APPLYING ACTIVITY PATTERNS FOR DEVELOPING AN INTELLIGENT PROCESS MODELING TOOL

Lucinéia Heloisa Thom, Manfred Reichert
Institute for Database and Information Systems, Ulm University, Oberer Eselsberg, 89069, Germany

Carolina Ming Chiao, Cirano Iochpe
Institute of Informatics, Federal University of Rio Grande do Sul, Av. Bento Goncalves 9500, 91501-970, Brazil

Keywords: Business function, process modeling, activity pattern, ontology.

Abstract: Due to their high level of abstraction and their reusability, workflow patterns are increasingly attracting the interest of both BPM researchers and BPM tool vendors. Frequently, process models can be assembled out of a set of recurrent business functions (e.g., *task execution request*, *approval*, *notification*), each of them having generic semantics that can be described as activity pattern. To our best knowledge, so far, there has been no extensive work implementing such activity patterns in a process modeling tool. In this paper we present an approach for modeling business processes and workflows. It is based on a suite which, when being implemented in a process modeling tool, allows to design business processes based on well-defined (process) activity patterns. Our suite further provides support for analysing and verifying certain properties of the composed process models (e.g., absence of deadlocks and livelocks). Finally, our approach considers both business processes designed from scratch and processes extracted from legacy systems.

1 INTRODUCTION

Organizations are increasingly interested in improving the efficiency and quality of their business processes as well as in finding ways to better cooperate with customers and business partners (Weske, 2007), (Lenz, 2007). To achieve these goals, enterprises are adopting Business Process Management (BPM) tools as well as emerging patterns for process modeling and change.

BPM technology (e.g., workflow systems) enable the definition, enactment and monitoring of the operational processes of an enterprise. Moreover, through Web service technology, the benefits of BPM can be created for cross-organizational business processes involving linked organizations as well. By automating processes BPM technology contributes to the reduction of costs, execution times, errors and redundancies with respect to process performance. At the same time, it improves control over the processes (Thom, 2006a).

Usually, business processes comprise a variety of business functions, with a specific and well-defined semantics. Thereby, a particular business function may occur several times within one or multiple process definitions (Thom, 2006b), (Thom, 2006c). As example consider a simple approval process for changing the layout of a product (cf. Figure 1).

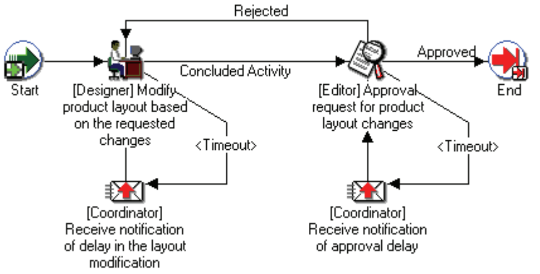
Rejected
Start
[Designer] Modify product layout based on the requested changes
Concluded Activity
[Editor] Approval request for product layout changes
Approved
End
<Timeout>
<Timeout>
[Coordinator] Receive notification of delay in the layout modification
[Coordinator] Receive notification of approval delay

Figure 1: Approval process for product layout change.

This process includes the following activities: (1) a designer modifies the product layout according to the requested changes; (2) the designer (optionally) receives a notification when Activity 1 is delayed; (3) an approval for the changed product layout is requested; (4) the requested editor (optionally) receives a notification when the approval is delayed. Altogether this process comprises four (recurrent) business functions with generic semantics that can be described in terms of activity patterns. In detail,

Heloisa Thom L., Reichert M., Ming Chiao C. and Iochpe C. (2008).
APPLYING ACTIVITY PATTERNS FOR DEVELOPING AN INTELLIGENT PROCESS MODELING TOOL.
In *Proceedings of the Tenth International Conference on Enterprise Information Systems - ISAS*, pages 112-119
DOI: 10.5220/0001704801120119
Copyright © SciTePress

the following business functions are used: *task execution request* (Activity 1), *notification* (Activities 2 and 4), and *approval* (Activity 3). We denote these recurrent functions as workflow activity patterns (activity patterns for short); i.e., activity patterns represent business functions that occur several times within one or multiple process models, and therfore might be reused when defining other business processes.

Recently, workflow patterns have been suggested capturing different process aspects: control flow (Russell, 2006a), data flow (Russell, 2004a), resources (Russell, 2004b), exception handling (Russell, 2006b), service interactions (Barros, 2005), process change (Weber, 2007), application–oriented aspects (Bancroft, 1998), and process compliance (Namiri, 2007). Yet, all these patterns have in common that they are relevant for implementing BPM systems and for defining adequate and expressive process modeling languages.

However, all these patterns provide only a partial answer to the question what business functions a modeler wants to use repeatedly when designing process models (Thom, 2007a), (Thom 2007b). In practice, respective business functions (Medina-Mora, 1992), (Flores, 1998), (Muehlen, 2002), (Malone, 2004) are often re-defined from scratch for almost every process definition. This, however, is inefficient, and also undesirable from a maintenance perspective. While some research has been reported on how metadata can be organized to manage large-scale modeling projects (see Thomas and Scheer 2006), we are not aware of any work evidencing the existence of activity patterns for defining business functions within real process models, or for investigating their necessity and completeness with respect to business process modeling. Besides that, contemporary process modeling tools do not provide functionalities that enable users to define, query, and reuse such patterns in a proper and effective way.

Related to these problems, in earlier work we proposed a set of seven workflow activity patterns. Each of these activity patterns captures a recurrent business function (such as the ones shown in Figure 1) we can find frequently in business processes. Combined with specific control flow patterns, respective activity patterns are suitable to design a large variety of process models in different domains. In this paper we briefly report on the results of an empirical study in which we analyze the frequency of activity patterns taking a set of 214 real-world process models from domains like quality management, software access control, and electronic change management. For specific process categories, we further discuss results of an additional analysis in which we investigate the frequency of co-occuring activity patterns. The result of this analysis is utilized for developing an intelligent suite for normalizing and modeling business processes based on the reuse of activity patterns. Given some information about the kind of process being designed, the results of our analysis can be further used by this suite to suggest a ranking of the activity patterns suited best to follow the last pattern modeled. With normalization we mean the definition of a standard description form to which the business processes are translated, i.e., a canonical format for describing process models.

This suite, which we denote as *Workflow Modeling Tool in the following*, can be added as an extension to existing process modeling components (e.g., Intalio (Intalio, 2006), Aris Toolset (IDS Scheer, 2007), or ADEPT Process Composer (Reichert, 2006)). Basically, the suite is intended to provide a number of advanced modeling functionalities, such as the: (1) extraction of business processes from legacy systems and their normalization, correctness checking and translation into a standard notation; (2) support for designing normalized process models by suggesting to the designer activity patterns relevant in the given modeling context (e.g., considering statistical co-occurences of multiple patterns) ; (3) construction of a knowledge base for storing and retrieving activity patterns.

The remainder of this paper is organized as follows: Section 2 gives an overview of the activity patterns we identified in prior research. Exemplarily, we present the *approval* and the *decision* patterns in more detail. In Section 3 we discuss the results of empirical studies we performed in different domains in order to verify how often activity patterns are used in the design of process models. In Section 4 we describe the suite that aims at supporting the reuse of these activity patterns. Finally, Section 5 concludes the paper and gives an outlook on future research.

2 ACTIVIY PATTERNS

In the context of this paper we use the term Workflow Activity Pattern (WAP; activity pattern for short) to refer to the description of a recurrent business function that can be frequently found in business processes (e.g., *notification*, *decision*, *approval*). Initially, we derived seven activity

Table1: Overview of activity patterns representing business functions.

WAP - Name	Description
WAP1: *Approval*	An object (e.g., a document) has to be approved by one or more organizational roles.
WAP2: *Question-answer*	WAP2 allows to formulate a question in the context of a process, to identify an organizational role who is able to answer it, to send the question to this role, and to wait for the respective response (*single-question-answer*)
WAP3: *Unidirectional Performative*	A sender requests the execution of a particular activity from another actor involved in the process. The sender continues execution of his process part immediately after having sent the request for performing an activity..
WAP4: *Bi-directional Performative*	A sender requests the execution of a particular activity from another actor involved in the process. The sender waits until the receiver notifies him that the requested activity has been performed.
WAP5: *Notification*	The status or result of an actvity execution is communicated to one or more process participants
WAP6: *Informative*	An actor requests a certain information from a process participant. He continues process execution after having received the requested information.
WAP7: *Decision*	WAP7 allows to include a decision activity in the flow with connectors to different subsequent execution branches. Those branches will be selected for execution whose transition conditions evaluate to true.

WAP1: APPROVAL

Description: An object (e.g. a document) has to be approved by one or more organizational roles. Depending on the context, the evaluation is executed only once or it is requested multiple times (and approval is done either isequentially or in parallel).

Example: In a change management process, for example, a particular change request may have to be concurrently approved by all organizational roles concerned by the change. If one of these roles rejects the change request, it will be not approved.

Problem: During the execution of a business process, object approval by one or multiple organizational roles is required before proceeding with the flow of control.

Issues:

a) The approval activity is executed only once and by one organizational role.

b) The single approval is executed multiple times in processes being executed in flat and descentralized organizations (or specific organizational units).

c) Final decision can be made manually (i.e., by a user) or automatically according to some rules.

Solution: The bellow process fragment illustrates a single approval using the BPMN notation; here an organizational role *reviewer* performs a document review either resulting in approval or disapproval.

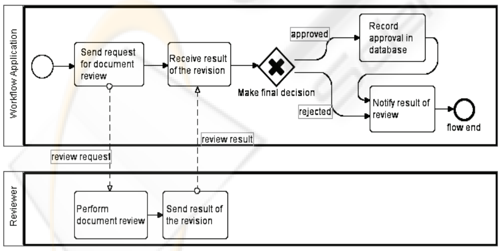

Figure 2: Approval activity pattern.

WAP7: DECISION

Description: In a process the execution of one or multiple activities is requested. Depending on the results the process continues execution with one or multiple branches.

Example: To get feedback from a user concerning a particular service, he shall indicate his satisfaction degree by giving grades from 0 to 10. Depending on the specified grade process execution continues with one or multiple branches depending on their conditions (e.g., grade between 0 and 4).

Problem: In a process an explicit decision step has to be included. The final decision is made based on the execution result(s) of requested activities.

Issues:

a. The decision pattern is usually combined with a performative bi-directional pattern .

b. Based on the response one or several subsequent branches are selected for execution..

c. The final decision is usually made automatically based on the execution result(s) of previous activities.

Solution: The bellow process fragment illustrates a single decision pattern using the BPMN notation.

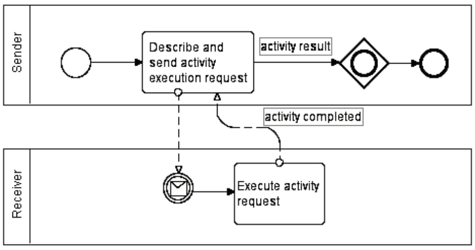

Figure 3: Decision activity pattern.

patterns based on an extensive literature study: **APPROVAL**, **QUESTION-ANSWER**, **UNI-DIRECTIONAL PERFORMATIVE**, **BI-DIRECTIONAL PERFORMATIVE**, **INFORMATIVE**, **NOTIFICATION** and **DECISION.** Table 1 shows an overview of the 7 activity patterns we identified. Each of them is specified taking the following attributes: *Description*, *Example*, *Problem* and *Issues*. (We have considered additional attributes as well like *Design Choices* (pattern variants), *Related Patterns* and *Pattern Implementation*. However, these attributes are outside the scope of this paper. Bellow we give two examples of pattern descriptions: the **APPROVAL** activity pattern for single approval (i.e., the approval pattern is executed only once) and the **DECISION** activity pattern.

3 ANALYZING THE FREQUENCY OF ACTIVITY PATTERNS IN REAL PROCESS MODELS

With the goal to check whether the identified activity patterns are present in real applications as well, we analyzed 214 process models. These process models have been modeled either with the Oracle Builder tool or anUML modeler. Altogether, the analyzed process models stem from 13 different organizations and are related to different applications like Total Quality Management (TQM), software access control, document management, help desk services, user feedback, document approval and electronic change management.

Amongst others, we have obtained the following results from our empirical studies (i.e. from the analysis of the 214 process models):

1. evidence with high probability that the activity patterns exist in real process models;
2. evidence that the set of activity patterns is both necessary and sufficient to model all 214 process models analyzed; and
3. identification of common ocurrences of activity patterns based on a classification of the respective processes into *Human-Intensive* and *System-Intensive* (Le Clair, 2007). For example, if the *Decision* activity pattern occurs in an intensive-system process it is most of the times followed by a *Notification* activity pattern.

For each activity pattern we calculate its *support value*. In the given context *support* corresponds to the number of occurrences of each activity pattern when looking at the total set of 214 process models. For those models comprising more than one occurrence of the same pattern we consider just one of these occurences.

First, we identify and annotate activity patterns within all analyzed process models. Second, for all process models we count the number of occurrences of each pattern. To get relative values, the obtained result is divided by the total number of analyzed process models (i.e. 214 in our study).

3.1 Frequency of Activity Patterns in Process Models

The following five activity patterns are not dependent on specific application domains or organizational structure aspects (e.g., the degree of centralization in decision making, standardization of work abilities): UNIDIRECTIONAL and BI-DIRECTIONAL PERFORMATIVE, DECISION, NOTIFICATION and INFORMATIVE . This fact mainly explains why these five patterns have been identified with high frequency in almost all analyzed process models. The same applies to the APPROVAL pattern. This can be explained by the high degree of centralization on decision-making existing in the organizational units for which we analyzed their process models. This high centralization implies the use of approval activities. By contrast, most of the process models analyzed do not comprise QUESTION-ANSWERING activities. Figure 4 graphically illustrates the frequency of each activity pattern with respect to the set of process models analyzed.

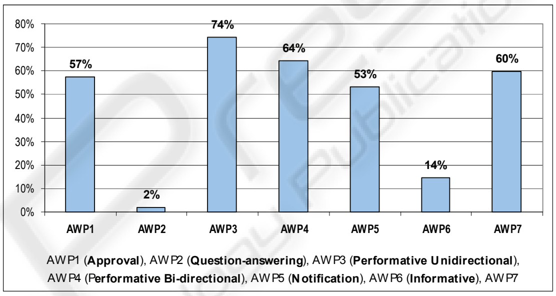

Figure 4: Frequency of activity patterns in real process.

3.2 Identifying Common Ocurrences of Activity Patterns in Process Models

One of the use cases for the *knowledge base* of our suite (cf. Section 4) is based on a mechanism that gives design time recommendations with respect to the most suited activity patterns to be combined with an already used pattern. This mechanism utilizes statistical data we gathered during our empirical study, which we also summarize in this section. To obtain the frequencies for pattern co-occurences, we analyze the sequences of the occuring activity patterns in 154 of the 214 process models studied.

In earlier work, we have shown that if we classify the process models into human–oriented (i.e., with human intervention during execution) and fully automated (i.e., with no human intervention during execution) we can identify certain activity patterns more often in one of the two categories. We tried to classify the processes based on common characteristics (e.g., application domain), also considering classifications from the literature in this context. However, most of the studied classifications (Dowson, 1987), (Harrington, 1991) and (Leymann, 1999) are based on specific application domains of the related process models. Accordingly, those approaches are not applicable to our analysis because the set of the process models we have been investigating does not cover all the categories covered by these approaches.

We decided then to use the approach of Le Chair who classifies business processes into *system-intensive* and *human-intensive* (Le Clair, 2007). The system-intensive processes are characterized by being handled on straight-through basis, this means that there is minimal or no human intervation and few exceptions migh occur. The human-intensive processes require people to get work done by relying on business applications, databases, documents as well as other people and interacting extensively with them. This type of process requires human intuition or judgment for decision-making during individual steps.

By classifying our set of process models in those two categories, we obtain 123 human-intensive process models and 31 system-intensive process models respectively. Remember that in this analysis we consider 154 of the 214 process models. The next step was to evidence the occurrence of the activity patterns in the two categories of process models. Figure 5 shows the frequency of the workflow activity patterns in the *system-intensive process models* and the *human-intensive*. Note that some patterns (i.e. approval, informative, question-answer) do not appear in the system-intensive process models at all. These patterns are frequently related to human activities, i.e. are executed by an organiyational role.

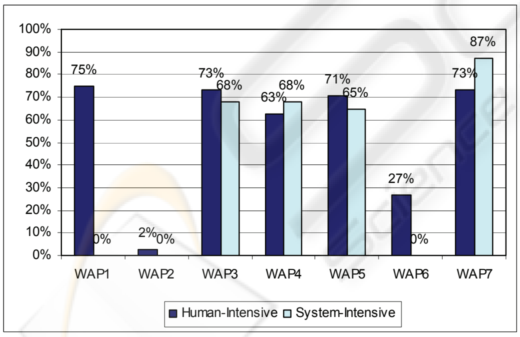

Figure 5: Frequency of Activity Patterns in Human-intensive and System-intensive Process Models.

In other analysis we search for frequent and recurrent occurences of activity patterns in the process models. Relying on these common ocurrences of activity patterns the knowledge base must show to users a ranking of the most frequent subsequent activity patterns which follow the activity pattern the user has recently modeled in the process.

Figure 6 shows how often the DECISION PATTERN is used immediately after the other workflow activity patterns in the set of process models we analyzed. Note that for each kind of process (i.e. human intensive and system intensive), specific pairs of process have more probability to occur. Per example the pair DECISION → NOTIFICATION (AWP5) is more often in *system-intensive process models.* On the other hand, the pair DECISION → APPROVAL is more often in the *human-intensive process models*

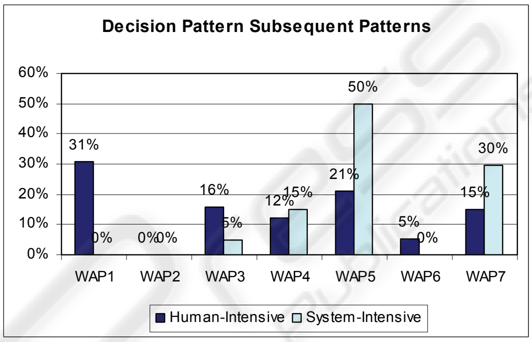

Figure 6: Subsequent Activity Patterns of the Decision Pattern (regarding both system- and human-intensive processes).

In Figure 7, we present the result of the analysis of the PERFORMATIVE UNIDIRECTIONAL subsequent patterns. In system-intensive process models, the most frequent pair of activity patterns is UNIDIRECTIONAL → UNIDIRECTIONAL. A considerable amount of the studied system-intensive process models presented a sequence of 2 or more UNIDIRECTIONAL PERFORMATIVE patterns as a way of modularizing distinct software functions. In human-intensive process models, the frequency of the pairs was very similar, except for the less frequent activity patterns.

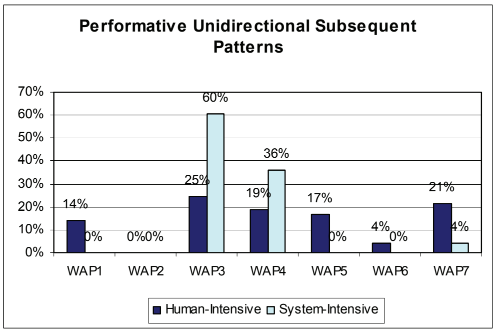

Figure 7: Frequency of performative unidirectional subsequent patterns on system-intensive and human-intensive processes.

As example consider a *system-intensive process* to delete an existent document (cf. Figure 8). This process includes the following activities: (1) the data used by an external program (*Bussines Event System)* to delete a document are set; (2) the properties of an event to delete the document are defined; (3) an event to delete the document is executed; (4) an event informs whether the document is deleted or an error ocurred; (5) in case of an error the main author of the document is notified. Activity 4 is then repeated. Note that the process include a sequence of 4 UNIDIRECTIONAL PERFORMATIVE patterns (activities 1, 2, 3 and 4 respectively).

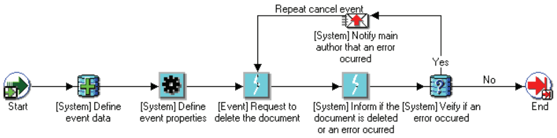
Repeat cancel event
[System] Notify main author that an error ocurred
Yes
No
Start
[System] Define event data
[System] Define event properties
[Event] Request to delete the document
[System] Inform if the document is deleted or an error ocurred
[System] Veify if an error occured
End

Figure 8: Example of system-intensive process.

4 TOWARDS AN INTELLIGENT BPM TOOL

We now present an *intelligent workflow designer* tool (intelligent suite for short) for normalizing and modeling business processes based on the reuse of activity patterns. This suite can be added as an extension to existing process modeling components (e.g., Intalio (2006), Aris Toolset (IDS, 2007), or ADEPT Process Composer (Reichert, 2006)).

Core functionalities of the intelligent suite are:

1. Extraction of business processes from legacy systems and their normalization: Comprises the extraction of business rules from the analysis of source code (e.g. COBOL, clipper, access, visual basic, C++) of legacy systems and subsequent generation of business processes in high-level notation (such as the BPMN). The process is then, validated by matching it with existent activity patterns stored in a knowledge database. The challenge here is to identify all embody activity patterns comprised by the process. As a result the process is translated into one or more activity patterns. Such procedure must benefit the translation of the processes to some execution language (e.g., BPEL4WS). Furthermore, with the scope of business process extraction, a model checking is performed, in order to test the correctness (accuracy) of the process model.
2. Support to process design: a user process is received by the *intelligent workflow designer* as an input. The process is then, matched with activity patterns stored in the knowledge database in order to identify the partial order of activity patterns it comprises. Having this information, the intelligent suite will recommend the most suitable activity patterns to be used together with the activity pattern designed before. In addition, it will inform how frequent each pair of activity patterns was used in earlier modeling. This module will be developed based on the analysis result we presented in Section 3.2.
3. Construction of a knowledge database of activity patterns: The activity patterns repository will store not only the activity patterns but also the frequency with each activity pattern is combined with an already used pattern. Through the analysis of new process models (e.g., from automotive as well as health care domain) we aim at increasing the support value of the sequences of activity patterns (cf. Section 3.2). Thus, in design time the accuracy, concerning the frequency associated with each recommendation of pair of pattern be correct may increase. Figure 9 illustrates the intelligent suite.

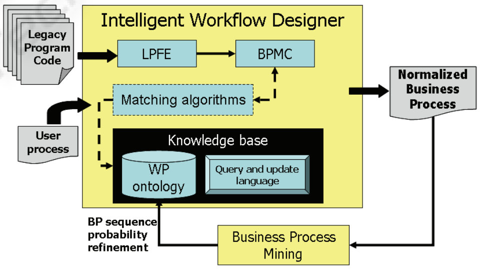
Intelligent Workflow Designer
Legacy Program Code
LPFE
BPMC
Normalized Business Process
Matching algorithms
User process
Knowledge base
WP ontology
Query and update language
BP sequence probability refinement
Business Process Mining

Figure 9: Intelligent Suite.

Core components of the intelligent suite are:

- *Legacy Program Flow Extractor (LPFE)*: component responsible by the extraction of business process rules from the source code of legacy systems. In addition, generation of corresponding process in high-level language (such as BPMN).
- *Business Process Model Checking (BPMC):* this component verifies how complete and correct the extracted process is. First the process is translated to some formal language (e.g., Pi-calculus). In case it is correct, the process is matched with the knowledge base so that the activity patterns comprised by the process can be identified.

- *Knowledge Base*: is the database where the activity patterns are stored. It is composed by an ontology which describes the activity patterns. Furthermore, it comprises a query and update language (mechanism). The mechanism gives design time recommendations with respect to the most suited activity patterns to be combined with an already used pattern. In addition, the update mechanism must be used to change the probabilistic results of each pair of activity pattern based on the process analysis results.
- *Matching Algorithms*: algorithms responsible by the identification (*matching*) of the activity patterns maintained in the ontology. The selected activity patterns are those comprised by either the user process or the processes generated by the LPFE and BPMC components.
- *Business Process Mining*: External tool to the Intelligent Workflow Designer which receives a set of normalized activity patterns as input. The output of this tool will permit the knowledge base to be updated, once it will have new information about the frequency of each pair of activity pattern.

5 CONCLUSIONS AND FUTURE WORK

While workflow patterns were defined for several aspects related to process execution, the aspect of recurrent business functions is only partially addressed by existing work. In prior work, we identified a set- of seven workflow activity patterns that appear necessary and sufficient to model a large variety of process models. In other work we investigated in how far process modeling tools can be tailored to provide a direct support for pattern reuse. Further we are working in the documentation of the activity patterns with Pi-calculus. In this paper we reported the results of an empirical work where we search how often activity patterns as well as common ocurrences of them (i.e. pairs of activity patterns) are present in a large set of real process models. In addition we proposed an approach for modeling business processes and workflows. It is based on a suite which, when being implemented in a process modeling tool, allows to design business processes based on well-defined (process) activity patterns. Our suite further provides support for analysing and verifying certain properties of the composed process models (e.g., absence of deadlocks and livelocks). Our approach considers both business processes designed from scratch and processes extracted from legacy systems.

The main advantages of this approach can be summarized as follows: (a) the completeness and necessity of the activity patterns for process design has already been evidenced in prior work; (b) the intelligent suite is tool-independent and can be adapted for any workflow modeling tool; (c) the business process model checking can be considered as a very important component which can help in the verification of how complete and correct is the process being designed. This can be accomplished through matching the current process model with activity patterns stored in the knowledge base.

As future work we intend to perform additional analyzes considering process models from different application domains (e.g., health insurance and automotive). Our goal is to identify more common ocurrences of pairs of activity patterns. In this context we also intend to continue studying the workflow classifications so that we can find more specific classification and with smaller granularity to divide the set of processes. A less generic classification will be useful when we try to converge on the user needs using just a few steps. We also consider making an experiment for comparing process modeling with and without activity pattern support. Finally, we intend to implement the intelligent suite in an existent workflow designer tool.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the Coordination for the Improvement of Graduated students (CAPES), the Institute of Databases and Information Systems of the University of Ulm (Ulm, Germany) and the Informatics Institute of Federal University of Rio Grande do Sul (Porto Alegre, Brazil).

REFERENCES

Bancroft, N., Henning, S., Sprengel, A.. 1998. *Implementing SAP R/3*. 2nd ed. Greenwich, Conn.: Manning, xxiv, 310 p. : il.

Barros, A., Dumas, M. and ter Hofstede. 2005. *Service Interaction Patterns*. In W.M.P. van der Aalst, editor, Proceedings of the BBPM, v. 3649 of Lecture Notes in Computer Science, pp. 302-318. Springer-Verlag.

Dowson, M. (1987) Interation in the Software Process Review of the 3rd International Software Process Workshop. In *Proceedings of the 9th international Conference on Software Engineering* (Monterey, California, United States). International Conference on Software Engineering. IEEE Computer Society Press, Los Alamitos, CA, 36-41.

Flores, F. 1998. *Computer Systems and the Design of Organizational Interaction*.

Harrington, H. J. (1991) "Business Process Improvement: The Breakthrough Strategy for Total Quality, Productivity, and Competitiveness". McGraw-Hill. 274p.

Malone, T. W. Crownston, K. Herman, G.A. 2004. *Organizing Business Knowledge*: The MIT Process Handbook. ISBN 0-262-13429-2.

Medina-Mora, R., 1992. *The action workflow approach to workflow management technology*.

IDS Scheer. 2007. *Aris Plataform*: Product Brochure. http://www.ids-scheer.com/.

Intalio. 2006. *Creating Process Flows*. http://bpms.intalio.com.

Le Clair, C.; Teubner, C. 2007. *The Forrester Wave*: Business Process Management for Document Processes, Q3 2007.

Lenz, R.; Reichert, M. 2007. *IT Support for Healthcare Processes – Premises, Challenges, Perspectives*. Data and Knowledge Engineering, 61:39-58

Leymann, F.; Roller, D. (1999) "Production Workflow: Concepts and Techniques". Prentice Hall. 479p.

Keller, G.; Nüttgens, M.; Scheer; A.-W., 1992. *Semantische Prozessmodellierung auf der Grundlage "Ereignisgesteuerter Prozessketten (EPK)"*. Heft 89, Institut für Wirtschaftsinformatik, Saarbrücken, Germany.

Namiri, K, Stoganovic, N. 2007. Pattern –Based Design and Validation of Business Process Compliance. In: *Proceedings of OTM*, Springer-Verlag, Heidelberg, p. 59-76.

Reichert M., Rinderle S., Kreher, U., Acker, H., Lauer, M., Dadam, P.: *ADEPT Next Generation Process Management Technology* (Tool Demonstration). Proc. CAiSE'06 Forum, Luxembourg, June 2006, pp. 11-14

Russell, N., ter Hofstede A.H.M, Edmond, D. 2004a. *Workflow Data Patterns*. In Proceedings of the Informatik 2004 - Informatik verbindet (Band 1), Lecture Notes in Informatics (LNI). Ulm. v. p-50.

Russell, N., 2004b. *Workflow Resource Patterns*. Technical report, FIT-TR-2004-01, Queensland University of Technology, Brisbane.

Russell, N., van der Aalst, W.M.P, ter Hofstede, A.H.M, 2006b.*Workflow Exception Patterns*. In Proceedings of 18th CAiSE, 18, p.288-302.

Russell, N., ter Hofstede A.H.M, van der Aalst, W.M.P and Mulyar, N. 2006a. *Workflow Control-Flow Patterns*: A Revised View. BPM Center Report BPM-06-22, BPMcenter.org.

Smith, H., and Fingar, P., 2003. *Business Process Management*: The Third Wave. Meghan-Kiffer Press.

Stefansen, C. Smawl. 2005. A Small Workflow Language Based on CCS. In 17th *Conference on Advanced Information Systems Engineering*, Porto, Portugal.

Weske, M. 2007. *Business Process Management*: Concepts, Languages, Architectures. Berlin: Spring-Verlang.

Thom, L. H., Iochpe, C., Amaral, V. L., and Viero, D. M. 2006a. In: *Workflow Handbook 2006* Including Business Process Management, Chapter Toward Block Activity Patterns for Reuse in Workflow Design, Workflow Management Coalition, p. 249-260.

Thom, L. H.. 2006b. Applying Block Activity Patterns in Workflow Modeling. In *Proceedings of the 8th International Conference on Enterprise Information Systems*, Paphos, Cyprus, p. 457-460.

Thom, L. H.. 2006c. *A Pattern –based Approach for Business Process Modeling*. Porto Alegre: PPGC. (PhD Thesis).

Thom, L. H ., Lau, J. M., Iochpe, C., Mendling, J.2007a. Extending Business Process Modeling Tools With workflow pattern Reuse. In: *Proceedings of the International Conference on Enterprise Information Systems*, ICEIS, Funchal (Portugal), v.9.

Thom, L. H. Iochpe, C., Reichert, M. 2007b. *Workflow Patterns for Business Process Modeling*. In: Conjunction with CAiSE'07, Workshop on Business Process Modeling, Development, and Support. Trondheim (Norway), v. 8.

Thomas, O.; Scheer, A.-W. 2006. Tool Support for the Collaborative Design of Reference Models - A Business Engineering Perspective. In: *39th HICSS-39 2006), CD-ROM / Abstracts Proceedings*.

zur Muehlen, M. 2002. *Workflow-based process controlling*: foundations, design, and application of workflow-driven process information systems. Logos Verlang Berlin: Berlin. 299 p.

Weber, B., Rinderle, S., Reichert, M. *Change Patterns and Change Support Features in Process-Aware Information Systems*. CAiSE 2007: 574-588