Ontology-driven Process Specialization

Abderrahmane Leshob[1,2], Hafedh Mili[2] and Anis Boubaker[2]

University of Quebec at Rimouski[1], University of Quebec at Montreal[2]

Abstract. Business process design is an important activity for the planning and analysis of information systems that support the organization's business processes. Our goal is to help business analysts produce detailed models of the business processes that best reflect the needs of their organizations. To this end, we propose to, a) leverage the best practices in terms of a catalog of generic business processes, and b) provide analysts with tools to customize those processes by generating, on-demand, new process variants around automatically identified process variation points. We use business patterns from the Resource Event Agent ontology to identify variation points, and to codify the model transformations inherent in the generation of the process variants. We developed a prototype, showing the computational feasibility of the approach. Early feedback from a case study with three *Business Process Management* (BPM) experts validated the relevance of the variation points, and the correctness of corresponding transformations, within the context of key *Enterprise Resource Planning* (ERP) processes. In this paper, we summarize the approach and report of the results of a larger experiment, gaining insights into the strengths and weaknesses of our approach, and suggesting avenues for improvement.

1 Introduction

Business process modeling is an important activity for organizational design and for the planning and analysis of information systems that support the organization's business processes. This work aims at helping business analysts develop precise business process specifications without having to become process designers specialists. Our approach consists of developing: 1) a catalog of broadly useful generic business processes that business analysts can use as a starting point for their organization-specific process, and 2) a set of specialization operators that business analysts can apply to a generic process to obtain a process model that accurately reflects their organization's needs.

With regard to the first point, the very existence of ERP systems and frameworks does suggest that. Nonetheless, to ensure that our process catalog has a "good coverage" with a "manageable" set of processes, we need a "good" representation of business processes that abstracts unimportant details and highlights business meaningful process variations.

With regard to the second point, our representation of the specialization operators should hide the technical details of the underlying model transformations, and focus on the business-level meaning of these specializations. To this end, we adopted the question-based approach to process specialization.

The rest of the paper is organized as follows. Section 2 surveys related work. Section 3 describes our approach for generalizing the specialization's questions. Section 4 presents the REA framework and patterns with a view towards automatic process specialization. Section 5 presents the design and implementation core of our approach. Section 7 discusses validation. We conclude in section 8.

2 Related Work

A number of business process cataloging efforts have used questions or options to manage process variability. Carlson's *Business Information Analysis and Integration Technique* (BIAIT [1]) proposed seven questions based on the concept of generic order to identify the major functional building blocks of the information system. This approach is interesting since variation points are meaningful to a business analyst. Unfortunately, it works only at a macroscopic level.

The MIT process handbook [2] organizes processes along *specialization dimensions*, framed as questions. This approach is helpful in navigating the process catalog, however, the questions and the corresponding process variants are process-specific.

Coad, Lefebvre and De Luca's [3] proposed a questionnaire-based framework to specialize generic business models, but they focused on model *fragments* instead of entire models. A number of researchers have explored the problem of managing variability within process models (see e.g. [4], [5]). However, these approaches focus on managing *previously codified process variants*, as opposed to *deriving* those variants.

3 An Approach to Question-Based Specializations

In order to be able to reason about the business process, we will use the business process perspectives as proposed by Curtis [6]. Indeed, Curtis argued that a complete representation of a business process requires four distinct views [6]: 1) the *dynamic view*, which provides sequencing and control dependency between the tasks of a process, 2) the *functional view*, which represents the functional dependencies between the process elements. In our work, we replaced this view by the *REA view* (see section 4), 3) the *informational view*, which includes the description of the entities that are produced, consumed or otherwise manipulated by the process, and, 4) the *organizational view*, which describes who performs each task or function, and where in the organization.

In order to apply our specialization approach to processes from different domains, we needed to define generic questions. Our first attempt consisted of generalizing the sets of questions proposed in the literature, including [2, 3, 7]. Idem for the underlying model transformations. This sounded reasonable on paper/toy examples, but proved unworkable with real life processes taken from the SAP/R3 blueprint [8]. Indeed, the generalized questions and the corresponding

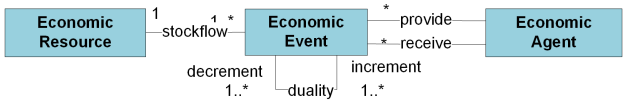

Fig. 1. REA metamodel

generalized model transformations had lost much of the original business semantics, and became meaningless. For a given process, this resulted into many 'question instances' that did not make sense and spurious model transformations.

We realized that we needed *BIAIT-style* questions. By *BIAIT-style* questions, we mean questions that relate to the *essence* of the business processes. Similar to Carlson's concept of *generic order*, as it pertains to an organizations core activities/processes, we needed a *business ontology* that would enable us to see the similarities between, say, a procurement process and a hiring process. We felt that the *Resource-Event Agent* (REA) framework [9] might offer such an ontology. McCarthy views the core processes of an organization as a sequence of *exchanges* or *transformations* of *resources* between *agents* [9]. In this context, it is easy to see how the concept of a *contract*, for example, becomes relevant to various process areas, as a way of *governing* those exchanges.

4 Variants with REA Business Patterns

In this section, we present the concept of an REA pattern. Finally, we show how such a pattern can be the source of, a) a variation point, framed as a generic question, and b) the corresponding model transformation.

4.1 The REA Framework

McCarthy proposed the REA framework as a way of capturing the economic phenomena that needed to be accounted, from an *accounting* perspective [9]. In REA, an enterprise can increase or decrease the value of its resources through either *exchanges* or *conversions* [10]. In an *exchange*, the enterprise receives economic resources from external economic agents, and provides other resources in return. In a *conversion,* an enterprise uses or consumes resources in order to produce new or modified resources.

Fig. 1 shows the basic REA metamodel. *Economic resources* are objects that are scarce, have utility, and are under the control of an enterprise [9]. Economic events are "a class of phenomena which reflect changes in scarce means resulting from production, exchange, consumption, and distribution" [11]. An *economic event* can be either an *increment* or a *decrement* in the value of economic resources. The duality relationship links increment events to decrement events. An *Economic Agent* is an individual or an organization capable of controlling economic resources, and transferring or receiving that control to or from other individuals or organizations [10].

4.2 REA Business Patterns

Work on REA has identified a number of business patterns. Such patterns are the focus of our generic questions and our specialization operators. Hruby identified more than twenty REA business patterns [10]. Geerts and McCarthy proposed several REA patterns as part of the REA ontology [12]. From these, we focused on a dozen patterns. Each such pattern involved one or more *variation points*, framed as *generic questions*, and *transformation rules* that generated the process variants, corresponding to different answers of those questions.

4.3 Running Example

Consider the procurement process of Fig. 2. It starts by filling out a requisition. The purchaser then sends a request for quotation (RFQ) to potential suppliers. After receiving the quotations, the purchaser selects a supplier, creates a purchase order (PO) and sends it back to that supplier. In turn, the supplier fulfills the order and delivers it to the purchaser. Once the product is received, a goods receipt is generated and payment is made.

A question raised by the exchange pattern is whether the exchange is governed by an *agreement*. A 'yes' answer would impact several views of the process. Indeed, we need to represent the Agreement object in the information view. We also need to simplify the dynamic view by removing the steps for supplier selection. This question can be represented by a one-parameter boolean function, verbalized as follows: *Is there an agreement that governs the business process {0}?* Hruby modeled agreements using the REA *contract pattern* [10], which introduces the related notions of *commitment*, *contract*, and *agreement* (see Fig. 3). A *commitment* is a promise of economic agents to perform an economic event in the future. A *contract* is a collection of related commitments. Terms are potential commitments that are instantiated if certain conditions are met (e.g. commitment not being fulfilled).

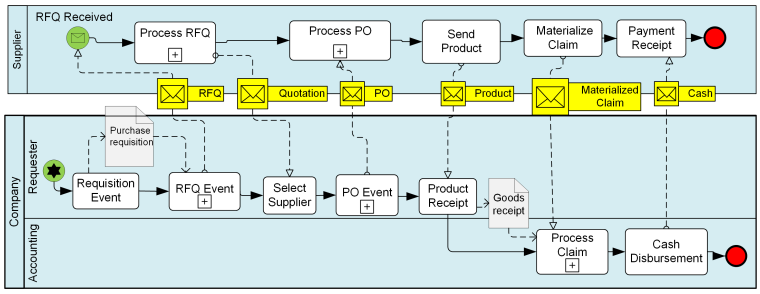

Fig. 2. Generic procurement process

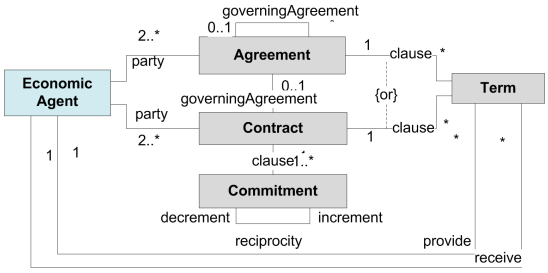

Fig. 3. Contract pattern

5 Business Process Specialization

The initial population of the 4-view generic process catalog relies on the existence of a catalog of REA patterns codified in a way that, 1) identifies the variation points, and 2) operationalizes the transformations corresponding to the different variants. Our main sources for REA patterns have been REA's ontology [12], and Pavel Hruby's twenty two REA patterns [10]. For the purposes of this experiment, we excluded some of the 'trivial patterns', as well as those patterns that dealt exclusively with the data model/the informational view. The end result was a dozen patterns.

To populate our generic process catalog, we explored a number of sources, including the MIT process handbook [2] and SAP R/3 blueprint[13]. To build the four process views (i.e. REA, dynamic, informational and organizational views), we have developed a number of heuristics to support a semi-automated process for generating the views from an annotated BPMN model, discussed in [8].

For process browsing and specialization, we envision a process where a business analyst looking for a particular process starts browsing the process catalog, drilling down progressively, until either they find a process that they can use as is, or they find a process that is close enough, and that they can specialize. To specialize a process, the analyst 'asks' the tool to identify the variation points and present them, along with the various alternatives (answers). The analyst then selects the appropriate answers, and the tool generates a specialized process, resulting from the application of a cascade of transformations. Fig. 4 shows the overall process of the proposed process specialization approach.

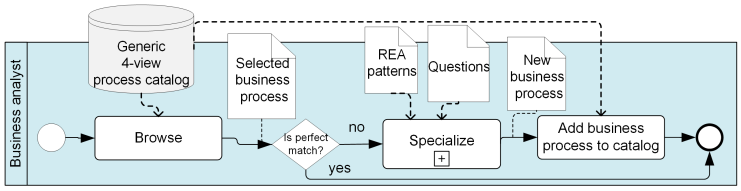

Fig. 4. Overall process of the proposed process specialization approach

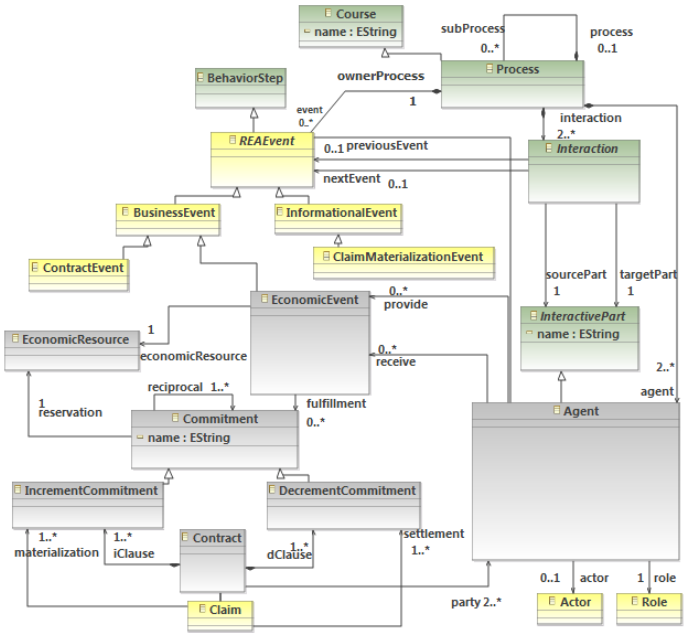

Fig. 5. Excerpts from the implemented 4-view business process metamodel

6 Preliminary Implementation

6.1 Process Model Representation

Our metamodel is based on the REA business ontology. The REA metamodel does not support the concept of orchestration, since REA is an economic-resource-centred view that focus solely on those resource altering phenomena. Hence, we extended the REA metamodel to cover the informational, organizational and the dynamic views of process models. We implemented our metamodel as an Eclipse plugin with the Eclipse Modeling Framework™ (EMF, version 2.7.2), which is an Eclipse-based Java modeling framework that implements a core subset of OMG's Meta Object Facility (MOF). With EMF, the informational view comes out-of-the-box, embodied in the core UML metamodel, with classes such as `EClass`, `EAttribute`, etc. The dynamic view was based on the *Business Process Definition Metamodel* (BPDM, [14]) behavioral model. The organizational view was implemented using the subset of the *Organization Structure Metamodel* (OSM, [15]) used in BPDM. The REA view was implemented with concepts from the REA ontology, including a) core concepts such as `EconomicResource`, the `REAEvent` subhierarchy, and the duality between economic events, and b) specialized concepts coming from various REA patterns, such as `Claim` and `Commitment`. Fig. 5 shows the main classes from the implemented metamodel.

```
<!-------------------------------------------------------------->
<!-- The agreement question  (Contract pattern) -->
<!-------------------------------------------------------------->
 <question  id="q4">
        <name>AGREEMENT_IN_THE_PROCESS</name>
        <description>
                Contract regulates the behavior among organizations and individuals.
                A contract is an entity containing commitments and terms. Agreement
                is a higher level contract that regulates the behavior of individual
                contracts. SLA is an example of agreement.
        </description>
        <core answerType="AnswerType.BOOLEAN" >
                Is there an agreement that governs the business process {0}?
        </core>
        <parameters >
                <parameter>Process</parameter>
        </parameters>
</question>
```

Fig. 6. The representation of generic questions

6.2 Implementing variation points

Variation points were designed as generic questions, which can be thought of as multi-parameter functions, whose values are taken from an enumeration, and whose parameter types correspond to business process element types. When a 'question' is 'instantiated' for a particular process model fragment, the parameters are bound to specific elements of the fragment, and the function call is presented in a *verbalized* form so as to make sense to the analyst who is then prompted for a value among the enumeration of possible answers.

For example, the question regarding whether a particular exchange were governed by an agreement would be represented by the template "*`Is there an agreement that governs the [exchange process] {0}`*", where `{0}` is a *positional parameter* that will be replaced by an REA exchange process within the value chain. For example, when applied to the procurement process of Fig. 2, this question will be formulated as "*`Is there an agreement that governs the process procurement`*". The question model is implemented using an XML schema. Fig. 6 shows the representaion the agreement question.

6.3 Implementing Process Specialization

We implemented our transformations as **if-then** transformation rules that manipulate EMF process models, where the **if** part matches a process model pattern, and the **then** part applies the relevant transformation to process model elements represented as EMF objects. Thus, we looked into open-source hybrid *object-rule* systems, and settled for JBoss Drools version 5.1. We wrote a *single* DROOLS transformation rule, *per* <process model view, answer> combination. Hence, the transformation that needs to be applied to the *informational view* in case of a `yes` answer to the `'is there an agreement that governs the [exchange process] {0}'` is represented by a single rule, shown in Fig. 7.

The *when* part binds the variable *$q* to the instance of the specific question that is in the rule engine's working memory (line 3). The *then* part describes the

```
1  rule "YES_AGREEMENT_INFORMATIONAL_VIEW"
2  when
3   $q : Question(gQuestion.name==AGREEMENT_IN_THE_PROCESS);
4  then
5   Message m = DynaHandler.getREAContractMessage(process, $q.param(0));
6   Agent agent1 = (Agent) m.getSourcePart();
7   Agent agent2 = (Agent) m.getTargetPart();
8   EClass clazz=InfHandler.addEClass(view,AGREEMENT_CLASS, BpPackage.
        eINSTANCE.getEResource());
9   InfHandler.addEAttribute(clazz, "dateFrom", eINSTANCE.getEDate());
10  ...
11  EReference ref1=InfHandler.addEReference(view,clazz,agent1, agent1.
        toLowerCase(),1,1,null, false);
12  InfHandler.addEReference(view,agent1,clazz,AGREEMENT_CLASS.toLowerCase
        (),1,UNBOUNDED_MULTIPLICITY,ref1,false);
13  ...
14  EReference ref3 = EMFInfViewHandler.addEReference(view, clazz, m.
        getResource(),"contract", 0, UNBOUNDED_MULTIPLICITY, null,false);
15  ...
16  InfHandler.removeElementsByEventType(process,"REAAgentIdentification");
17 end
```

Fig. 7. Transformation rule of informational view for yes answer to agreement question

actions. Lines 5 to 7 identify the economic agents. The expression *$q.param(0)* in line 5 refers to the REA process. Lines 8 to 10 create an `Agreement` class, and add it to the informational view. Lines 11 to 13 add bidirectional associations between the `Agreement` class and the classes that represent contract's parties. Lines 14 to 15 add bidirectional associations between the `Agreement` class and the REA contract class. Finally, line 16 removes the classes that represent objects that are used solely for identifying the exchange partner (i.e. objects related to the business events of type `REAAgentIdentification`).

7 Validation

We are proposing a methodology for representing and specializing business processes that enables business analysts to find, or derive, a business process that matches their needs. Viewing our work within the context of *method engineering*, we could evaluate the extent to which our approach enables business analysts to produce *better* process models *faster*.

At this stage of the research, we propose to validate: (i) our *representation of business processes,* (ii) our *variation points,* and *(iii)* our *specializations.*

With the exception of the syntactic correctness, the other aspects can only be validated by *human subjects*, who have the expertise to walk through the methodology, and to evaluate the various processes, models, and questions. Of the many experts we solicited, three generously volunteered to participate. Expert 1 is a business consultant at a major Canadian bank, with an extensive experience with SAP processes. Expert 2 is specialized in business process management (BPM) at a major consulting multinational. Expert 3 is a university

Table 1. Experimental processes.

	Business Process	ERP Classification
P1	Procurement	Buy
P2	Sales & Distribution	Sell
P3	Production & Inspection	Make
P4	Maintenance & Customer Service	Service
P5	Hiring	Human Capital Management
P6	Financial Loan	Finance
P7	Insurance	Service
P8	Payroll	Human Capital Management

Table 2. Experimental generic questions

Pattern	Question
Exchange	Q1: Does the process {0} support instantaneous exchange?
	Q2: Does [increment events] {0} precede(s) [decrement events] {1} for the exchange process {2}?
Outsourcing	Q3: Does the organization plan outsourcing the process {0} to a partner?
Contract	Q4: Is there an agreement that governs the process {0}?
Commitment	Q5: Does the process {0} support future obligations?
Claim	Q6: Does the process {0} support a Two-way match strategy for claim processing?
Posting pattern	Q7: Does the organization keep transactions history between partners for the process {0}?

professor with an extensive business consulting experience. In addition to the qualitative results obtained from our experts, we conducted a quantitative experiments with twelve graduate students from a business school. All students have a strong experience in aligning business processes in the context of small and large organizations. They have also a very good background in business process modeling with BPMN and information systems design with UML.

7.1 Experimental Data

We studied 22 processes from ERP systems [13, 16]. From these processes, we selected 8 processes, one from each ERP process area, shown in Table 1. To validate the applicability of questions and transformations, out of the 12 questions that our study of Hruby patterns identified, we started encoding the ones that were related to the REA patterns that we felt were the most useful. This resulted into encoding seven (7) questions, and the view transformation rules corresponding to the various answers. Table 2 shows the REA patterns, and the corresponding generic questions.

7.2 Validation of Process Representation

Our business process metamodel resulted from a number of iterations. The informational view is handled with EMF's ECore package. The REA metaclasses

were added as needed by the new encoded REA patterns. For the dynamic view, the subset of the BPDM metamodel [14] that we implemented proved sufficient.

As expected, the derivation of the REA view from the BPMN view was challenging, for three main reasons. First, there is no one to one mapping between REA concepts and BPMN concepts. Indeed, not all BPMN *data objects* represent *REA resources*, and not all *BPMN activities,* represent *REA economic events*. Second, BPMN models *typically* ignore classes of resources that are nonetheless needed to perform activities, such as *labor* or *equipments*. Finally, the REA patterns rely on the concept of *duality* between resource increment events and decrement events. This was not an issue with exchange-type processes, but was an issue with conversion-type processes where the BPMN models were missing *both* resources *and* dual economic events, which had to be added manually. In total, our 8 processes contained 7 REA exchanges and 4 conversions. We had to complement the initial BPMN models for three of the conversions, to add the missing resources and events.

7.3 Validation of Questions Applicability

A key aspect to our approach is our claim that variation points/questions identified within the context of REA patterns, a) made sense from a business view, and b) were applicable to *many* process areas. To validate this claim, we presented our experts with the 7 questions that we had encoded, and the 8 generic processes that we had modeled, and asked them whether a particular question, as instantiated for a particular process, made sense. If the process contained several REA patterns, then the question was instantiated for each pattern. Experts 1 and 2 volunteered for the exercise, and took 45 minutes to produce Table 3. A value of 1 meant that the question, as instantiated, made sense for the process at hand. A value of NA (Not Applicable) meant that the question did not apply to this kind of process. A value of Partial was used when the same question was instantiated several times for the same process (REA value chain), and where some instantiations made sense while others did not. A value of 0 meant that the questions did not make sense/should not have been instantiated. A closer inspection revealed that all NA values were due to the fact that exchange-type questions were instantiated for conversion-type processes. This also explains the 'Partial' values, which were used when an exchange-type question was instantiated twice for processes that consisted of a combination of an exchange and a conversion, and the experts felt that the question did not apply to the conversion leg of the process (e.g. Sales & Distribution). With hindsight, we could have filtered the instantiation of exchange-type questions (Q1, Q2, Q4, and Q6) for conversions, which would have yielded 95.08 % of sensical instantiations.

7.4 Validation of Process Specializations

To validate the process specializations, we needed to: a) validate that the models produced by the corresponding transformations conform to the process metamodel syntax and semantics, and b) validate whether the models produced re-

Table 3. Questions applicability.

Business Process	Q1	Q2	Q3	Q4	Q5	Q6	Q7
Procurement	1	1	1	1	1	1	1
Sales & Distribution (S&D)	Partial	Partial	1	Partial	1	Partial	1
Production & Inspection	NA	NA	1	NA	1	NA	1
Maintenance & Customer Service	Partial	Partial	1	Partial	1	Partial	1
Hiring	0	1	1	1	1	1	1
Financial Loan	0	1	1	1	1	1	1
Insurance	1	1	1	1	1	1	1
Payroll	0	1	1	1	1	1	1

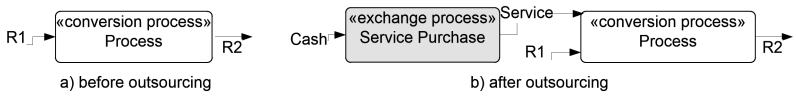

Fig. 8. REA outsourcing pattern

flect the *expected business semantics*. Metamodel conformance was validated using EMF's *validation framework* (EMF-VF). EMF-VF provides a capability to parse an EMF model and return true if the model satisfies the constraints of its metamodel, and false otherwise. We were able to ascertain that all the generated models conformed to the metamodel.

With regard to *business semantics* correctness, we had to rely on our experts. They had to answer questions of the type 'given initial process P, and answer A to question Q, is the [generated process] P_{qa} what you would have expected?'. Thus, we had to generate a number of specialization scenarios using our 8 processes, 7 questions, and the possible answers for each question. Notice that taking all the applicable <process, question> combinations from table 3 (49 out of 52) does not guarantee *complete test coverage* because we wanted to evaluate, among other things, combinations or cascades of elementary specializations that may transform the same elements, i.e. *confluent transformations*. Thus, we relied on Expert 1 to generate what he felt were 'interesting specializations', while ensuring that each question (transformation) was exercised at least once. In the end, Expert 1 generated twenty-five (25) new processes, by executing 78 transformations, with an average of about 3 transformations per process. Those 25 processes were then handed to Experts 2 and 3 for validation.

Experts 2 and 3 confirmed that 19 out of 25 processes (76%) corresponded to what they expected. As for the remaining 6 (24%), which all involved a yes answer to question Q3 (outsourcing), they felt that while the net flow of economic resources was correct for all 6 processes, the resulting models were not the ones they expected. Specifically, they argued that the obtained models were unnecessarily complex. This is because in REA (see REA outsourcing pattern of Fig. 8), we can only exchange (buy/sell) resources, and not processes.

Table 4. Evaluation results of G1 and G3

Group	Participant	Evaluation Phase			Generation& Matching Phase		
		P1q1	P3q3	P2q4	P7q6	P6q7	P5q4
G1	Student 1	1	1*	1	1	1	1
	Student 2	1	1*	NA	1-	1	1
	Student 3	1	1	1	1	1	1
G3	Student 7	1	1*	1	1-	1	1
	Student 8	1	1*	1	1-	1	NA
	Student 9	1	1*	1	NA	1	1

Table 5. Evaluation results of G2 and G4

Group	Participant	Evaluation Phase			Generation & Matching Phase		
		P7q6	P6q7	P5q4	P1q1	P3q3	P2q4
G2	Student 4	1	1	1	1	1+	1
	Student 5	1-	1	1	1	1+	1
	Student 6	1-	1	1	1	1+	1
G4	Student 10	NA	1	1	1	1+	1
	Student 11	1	1	1	1	NA	NA
	Student 12	NA	1	1	1	1+	1

While the experts provided us with a *qualitative* results, we also conducted a *quantitative* experiments with twelve graduate students from a business school. All participating students have a strong knowledge of BPM methodologies. To validate the correctness of our specializations, we first selected 6 generated processes from our repository and then conducted an experiment in two phases: 1) *process evaluation phase* and 2) *process generation & matching phase*. In the process evaluation phase, the students had to answer the same type of questions we have asked our experts i.e. 'given initial process P, and answer A to question Q, is the generated process Pqa what you would have expected? In the process generation & matching phase, they were asked to provide their own transformations of the base processes by applying a yes answer to selected questions and then, compare the resulting processes to our automatically generated processes. To conduct our experiments, we formed 4 groups (G1, G2, G3 and G4) of 3 students each. We presented to each group a set of processes and asked each participant to proceed individually as follow: participants of G1 and G2 started with the evaluation phase and proceeded thereafter with the generation phase while participants of G3 and G4 proceeded inversely. As shown in tables 4 and 5, we asked the groups to validate our specialization methodology using the same set of processes, questions and answers in different sequences. Thus, each student evaluated 3 processes and generated 3 processes. The participants took 3 hours to complete the experimentation. Tables 4 and 5 illustrate the evaluation results of groups (G1, G3) and (G2, G4) respectively. The notation **Piqj** denotes the specialized process obtained/to obtain by transforming the process *Pi* after applying a yes answer to the question *Qj* (see table 2).

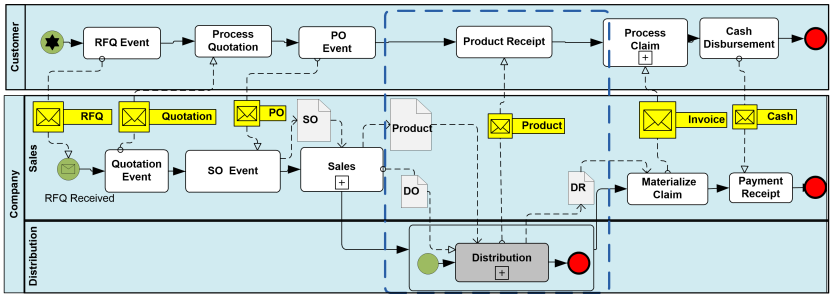

Fig. 9. Generic Sales & Distribution process

The meaning of the value 1 depends on the experimentation phase. In the evaluation phase, the value of 1 meant that the process generated by our approach is what the student expected. In the generation phase, it meant that both our process and the process generated by the student are identical. The value of 1* meant that the student found the generated process correct but more complex than what he/she expected. The value of 1+ indicated that the student's generated process highlighted correctly, but differently, the outsourcing concept. The value of 1- indicated that the student kept the *informational event* (see [17]) of the invoice value calculation without sending it to the partner while our transformation rule removed the whole *business event* (see [17]) which (1) calculate the invoice value and (2) send it to the partner. Finally, a value of NA (Not Applicable) meant that the student did not evaluate the resulting process or did not generate a complete and a valid process. Thus, the NA values were discarded from our results.

With regard to the value of 1*, all students, except student 3, who evaluated the generated process after a yes answer to the outsourcing question confirmed the results obtained by expert 2 and expert 3. They argued that the generated models were unnecessarily complex. This is because in REA, we can only exchange economic resources. Thus, to outsource a process, we consider its 'performance on our behalf' as a *service*, that we *purchase* (*exchange*), and then *consume* (*conversion*) (see the outsourcing business pattern in Fig. 8). For example, to outsource the *distribution* part of Sales & Distribution process (Fig. 9), our tool generated the model shown in Fig. 10 where *SO*, *PO, DO, SPO, DS* and *DR* stand for *Sale Order*, *Purchase Order, Distribution Order, Service Purchase Order*, *Distribution Service* and *Distribution Receipt,* respectively. Expert 2, expert 3 and all students that generated new processes after outsourcing a part of a process (value of 1+ in Table 5) proposed a simplified, but REA-invalid, process similar to the process in Fig. 11.

Finally, regarding the 1- values, both processes are *semantically equivalent* as the invoice calculation (i.e. informational event) does not affect the business process behavior (see [18]). Fig. 12 illustrates the result of the evaluation phase. Among the 36 evaluations, students confirmed that 72.22 % of the evaluated processes are exactly to what they expected, while 5.56% are different

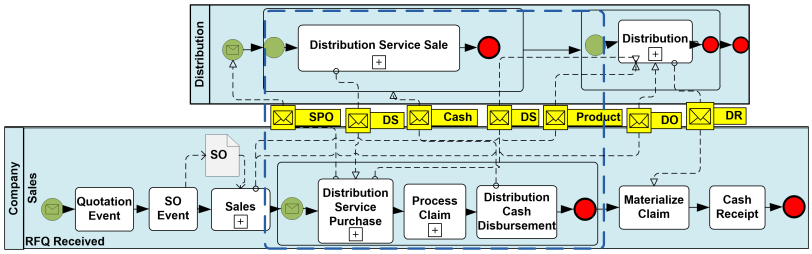

Fig. 10. B2B collaboration after outsourcing the Distribution in S&D process

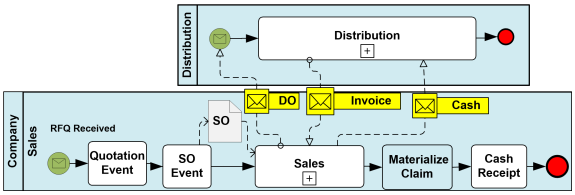

Fig. 11. B2B collaboration after outsourcing the Distribution in S&D process according to experts 2 and 3

but semantically equivalent. As for the remaining evalutions (13.89%), which all involved a yes answer to the outsourcing question, students felt that while the net flow of economic resources was correct for all processes, the resulting models were not the ones they expected.

7.5 Threats to Validity

This section presents the threats to the validity of the performed experiments.

Threats to external validity The external threat comes from the limited set of generic questions (7 questions) which are obtained from REA patterns. To date, we have identified 16 questions but automated transformation rules for 7 of them. To mitigate this concern, a set of 30 variation points were analyzed from

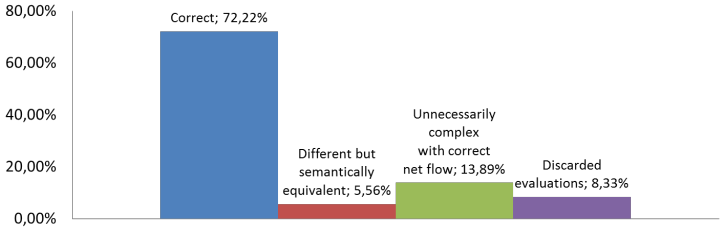

Fig. 12. Correctness of the transformations in the process evaluation phase

the industry and literature, including the specialization patterns from e3-value business ontology [19] and process re-engineering domain [20]. The experimental questions were selected from question-based specialization approaches that were studied in previous work on business process adaptation area to facilitate comparison with previous results.

Threats to construct validity Construct validity refers to the validity that observations or measurement tools actually represent or measure the construct being investigated. In this paper, one possible threat to construct validity arises from the evaluation method used to prove semantic equivalence of source and resulting business process models. There are several general approaches like ([21, 22]) that can be used to evaluate semantic equivalence between models. Therefore, conclusions obtained from our correctness evaluation might not be representative of other evaluation methods. A mature and widely used technique (semantic consistency) is used to mitigate this concern. Indeed, our experts used the semantic consistency approach based on a set of business constraints to evaluate the semantic correctness of the resulting process models.

8 Conclusion

Business process modeling is an important activity for both organizational design and for the planning and analysis of information systems that support the organization's business processes. The purpose of this work is to assist business analysts in the process of modeling the business processes that best reflect the practice of their organizations. To this end, we proposed to, a) leverage the best practices in terms of a catalog of *generic business processes*, and b) provide business analysts with tools to customize those processes by generating, on-demand, new process variants around automatically identified process variants. Our approach for *automatically* identifying process variants and *automatically generating* user-selected process variants relies on the *projection* of the generic processes along the REA *business ontology* which focuses on *key business abstractions* such as *resource exchanges* and *conversions* that underlie seemingly different business processes.

We developed a prototype of our approach that relies on available standards and open source software. We also conducted experiments with expert analysts and graduated students to validate the *soundness* of the *conceptual ingredients* that underlie our approach. Notwithstanding the small size of our experimental data set, the results support the soundness of the approach. Much remains to be done, both in terms of core functionality, dealing with *confluent* transformations [8], and in terms of *usability* before we can make this into a tool that process analysts will readily use.

References

1. W. M. Carlson, "Business information analysis and integration technique (BIAIT): the new horizon," *ACM SIGMIS Database*, vol. 10, no. 4, pp. 3–9, 1979.

2. T. W. Malone, K. Crowston, and G. Herman, *Organizing Business Knowledge: The MIT Process Handbook*. MIT Press, 2003.
3. P. Coad, J. De Luca, and E. Lefebvre, *Java Modeling In Color With UML: Enterprise Components and Process*. Prentice Hall PTR, 1999.
4. A. Hallerbach, T. Bauer, and M. Reichert, "Capturing variability in business process models: the Provop approach," *Journal of Software Maintenance and Evolution: Research and Practice*, vol. 22, no. 6-7, pp. 519–546, 2010.
5. A. Schnieders and F. Puhlmann, "Variability Mechanisms in E-Business Process Families," in *Proceedings of the 9th International Conference on Business Information Systems (BIS'06)*, pp. 583–601, 2006.
6. B. Curtis, M. I. Kellner, and J. Over, "Process modeling," *Commun. ACM*, vol. 35, pp. 75–90, Sept. 1992.
7. P. Wohed, "Conceptual Patterns-A Consolidation of Coad's and Wohed's Approaches," in *Natural Language Processing and Information Systems*, vol. 1959, pp. 340–351, Springer Berlin Heidelberg, 2001.
8. A. Leshob, *Representating, classifying and specializing business processes*. PhD thesis, UQAM, Montreal, 2013.
9. W. E. McCarthy, "The REA accounting model: A generalized framework for accounting systems in a shared data environment," *Accounting Review*, vol. 57, no. 3, pp. 554–578, 1982.
10. P. Hruby, *Model-Driven Design Using Business Patterns*. Berlin/Heidelberg: Springer-Verlag, 2006.
11. S. C. Yu, *The Structure of Accounting Theory*. The University Press of Florida, 1976.
12. G. Geerts and W. McCarthy, "The ontological foundation of REA enterprise information systems," *Annual Meeting of the American Accounting Association*, no. March, pp. 1–34, 2000.
13. T. A. Curran, G. Keller, and A. Ladd, *SAP R/3 Business Blueprint: Understanding the Business Process Reference Model*. Prentice Hall, 1998.
14. OMG, "Business Process Definition MetaModel (BPDM)," 2008.
15. OMG, "Organization Structure Metamodel (OSM)," 2006.
16. M. Murray, *Discover Logistics with SAP ERP*. SAP PRESS, 2009.
17. J. S. David, "Three Events That Define an REA Methodology for Systems Analysis, Design, and Implementation," in *Proceedings of the Annual Meeting of the American Accounting Association*, (Dallas, TX), 1997.
18. W. M. P. van der Aalst, A. K. A. de Medeiros, and A. J. M. M. Weijters, "Process equivalence: Comparing two process models based on observed behavior.," in *Business Process Management* (S. Dustdar, J. L. Fiadeiro, and A. P. Sheth, eds.), vol. 4102 of *Lecture Notes in Computer Science*, pp. 129–144, Springer, 2006.
19. J. Gordijn and H. Akkermans, "Designing and Evaluating E-Business Models," *IEEE Intelligent Systems*, vol. 16, no. August, pp. 11–17, 2001.
20. H. A. Reijers, M. Dumas, W. M. P. van der Aalst, and A. H. M. ter Hofstede, "Process Design and Redesign," in *Process-Aware Information Systems: Bridging People and Software through Process Technology*, pp. 207–234, New Jersey: John Wiley & Sons, 2005.
21. G. Engels, R. Heckel, and M. K. Jochen, "Consistency-Preserving Model Evolution through Transformations," in *<UML 2002>-The Unified Modeling Language*, pp. 212–227, Springer Berlin Heidelberg, 2002.
22. D. Varró and A. Pataricza, "Automated Formal Verification of Model Tranformations," in *Critical Systems Development in UML*, no. Otka 038027, pp. 63–78, Technische Universität München, 2003.