ProcessGene Query – a Tool for Querying the Content Layer of Business Process Models

Avi Wasser[1], Maya Lincoln[1] Reuven Karni[1]

[1] ProcessGene Ltd.
15303 Ventura Boulevard, Sherman Oaks, California, 91403, USA
{avi.wasser, maya.lincoln, Reuven.karni}@processgene.com

Abstract. One of the main challenges currently facing the world of enterprise information technology in general and ERP/SCM/CRM systems in particular, is visibility into the business of organizations. While the phenomena of devising supporting tools for process execution frameworks is widespread in academia and practice, there have been few attempts to develop methodologies and software tools that support structured analysis of the business process content layer. The incorporation of content into a business process model produces complexity in the sense that it adds semantics and relationships of actual business data. To confront this complexity, this research suggests a framework and a supporting software tool "ProcessGene Query" for conducting search-queries on business process models.

1 Introduction

One of the main challenges currently facing the world of enterprise information technology, and ERP systems in particular, is visibility into the business of organizations, [10]. The prevalent approach utilizes conceptual business process modeling as the foundation for creating and managing this visibility, aiming to connect the business activity and its supporting information technology (IT) systems [6].

The current main thrust of business process modeling research has focused on the study of structural frameworks and execution patterns [9], putting little emphasis on the content layer that is supposed to populate these frameworks. "Real life" business process models, which contain practical content objects, have been disregarded [9], except in illustrative examples.

Structural process frameworks define formal architectures and standards for representing business activities and processes. The spectrum (Fig. 1) ranges from simple descriptive frameworks such as activity diagrams, suitable mostly for business users, through more formal frameworks such as OPM [11] and Petri-nets, suitable mostly for software implementers and IT system analysts, to code-compatible structures such as BPEL and XLANG [12], suitable for software developers.

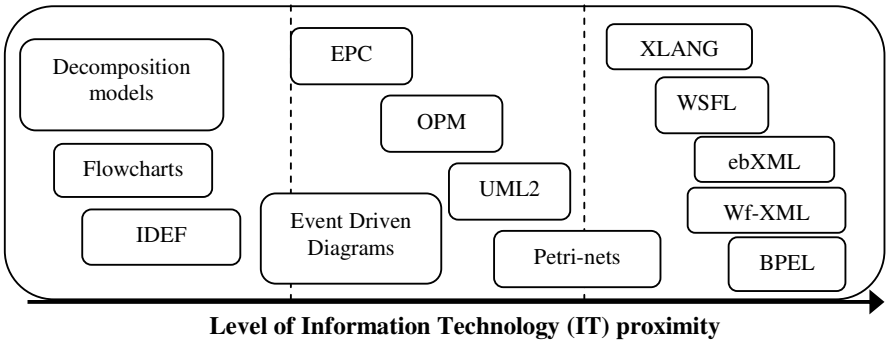

Fig. 1: The structural frameworks spectrum

The practical deployment of these frameworks, involves an attempt to enumerate *actual* business processes carried out within enterprises. Modeling in this context focuses on the *content* layer of business process models. We define the content layer as the itemization of the suite of actual business processes constituting the framework of business-related activity within a particular industrial sector, or, alternatively, within a particular enterprise. Only a few scientific publications address the topic of business process content [7]-[9]. On the other hand the initiative has been taken and business process content was developed and applied, by enterprise software vendors, IT integrators, and BPM commercial firms.

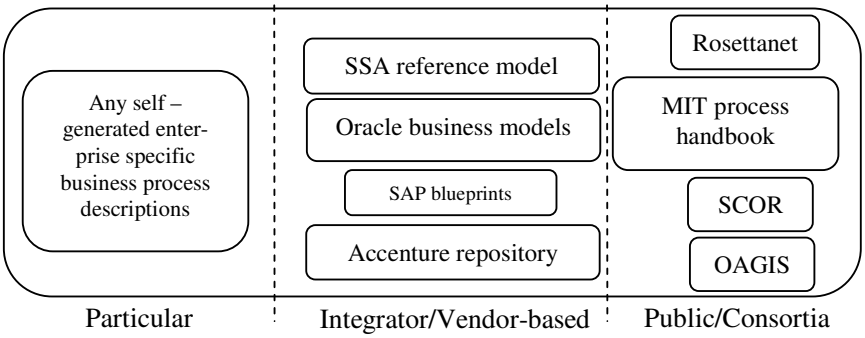

Fig. 2: Business process content examples

Fig. 2 presents some business process content compendia, divided into three main types: (a) particular, enterprise specific content; (b) vendor/integrator content such as the OBM (Oracle Business Models) library [4] and SAP solution maps [5] and (c) collaborative/consortia content frameworks such as the MIT process handbook [1], OAGIS [13] and Rosettanet [14]. Thus, while the phenomena of formulating structural execution frameworks is widespread in academia (e.g. [15]), there seem to be few attempts to develop theories, empirical studies and supporting tools [9] (such

as generation, customization, validation and search mechanisms) for "complete" business process models which incorporate an actual content layer (Fig. 3).

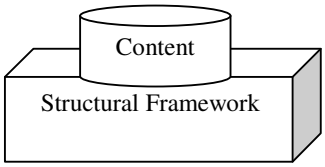

Fig. 3: A process model as a combination of structure and content layers

When this research addresses *business process models* it refers to "complete" models that also include a *content* layer, so that the combination of structure and content can display the actual suite of business processes constituting the framework of activity within the enterprise and enable subsequent implementation through IT. For example: a flowchart describing bottleneck leveling in production, or a Petri-net describing the process of managing a service request in CRM. Such business process models are considered complex since they include a large number of interconnected data objects (processes, roles, events, related data, etc.). This complexity increases when the models are to be expressed and actualized by a corresponding IT system (e.g. ERP/SCM/CRM), which requires verification and validation of the business process models from a functional and managerial point of view prior to actual implementation and subsequent execution. To confront this complexity, and in order to enable effective handling of the business process models content, this research suggests a framework and a supporting software tool for conducting search-queries on business process models.

The paper features the following sections: a demonstration of a standardized format for describing the content of a business processes based on current offerings of ERP vendors – (section 2); the "ProcessGene Query" methodology and tool for searching the content of process models (section 3); an example for running content search queries (section 4); conclusions and suggestions for further work (section 5).

2 Describing the Content of Business Process Models

Due to their dominance in industry, we will focus on content layers from vendor/integrator commercial business process models. These include, for example, SAP's industry and cross-industry Business Solution Maps [5], Lawson-Intentia's ERM (Enterprise Reference Models) [2] and Oracle's OBM (Oracle Business Models) library [4]. In the Oracle business process flows, for example (Table 1), the top level "high level flow" for an industrial sector presents names and descriptions of the high level functionalities for that industry (about 7), and their corresponding business flows (about 7). Business flows are then broken into activities and tasks, holding similar amount of items at each level.

Table 1: Oracle E-Business suite process content hierarchy

(1) High Level Flow = "Procure to Pay" (top hierarchal level)
(2) Business Flow = "Analyze to Agreement" (second level)
(3) Activity/Procedure = "Negotiate and Select Suppliers" (third level)
(4) Task = "Enter supplier information" (fourth level)- with a link to corresponding IT components such as setups and customizations of datasets

From these categorizations vendors and integrators develop a suite of processes, reflecting what an enterprise does, or needs to do, in order to achieve its objectives [3]. Furthermore- the content includes pointers to additional content items that are in use during an implementation process such as user requirements, test scripts, setup parameters, flow diagrams, workflows and related documents. If we assume an amount of seven items at each hierarchal content level we would reach almost 20,000 interconnected data items, not counting the additional process-related content items. It is also important to realize that each item holds a certain amount of metadata, which users may need to retrieve and review. Research into a vendor/integrator defined commercial business process models has introduced several concepts: (a) the necessity for a compendium of realistic business processes in order to be able to generate practical enterprise models; (b) the inclusion of cross-references between business processes, additional content items and IT components offered by software vendors; and (c) the complexity in a concurrent management of a relatively large dataset. To confront complexity, this research suggests a query methodology and supporting tool for assisting in the retrieval and management of the business process content layer.

3 Methodological Framework

In order to formulate and demonstrate the proposed query framework, we present two data models that organize the business process data and form the foundation for running search-queries on the content layer. Then we elaborate on the query method.

3.1 Process Data Structure Models

Process Descriptor Decomposition Model. This model introduces the basic ideas and notations for formally representing business process model content objects by a hierarchal graph of descriptors, as shown in Fig. 4. The process model contains n levels of process hierarchy (L1, L2, … , Ln). At each level, each process is represented by a single process descriptor, and each process descriptor consists of one action, one object that the action acts upon, and possibly one or more action qualifiers, object qualifiers and means.

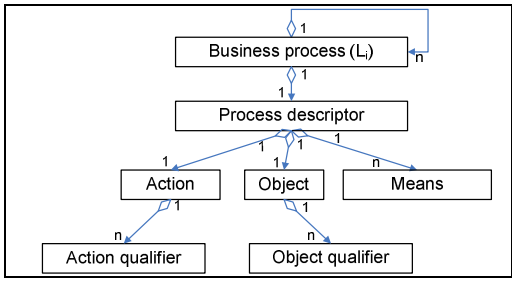

Fig. 4: The process descriptor decomposition model

For example, a process descriptor can be defined as: "Issue confirmed purchase order to local supplier by e-mail", comprising an object, an action and their qualifiers.

Business Action and Object Taxonomy Model. This model organizes a set of process descriptors, attempting to determine the relationships between business actions and objects both longitudinally (hierarchically) and latitudinally (in terms of execution order) as described in Fig 5. In this model an action is related to an object by an operability connector, e.g. the action "receive" is related to the object "invoice". Longitudinally- the action "issue" is considered a subclass (a more specific form) of "produce", and the object "purchase order" is a subclass of "purchasing document" (note that the operability connectivity applies also to relations between different hierarchy levels). Latitudinally, each object holds a list of ordered actions applied on that object (e.g. the object "product" is related to the actions "plan" followed by "produce"); (b) a list of ordered objects that express the object lifecycle (e.g. the following lifecycle sequence: "raw material"→, (…)→, "product"→, (…)→, "returns"→, (…)).

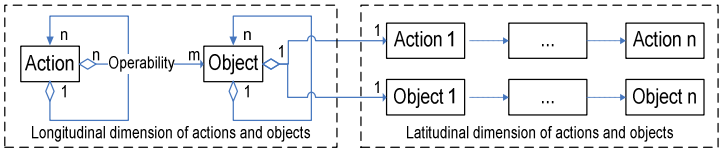

Fig 5: Business action and object taxonomy model

These longitudinal and latitudinal viewpoints contribute another dimension for analyzing and learning the business process model content layer in terms of identifying action and object hierarchies and execution sequences.

3.2 The ProcessGene Content Query Method

The method aims to provide a simple yet powerful query interface in which users are able to express and perform a large set of queries using intuitive definitions.

The ProcessGene Query mechanism includes four main components. At the front-end: a Scoping-Assistant (SA), for defining the content query range; and a Query Specification Interface (QSI), for expressing the user's data extraction requirements. At the back-end: a Query Interpreter (QI) for interpreting the user specification into a set of normalized queries; and a Query Results Packager (QRP) for packaging the retrieved results to include only data that is of interest to the user. The SA uses business processes as means for query focusing, since at any hierarchy level, these objects are related with all other data components. After defining the query's underlying data scope, the QSI enables users to specify data requirements. This module is based on two specification layers, offering at the first layer a simple interface, which enfolds more advanced options for users that wish to drill-down and expand the query capability. The first query specification layer presents all business process model component types as a flat checklist, enabling the user to select query components. Each component can then be expanded, presenting additional data fields and enabling the user to specify different criteria for each field. Conditions are expressed using regular expressions (strings, keywords, wildcards), or by selecting one or more values from a list of values, depending on the data field type. In addition, the QSI also assists in defining the query result structure and content. Instead of generating pre-defined result segment structures, the user can define which data components are to be included in the result set. After the specification phase, the QI analyzes the user request and composes a set of all compatible normalized queries. The QRP then modifies the retrieved results to include only data fields that were required by the user. These manipulated results are eventually presented to the user according to the business process model hierarchy.

4 Example: Process Content Query

To illustrate the proposed framework for supporting a search query on business process content we present an example, in which a user is interested to find out "how order-based decisions are handled by sales representatives". Using the SA, the user selects a level 1 process, "Order to Cash", based on the information that orders can be handled by sales representatives at pertaining lower-hierarchy business processes (e.g. Order Management, Shipping Management, …) (Fig. 6).

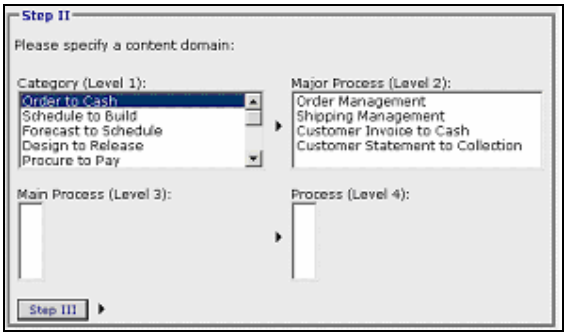

Fig. 6: The query scoping assistant (SA)

At the next step, the user uses the QSI to select process levels and define content requirements for the relevant process fields (Fig 7).

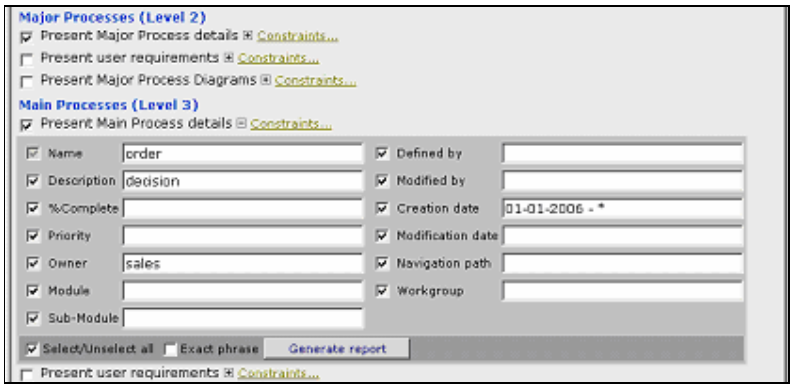

Fig 7. The query specification interface (QSI)

Following our example, the user will limit the "Name" field to include the string "order", the "Description" field to include the string "decision", and the "Owner" field to include the string "sales". He leaves the "exact phrase" option unchecked in order to retrieve more results. If, in addition, the user is interested only in "new" processes defined in the organization after a new sales strategy was implemented during 2006, he will add to the "Creation date" field the expression: "01-01-2006 - *". On top of these data fields the user can also check other required data fields. At the next step, the QI interprets QSI definitions into a set of normalized SQL queries, and the QRP joins all resulted data fields into query results ordered by hierarchal location witin the business process model. The example demonstrates how a user without any in-depth understanding of the data structure can extract relevant results for a relatively complex query – all by using the SA and the QSI.

5 Summary

The ProcessGene Query system provides a method for searching business processes, allowing users to phrase queries without extensive knowledge of the underlying database structure. Although the system provides a good starting point for developing the field of business process search queries, many innovations are needed to exploit open issues such as optimization of result sets, adding business logic for determining semantically related answers, query relaxation and the ranking of results. These issues were discussed extensively in the literature, but have not been addressed yet within the context of business process management.

It is hoped that by expanding the search and query capabilities on business processes content, researchers and IT practitioners will be able to generate complete and consistent business process models as part of their services to ERP/CRM/SCM community.

6 References

[1] Malone, T. W. *The Future of Work: How the New Order of Business Will Shape Your Organization, Your Management Style, and Your Life*. Boston, MA: Harvard Business School Press, 2004.

[2] Intentia. Reference Models, *http://www.intentia.com/WCW.nsf/pub/tools_index*, 2004.

[3] M. Lincoln, Karni R. A Generic Business Function Framework for Industrial Enterprises. *CD Proceedings of 17th ICPR Conference*, Blacksburg, VA, USA, October 2003.

[4] Oracle. Business Models (OBM), *http://www.oracle.com/consulting/offerings/implementation/methods_tools/*, 2006.

[5] SAP Solution Composer, *http://www.sap.com/solutions/businessmaps/composer/* 2006.

[6] C.P. Holland, B. Light, *A critical success factors model for ERP implementations*, IEEE Software 16, 1999, 30–35.

[7] Fettke, P.; Loos, P.; Zwicker, J.: Business Process Reference Models - Survey and Classification. In: Kindler, E.; Nottgens, M.: Business Process Reference Models – BPRM workshop proceedings: 1-15, 2005, BPM2005 workshop.

[8] A. Bernstein. Process Recombination: An Ontology Based Approach for Business Process Re-Design. *SAP Design Guild*, Vol. 7, October 2003.

[9] Avi Wasser, Maya Lincoln, Reuven Karni: Accelerated Enterprise Process Modeling Through a Formalized Functional Typology. BPM 2005: LNCS 3649 446-451.

[10] C. Rolland, N. Prakash, Bridging the gap between organizational needs and ERP functionality, Requirements Eng. 41, 2000, 180–193.

[11] Dov Dori, Iris Reinhartz-Berger: An OPM-Based Metamodel of System Development Process. ER 2003: 105-117

[12] G. Yang. Towards a Library for Process Programming. In W.M.P. van der Aalst, A.H.M. ter Hofstede, and M. Weske, editors, *BPM 2003*, volume 2678 of *Lecture Notes in Computer Science*, pages 120-135. Springer-Verlag, Berlin, 2003.

[13] OAGIS. Best Practices and XML Content for Everywhere-to-Everywhere Integration, http://www.openapplications.org/, 2004.

[14] Rosettanet. Lingua Franca for Business, *http://www.rosettanet.org/*, 2004.

[15] Wil M. P. van der Aalst, Arthur H. M. ter Hofstede: YAWL: yet another workflow language. Inf. Syst. 30(4): 245-275, 2005.