, organizations seeking to leverage technology to improve their business have had to choose between buying packaged software, or building custom solutions in-house. Each option carries some pretty heavy baggage.

Packaged software can be inflexible, expensive, and hard to use. The procurement process alone can take months, and implementation even longer. As the packaged apps pile up, so does the expense of managing the many vendors, licenses, and maintenance contracts associated with them.

In-house development provides the company with the most flexibility, and the best opportunity to match technology to business need. Unfortunately, the overhead can be staggering. Programmers are expensive resources with arcane skills that are often tangential at best to the core mission of the organization. Talented coders and development managers are hard to spot, hard to recruit, and hard to keep. The departure of even a single key programmer can sometimes render individual or even multiple applications unmaintainable.

Faced with this dilemma, savvy business technology leaders are turning to Process Director from BP Logix. Process Director is an intelligent business process management (BPM) solution that acts as a low-code/no-code platform for rapid application development. In other words: Process Director enables business users (often referred to as "citizen developers"), rather than professional programmers, to quickly create custom applications for their specific needs, without the risks and overhead of traditional in-house development.

This paper is one of a series that will examine how Process Director can help you rapidly build and deploy robust, highly customized applications in a fraction of the time, and at a fraction of the total cost of ownership (TCO), of traditional development efforts.

Systems of Record and Systems of Engagement

As organizations produce and store increasingly large volumes of data, the pressure to transform those trillions of bits and bytes into actionable business information continues to increase. As a first step, most companies invest in systems of record, such as CRM, ERP, or HR systems, to help them organize their data. They may also deploy business intelligence (BI) systems to help them understand and visualize the story behind the numbers.

But organization and visualization aren't the same as action. At any given moment there may be hundreds, thousands, even millions of processes underway in your organization, each of them depending on business data. For those processes—many of which operate in the form of custom business applications—to succeed, the information they access must be up-to-date and accurate. Such information may be generated in the course of the process itself (for example: the item(s) the customer is ordering), or may be found in the company's systems of record (for example:



the customer's outstanding balance). The data stored in the systems of record thus become a vital component of the applications that facilitate all of the organization's business activities. Such applications are often referred to as systems of engagement, because they enable employees, customers, and others to engage directly in the business at hand.

It's clear, then, that systems of engagement, such as business applications, need a way to exchange information with systems of record. And yet, if

your business applications are being built by citizen developers, it may not be easy for those individuals to access and manipulate such data. After all, systems of record, databases, and external data sources (such as web services) are complex: extracting data from them often requires detailed technical knowledge of query mechanisms, APIs, or programming languages. If our developers are required to have such skills, we may be forced to abandon our citizen developer model altogether.

Data Virtualization with Business Values

Process Director addresses these issues by separating the details of **accessing** systems of record from those of **using** the resulting data within a business application. As a result, your technical experts can configure the appropriate access mechanism once, and citizen developers can reuse the information provided anywhere at all within the application, without having to know or understand how the information was obtained.

A simple example may be useful. Imagine an application for submitting and approving employee time off requests. The user submits a form specifying how much time they would like to take. The request makes its way from the employee to their manager. If the request exceeds the employee's remaining time off allowance, it then has to receive further approval from the department head and HR. An example of such an application, implemented using Process Director's Process Timeline™ technology, is shown in Figure 1.

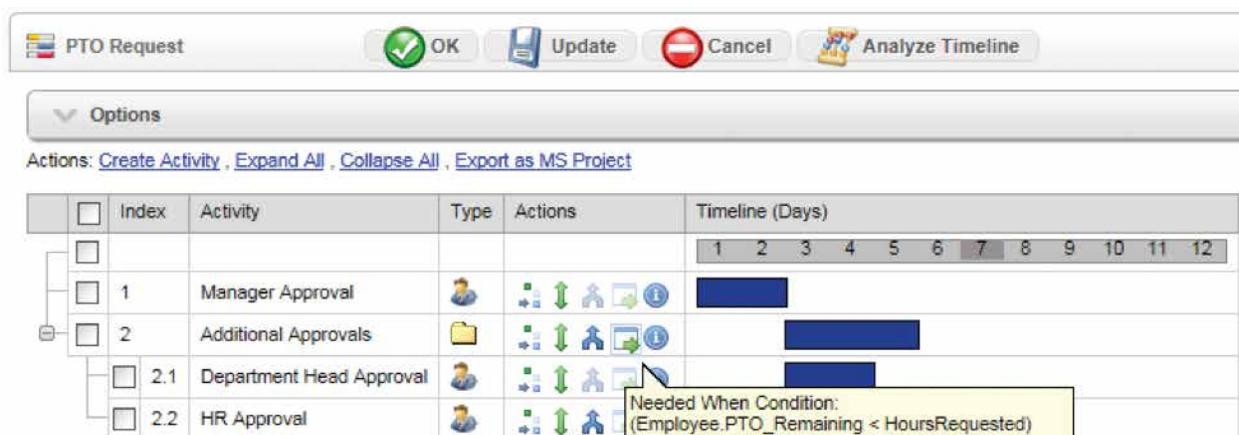


Figure 1 Conditional routing based on external data

How did the developer—let’s call her Diana—create this application? Note that this process, like most, combines instance-specific information provided by the user (how much time off is requested) with data obtained from a system of record (the amount of time remaining). The routing of the request depends on the employee’s time off allowance. Although that information may reside in the company’s HRIS (for example, PeopleSoft), we want Diana to be able to access it, even though she doesn’t know how to connect to PeopleSoft, or even that the information is stored in PeopleSoft in the first place.

In order to obtain the employee’s time off balance, Diana uses a Process Director **business value**. When constructing the rule determining whether the form should be routed for additional approvals, she merely selects the desired data element from a dropdown menu. She uses the selected business value as part of the rule used to govern when additional approvals are required, as illustrated in the “Needed When” tooltip in Figure 1. The process segment entitled “Additional Approvals” is executed only when the number of hours available for the employee is less than the number of hours requested.

Diana created a rule using information about the employee’s paid time off balance, without having to know anything about PeopleSoft, or how to retrieve the data stored within it. Instead, she used an existing resource, the business value, previously configured by an individual with the knowledge and access rights to do so. In our example, Paul, the PeopleSoft guy, created this business value. Once created, the business value was automatically available to Diana through a dropdown menu.

Paul configured the business value, in this case called “Employee,” with the ability to retrieve useful information on demand from the PeopleSoft suite. This technique—making it easy to use external data without having to know the details of how that data is obtained—is referred to as “data federation” or “data virtualization.” Using Process Director’s data virtualization capabilities, Diana was able to use the “PTO_Remaining” property of the “Employee” business value for her comparison, even though she is not familiar with PeopleSoft and may not even be aware that PeopleSoft is the source of that information in the first place.

In fact, the company could even change how that information is obtained (say, if they switch to another vendor’s HRIS). By simply reconfiguring the “Employee” business value to access the information in its new location, Diana’s application continues to operate with no changes required.

As shown in Figure 2, Paul configured some additional properties (such as “Name” and “ID_Num”) within the “Employee” business value, each of which can be accessed by a process developer like Diana as needed.

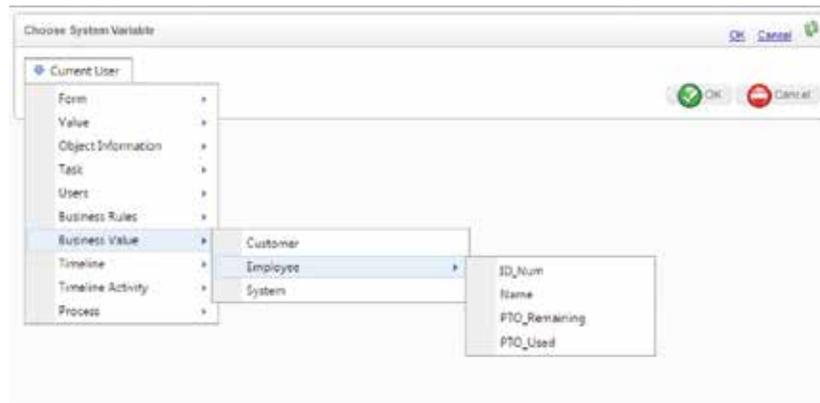


Figure 2 Selecting a Pre-Configured Business Value

Business values enable you to drive your applications with real-time data. They are updated whenever they are used; that is, each time a form or process references a specific business value, Process Director retrieves the associated external value. Of course, like all Process Director objects, business values are controlled by fine-grained permissions. Diana can only see and use those business values that she is authorized to access. Paul configured these permissions as the custodian of the company’s PeopleSoft data in such a way that Diana could use the data, but not change how it is obtained, thus ensuring that appropriate governance is in place.

Tabular Data

We’ve seen how to use Process Director business values to extract a single value (sometimes called a “scalar value”), such as an employee’s paid time off balance, from an external data source. But what about lists, sets of related data (“records”), or even lists of records (“tables”)?

Business values have you covered. They can be configured to represent scalars or lists, and by combining business value properties, can generate records and tables. For example, let’s say Diana is developing a form to help customers find branch locations in their city (Figure 3). Bob, who manages the branch database, has configured the “Branch” business value to retrieve information about branch offices from the corporate database. Once again, Diana doesn’t know anything about that database, or how to get data from it. But she knows that she can use various properties of the “Branch” business value (e.g., “Branch.Address,” “Branch.Phone”), to produce a table of branch records that’s immediately available for display on her form.

For those with some familiarity with Process Director, the image on the right side of Figure 3 illustrates the configuration Diana used in order to populate the addresses and phone numbers in the form on the left. Note that she simply selected the two relevant properties of the “Branch” business value, and that she didn’t have to be concerned with how many rows of information were returned.



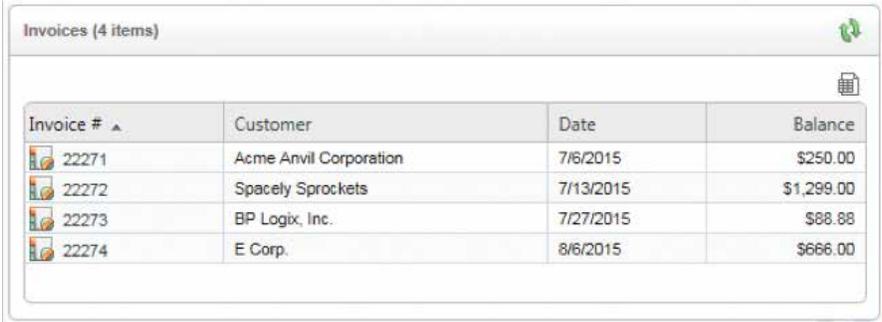
Figure 3 Sample Form and Configuration Showing Tabular Data Filled Using a Business Value

Summary Business Values

Business values are an incredibly powerful tool for retrieving and leveraging external data within your application. But they also provide a great way to gain control of the mass of information available within Process Director by enabling you to easily apply operations to lists of data. Process Director is very good at producing data sets using **knowledge views**—parameterized, tabular reports and search results created without programming through a simple user interface. In fact, in addition to eForms, knowledge views are the primary way that Process Director presents information. Examples of knowledge views include:

- Time off requests exceeding 16 hours
- Purchase orders approved by Penny
- Tasks currently assigned to members of my team

Knowledge views can be configured to return all sorts of information, such as form data, process metadata, business values, etc. But sometimes we want to boil that information down a bit. For example, let's say Katy has created a knowledge view that generates a list of customers with outstanding invoices. As shown in Figure 4, the knowledge view displays the customer name, as well as the invoice number, amount, and date.



Invoice #	Customer	Date	Balance
22271	Acme Anvil Corporation	7/6/2015	\$250.00
22272	Spacely Sprockets	7/13/2015	\$1,299.00
22273	BP Logix, Inc.	7/27/2015	\$88.88
22274	E Corp.	8/6/2015	\$666.00

Figure 4 Knowledge View Displaying Open Invoices

Diana, in the meantime, is developing a process that examines various KPIs and issues notifications if they are out of range. One of these KPIs is the average of all outstanding invoices: if this amount is above a certain threshold, her process will notify senior management so that further action can be taken. Fortunately, Katy also created a business value that yields the average of all of the invoice amounts produced by her knowledge view. Diana can now use that business value to govern whether or not her process issues a notification.

This example demonstrates a key strength of business values: the ability to extract derivative values from data sets. Business values can produce counts, sums, averages, and other results of operations on data, whether produced from Process Director knowledge views or stored in external data sources. These results are determined in real time, just like all business values. Summary business values can be very useful in a wide variety of scenarios, such as:

- Directing call center processes whose behavior is influenced by the average call duration over the past 10 minutes
- Determining red/amber/green status based on the number of processes currently predicted to complete after their due date
- Launching remedial actions when an overall risk assessment crosses a certain threshold

Conclusion

As enterprises evolve and mature, they are shifting away from high risk custom development projects as well as restrictive packaged applications, instead taking advantage of low-code/no-code BPM solutions like Process Director. Process Director empowers organizations to gain the benefits of highly customized business applications without the risks or restrictions of traditional application development or procurement.



Business applications depend on real-time business data. Process Director business values are an easy and elegant way to access information from anywhere in the organization—or beyond—and to use that information to drive custom applications. Business values use data virtualization techniques to separate the skills required to access systems of record and other data sources from those used to configure applications. As a result, once a business value is configured, it can be used in one or many applications, subject to the appropriate access controls.

If you'd like more information about business values or Process Director in general, please visit our web site at www.bplogix.com, or contact us via email at sales-info@bplogix.com, or contact the author of this white paper at scott.menter@bplogix.com. We look forward to speaking with you.



Process Driven, Business Focused
www.bplogix.com

Copyright © 2015 BP Logix, Inc. Process Director and Process Timeline are trademarks of BP Logix, Inc.

