2012 IEEE 36th International Conference on Computer Software and Applications Workshops

Simplifying the Design of Signature Workflow with Patterns

Hwai-Jung Hsu
Institute of Information Science
Academia Sinica
Taipei, Taiwan, R.O.C
hjhsu@iis.sinica.edu.tw

Ming-Wei Tsai and Feng-Jian Wang
Dept. of Computer Science
National Chiao Tung University
Hsinchu, Taiwan, R.O.C
{mwtsai, fjwang}@cs.nctu.edu.tw

Abstract—**Signatures responsible for authentication, authorization, etc, are important in many workflow applications. Most studies associated with signatures are focused on digital signatures only, and modeling of signature workflows is seldom studied. However, the dependencies between signatures can be complex, and thus modeling signature workflows becomes time consuming and error-prone. In this paper, we propose six patterns to simplify the design of signature workflows. All the patterns are described in BPMN and a case study is made to illustrate how to apply these patterns in construction of a workflow. A method for applying these patterns in development of workflows is sketched, and the advantages, simplifying the construction of a workflow with BPMN, are also revealed with the case study.**

Keywords- BPMN, workflow, signature workflow, signature pattern, transaction

I. INTRODUCTION

Workflow is the automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another or more for action, according to a set of procedural rules [17]. A signature on a document is a piece of data asserting that a named participant agrees with the content of the document at a workflow [7]. Signatures in different workflows are used for significant purposes such as authentication, authorization, etc [5][6][7]. On the other hand, signatures in enterprises are managed with a certain type of workflows called signature workflows. In [20], Wu describes signature workflows as document circulation, and extends an aggregate signature scheme in order to combine many signatures into one whether the document routing is sequential or parallel. Liu et al. propose an extended mathematical model based on workflow resolution for multi-signature workflows handlings [8].

Modeling signature workflows helps understanding of the signature systems, and reduces the communication costs among system developer. Mostly, the participants of the workflow sign the document after a transaction is accomplished. The dependencies and compensation behaviors between the transactions of the workflows make modeling signature workflow a complex task. For example, a document signed by a clerk might not take effect until his manager approves it. If the manager does not agree to the content of the signed document, the clerk revises the document and some relevant compensation processes must be executed along with the revision. Moreover, the flow of signatures in a signature workflow can be either sequential or parallel. The phenomenon indicates that the modeling signature workflows can be error-prone, highly time consuming and thus significant.

In this paper, we develop six patterns of signature workflows, and sketch a method to apply the patterns in construction of various workflow applications. All the patterns are described in Business Process Modeling Notation (BPMN) [12]. BPMN [12], defined by Object Management Group (OMG) [13], is a standardized graphical notation for business processes modeling, and is powerful in expressing advanced concepts necessary for modern workflow management such as transactions, multiple instance and compensations, etc [3][4][14][16][19]. The patterns provide the foundation of development of signature workflows. The workflow developers may choose one of the patterns as the entry point of signature workflow development, and devote their efforts on filling in the non-signature functions necessary by the enterprise applications.

The rest of the paper is organized as follows. In section II, signature workflow is featured, and the related works are described. In section III, the basic structure of a signature workflow is first introduced in BPMN, and all the six patterns are depicted on the basis of the basic structure. In section IV, a procedure to apply our patterns in development signature workflow is first sketched, and then a case study is established to demonstrate our methodology. Finally, the conclusion is made in section V, and the future works are discussed.

II. BACKGROUND

A. Signature Workflow

A signature workflow, the document circulation described in [20], involves multiple participants in several signature activities to sign on a document (*s-document*) to be associated with necessary authentication signatures finally. An s-document might be reviewed by one or more participants in sequential or in parallel. In this paper, we assume that the signature activity associated with one participant (an *ASA*) is atomic. In other words, all the tasks in one ASA are grouped as a transaction. Participants in a signature workflow can be

978-0-7695-4758-9/12 $26.00 © 2012 IEEE
DOI 10.1109/COMPSACW.2012.60

classified into (1) *originator*: the one who initialize the workflow and related documents, (2) *reviewer*: the one who review and authorize the documents, and (3) *approver*: the last reviewer of the signature process and the workflow is completed after the approver's approval.

In an ASA, each participant should basically complete the following tasks in sequence:

(1) Initializing an s-document or receiving the s-document from the other participants;

(2) If not the originator, verifying and reviewing the s-document from the former reviewer;

(3) Accomplishing the predefined works;

(4) Signing on the s-document; and

(5) Sending the s-document to the following ASAs.

An s-document is passed between ASAs in sequential or in parallel. When parallel signatures are necessary, the s-document is first duplicated and distributed to all or parts of the concurrent ASAs, and the duplicates are finally merged accordingly [5][6][7].

An ASA can be viewed as a kind of compensation-based transaction and the whole tasks in an ASA is considered as a kind of ACID transaction [18]. On the other hand, a signature workflow involving multiple participants is considered as a long-time process. If an s-document is rejected, the s-document must be revised, and the corresponding compensation activities are executed to keep the consistency of entire workflow.

B. Related Works

In the past, three methods were developed to facilitate the development of signature workflows. Shen et al. [22] analyze the structures of a signature workflow and construct a related model on the basis of Petri net [10]. The signature workflow is divided into five phases: the issuing draft, the department head, the chief of office, the countersign and the executive leadership. Stork et al. propose a hierarchical extension of Petri net to model the active signature workflows [15]. The approach allows the process designers to specify which activities may be refined and enables the participants of the signature workflow to modify these activities. In [9], Marchetti et al. define an agent-based XML extension called XFlowML to describe signature workflows. XFlowML is claimed more straightforward to the participants than graphs. However, both Petri net and XFlowML are not friendly for process designers, and modeling signature workflows with common workflow model is still necessary.

III. SIGNATURE WORKFLOW PATTERNS

To reduce the design work of signature workflows, we present a set of signature patterns in BPMN. The designers can adopt the patterns for various purposes of the signature workflows.

A. Basic Structure of Signature Workflow for a Participant

In this paper, a workflow signature pattern is composed of several Structures of Signature Workflow for a Participant (SP). All the SPs are derived from the Basic Structure (BSP). As Figure 1 illustrates, a BSP is composed of an ASA, several control structures, and a participant in the corresponding signature workflow. Region (A) represents the entry point of the BSP where a participant starts executing an ASA. If an ASA is completed, the BSP is directed to the path leading to region (B); otherwise, to region (C).

An ASA fails when (1) the participant cancels the execution, (2) the participant disagrees with the content of the s-document passing to him, and (3) an exception occurs. For the circumstance (3), the ASA first tries to correct the exception by restarting itself, and once the ASA reaches its maximal limitation of restarting, it fails.

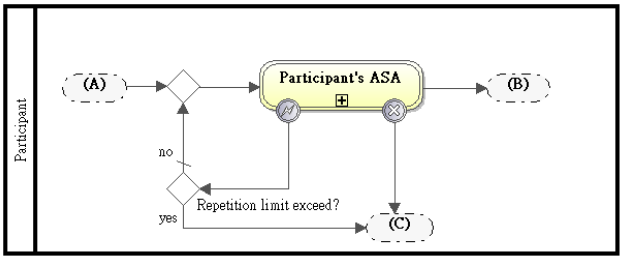

Figure 1 Basic Structure of Signature Workflow for a Participant, BSP

B. Sequential Signature Pattern

Sequential Signature Pattern is composed of sequential participants. The originator, the reviewer(s) and the approver simply signs the s-document one by one.

- **Intent:**

While the originator completes the corresponding ASA in a signature workflow, the s-document is sent to the reviewer(s) in sequence. If any of the reviewers disagrees with the s-document, the ASA(s) accomplished ahead are compensated.

- **Applicability:**

This pattern is applicable for the signature workflows where the s-document is reviewed and approved successively in a hierarchical organization.

- **Structure:**

Figure 2 illustrates the structure of Sequential Signature Pattern with (1) OSP: the originator's SP, (2) RSP_i: the *ith* reviewer's SP and (3) ASP: the approver's SP.

The signature workflow is initialized at the entry point of OSP. After the originator completes his ASA, the s-document is passed to the first reviewer (or the approver if none). On the other hand, if the originator's ASA fails, a cancellation message is sent to workflows concerning about the cancellation.

As for the *ith* reviewer (R_i) in the signature workflow, the s-document agreed by R_{i-1} (or the originator) is passed to R_i. After R_i completes his ASA, the s-document is passed to R_{i+1} (or the approver). If R_i's ASA fails, the s-document is returned to the originator and the signature workflow is restarted. The returning flow, notated as the *cancellation path*, includes a series of undo event(s) for R_{i-1}, R_{i-2}, ..., R_1 and the originator. Besides, the order of undo events is arranged in reverse order,

The study is sponsored by National Science Council.
(Project ID: 97-2221-E-009-063-MY3)

and the signature workflow succeeds after ASP is accomplished.

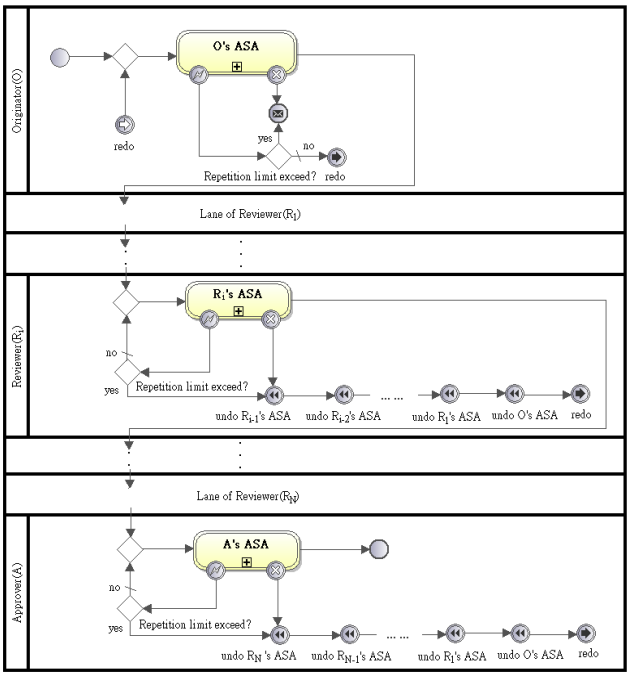

Figure 2 Sequential Signature Pattern

Example:

When a clerk of the bank completes a withdrawal job about large amounts of cash, a document associated to the withdrawal job should be reviewed by the assistant manager. After the agreement, the manager makes final approval.

C. Jump Signature Pattern

Sequential Signature Pattern can be furthered improved for emergencies. When a signature workflow is urgent, the flexibility to neglect some or all of the reviewers can be necessary, and the issue is considered in this pattern.

- **Intent:**

This pattern is derived from Sequential Signature pattern for emergent situations. To speed up the process, the s-document is allowed being reviewed by only key reviewer(s), i.e., non-critical reviewer(s) are skipped. If any key reviewer disagrees the s-document, only the ASAs of the originator and the previous key reviewer(s) are compensated.

- **Applicability:**

This pattern is useful for the signature workflows where the significance of each job is various among different cases.

- **Structure:**

As Figure 3 illustrates, OSP and each RSP_i (besids the last RSP) contains an extra exclusive gateway behind its ASA to check whether the next reviewer is skipped. If yes, the s-document is sent to another check point to see whether its following reviewer is also skipped. However, the ASP is allowed to be skipped. Besides, when failure occurs, only the ASAs of the previous key reviewer(s) and the originator are compensated.

- **Examples:**

An urgent tender case is applied by some salesman. However, the review process by the assistant managers is time consuming, and the application cannot be approved in time if it is reviewed by all the assistant managers as usual. According to the policy for emergent cases, the salesman may skip the assistant managers and sends the application to the manager for approval directly to catch up the deadline.

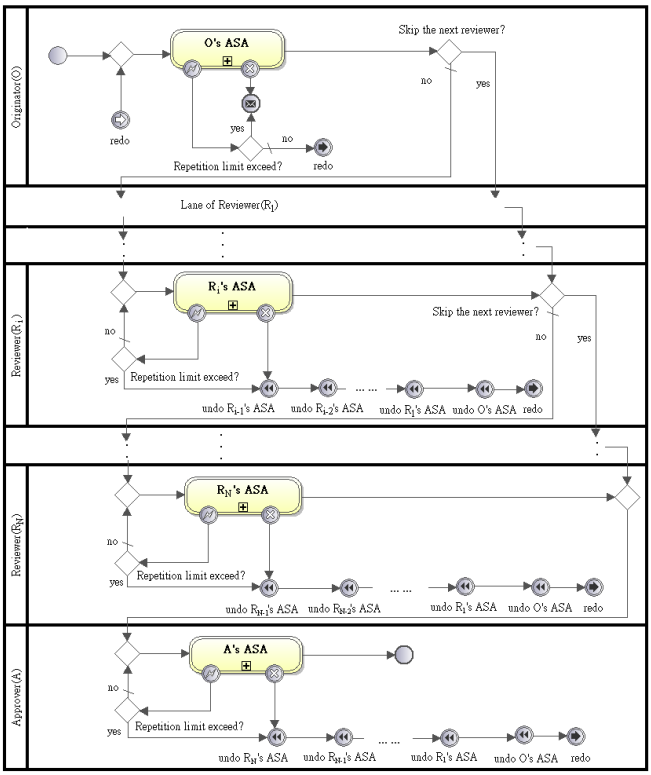

Figure 3 Jump Signature Pattern

D. Return Signature Pattern

While failure occurs, to compensate all the previous works and to start over might be costly. To solve this problem, a rejected signature workflow is enabled being recovered from the middle of its process to reduce the penalty of failure.

- **Intent:**

In the pattern, when a reviewer disagrees with the s-document, the s-document can be sent back to the originator or one of the previous reviewer(s). Only the ASAs between the disagreed reviewer and the returned target are compensated. The returned reviewer may correct the s-document and the costs for total roll-back are saved.

- **Applicability:**

This pattern is applied for the signature workflows with many reviewers and costly cancellation paths.

- **Structure:**

As Figure 4 illustrates, for each RSP, an exclusive gateway is inserted between two continuous undo events in the cancellation path. The extra exclusive gateways are responsible for deciding where to send the rejected s-document. For example, When R_i disagrees the s-document and decides to return the s-documet to some previous reviewer, R_j. A pair of redo events named "redo R_j" leads the signature workflow to RSP_j, and before the s-document is sent back, the ASAs completed between R_j and R_i are compensated in reverse order.

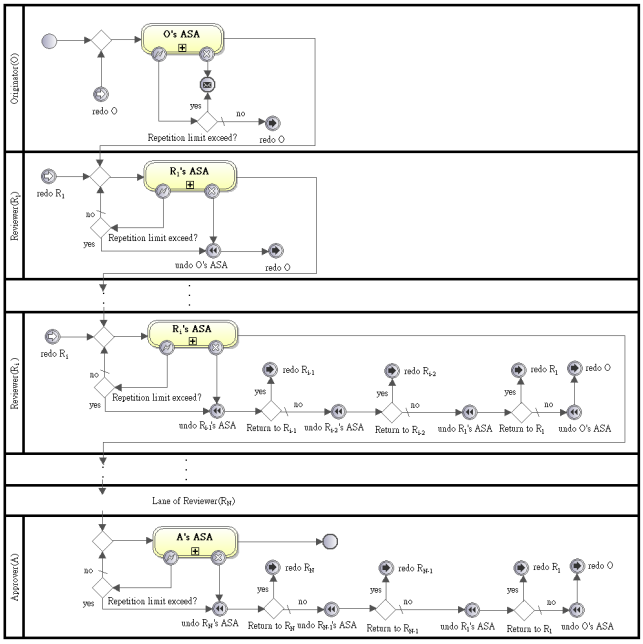

Figure 4 The Return Signature Pattern

- **Examples:**

Ten assistant managers are expected to review the examination of annual budgets. To save the cancellation costs, any of the assistant managers can be chosen to revise the document when the manager disapproves the annual budget.

E. Static Countersignature Pattern

Countersignature in which the reviewers sign on a document in parallel is common in modern enterprise, and three corresponding patterns are introduced. In such patterns, after the originator completes his ASA, the s-document is duplicated into several copies, notated as *mv-s-documents*, and is sent to reviewers in parallel. Besides, if any of the ASAs fails, the other ASAs are forced to be compensated if completed, or rolled back otherwise. The exclusive gateway at the very beginning of the ASP checks whether any ASA fails and the originator would have to compensate his ASA and redo the whole process if any. Otherwise, all the mv-s-documents are merged into one single s-document to start the final approval.

- **Intent:**

All the parallel reviewers are statically decided before the workflow enactment, and cannot be removed during the signature workflow.

- **Applicability:**

This pattern is applicable for the signature workflows with essential reviewers approving the s-document simultaneously.

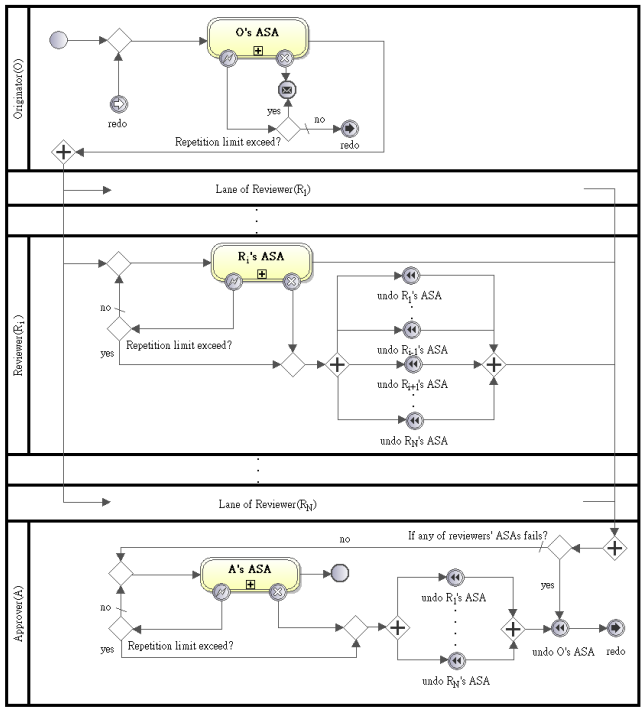

Figure 5 Static Countersignature Pattern

- **Structure:**

Figure 5 shows the structure of this pattern. OSP contains a parallel gateway after its ASA to dispatch the mv-s-documents to the corresponding reviewers. In an RSP, when the ASA fails, undo events are thrown to the rest reviewers simultaneously for compensation. As for the ASP, three structures are included as the following description:

(1) A parallel gateway at the entry point of ASP merges the input mv-s-documents.

(2) An exclusive gateway after the parallel gateway checks whether any reviewer disagrees the mv-s-document.

(3) For cancellation or disapproval, after all the reviewers' ASAs are compensated simultaneously, an undo event is sent to OSP to restart the process.

- **Examples:**

The human resource and the accounting department in the company are necessary departments to agree the leaving application. When an employee leaves, the human resource and the accounting department sign the leaving applications before the manager approves the application.

F. Dynamic Countersignature Pattern

In the following pattern, the originator is allowed to select the reviewers from a pre-defined reviewer list, and only make the selected reviewers to review the s-document.

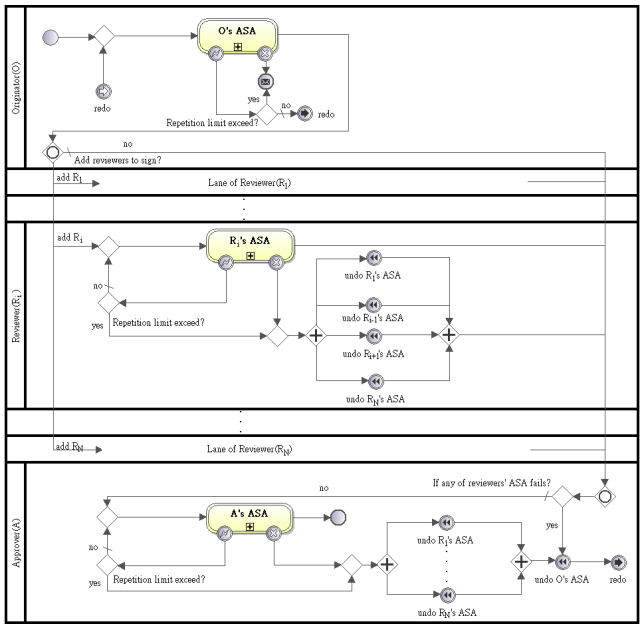

Figure 6 Dynamic Countersignature Pattern

- **Intent:**

Dynamic Countersignature pattern, an extension of Static Countersignature pattern, allows the originator to determine the necessary reviewer(s) after completing its ASA.

- **Applicability:**

During design time, the designers put all possible reviewers in the parallel signature workflow.

- **Structure:**

As Figure 6 illustrates, after the originator completes his ASA, an inclusive gateway in OSP duplicates the s-document and dispatches the mv-s-documents to the necessary reviewers. The conditions describing the necessary reviewers are decided by the originator before the inclusive gateway is fired. The input mv-s-documents of the ASP are merged by another inclusive gateway after all the necessary reviewers finish their review. The flow connecting the two inclusive gateways allows the originator passing the s-document to the approver directly.

- **Examples:**

When the cost of an advertising application exceeds the budget cap, the sales department may determine either or both of the auditing and the accounting department should review the application before the manager. The application may also be sent to the manager directly if the cost is low.

G. Additional Countersignature Pattern

The system may allow the reviewer to invite other reviewers to join the review process dynamically.

- **Intent:**

Additional Countersignature pattern is an extension of Dynamic Countersignature Pattern. The reviewers not being selected by the originator can be involved into the review process by any of the selected reviewers. However, each reviewer performs the review work at most once.

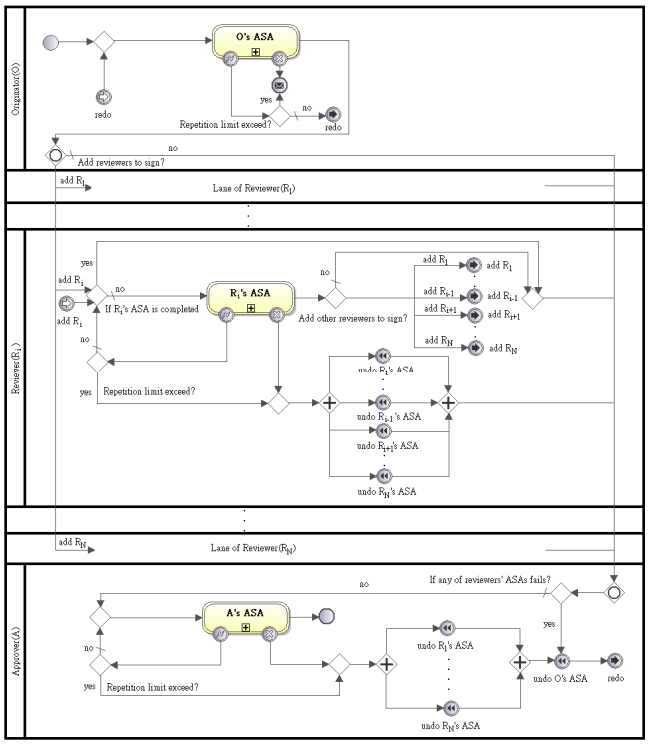

Figure 7 Additional Countersignature Pattern

- **Applicability:**

This pattern is applicable for the signature workflows where both the originator and the reviewer(s) are allowed to determine the necessary reviewer(s).

- **Structure:**

Additional Countersignature Pattern is extended from Dynamic Countersignature Pattern as figure 7 illustrates with three additional control structures to each RSP.

(1) A condition is added to the exclusive gateway in front of R_i's ASA to check to avoid redundant approval.

(2) An exclusive gateway after R_i's ASA is added to allow R_i to involve another reviewer.

(3) A redo event including generator and catcher named "add R_j" is added to involve R_j to review the mv-s-document.

- **Examples:**

The HR department, the accounting department, and the risk management department are the potential reviewers of an investment application. The investment department may include the HR and the accounting department for reviewing an investment application. If HR department cannot distinguish whether the application may bring damages, it can request the risk management department as an additional reviewer.

IV. How to Apply the Patterns

A. The Guideline

Here we sketch the procedure to apply the patterns proposed above to develop a signature workflow: (1) Choose the corresponding signature pattern according to the requirements. (2) Adjust the number of RSPs in the chosen signature pattern according to the real organizations manually. (3) Separately design the ASAs for the originator, the approver and each reviewer in the signature workflow. (4) Fill all the ASAs into the adjusted pattern for the corresponding participants.

B. Case Study

To demonstrate our methodology, a resignation process is established as an example in this section. Figure 8 shows the final process schema of the resignation process. The process designer simply adopts the Sequential Signature pattern as the skeleton of the process and modifies the structure of the pattern for the participants, the employee, the manager, and the boss. Next, he designs the ASA for each participant as the small figures in each participant's swim lane. Since all our patterns are based on BPMN, our methodology can be adopted by any BPMN-based workflow design tools or WfMS.

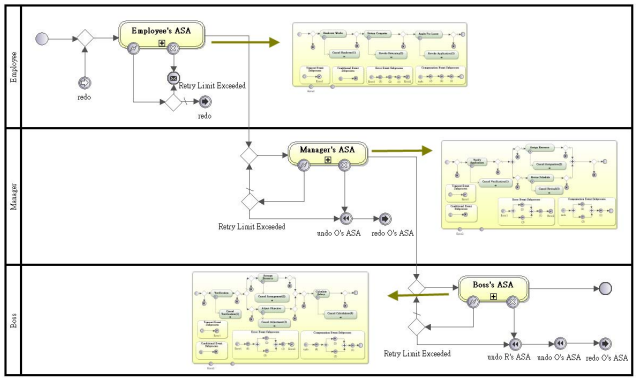

Figure 8 The Final Diagram of the Resignation Process

V. Conclusion and Future Works

In this paper, on the basis of BPMN, six signature workflow patterns are proposed. The patterns are useful in reducing the design efforts of signature workflows, and represent the basic workflow structures for various signature semantics. However, the patterns can be further developed as following. (1) The categories of signature patterns may be enriched, and more dynamic problems can be considered in countersignature patterns. (2) The patterns can be further combined to match the sophisticated requirement such as jump countersignature, etc. Besides, the specification need be transformed into WS-BPEL [11] for execution. In addition, since each ASA should be considered as a transaction. A methodology helping the designers to transform a normal business process into a transaction can be developed.

References

[1] Dijkman, R. M., Dumas, M. and Ouyang, C., "Formal Semantics and Analysis of BPMN Process Models," Information and Software Technology, 2009.

[2] Gamma, E., Johnson, R., Vlissides, J. and et al., "Design Patterns: Elements of Reusable Object-Oriented Software," Addison-Wesley, 1994.

[3] Kous, K., "Comparative Analysis Versions of BPMN and its Support with Control-flow Patterns," MIPRO, 2010 Proceedings of the 33rd International Convention, 2010, pp. pp.315-319.

[4] Lerner, B. S., Christov, S., Osterweil, L. J. and et al., "Exception Handling Patterns for Process Modeling," IEEE Transactions on Software Engineering, vol. 36, no. 2, pp. 162-183, Mar-Apr, 2010.

[5] Leung, K. R.P.H. and Hui, L. C.K., "Signature Management in Workflow Systems," Proceedings. Twenty-Third Annual International Computer Software and Applications Conference, pp. 424-429, 1999.

[6] Leung, K. R.P.H. and Hui L. C.K., "Multiple Signature Handling in Workflow Systems," Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, 2000.

[7] Leung, K. R.P.H. and Hui, L. C.K., "Handling Signature Purposes in Workflow Systems," Journal of Systems and Software, vol. 55, no. 3, pp. 245-259, Jan, 2001.

[8] Liu, J.B., Qiu, Z.L., Zhou, H. and et al., "A Schema for Multi-Signature Handling Based on Workflow Model," 2009 Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, pp. 378-381, Sep. 2009.

[9] Marchetti, A., Tesconi, M. and Minutoli, S., "XFlow: An XML-Based Document-Centric Workflow," Web Information Systems Engineering - Wise 2005, vol. 3806, pp. 290-303, 2005.

[10] Murata, T., "Petri Nets - Properties, Analysis and Applications," Proceedings of the IEEE, vol. 77, no. 4, pp. 541-580, Apr, 1989.

[11] OASIS, "Web Services Business Process Execution Language Version 2.0," OASIS Standard, 2007.

[12] Object Management Group, "Business Process Model and Notation Version 2.0," OMG Final Adopted Specification. Object Management Group, January. 2011.

[13] Object Management Group, http://www.omg.org/

[14] Rosa, M. L., Reijers, H. A., Van Der Aalst, W. M.P. and et al., "APROMORE: An Advanced Process Model Repository," 2010.

[15] Stork, D. G. and Glabbeek, R. V., "Token-Controlled Place Refinement in Hierarchical Petri Nets with Application to Active Document Workflow," Applications and Theory of Petri Nets 2002, vol. 2360, pp. 394-413, 2002.

[16] Takemura, T., "Formal Semantics and Verification of BPMN Transaction and Compensation," Los Alamitos: IEEE Computer Society, 2008.

[17] The Workflow Management Coalition, "Terminology & Glossary," Management, no. 3. pp. 1-65, 1999.

[18] Wang, T., Vonk, J., Kratz, B. and et al., "A Survey on the History of Transaction Management: From Flat to Grid Transactions," Distributed and Parallel Databases, vol. 23, no. 3, pp. 235-270, Jun, 2008.

[19] Wohed, P., Van Der Aalst, W. M.P., Dumas, M. and et al., "Pattern-based Analysis of BPMN," Technology, 2005.

[20] Wu, Y., "Efficient Authentication of Electronic Document Workflow," Information Security and Cryptology, Proceedings, vol. 3822, pp. 101-112, 2005.

[21] Ye, J.H., Sun, S.X., Wen, L. and et al., "Transformation of BPMN to YAWL," 2008 International Conference on Computer Science and Software Engineering, pp. 354-359, 2008.

[22] Shen, Z. and Wang, H., "Research on the Application of Workflow Technology in OA System," in Computer Science and Information Technology, 2010 3rd IEEE International Conference on, 2010, vol. 7, pp. 519-52