Management Studies, December 2015, Vol. 3, No. 11-12, 273-280
doi: 10.17265/2328-2185/2015.1112.002

Generation of Interactive Questionnaires Using YAWL-based Workflow Models

Raimond Wüst, Stephan Bütikofer, Jürgen Spielberger
Zurich University of Applied Sciences (ZHAW), Winterthur, Switzerland
Jörg Sigrist
AM-TEC Switzerland AG (AM-TEC), Rafz, Switzerland

A concept is introduced in this article which has strong practical impact for computer aided system configuration. System configuration is a cumbersome and fault sensitive task while setting up systems in a broad range of business applications like ERP (enterprise resource planning) and other workflow-systems. Given a generic process or workflow model in YAWL-notation (yet another workflow language) or any other process modeling language like business process model and notation or WFMC (workflow management coalition), it could be stated that, by using a set of reduction rules as introduced, it is possible to generate a hierarchically structured tree of sub graphs of the workflow graph-representation. According to the notation used, authors call these sub graphs facts. The tree structure of the graph-representation on one hand and the logical relation between the branches and leafs of the tree on the other can be utilized to create a set of constraints and dependencies among the single facts. Some researchers showed that the nested branches can be associated to (predefined) questions with respect to the configuration of a workflow management system, for instance an ERP-application. They presented an algorithm which dynamically sorts the questions and answers in a maximum efficient configuration path, while working through the corresponding questionnaire. By combining the different elements as facts, constraints on questions, and configuration space, it is thus possible to algorithmically generate the efficient structured and interactive questionnaire for the configuration of workflow systems and algorithmically check the consistency (dead lock free, free of synchronization structural conflict) of the underlying workflow model. The concept was tested in the prototype of the interactive questionnaire for configuration of the web-service based ERP-Application Posity.

Keywords: workflow modeling, interactive questionnaire, algorithmic process analysis, workflow graph reduction, formal workflow consistence, ERP-system setup, system automation

Raimond Wüst, Ph.D., Zurich University of Applied Sciences (ZHAW), Institute for Data Analysis and Process Design (IDP), Rosenstr, 3, 8401 Winterthur, Switzerland.

Stephan Bütikofer, Ph.D., Zurich University of Applied Sciences (ZHAW), Institute for Data Analysis and Process Design (IDP), Rosenstr, 3, 8401 Winterthur, Switzerland.

Jürgen Spielberger, M.Sc., Zurich University of Applied Sciences (ZHAW), Institute for Applied Information Technology (InIT), Obere Kirchgasse 2/Steinberggasse 12/14, CH-8401 Winterthur, Switzerland.

Jörg Sigrist, M.Sc., AM-TEC Switzerland AG (AM-TEC), Bahnhofstrasse 18, CH-8197 Rafz, Switzerland.

Correspondence concerning this article should be addressed to Raimond Wüst, Zurich University of Applied Sciences (ZHAW), Institute for Data Analysis and Process Design (IDP), Rosenstr, 3, 8401 Winterthur, Switzerland. E-mail: Raimond.wuest@zhaw.ch.

Introduction

Many companies make the experience, that setting up or migrating an ERP-system is very expensive and risky. A considerable part of the costs is generated by hiring know-how of ERP-experts which need special skills for the configuration of the target ERP-system, on one hand. On the other hand, they have to learn about the specific implementation of business processes in each company which is a very tedious job. Therefore, it is reasonable to invest in the development of systems which enable modeling and software generation in an easy and configurable manner to achieve reuse and adaptability. In the case of an ERP-system, a framework that enables company business staff to configure the system on their own gives a significant advantage. Recently, the Swiss start-up company Posity AG has developed such a framework, using the Posity-Domain-Specific Language (Spielberger, Bärtschi-Rusch, Mürner, Perellano, & Wüst, 2014). In addition, with the support of the Zurich University of Applied Sciences (ZHAW), Posity developed a concept to set up an instance of the Posity ERP-system with the help of a questionnaire and without manual. The basic principle is that the process expert (user) can select the relevant workflow for his company from a reference process model by answering dynamically activated questions of a questionnaire.

Literature Review

The principle of reference model based workflow configuration has been described in detail by Rosemann and Van Der Aalst (2006). In PDSL, the reference process model is generated using process diagrams, the syntax of which corresponds to the business process model and notation (Grosskopf, Decker, & Weske, 2009). The concept on how to relate a set of meaningful questions to a reference process model has been described by La Rosa, Van Der Aalst, Dumas, Ter Hofstede, and Gottschalk (2006). The answers to these questions represent the configuration space. The text for the questions themselves has to be defined manually by the author of the configuration space. The questions relate to facts. The interrelations between facts are defined by logical conditions which are inherent to the workflow model. Each fact represents a set of processes that have to be executed together. The sequence of the questions, which are dynamically selected from the initial set, depends on the logical conditions and the answers to the already answered questions. The set of answers represents the configuration path and results in the set-up for a specific workflow. The configuration concept that they describe is universally applicable and can be used to configure the Posity ERP-system.

The pre-condition for an efficient configuration path which at the end will result in meaningful questions and a non-redundant short sequence of questions during a configuration run of the questionnaire is that there is a well-structured reference model given. The reference model, which is used in work of La Rosa et al. (2006) is structured into different process layers which are segregated into subtasks which are associated to facts. In work of La Rosa et al. (2006), even a demonstrator tool, the VICS EDI (VICS: voluntary industry commerce standards; EDI: electronic data interchange) framework, is presented. This tool demonstrates how the questionnaire works in detail. For their demonstration, they make use of the order fulfillment collaborative process model which is a standard logistic workflow model implemented in the YAWL notation (Van Der Aalst & Ter Hofstede, 2005).

The association of questions and facts as well as the constraints which determine the dependencies of questions and facts from each other are shown in Figure 1.

Although the Posity ERP tool-environment contains an easy-to-use graphical process library which facilitates the structured input of the business process at different abstraction layers, it remains a complex task to segregate workflow processes at the lowest level of the reference model of PSDL into the higher level abstract business processes.

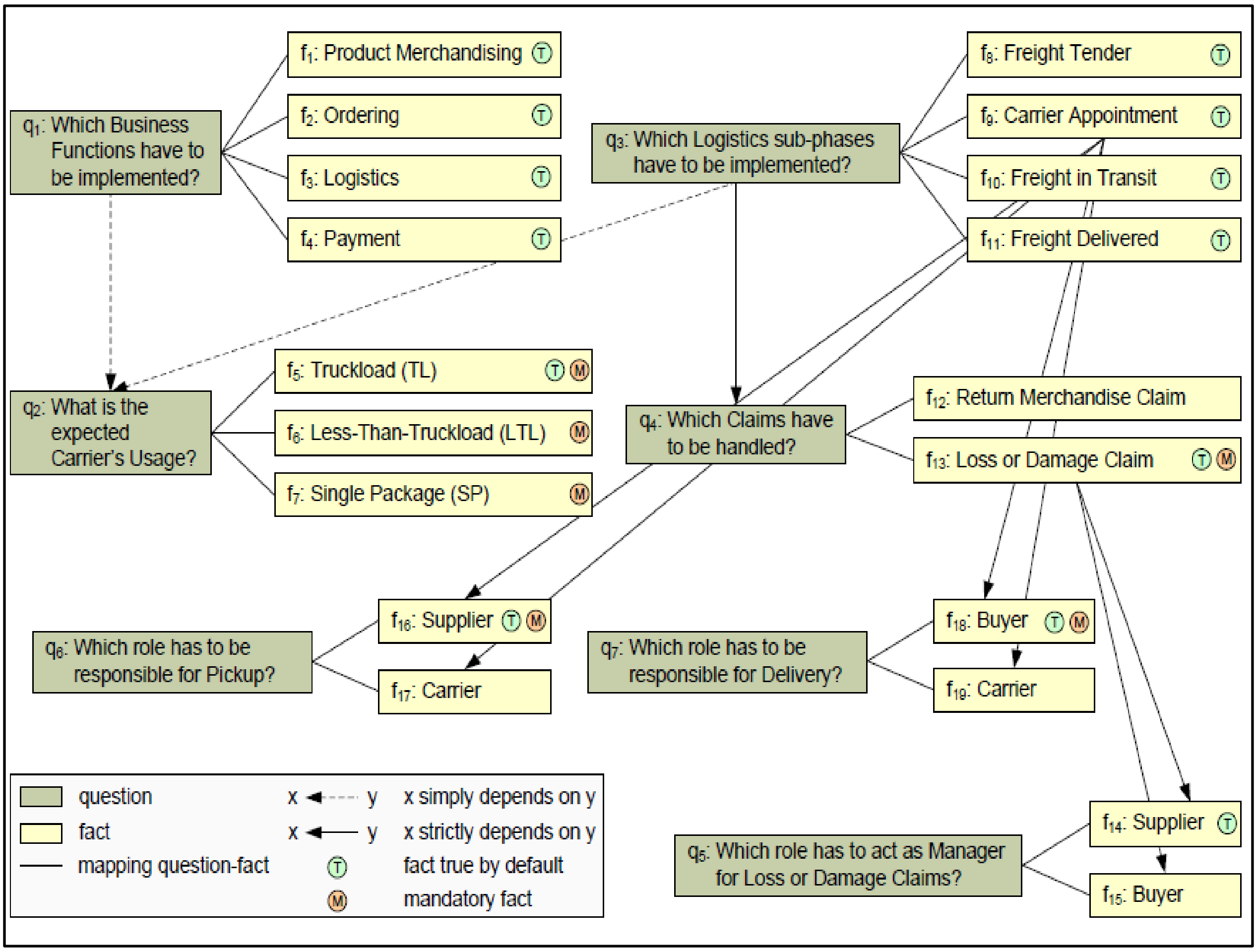
q1: Which Business Functions have to be implemented?
f1: Product Merchandising T
f2: Ordering T
f3: Logistics T
f4: Payment T
q3: Which Logistics sub-phases have to be implemented?
f8: Freight Tender T
f9: Carrier Appointment T
f10: Freight in Transit T
f11: Freight Delivered T
q2: What is the expected Carrier's Usage?
f5: Truckload (TL) T M
f6: Less-Than-Truckload (LTL) M
f7: Single Package (SP) M
q4: Which Claims have to be handled?
f12: Return Merchandise Claim
f13: Loss or Damage Claim T M
q6: Which role has to be responsible for Pickup?
f16: Supplier T M
f17: Carrier
q7: Which role has to be responsible for Delivery?
f18: Buyer T M
f19: Carrier
q5: Which role has to act as Manager for Loss or Damage Claims?
f14: Supplier T
f15: Buyer
question
fact
mapping question-fact
x y x simply depends on y
x y x strictly depends on y
T fact true by default
M mandatory fact

Figure 1. A possible structure of questions and facts drawn from the VICS EDI framework for the configuration space of the order fulfilment collaborative process model. Source: La Rosa et al. (2006).

Structure of This Article

This paper presents a method that shows how the hierarchical structure of the process model can be generated in an automatic way by evaluating the logical workflow syntax at the lowest level of the process nesting hierarchy which is defined by the business process model and notation-workflow graph. In the following section, authors give a short overview of relevant literature on workflow graph representations and summarize existing graph reduction techniques used in formal business process analysis based on workflow graph notation. Then, authors introduce an algorithmic approach based on extensions of existing rules. Later, authors show the result of applying the new graph reduction rules in order to structure the workflow example of the commonly known order fulfillment collaborative process model. In the last section, authors conclude by summarizing the introduced concept and proposing steps for further research.

Existing Graph Reduction Techniques

Already 15 years ago, Sadiq and Orlowska (2000) showed how workflow graph representations of process models can be used on one hand to reduce complex workflow graphs consisting of (a) modeling objects (tasks and choice/merge coordinators as nodes and control flows as directed edges) and (b) task types (activities, subprocesses, blocks, and null tasks) into new simplified graph structures by defining reduction rules. On the other hand, by the application of these reduction rules, they claimed to be able to evaluate the logical syntax of the process modeling graph structure and thus to verify consistency of the workflow. Regarding the reduction rules, they distinguished among terminal, sequential, adjacent, closed, and overlapped reduction rules. They also showed that through the application of the reduction rules, they could give a formal representation of the model hierarchy of the original process modeling graph (the graphs nesting structure) by counting the instances of subgraphs. Two years later, Van Der Aalst, Hirnschall, and Verbeek (2002) pointed out that the verification algorithm presented in work of Sadiq and Orlowska (2000) is not complete, because it cannot handle cancellation regions and OR-joins. Actually, in the same article, they presented an alternative verification algorithm which adapts the principle of the reduction rules but can handle cancellation regions and OR-joins. The method to generate the configuration space of the questionnaire that is presented here is also based on the idea of the reduction rules and is consistent with the concept of Van Der Aalst et al. (2002). In the following section, authors demonstrate how the procedure works in practice, again referring to the example of the order fulfillment collaborative process model. As suggested by Verbeek, Van Der Aalst, and Ter Hofstede (2005), authors used the YAWL notation to analyze the workflow. Still, for the illustration of the nesting structure, authors used the original workflow graph representation by Sadiq and Orlowska (2000).

Extended Graph Reduction Methods

The derivation of facts and their preconditions was an open question by La Rosa et al. (2006). Therefore, authors first want to show the relation between facts and their preconditions on one side and the reduction rules mentioned in below section on the other. They want to demonstrate that the configuration space of a questionnaire can directly be derived from the structural dependencies which are given by a consistent workflow graph and its inherent logical constraints. The relations between the facts of La Rosa et al. (2006) can be algorithmically generated by applying the sequential and adjacent reduction rule from Sadiq and Orlowska (2000). The terminal, the overlapped, and the closed reduction rule are used to detect structural conflicts in the workflow graph. These latter rules do not influence the facts and can therefore be ignored for the moment. For the detailed definitions of configuration models, workflow graphs, and the algorithms belonging to the reduction rules, the authors of which refer to La Rosa et al. (2006) and Sadiq and Orlowska (2000). Here, this paper wants to demonstrate the basic idea with the example of the order fulfillment collaborative process model used by works (La Rosa et al., 2006; Sadiq & Orlowska, 2000).

As in Sadiq and Orlowska' works (2000), this paper will also make use of the following basic process modeling objects (see Figure 2). The atomic parts of the workflow graph are sequences, forks, and joins.

A task, graphically represented by a rectangle, represents the work to be done to achieve some objectives. It is also used to implicitly build sequence, fork, and synchronizer structures. It is the primary object in workflow specifications and could represent both automated and manual activities.

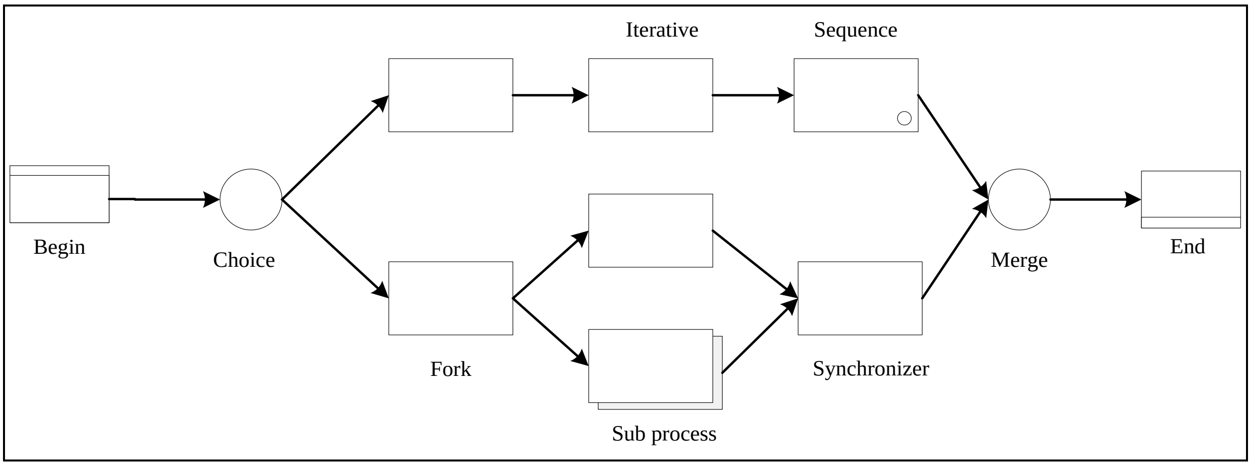

Figure 2. Process modelling structures.

A choice/merge coordinator, graphically represented by a circle, is used to explicitly build choice and merge structures.

A transition links two nodes in the graph and is graphically represented by a directed edge. It shows the execution order and flow between its tail and head nodes. By connecting nodes with transitions through modeling structures, this paper builds DAGs (directed acyclic graphs) called workflow graphs, where nodes are represented by vertices and transitions are represented by directed edges.

A sequence is the most basic modeling structure and defines the ordering of task execution. It is constructed by connecting at the most one incoming and one outgoing transition to a task.

A fork (and-split) structure is used to represent concurrent paths within a workflow graph and is modeled by connecting two or more outgoing transitions to a task. A fork does not enforce concurrent execution of the nodes that follow its outgoing transitions. It merely models the execution independence among concurrent paths.

A synchronizer (and-join) structure, represented by attaching more than one incoming transition to a task, is applied to synchronize those paths, where the execution of the workflow must be stopped until the execution of other concurrent paths has completed.

A choice (or-split) structure is used to model mutually exclusive alternative paths and is constructed by attaching two or more transitions to a choice/merge coordinator object. At run-time, the workflow selects one of the alternative execution paths for a given instance of the business process by activating one of the transitions originating from the choice coordinator object.

A merge (or-join) structure closes the choice structure. It is applied to join mutually exclusive alternative paths into one path by attaching two or more incoming transitions to a merge coordinator object. Since a workflow model is represented by a directed acyclic graphs, it has at least one node that has no incoming transitions (a source or begin node) and at least one node that has no outgoing transitions (a sink or end node). In a questionnaire, a question is associated to facts and the corresponding answer will set the facts to true or false. This research wants to associate facts to parts of the workflow graph. By setting a fact to true, it activates the corresponding part of the workflow graph. It assumes that they have a workflow graph without structural conflicts. The assignment of facts has to fulfill two main features. First, it should not introduce structural conflicts in the workflow (i.e., breaks). Second, it should identify instance subgraphs or blocks of the workflow, which can be activated by just answering a yes-or-no-question or setting the corresponding fact (see Figure 1) to true.

In this approach, this paper starts with the association of a fact to each modeling object of the workflow graph—basic facts. By applying the reduction rules from Sadiq and Orlowska (2000) iteratively, it builds up new facts and their preconditions out of the basic facts. Answering all basic facts by a single question is often not possible and furthermore not necessary, since the facts depend on each other. It puts the requirements which are additional compared to the original rules in Sadiq and Orlowska' works (2000) in the followings bullet sections:

- sequential reduction rule: If the current node is part of a sequential structure, the current node will be removed from the graph by the sequential rule and the edge from the predecessor node becomes directly connected to the successor node. In addition, authors associate a new fact to this new edge. This new fact is precondition for the basic facts of the removed node and its incoming and outgoing edge (see Figure 3);
- adjacent reduction rule: The adjacent reduction rule targets two types of components. If the current node is not removed by the sequential reduction rule, it means that it is forming either a split or join structure, since it would either have out degree or in degree or both that is more than one. In this case, the authors check if the current node has a single incoming transition and is introducing a split structure by having more than one outgoing transition. If the type of the current node is the same as its preceding node, they move outgoing transitions of the current node to the preceding node and remove the current node.

From Figure 3, the whole workflow is partitioned into four sub-processes: product merchandising, pre-ordering, ordering, and logistics, which can be implemented independent from each other. Therefore, they all have a direct or-connection to the end-node or to a subsequent sub-process. Inclusion of rectangles represents nesting of sub-processes, for instance logistics—carrier appointment—truck load—pickup role. Facts indices (e.g., Figures 1 and 2) are correspond to those used in the works of La Rosa et al. (2006). Facts with equal numbers belong together and have to be executed in combination. These dependencies can be resolved in the reference model or have to be manually combined.

If the last criterion is not met, authors check if the current node has a single outgoing transition and is introducing a join structure through more than one incoming transitions. Then they merge join structures analogously to the split case before.

- If the current node is a fork and all its outgoing edges end directly in a synchronizer structure, authors introduce a new fact, which is associated to the fork, the synchronizer, and all edges inbetween. This new fact is precondition for all the facts associated to the parts of this fork/synchronizer block.
- If the current node is a choice structure and all its outgoing edges end in a merge structure, authors introduce a new fact for each path between the choice and the merge structure (i.e., the choice and the merge structure and one edge inbetween). Each new fact is a precondition for the parts of the associated path (see Figure 3). Furthermore, authors add a fact to the whole choice/merge block and a constraint that this new fact can only be set if and only if one of the facts associated to the paths is set (i.e., this fact forms an XOR type question).

Results

By applying these rules iteratively, it is possible to consider the whole workflow graph starting from its interior atomic units. The resulting nesting structure is shown in Figure 3 together with the facts of the questionnaire in the works of La Rosa et al. (2006).

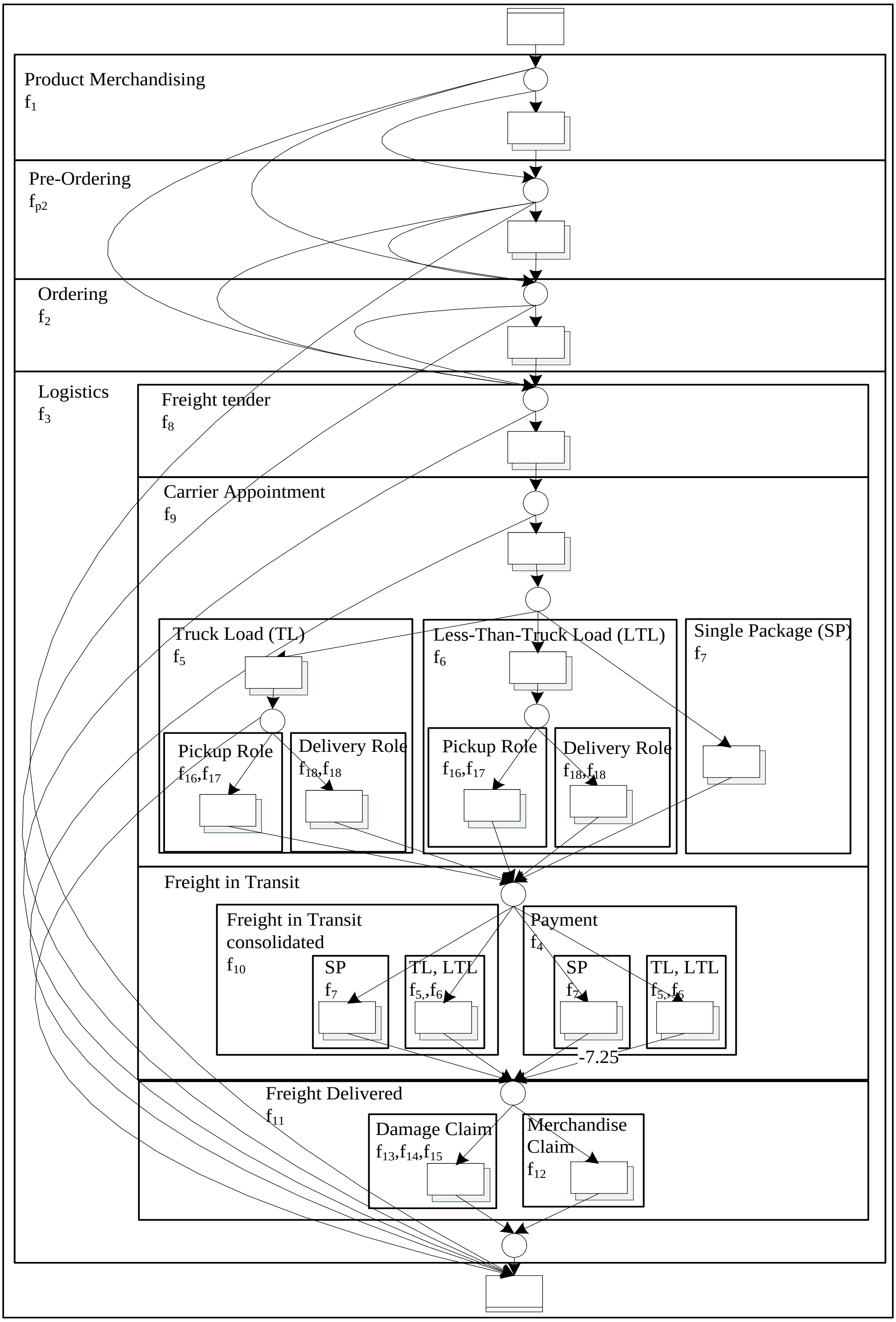

Figure 3. Nesting structure of the order fulfilment collaborative process model.

Conclusions

The utilization of reference process models has a high potential for the easy configuration of workflow- and ERP-systems. A real instance of such a system configuration can be generated by translating a configuration path obtained from the questionnaire algorithm of La Rosa et al. (2006). Authors demonstrated how the relation of facts and conditions, which are needed for the questions of the questionnaire, can be derived from the structure of the underlying workflow graph by algorithmically applying an extended version of the graph reduction techniques summarized in mentioned section. They applied this technique in order to generate a prototype implementation of a Posity ERP-system configuration based on PDSL as the workflow modeling notation. In a next step, they want to develop a configuration algorithm, which can be universally applied for solving workflow system configuration problems. Also they want to improve the formalism to avoid any additional manual interaction (e.g., grouping facts together) in the future.

References

Grosskopf, A., Decker, G., & Weske, M. (2009). *The process: Business process modelling using BPMN.* Tampa: Meghan-Kiffer Press.

La Rosa, M., Van Der Aalst, W. M., Dumas, M., Ter Hofstede, A. H., & Gottschalk, F. (2006). *Generating inter-active questionnaires from configuration models.* Retrieved from http://eprints.qut.edu.au

Rosemann, M., & Van Der Aalst, W. M. P. (2007). A configurable reference modeling language. *Information Systems, 32*(1), 1-23.

Sadiq, W., & Orlowska, M. E. (2000). Analysing process models using graph reduction techniques. Proceedings from *The 11th International Conference on Advanced Information System Engineering*.

Spielberger, J., Bärtschi-Rusch, M., Mürner, M., Perellano, G., & Wüst, R. M. (2014). Rapid development of ICT business services by business engineers independent of computer scientists. Proceedings from *Science-to-Business Marketing Conference*. Retrieved from http://www.s2b-conference.com/

Van Der Aalst, W. M. P., & Ter Hofstede, A. H. M. (2005). YAWL: Yet another workflow language. *Information Systems, 30,* 245-275.

Van Der Aalst, W. M. P., Hirnschall, A., & Verbeek, H. M. W. (2002). An alternative way to analyze workflow graphs. In A. Banks-Pidduck, J. Mylopoulos, C. C. Woo, and M. T. Ozsu (Eds.), Proceedings from *The 14th International Conference on Advanced Information Systems Engineering (CAiSE'02)* (pp. 535-552). Berlin: Springer.

Verbeek, H. M. W., Van Der Aalst, W. M. P., & Ter Hofstede, A. H. M. (2005). *Verifying workflows with cancellation regions and OR-joins: An approach based on relaxed soundness and invariants*. Oxford: Oxford University Press.