Goal-based and Risk-based Creation of Adaptive Workflow Processes

Dr. N.C. Narendra

Software Engineering Process Group (SEPG)
Hewlett-Packard India Software Operations Ltd.
29, Cunningham Road
Bangalore - 560 052, INDIA
ncnaren@india.hp.com

Abstract

Adaptive Workflow Management is being increasingly looked upon as one of the cures for the problems plaguing business organizations, which are being forced to continually restructure their business processes in response to changing market conditions. In such a situation, techniques for designing adaptive workflow processes from business goals become essential.

In this paper, we present a knowledge-based technique, incorporating goals and risks, to derive portions of workflow process schemas from user-defined goals and scenarios. We also present a technique for composing these portions to create a complete workflow schema. We also show that our technique can create adaptive workflow processes. Another contribution of this paper, is that it also provides a way to model risks into the workflow process.

1. Introduction

Adaptive Workflow Management is being increasingly looked upon as one of the cures for the problems plaguing business organizations, which are being forced to continually restructure their business processes in response to changing market situations. In such a situation, techniques for designing adaptive workflow processes from (continually changing) business goals become essential.

In this paper, we present a knowledge-based technique, incorporating goals and risks, to derive portions of workflow process schemas from user-defined goals and scenarios. We also present a technique for composing these portions to create a complete workflow schema. We also show that our technique can create adaptive workflow processes. Another contribution of this paper, is that it also provides a way to model risks into the workflow process.

Our technique has the following steps [Keung]: elicitation of goals from customers, derivation of process-related goals from customer goals, scenarios for meeting the process-related goals, refining the scenarios into their constituents, and integration of all the refined scenarios (along with the risks) to determine the overall workflow schema.

This paper is organized as follows. The next section provides some preliminary definitions. Section 3 describes our technique, and the paper concludes in Section 4.

2. Preliminaries

2.1 Goals and Scenarios

The definition of a *goal* is obvious: it is an end that needs to be achieved, for the sake of meeting some customer requirement. For example, for an insurance company, one of the goals could be to process all insurance claims within 2 weeks. Hence goals, which are to be derived from the customer's point of view, impose certain constraints within which the insurance company has to define its business processes.

Goals can also be classified into customer-specified and process-related. For example, the requirement that all insurance claims should be processed within 2 weeks, could be a customer-specified requirement. From such a goal, process-related goals can be derived, viz., that the insurance investigator should complete his/her job within 1 week, that insurance clerks should be able to process all customer requests within a day, and so on. A workflow schema should then be derived that satisfies the process-related goals.

A *scenario* is a description of a possible interaction between the customer, and different departments in the insurance company, for the purpose of fulfilling a part of the process-related goals. For example, recording a customer's insurance claims, is a scenario for an insurance company. Another scenario could be the verification of the customer's claim by the insurance investigator, and a third scenario could be the interaction with the customer, to notify him/her as to the acceptance/rejection of the insurance claim.

Hence, while developing workflow schemas, we need to first determine the customer's goals, then develop the appropriate process-related goals, develop the scenarios that satisfy the goals, and then appropriately refine and compose the scenarios to develop the workflow schema that will satisfy the goals.

2.2 *Risk Modeling*

Risks are inherent in any business enterprise. Hence it is essential that business processes take into account potential risks, and that these risks be suitably modeled therein. This would involve determining and modeling the following [Riskit]:

- the *risk factor*, i.e., the characteristic that affects the probability of a risk event occurring
- the *risk event*, which represents the occurrence of a negative incident - or a discovery of information that reveals negative circumstances
- the *risk outcome*, which describes the state of the business after the risk has materialized
- the *risk consequences*, which represents the state of the business after corrective reaction has been taken
- the *risk effect*, which represents the impact of the risk on the customer and process-related goals
- the *utility loss*, which captures the severity of the loss to the business

For our insurance company, some risks could be:

- non-availability of insurance investigator to check out the accident claim
- lack of timely and accurate information from police authorities
- lack of timely legal advice from company's legal experts

The advantage of modeling risks in this fashion, is that we can is to thoroughly analyze each risk and determine its utility loss. This will assist in adopting contingency plans for addressing the risks.

3. Our Approach

In this section, we describe our technique in detail.

3.1 From Customer Goals to Process-related Goals

The first step is to get customers/users to specify the goals that the workflow schema should satisfy, from their point of view. For our (highly simplified and contrived) insurance company example, it could be something like: "I want you to get back to me about my insurance claim within 2 weeks." Hence the workflow schema should be designed so that it should deliver its results to the customer within 2 weeks.

These goals should then be translated into process-related goals, by asking questions such as "What process goals should the insurance investigator satisfy in order to meet the customers' goals?" Many such goals may need to be derived [Keung], and some of them may even contradict each other. Some goals are interdependent, and some could also be complementary. Furthermore, we may have to deal with the fact that different goals get different priorities.

Hence, in this step, it is essential that all these contradictions/conflicts be resolved, so that our insurance company can derive the appropriate process-related goals from the customer goals.

3.2 From Process-related Goals to Scenarios

The next step is to derive scenarios from the goals.

The scenario derivation should follow the following steps:

- A repository of old workflow processes and schemas should be maintained, as described in the Process Handbook [PH] project. This repository will have information on the different scenarios and their related processes, and how they relate to each other and the goals that they meet.

  This can also be accomplished by means of Knowledge-based techniques such as Case-Based Reasoning (CBR); this is possible since each scenario and its accompanying process, can be represented as a "workflow case", which is being investigated as a candidate for reuse.

- This repository can be used to select the scenarios that meet process-related goals similar to that derived here.
- The selected scenarios can then be suitably modified to create the scenarios needed for the current set of customer goals
- If there are no applicable scenarios in the repository, then they will have to be developed from scratch

Our scenario approach is similar to that in [Aalst2], where Message Sequence Charts are used to specify workflow processes as interaction structures, and Petri Nets are used to model the workflow processes themselves.

Example scenarios for our insurance company are depicted in Fig.1.

A brief explanation of Fig.1 is in order. In Scenario #1, the customer applies for an accident insurance claim, where she will interact with the Clerk, who will in turn log her request into the company's computer system (say, an MIS). In Scenario #2, the system will prompt the insurance Investigator to investigate the insurance claim with the police. In Scenario #2, the claim is also investigated for validity with the company's lawyer. In both Scenario #2 and Scenario #3, the results of the investigation are fed into the computer system. Finally, in Scenario #4, the results of the insurance claim are reported to the customer, and the customer's feedback on the service (if any) are also recorded into the system.

One point to note in this example, is the absence of iteration; that is, process activities are not repeated. However, scenarios can also represent iteration. One easy way to represent iteration, i.e., loops, in scenarios, is to use specially shaded arrows for each loop.

However, this naturally brings us to the question: how to represent sub-scenarios within scenarios, especially sub-loops within loops? For the sake of clarity, scenarios can be refined into sub-scenarios in a manner similar to that used to decompose Message Sequence Charts (MSCs) [MSC]. The sub-scenario represents a decomposition of the parent scenario without affecting the parent scenario's observable behavior.

Generally, the rule to follow here, would be to ensure that each scenario represented as in Fig. 1, will contain only one level of iteration. Any second-level iteration, will have to be represented separately, as a sub-scenario.

3.3 Refining Scenarios - Derivation of Role Activity Diagrams

One thing to note, is that the scenarios are "role-based", i.e., they depict interactions between people or computers playing particular roles as part of the workflow schema. However, typical workflow representations are "activity-based", i.e., they depict the activities being performed, with only peripheral reference to the agents performing that particular activity.

In order to address this mismatch, we use Role Activity Diagrams (RADs) [Kueng]. RADs are a powerful way of representing both activities and roles in the same diagram, so as to depict the relation between them.

Scenarios can be pairwise composed into RADs, using one of the following three operations, as appropriate:

- *Sequential composition*, where scenarios are combined sequentially, such that the output from the first scenario is the input to the second scenario; this also includes the case of sub-scenarios, which will have to be embedded into their respective parents via sequential composition
- *Choice composition*, where either scenario can be executed, depending on particular predicate values
- *Parallel composition*, where both scenarios should be executed together, in parallel

The RAD for the scenarios presented in Fig.1, is presented in Fig.2.

Once Fig.1 has been understood, Fig.2 is quite straightforward. Most scenario composition in our example is sequential, except for the insurance Investigator interacting with the Police and Lawyer, which can be parallelized. The resulting parallel representation is also shown in Fig.2.

3.4 Composing RADs into Workflow Schema

Composing RADs into workflow schemas is quite straightforward. By following the activities from top to bottom in the RAD, each transition (represented as an arrow in the RAD) is converted into an activity. Since sequential, choice and parallel composition are already represented in the RAD, this will make it easy to depict these features directly in the workflow schema.

The workflow schema resulting from composing the RADs in Fig.2 is presented in Fig.3. When RADs are converted into workflow schema, care is to be taken to ensure that organizational "white-space", i.e., inefficiencies arising out of interface mismatches between scenarios, are minimized [Gruhn]. Otherwise, the resulting workflow schema will be sub-optimal.

3.5 Modeling and Managing Risk

A complete development of the workflow schema is not possible without analyzing and modeling the risks involved in executing the process steps. Basically, a risk is an occurrence that could jeopardize the successful operation of any of the activities; hence it is highly essential that the risks be identified and modeled in the workflow schema, in the form of "what-if" analyses.

From a workflow perspective, a risk is a possible negation of one of the preconditions or postconditions of an activity in the workflow schema. Hence, modeling risks in a workflow schema, would involve inserting additional activities for mitigating the risks.

Risks can materialize in three different ways, with respect to any activity:

- Just before an activity is supposed to begin
- While it is executing
- Long before it is supposed to begin; however, it would be better to address the risk right away, since it may take time to address the risk

Hence it is essential to consider risks from the aforementioned three viewpoints, and model them accordingly. For this purpose, we adapt the Riskit methodology presented in [Riskit] and already discussed in Section 2.2.

Our basic approach to adapting Riskit in our technique, is the following:

- determine the risk factors, risk events, risk outcomes and risk consequences of any activity
- model the risk events as negations of preconditions of the activity
- model the risk outcomes as negations of postconditions of the activity

- model the risk consequences, risk effect and utility loss as alternative paths in the workflow schema, in order to deal with the risk

The complete workflow schema, incorporating a few risks, is given in Fig.4 below. Please note that Fig.4 depicts only successful risk mitigation activities (not described in detail for the sake of simplicity); for the sake of brevity, the case when these activities are not successful and the workflow execution may have to be restarted, is not shown in the figure.

3.6 Adaptivity

In this section, we show how adaptivity is built into our technique; i.e. we show how our approach builds adaptive workflows.

Basically, adaptivity refers to the ability of a workflow management system to respond quickly and effectively to changes from (external or internal) sources, while maintaining customer requirements. Hence there are three levels of adaptivity [Narendra, Han].

- *Changes in workflow processes:* these could be necessitated by operational reasons, such as the need to improve the efficiency of existing processes. Generally, these are easiest to implement
- *Changes in workflow schema:* these could be necessitated by changing business environments, or changes in existing business relationships (for example, our insurance company may decide to outsource some of its work to an external agency). These require major re-engineering of the existing workflow schema itself
- *Changes in goals:* these could be necessitated by changing external situations. For example, our insurance company could discover that its competitors are able to process insurance claims within 10 days instead of 2 weeks, and this could become the new goal of our insurance company in order to attract new customers (and retain the old ones!). This is typically the most disruptive change.

We can see that our technique is able to satisfy changes at all the three levels of adaptivity, since our approach mirrors these levels - goals, scenarios, and workflow schema. Hence our technique will be able to derive workflows that are easily adaptive.

4. Conclusions and Future Work

In this paper, we have presented a knowledge-based technique, incorporating goals and risks, for deriving workflow process schemas from customer-specified goals. We have also shown that our technique can derive adaptive workflows.

Our work throws up several important and interesting topics for future work:

1. *How to analyze workflow processes, to ensure that they do meet customer requirements?* Some work in this direction, especially in the area of structural analysis of Petri-Net Based Workflows, has been accomplished in [Aalst]. However, the larger problem is to analyze adaptive workflows, i.e., how to analyze a workflow process whose detailed picture becomes known only after execution [Han].
2. *How to derive and analyze distributed workflow processes?* In other words, how do we adapt our technique to derive workflow processes that can be distributed across several organizations? A beginning has been made in the MENTOR project [MENTOR], where techniques for centralized workflow specification and distributed workflow execution have been developed; adapting those techniques to our work, would be fruitful. An excellent beginning in this direction has also been made in [Aalst2], although more work needs to be done.
3. *Automation and incorporation of Agent Technology* - since our technique involves manually determining scenarios, and converting them into RADs and workflow schema, it would be useful to look at how much this can be automated. In the case of distributed workflows, the incorporation of Software Agent technology [BT], could prove to be fruitful. We could envisage a situation where each agent will define its scenario, and another agent will put them all together to form the RAD and workflow schema. This community of agents can then be entrusted with the task of continuously monitoring the workflow execution, and adapting the workflows (as in Section 3.6) in case of changes.

5. Acknowledgments

The author wishes to thank Indradeb P. Pal and C. Laxmi Kutty for useful comments on an earlier draft of this paper.

6. References

[PH] A. Bernstein, "The Product WorkBench: An Environment for the Mass Customization of Production-Processes," WITS '98, Helsinki, Finland, 1998; available from http://web.mit.edu/avi/www/publ.htm

[Aalst] W.M.P. van der Aalst, "Diagnosing Workflow Processes Using Woflan," available from http://wwwis.win.tue.nl/~wsinwa/index.html

[Aalst2] W.M.P. van der Aalst, "Interorganizational Workflows: An Approach Based on Message Sequence Charts and Petri Nets", available from http://wwwis.win.tue.nl/~wsinwa/index.html

[Han] Y. Han, A. Sheth and C. Bussler, "A Taxonomy of Adaptive Workflow Management," Workshop of the 1998 ACM Conference on Computer Supported Cooperative Work, Seattle, Washington, USA, November 1998

[MENTOR] P. Muth, D. Wodtke, J. Weissenfels, A.K. Dittrich and G. Weikum, "From Centralized Workflow Specification to Distributed Workflow Execution," available from http://www-dbs.cs.uni-sb.de/~mlite

[Keung] P. Keung and P. Kawalek, "Goal-based Business Process Creation and Evaluation," Business Process Management Journal Vol. 3, No. 1, 1997, pp. 17-38, available from http://www2-iiuf.unifr.ch/is/peter/peter.html

[Riskit] J. Kontio, D. Getto and D. Landes, "Experiences in improving risk management processes using the concepts of the Riskit method," available from http://mordor.cs.hut.fi/~jkontio/fse6-rm.pdf

[BT] P.D. O'Brien and M.E. Weigand, "Agent-based Process Management," available from http://www.labs.bt.com/projects/ibsr/papers/aew/ker.ps.gz

[Gruhn] V. Gruhn and U. Wellen, "Process Landscaping: Modelling Complex Business Processes," available from http://www.nectar.org/journal/03/012.htm

[Narendra] N.C. Narendra, "Adaptive Workflow Management - An Integrated Approach and System Architecture," ACM Symposium on Applied Computing, Special Track on Coordination Models, Languages and Applications, 2000, to appear

[MSC] E. Rudolph, P. Graubmann and J. Grabowski, "Tutorial on Message Sequence Charts,", available from ftp://ftp.win.tue.nl/pub/techreports/sjouke/msctutorial.ps.Z and http://www.win.tue.nl/cs/fm/sjouke/msc.html

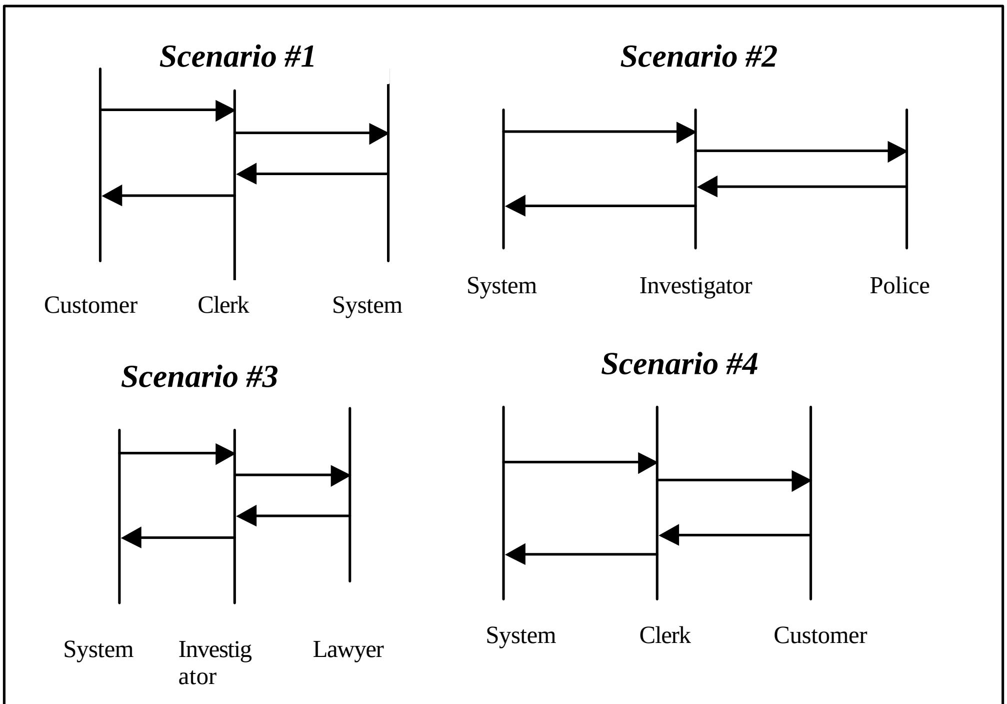
Scenario #1
Customer
Clerk
System
Scenario #2
System
Investigator
Police
Scenario #3
System
Investig
ator
Lawyer
Scenario #4
System
Clerk
Customer

Fig. 1

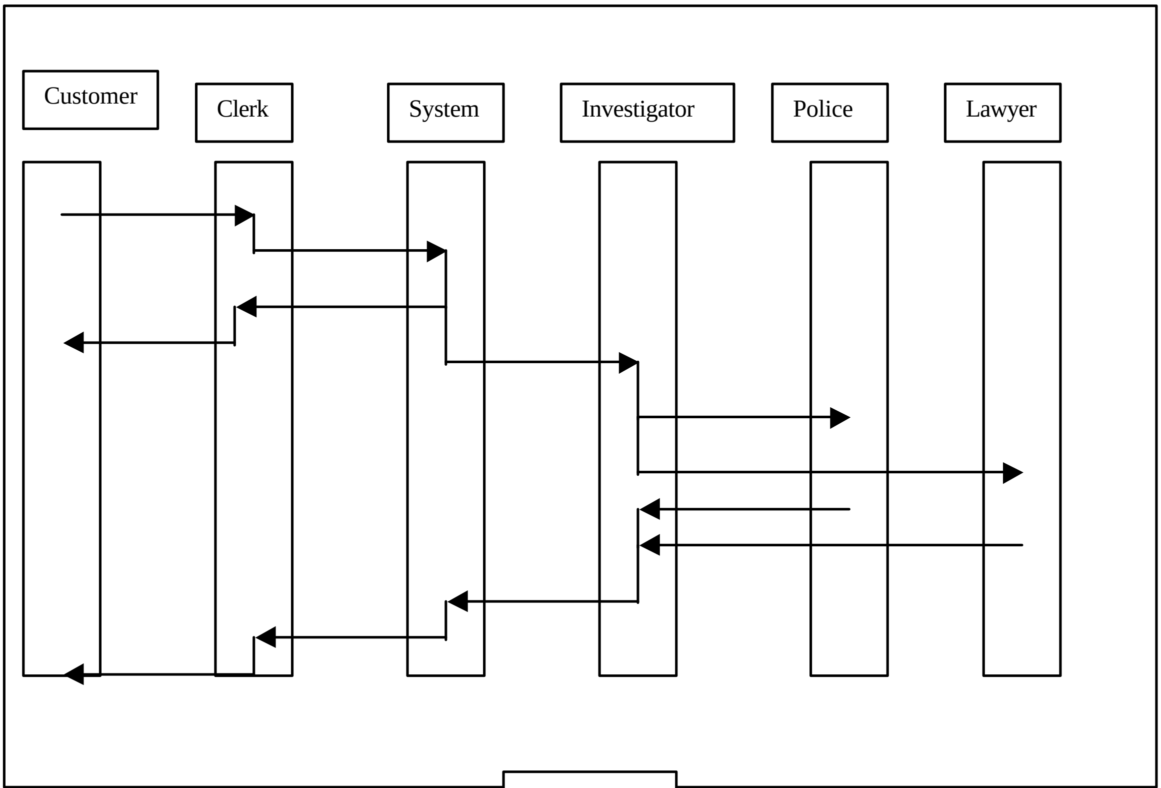
Customer
Clerk
System
Investigator
Police
Lawyer

Fig. 2

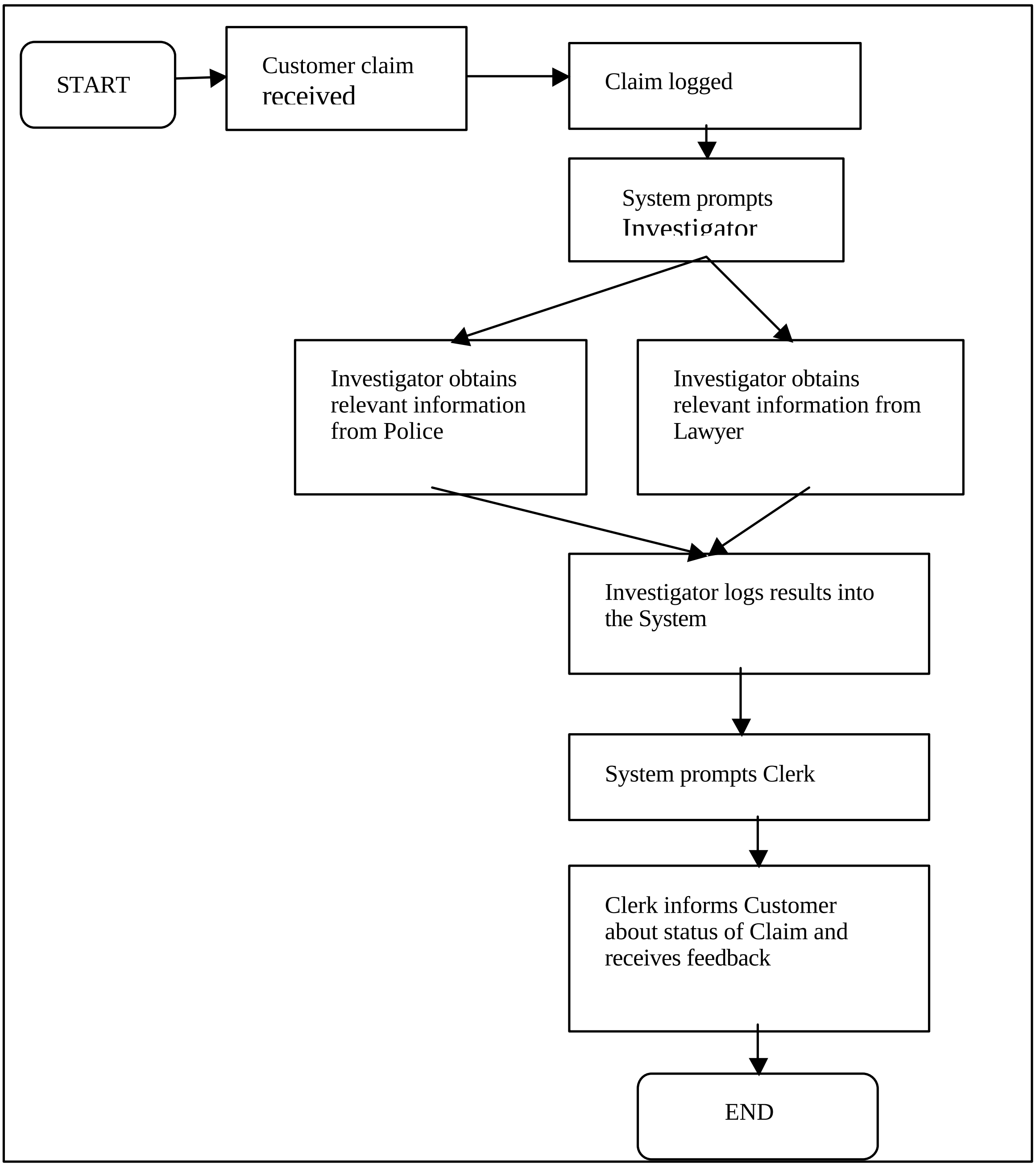
START
Customer claim received
Claim logged
System prompts Investigator
Investigator obtains relevant information from Police
Investigator obtains relevant information from Lawyer
Investigator logs results into the System
System prompts Clerk
Clerk informs Customer about status of Claim and receives feedback
END

Fig. 3

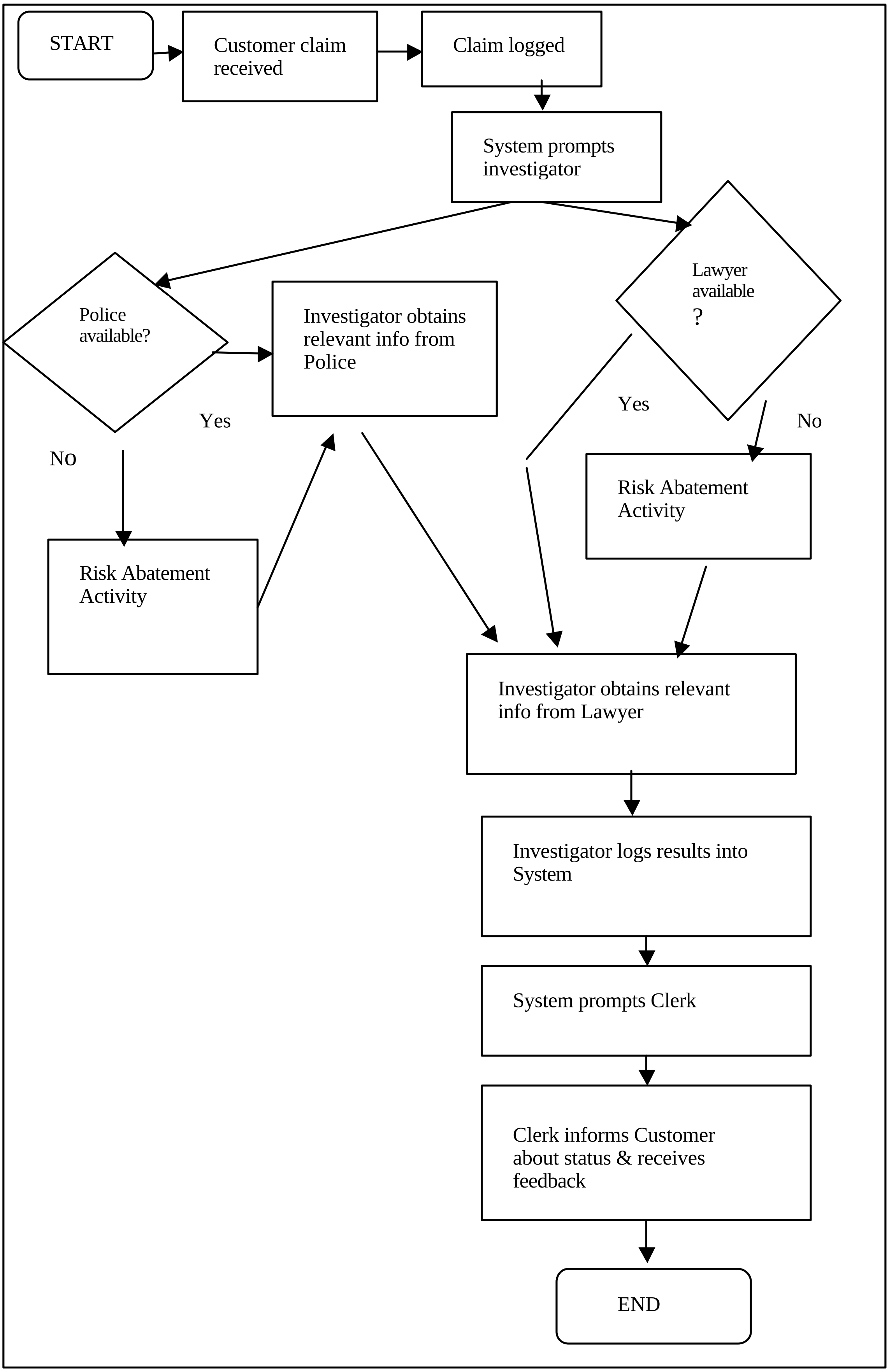
START
Customer claim received
Claim logged
System prompts investigator
Police available?
Investigator obtains relevant info from Police
Lawyer available ?
Yes
No
Yes
No
Risk Abatement Activity
Risk Abatement Activity
Investigator obtains relevant info from Lawyer
Investigator logs results into System
System prompts Clerk
Clerk informs Customer about status & receives feedback
END

Fig. 4