Business Process Cockpit

(EXTENDED ABSTRACT)

Mehmet Sayal, Fabio Casati, Umeshwar Dayal, Ming-Chien Shan

Hewlett-Packard
1501 Page Mill Road
Palo Alto, CA, 94304
USA
[sayal,casati,dayal,shan]@hpl.hp.com

1. Overview

Business Process Management Systems (BPMSs) are software platforms that support the definition, execution, and tracking of business processes. They are often used to support administrative and production processes as well as to implement complex web services, delivered by composing existing ones according to some process logic. BPMSs have the ability of logging information about the business processes they support, including for instance the start and completion time of each activity, its input and output data, the resource that executed it, as well as events (messages) sent or received. This information is a gold mine for business and IT analysts: in fact, its analysis may reveal problems and inefficiencies in process executions and identify solutions in order to improve process execution *quality*, both as perceived by the users in terms of better and faster processes, and as perceived by service providers in terms of lower operating cost. In addition, information on active processes can be used for notifying users and processes of quality degradations.

This paper describes the *Business Process Cockpit* (BPC), a tool that supports real-time monitoring, analysis, and management of business processes running on top of HP Process Manager, the BPMS developed by Hewlett-Packard [HPPM]. The BPC is a component of the Business Process Intelligence (BPI) tool suite, a set of tools that support off-line and real-time business and IT decisions to improve business process quality. The BPI suite operates by applying data warehousing and data mining techniques to business process execution data, as shown in Figure 1. An Extract, Transfer, and Load (ETL) application collects data from the log and loads them into a Process Data Warehouse (PWD). Besides performing traditional warehousing functions such as data cleaning, ETL also labels process executions with quality information, based on user-defined criteria of quality. Details on the BPI, ETL, PDW, and process quality labeling are provided in [BCDS02]. Data in the PDW can be analyzed using BPC, directly accessed with a commercial reporting tool (such as Crystal Reports or Oracle Discoverer), or input to BPI simulation component to derive simulation parameters from real data [JCS02].

Proceedings of the 28th VLDB Conference,
Hong Kong, China, 2002

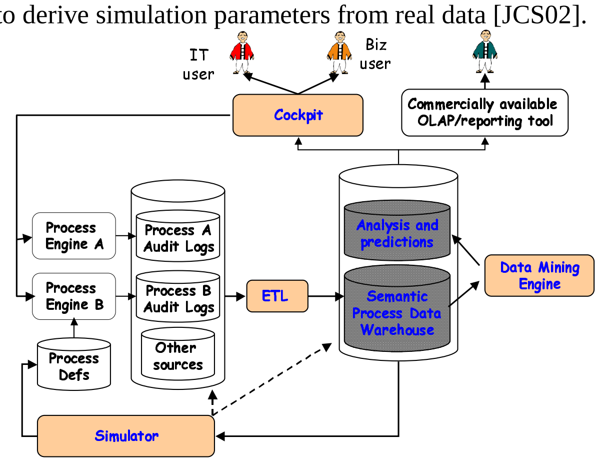

Figure 1 - Architecture of the Business Process Intelligence tool suite

The BPC was developed at Hewlett-Packard in response to the need of customers, in relation to several projects in the supply chain automation and collaborative exchange domains. Supply chain automation and collaborative exchanges address two different sides of the same problem: how to make the supply chain more efficient and cost-effective, both in terms of automation of internal processes and interaction with business partners. Supply chain automation concentrates on perspective of one company, while collaborative exchanges aim at developing market exchanges where companies interact to define and execute business agreements. Processes in these domains have some distinguished *characteristics*, such as having high volume, involving multiple companies in the execution, and providing significant business value and impact as they are immediately related to sales, purchases, profits and revenues.

These characteristics pose demanding *requirements* to process analysis and management tools. For example, it is essential that users are able to quickly and easily analyze *business-level indicators*. Moreover, unlike most practical cases, where the BPMS administrator is the main or only person that analyzes and monitors processes, here the analysts are business users, who have no knowledge of the process details, of the specifics of the BPMS, and of how reports can be extracted based on business process execution data. We experienced that there exist many different kinds of analyses that business users may want to conduct, and it is difficult to satisfy users by canning some queries and pre-package few reports. Finally, a system that is used for analyzing processes in supply chain automation and collaborative exchange domains should exhibit semantic reliability, i.e., it should be able to react to predicted or actual semantic problems, such as delays in procuring or shipping the goods or, more generally, violations of service level agreements.

The characteristics of the processes and the consequent requirements translate into concrete issues and challenges that we had to face in the design and development of the BPC:

- Identify a way to enable business users to easily (i.e., without writing any code) define and extract business-level metrics from execution data, hiding the details of the process. Provide semantic analysis of the data by enabling users to define new concepts and view the data using those concepts.
- Identify how to let business users create a large variety of business reports on process execution data and determine the best way to present such data.
- Design a monitoring and management console that helps in achieving semantic reliability in two ways: it notifies users of alerts, quality degradations, and violations of service level agreements, so that users can take proper actions; and it automates the reaction to the problem by providing feedback to the BPMS.

In the following sections we describe the concepts, features and architecture of the BPC, thereby describing how we addressed these challenges. We also observe that similar concepts and architectures can be reused and extended to manage, in a uniform fashion, web services, enterprise applications, and in general any e-business system. We initially focused on business process because of its value, both in general and for Hewlett-Packard in particular, and because we had access to customers' requirements and customers' data, a crucial help in design and development of such a system.

2. Concepts and Features

2.1 Defining business metrics

In order to understand what kinds of process metrics users were interested in, we have conducted investigations with several potential users of the BPC in the supply chain contexts. In all cases, users expressed interest in *qualitative* metrics. In particular, the main requirement involved the possibility of classifying process instances into user-defined taxonomies. In order to meet these requirements, BPC includes the notion of *taxonomy*.

Taxonomy is a user-defined way to classify instances of a process depending on its characteristics. Multiple taxonomies can be defined for the same process and each taxonomy can have several *categories*. A process instance can belong to only one category within a taxonomy at any time. For example, the taxonomy *negotiation outcome* may include the categories *success* and *failure* (depending on whether an agreement is reached in a negotiation). Another taxonomy, say *duration*, may provide a qualitative indication of the process execution time, with four categories: *fast* (processes that last less than 5 days), *acceptable* (between 5 and 10 days), *slow* (between 10 and 15 days), and *very slow* (more than 15 days).

BPC taxonomies are defined as a list of categories, and semantic description of every category. This can be done by instantiating a *behavior template* for each category. A *behavior template* identifies a generic process behavior, such as *"Processes **P** in which node **N** has been executed."* A template is instantiated by specifying the parametric parts (P and N in this example). For example, the category *success* of taxonomy *negotiation outcome* can be defined by associating it to the template with parameters P="Supply chain negotiation" and N="notify acceptance". Templates are implemented by means of SQL statements, which analyze data in the PDW.

2.2 BPC process analysis: focus points, perspectives and restrictions

The design of the BPC analysis and reporting features proved to be a challenging task, requiring several iterations with business users. The main challenge was enabling users to define and visualize a wide variety of business process reports in an extremely simple way. We summarize the outcome of this design process.

The BPC visualizes process execution data and related quality metrics according to different *focus points.* A *focus point* identifies the process entity that is the focus of the analysis. For example, under the *service* focus, users can see information about the a particular service. We have identified the following focus points as relevant in business-level analysis:

- *Processes*: displays information about a specific process or set of processes.
- *Resources*: shows data related to an individual resource or group of resources. A resource is a human or software component that participates in the execution of process activities.
- *Services*: displays data related to the services invoked within the execution of processes in order to carry out individual activities within process definitions.

For each focus point, the BPC presents basic statistical information (such as average execution time and performance), value-related information (associated revenues and costs), and quality (taxonomy) information such as resource performance and rankings. Within a focus point, the information can be analyzed according to different *perspectives*. A *perspective* defines a way to look at the information of the selected focus point. For example, looking at the *process* focus point under the *time* perspective, users can view process statistics by year, week, day of the week, etc. BPC provides the following perspectives:

– *Resources*: shows data depending on the resource that took part in the execution. For example, by looking at the *service* focus point under the *resource* perspective, users can see service execution data aggregated depending on which resource (i.e., service provider) executed the service.
– *Services*: shows data aggregated depending on the service that has been executed.
– *Time*: shows data by time. This perspective has several sub-perspectives, allowing users to view data aggregated by calendar or fiscal year, quarter, or month, by week, by day of the week, hour of the day.
– *Processes*: displays information aggregated by process.
– *Taxonomies*: shows data aggregated by the categories of a given taxonomy.

As an example, Figure 2 shows the distribution of the number of instances of an "Electronic order processing" process (called EOPG in the figure) among the categories of a *duration* taxonomy. The figure shows that many instances of the selected process are very slow. In order to find out the reason for very slow execution of the process instances, the user can view the data about the instances of that process from different perspectives.

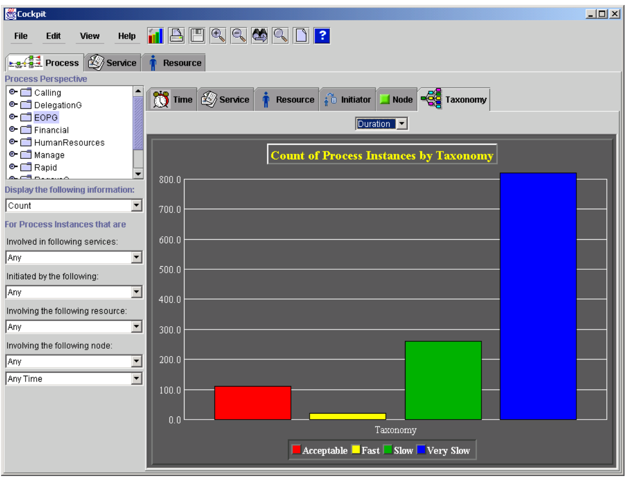

Figure 2 - Screen shot from Business Process Cockpit: *Process* focus and *Taxonomy* perspective

Figure 3 shows the percentage of slow process instances, which are the instances whose execution took longer time than a user-defined limit, aggregated by the resources that have been involved in process execution. The user can view this report to identify the resources that cause slow process executions. The figure shows that 55% of process instances are slow when a particular resource (the third one from left) is involved in the execution of the process.

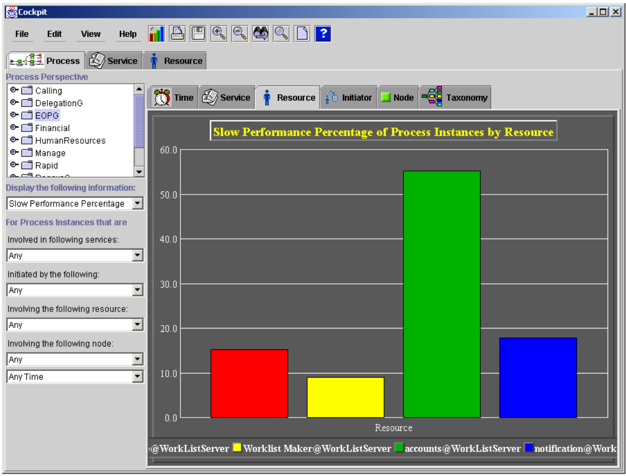

Figure 3 - Screen shot from Business Process Cockpit: *Process* focus and *Resource* perspective

After finding out the resource causing slow execution of the selected process, the user may switch to *resource* focus in order to analyze the data further for the particular resource that is found to cause slow process execution. While inside resource focus, the user can look at the data from different perspectives in order to find out if that resource works slow only under certain circumstances (e.g., certain days of the week) or if she is always slow.

Besides determining aggregation, the process characteristics described in the *perspectives* above can be used to *restrict* (filter) the data. For instance, the BPC can show data related to processes started within a given time window, or classified in certain categories, or involved the execution of a certain service. The restriction attributes are the same as those of perspectives. We experienced that this symmetry proved to be helpful in improving the usability of the tool by business users.

2.3 Process monitoring and management

BPC monitors active process executions, particularly to detect the occurrence of critical situations. For this purpose, BPC allows analysts to define *alerts*, i.e., situations that need to be detected. Alerts are defined in a way analogous to that of categories, i.e., by instantiating templates. An example of an alert template is *"Processes **P** that stopped at node **N** for more than **M** minutes."* The BPC periodically monitors alerts on running processes by reading data in the HPPM logs. Users can get the most benefit from alerts by defining *policies*. A *policy* is a rule that defines which set of actions should be automatically executed by the BPC as soon as an alert is detected. There are three kinds of actions. First, BPC can be instructed to notify a given user, either via email or cellular phone short message. Second, BPC can send a message to a JMS bus, to be consumed by any java client interested in receiving the event. Third, and most interesting, BPC can provide feedback to HPPM, in order to modify system parameters or change the course of the process. After having experimented different options, we identified two ways of providing automatic feedback to HPPM: raise the process *priority* in order to speed up its execution, or send an event message to the process instance for which the alert has been detected. Events sent to HPPM process instances can cause alternative execution paths to be followed in order to resolve the detected problem. In summary, BPC monitoring component notifies users or software components so that critical situations, such as performance degradations, can be detected and handled properly.

3. Architecture and Implementation

Figure 4 shows the "external" architecture of the Cockpit, that is, how it interacts with HPPM and other management components. The cockpit reads "live" data about active process executions from the HPPM logs and it reads data about completed process executions from the PDW. Both databases are accessed through JDBC.

In terms of output, the BPC communicates (in addition to the users) with HPPM to optimize executions, as discussed in the previous section. BPC also outputs configuration data to warehouse (to define semantic mappings between execution data and quality metrics), and notifies alerts to a JMS bus, used to connect the cockpit with other entrprise management platforms such as HP OpenView.

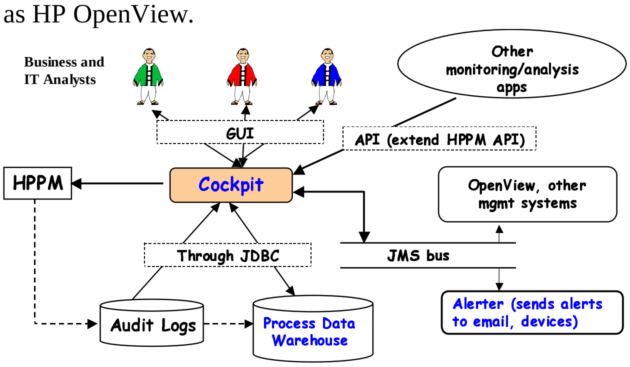

Figure 4 - External architecture of Business Process Cockpit

The internal architecture of BPC is shown in Figure 5. The GUI manages the interaction with the users. The Query Builder generates SQL queries that can be submitted to a Database Management System (DBMS). DBMS consists of the HPPM Log Database and PDW. The Query Runner submits queries to the DBMS and retrieves the results. The Chart Builder is used for managing Java based chart building and viewing tools that are used for displaying the data that is requested by the user. The Chart Builder can be integrated with more than one chart building and display tools. The Feedback Provider provides feedback to HPPM processes, based on the user-defined reaction policies. The API is in charge of interfacing with the external software components. The Data Encryptor/Decryptor provides secure data exchange with the databases, HPPM, and other external components. The Data Converter translates data among various formats that are used by BPC components.

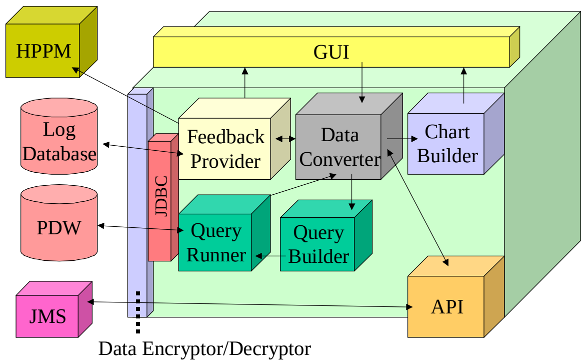

Figure 5 - Internal architecture of Business Process Cockpit

4. Main Contributions

We have designed and developed a tool that provides semantic analysis and monitoring of business process executions by business managers and IT professionals. We have collected user requirements, which may apply to many other domains that involve transactional data analysis. In general, the users would like the analysis tools to provide the basic concepts and statistics, and allow the users to define new ones. The users also like to be notified about the problems, and be provided with automatic handling option for certain problems. In this paper, we describe the design and implementation of BPC, which concentrates on the business process specific data monitoring and analysis, but it can be used for any other domain that involves similar requirements.

5. References

[BCSD02] Angela Bonifati, Fabio Casati, Umesh Dayal, Ming-Chien Shan. Warehousing Workflow Data: Challenges and Opportunities. *Procs of VLDB'01*, Rome, Italy, Sept 2001.

[HPPM] Hewlett-Packard. HP Process Manager Process Design Guide. 2001.

[JCS02] LiJie Jin, Fabio Casati, Ming-Chien Shan. Business Process Simulation with HP Process Manager. *Procs. of the Collaborative Technologies Symposium*, San Antonio, Texas, January 2002 (to appear).