Towards Configurable Process Visualizations with Proviado

Ralph Bobrik
University of Ulm, DBIS, D-89069 Ulm
ralph.bobrik@uni-ulm.de

Thomas Bauer
DaimlerChrysler, GR/EPD, D-89013 Ulm
thomas.tb.bauer@daimlerchrysler.com

Abstract

Visualizing complex business processes is an important task of process-aware information systems (PAIS). Current PAIS, however, fail in providing adequate mechanisms for visualizing and monitoring business processes. In particular, PAIS do not support personalized or adaptable process drawings, which is particularly important for large processes. In the Proviado project we are developing a comprehensive framework for this purpose. It allows for flexible process visualization along three dimensions: process views, process notations, and process representation forms. In this paper we summarize selected concepts and features of the Proviado demonstrator.

1. Motivation

Business process (BP) implementations are often scattered over heterogeneous information systems (IS) spanning different organizational units. Usually, process owners have to gather relevant information manually from distributed IS in order to monitor the progress of their processes. In addition, a multitude of different stakeholders may be involved in the execution of a particular BP. Each of them needs a different view on the process with a personalized visualization and an adapted granularity of information. For example, while managers usually prefer an abstract overview of the process, process participants need a more detailed view on the process parts they are involved in.

In order to elaborate basic requirements for BP visualization we conducted several case studies in the automotive domain [1]. This, in turn, has led us to three dimensions for realizing configurable BP visualizations. First, it must be possible to reduce process complexity by discarding or aggregating information not relevant in the given context. Second, the appearance of process elements (e.g., activities, data objects, control and data connectors) must be customizable independent from the representation of the source process model. Third, different diagram types (e.g., process

graph, swim lane, calendar, Gantt, table) should be supported.

Current approaches do not provide this degree of flexibility. While some of them focus on general issues related to process visualization [5,6,7], others deal with more specific aspects like process graph layouting [8]. In existing PAIS, process models are usually presented to the user in exactly the same way as they have been drawn by the process designer [3]. Recently, BP modeling tools have emerged, which additionally allow to alter the graphical appearance of a process model and to hide selected process aspects (e.g., data flow) [4]. Sophisticated concepts for creating process views are missing in existing tools.

2. The Proviado Framework

The Proviado framework aims at flexible and configurable BP visualizations. In particular, it enables BP visualizations which are adaptable to the needs of different user groups along the aforementioned dimensions [1]. For realizing a particular drawing of a BP model or BP instance respectively, a *visualization model* has to be specified separately from the process. Among other things, a visualization model allows to configure which process elements are to be displayed and which notation shall be used. This configuration can be specified at a high level based on a powerful view concept and a flexible template mechanism.

Process View Concept. The *process view* concept we developed allows reducing the complexity of a BP visualization. This is achieved by applying well-defined transformation rules based on graph reduction and graph aggregation. The *reduction* operation can be used to remove process objects from a process model. As an example consider Fig. 1 where activities E, F and G are removed from the given process model and a new control edge is inserted instead. Fig. 1 also gives an idea of the *aggregation* operation. Aggregate(B,C,H,K), for example, aggregates four activities by replacing them with one abstract node. Depending on the concrete structure of the sub-graph induced by the set of activities to be aggregated, different graph
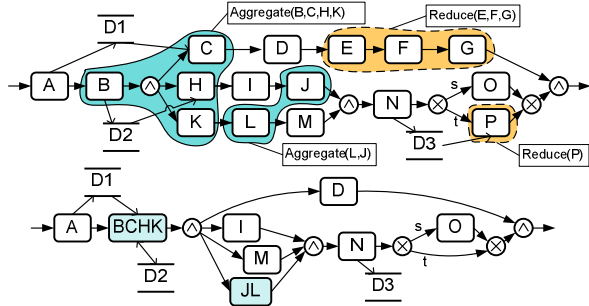
Fig. 1 Proviado Process View Concept

transformations may have to be applied. While in some cases the aggregated process view can be realized by simple graph transformations, in other scenarios this necessitates a more complex restructuring of the process graph. Generally, aggregated process views are more difficult to realize than reduced ones. In particular, relations to satellite objects (e.g., data elements, org. roles) have to be preserved (cf. Fig. 1) and attribute values for the abstract activity node resulting from the aggregation have to be calculated. Finally, aggregation operations are provided for all process aspects including data flow, and actor assignments.

It is important to mention that view building operations as provided by Proviado maintain a well-structured process model if desired. However, to introduce additional flexibility for BP visualizations, operations "violating" structural model constraints (e.g. *DeleteEdge*) are considered as well. Higher level operations built on top of aggregation and reduction operations exist that automatically derive the set of activities to be processed. This facilitates maintenance of view definition when changing the base process model.

Template Mechanism. While the described view concept allows us to define which process elements shall be displayed, the *Template Mechanism* (for details see [2]) enables us to configure the graphical appearance of the different elements. In this context a *template* represents the concrete notation (i.e., the symbols) to be used for visualizing a particular process element (e.g., activity, data object). Its graphical appearance (e.g., shape, arrow) is described based on Scalable Vector Graphics (SVG). By using this XML-based format, to a large degree, we can define templates graphically with a standard SVG Editor.

Each template comprises a set of data fields (i.e., parameters) which can be filled with concrete process data values (e.g., activity name or state) at visualization time. We use XPath expressions to establish the relationship between symbol definition and data fields. Required data transformations (e.g. date format conversion) can be realized via ECMA-Script expressions.

Altogether, a complete notation for BP visualization

consists of a set of templates. More precisely, each process element has to be linked to a template. This link can be established statically (i.e., remain unchanged) or dynamically based on selected process data (e.g., the runtime status of the process element). The latter allows, for instance, to use different symbols for activities, e.g., depending on their state or on the actor working on them. Finally, *Cascading Style Sheets* are used to vary the look of process drawings.

All in all the sketched Template Mechanism allows us to use a process notation in an unambiguous and easy to maintain manner. In combination with the view concept personalized BP visualizations become possible. While non-relevant process elements can be removed or aggregated with other objects, the visualization of relevant process elements can be adapted to specific user or application needs.
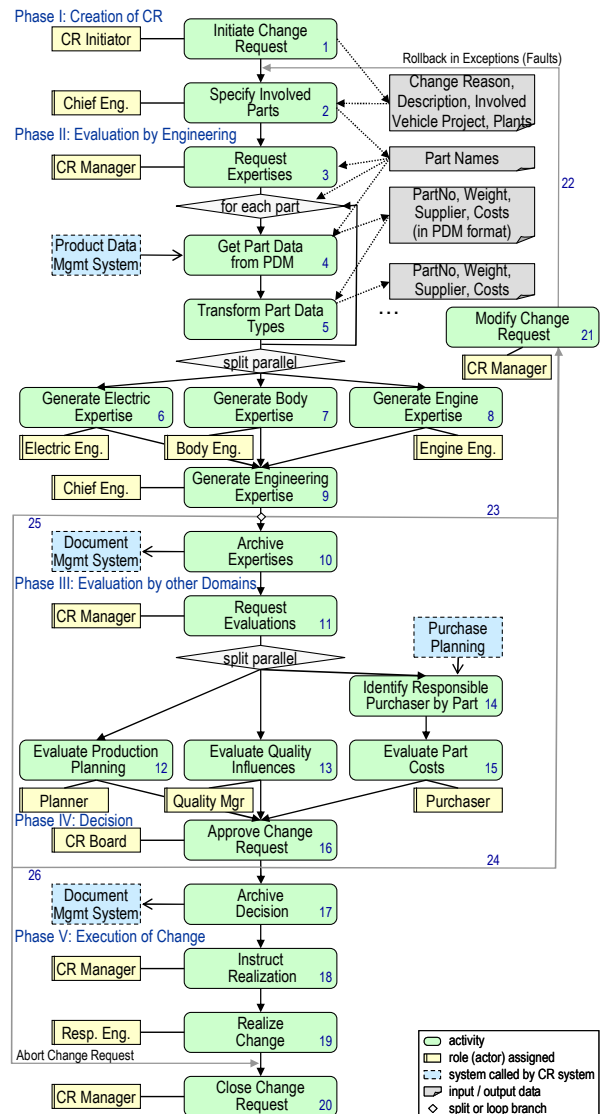

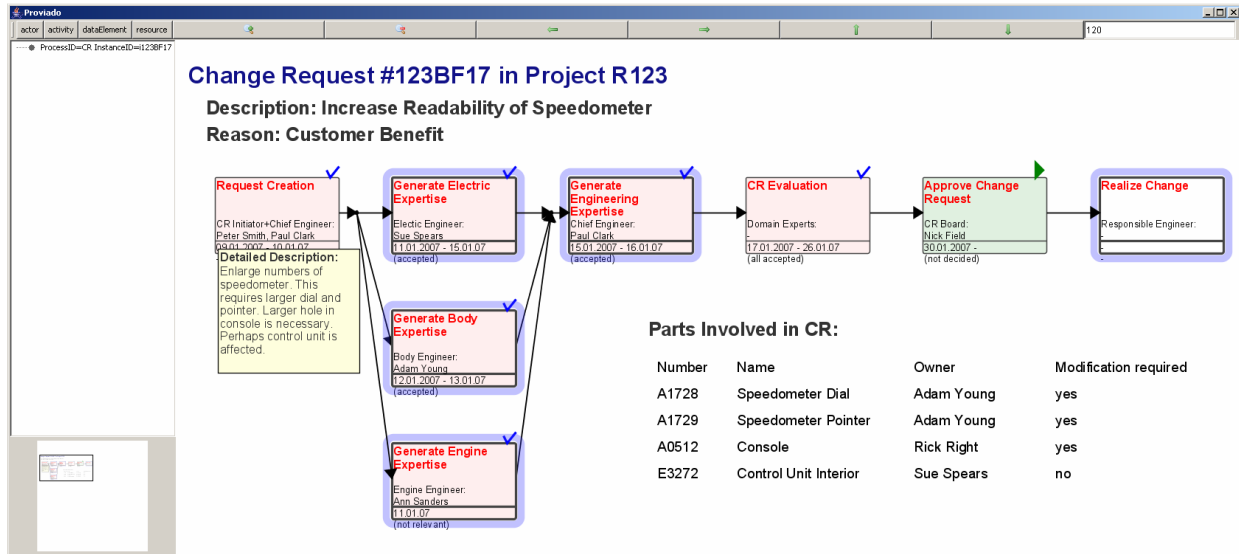**Fig. 2** Base process of the Change Request (CR)

Fig. 3 Process visualization for an engineer involved in the process as participant

3. The Proviado Demonstrator

For illustration purposes we consider a change request (CR) process from the automotive domain. The *base process model* is depicted in Fig. 2. It comprises activities and their assignment to five process phases. Furthermore, control and data flow, exceptional paths, role assignments, and IT system resources are depicted.

Assume that this process shall be visualized for an actor from the engineering domain. For this purpose non-relevant process elements have to be discarded. Automated steps for transforming and exchanging data (e.g. 4 and 5), for example, shall be not displayed. The same applies to selected interactive steps (e.g. 2 and 3). Finally, control edges capturing forward and backward jumps shall be removed. Altogether this view can be realized by applying the following view operations:

Aggregation:[1] {1, 2}, {11, 12, 13, 14, 15}
Reduction: {3}, {4, 5}, {10}, {17, 18}, {20}, {21}
DeleteEdge: {22, 23, 24}, {25, 26}

The resulting process view would still contain a large number of satellite nodes (representing actors, systems, etc.) which usually shall not be displayed. Our visualization model allows to omit such nodes and to assign their data values to other visualization objects, e.g., activity boxes (cf. Fig. 3). Furthermore, with our *Template Mechanism* any desired appearance of the process view to be displayed can be realized. For example, the visualization from Fig. 3 contains information like change reason, change description, and involved parts. Furthermore, a header has been added. Other data like a detailed CR description can be accessed via a tooltip. Finally, activities which are of particular importance for engineers are highlighted.

Note that the created process drawing as depicted in Fig. 3 constitutes one possible visualization of our base process model from Fig. 2. Depending on specific user requirements, for example, Proviado allows to provide different visualizations of the same process view (based on the described Template Mechanism, e.g., using a standardized notation like BPMN). In this context, different information and layouts can be presented. Furthermore, new process views (with same or different appearance) can be easily realized. For example, for managers each of the five phases of the CR process (cf. Fig. 2) could be aggregated to one single activity and only information about deadlines, delays, resources, and the final decision be visualized.

---

[1] Each operation is listed in brackets. The aggregations result in the activities "Request Creation" and "CR Evaluation".

References

[1] R. Bobrik, M. Reichert, T. Bauer: Requirements for the Visualization of System-Spanning Business Processes. *Proc. DEXA-Workshops,* Copenhagen, 2005.

[2] R. Bobrik, T. Bauer, M. Reichert: Proviado - Personalized and Configurable Visualizations of Business Processes. *Proc. EC-Web,* Krakow, 2006.

[3] IBM: IBM WBI Monitor V. 4.2.3. IBM Report, 2003.

[4] IDS Scheer AG: ARIS Business Publisher ver.7.0, 2007.

[5] P. Luttighuis, M. Lankhorst, R. van de Wetering, R. Bal, H. van den Berg: Visualising Business Processes. *Computer Languages* (27), 2001.

[6] J. Mendling, A. Brabenetz, G. Neumann: EPML2SVG - Generating Websites from EPML Processes. EPK 2004.

[7] A. Streit, B. Pham, R. Brown: Visualization Support for Managing Large Business Process Specifications. *Proc. 3rd Int. Conf. Business Process Management*, LNCS 3645, 2005.

[8] J.M. Six, I.G. Tollis: Automated Visualization of Process Diagrams. *Proc. Symp. on Graph Drawing* (GD 2001).