

Real-Time Business Process Performance Management

Yushun Fan¹, Xinxin Bai²

Dept. of Automation, Tsinghua Univ., Beijing 100084, China

¹fan@cims.tsinghua.edu.cn, ²bx02@mails.tsinghua.edu.cn

Abstract

Execution power is competition power. Facing dynamic market, real-time business process performance management (RTBPPM) is much important to enterprise success. This paper provides a RTBPPM framework and discusses relevant design issues. This framework allows for easy redesign of performance control formula and proactive response to unpredictable business situation. It is an effective method to meet the agility and intelligence requirements of contemporary enterprises.

1. Introduction

Execution power is competition power. Dynamic demands from market and customer make the success of enterprises rely more and more on executing core business process correctly and continuously, as necessitates a fundamental transformation of enterprises from conventional passive ones to intelligent and proactive ones. Therefore, implementing real-time business process performance management (RTBPPM), aiming at timely and intelligently monitoring business process execution, preventing business levels from sliding down and continuously improving process quality, is meaningful not only to enterprise but to its partners. Concepts and technologies like BPA, BAM and BI have illustrated the trend of people's focus on monitoring operational business processes. In this field, postmortem analysis method prevails. ARIS PPM^[1], BPI suite^[2] and Process Mining^[3] all fall into corresponding

representatives. However, run-time measurement and control of business process has not been taken into consideration. PDS based approach^[4] is an original one for real-time process monitoring, yet the including of metric calculation logic (ETLets) in ETL container cannot adapt the changes of workflow metrics. Till now, as far as we know, none of the existing WFMS or BPMS really addresses the issues of intelligent real-time process performance management. In this case, a performance-driven management framework to business processes is presented in this paper. Corresponding performance measurement framework, system architecture, performance control logic and calculation are discussed as well.

2. Business Process Performance Measurement Framework

“You get what you measure.”^[5] Around business process, many frameworks, including the famous Balanced Scorecard and EFQM's Excellence Model, have proposed measure metrics that indicate the business performance. In China, we proposed a framework^[6] (Fig.1) for business process performance measurement, which is specified for Chinese enterprises. .

It measures business process from two levels: strategy level and business level. Through analyzing strategy, customer requirement and profit at the top level, specific metrics on business level is set. From both static and dynamic point of view, organization structure, resource allocation and business logic is evaluated. We adopt this framework as a guidance to construct performance

metrics and corresponding control rules. Performance control objects include processes and activities, and the measures are mainly dynamic ones that is a subset of this

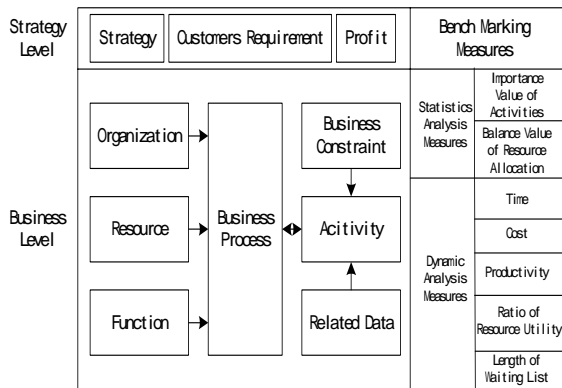


Fig.1 Business process performance measurement framework

framework, i.e. time, status, workload and cost.

3. Framework Supporting RTBPPM

“Real-time” character of RTBPPM embodies the meaning of timely and correct response to performance metric violation and exceptions. “Timely” indicates as little human interference as possible and also the ability to feed back performance information and control process route before performance decline. “Correct” calls for conformance of subsequent actions to enterprise strategy and business execution laws. To meet those two requirements, we draw a four-layer framework supporting RTBPPM as illustrated in Fig.2.

- Business Process Planning Layer (BPPL): The macro-control layer, in charge of planning of business process and setting of performance goals. Hereinto, business knowledge comes from operational experiences of stuff and business execution laws. Planning goals and business knowledge direct business modeling.
- Process Performance Management Layer (PPML): Tuner for optimizing process execution route. ECA form, parsed and executed by rule engine, expresses performance control goals and methods.
- Process Performance Watching Layer (PPWL): Map

process execution data to performance data, and act as the data source for PPML. Bearing multi-layer and multi-dimensional analysis ability, Process warehouse (PW) takes on the aggregation tool for performance data. Process history warehouse (PHW) fights for strategy evaluation and real-time process warehouse (RTPW) exhibits current information.

- Process Execution Layer (PEL): Execute business process and adjust its running at proper point of time. Workflow engines produce and respond to process execution event, which involves original business event and performance deviation event as well.

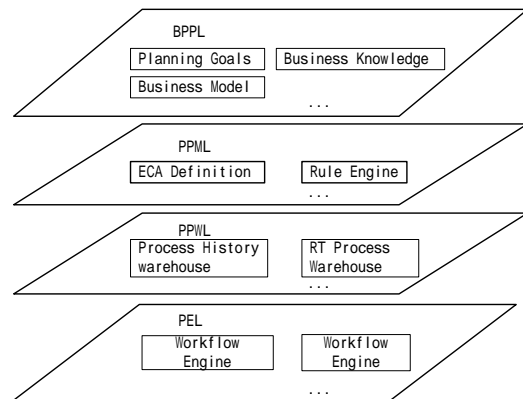


Fig 2

Framework supporting RTBPPM

Four layers shape a continuous improving loop in implementation by introducing process execution analysis. This loop is different from traditional workflow lifecycle loop (design-configuration-running-diagnose) in that it provides process with the ability of dynamic and intelligent adjustment. In a RTBPPM system, business is not only driven by business event, but at the same time by ECA rules relevant to performance metrics.

4. System Architecture

Based on framework detailed in last section, system architecture is depicted in Fig.3. It has four layers. The top layer is *Application Layer* that enables system to interact with outside (including enterprise users and external systems); implementation of business process planning in Fig.2 is accomplished through tools provide

by this layer and process performance management tool provide the capability of obtaining performance info and setting of ECA rules from the business point of view. *ECA Execution Layer* corresponds to PPML in Fig.2. *Execution Monitoring Layer* converges PPWL and process execution logic of PEL; it can be viewed as extensions to traditional process monitoring mechanism. *Data Layer*, as the bottom layer, covers all process execution data separated from process execution logic.

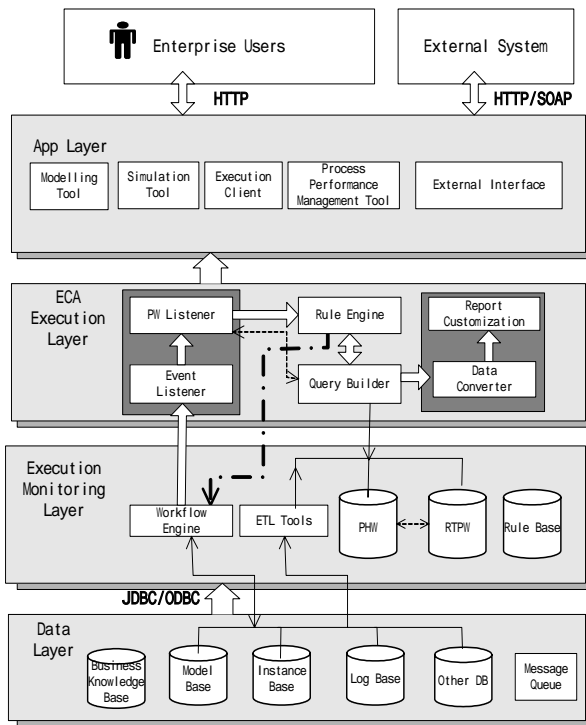


Fig.3 System architecture of RTBPPM

Here, we only focus on ECA execution mechanism realized in *ECA Execution Layer*. Given an ECA rule, listener module analyzes its event part and generates an event listener. Query Builder generates performance metric queries that can be submitted to an OLAP server. Joint query of PHW and RTPW may be needed. When an event occurs, the corresponding event listener notifies rule engine by pushing it into the event queue. Then after data processing in PW, PW listener submits performance metric queries to the OLAP server and retrieves the results, which can be evaluated by the rule engine. Action is determined through comparison of query outcome and

corresponding condition part in ECA rule. For message actions, rule engine sends defined messages to workflow engines; while for report ones, rule engine delivers report to relevant stuff through report service.

5. Process Performance Control Rules

As entity describing process performance control goals and methods, ECA rule affects process execution route and keeps up process performance level; it is the kernel for the whole RTBPPM. In ECA design, we take into account of conforming to our performance measurement framework and the extensibility of process running laws.

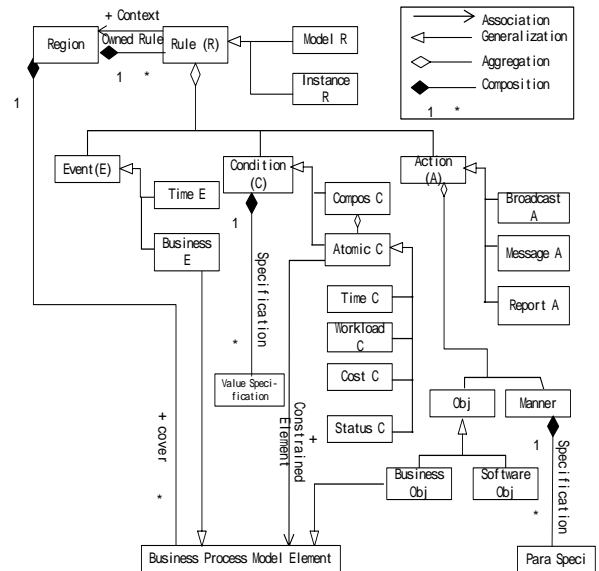


Fig.4 ECA meta-model in UML

Fig.4 shows the meta-model of ECA rules in UML.

ECA is a mechanism of sense-and-respond. One ECA rule consists of three parts: *Event (E)*, *Condition (C)*, and *Action (A)*. In the RTBPPM context, it represents when, what, how and who contents of process performance control logic. *Region* defines the range of process model elements acted on by an ECA rule, involving event range and action object range. *Event* may be a *Business Event* directly from process model, or a *Time Event*, such as time arrival. Event setting influences performance

control effect in respect that many trifling events crowded in event queue would delay real-time performance calculating, so it can only be concentrated on key business events in key processes. *Condition* describes the numerical constrains of the attribute of model element instance, corresponding to constraint of performance metrics. *Atomic Condition* can form *Composite Condition* through composite operation including AND, OR and etc. *Action* represents how and to whom of an action, it may send out a message to other components, or distributes specific report to relevant staffs. ECA rules are separated into two classes: *Model Rule* and *Instance Rule*. The former is most directly distracted from model definition, mainly including timing report distribution and time constraints of workflow; while the latter, being emergency measures for incoming exception, is from part of business knowledge.

6. Data Modeling

This section discusses hyper cube design of process warehouse. Considering speed constraint in performance calculation, RTPW stores only latest (i.e. the current day) process running data, and is supplemented during process execution. Log tracking strategy is used in data extraction of RTPW. Data granularity of RTPW is smaller than that of PHW; data of completed process instance in RTPW is transported after aggregation into PHW in business idle time, and then cleared from RTPW.

RTPW design adopts the data-driven modeling method^[7], which refer to workflow meta-mode. Hyper cube of RTPW involves a few dimensions and many measures; measures fall into measurement framework and are closely related to time, so this kind of model focuses on short-term behavior. As for the design of PHW, in order to meet deep analysis requirements, PHW must have a lot of dimensions as well as a lot of measures, thus additional data source is indispensable.

7. Conclusion and Future Work

To meet the agility and intelligence requirements of contemporary enterprises , we propose a framework supporting RTBPPM built upon current distributed WFMS. The separation of performance control (ECA rule) from performance calculation (PW) made convenient the updating of performance control scheme, thus the whole system becomes strongly adaptable. Moreover, ECA rules and RTPW are both in line with our process performance measurement framework; this top-down design pattern ensures process performance obeying enterprise strategy.

Further research includes rule conflict inspecting and combining mode of RTBPPM with other management views, such as organization performance management.

Acknowledgement:

This work is funded by Grant 2003AA4Z2020 from the National High-tech. R&D Program for CIMS of China and by Grant 60274046 from the National Natural Science Foundation of China.

References:

- [1] IDS SCHEER Whitepaper, "ARIS Process Performance Manager", <http://www.ids-scheer.com/sixcms/media.php/1186/>, 2002.
- [2] Fabio Casati, Umesh Dayal, Mehmet Sayal, and Ming-Chien Shan, "Business Process Intelligence", <http://www.hpl.hp.com/techreports/>, 2002.
- [3] W.M.P. van der Aalst, A.J.M.M. Weijters, "Process Mining: A Research Agenda", *Computers in Industry*, 2004, 53: p.231-244.
- [4] Josef Schiefer, Beate List, Robert M. Bruckner, "Process Data Store: A Real-Time Data Store for Monitoring Business Processes", *DEXA 2003*, Springer-Verlag Berlin Heidelberg 2003: p.760-770.
- [5] Neely, A. D., *Business Performance Measurement : Theory and Practice.*, NY : Cambridge University Press, 2002.
- [6] Huiping Lin, Yushun Fan, Cheng Wu. "Research on Enterprise Business Process Performance Evaluation System", *CIMS*, 2001, 7(12): p.47-32. (in Chinese)
- [7] Beate List, Robert M. Bruckner, Karl Machaczek, et al., "A Comparison of Data Warehouse Development Methodologies

Case Study of the Process Warehouse”, *DEXA 2002*,
Springer-Verlag Heidelberg, 2002, 2453: p.203-215.