Coordination Issues in Artifact-Centric Business Process Models

Giorgio Bruno

Politecnico di Torino, Torino, Italy
giorgio.bruno@polito.it

Abstract. In recent years, research in the field of business processes has shown a shift of interest from the activity-centric perspective to the artifact-centric one. The benefits, such as improved communication among the stakeholders and higher potential for flexibility, come from the focus on the key business entities (called artifacts) and on the distribution of the control flow in the life cycles of the artifacts. However, this perspective also entails a number of challenges, such as the coordination between the life cycles of the artifacts. This paper proposes an approach based on correlated transitions, i.e., transitions that belong to different life cycles and must be performed jointly. A new notation called Acta is illustrated with the help of two motivating examples.

Keywords: business processes, artifacts, life cycles, correlations, tasks.

1 Introduction

While it is commonly accepted that a business process is a standard way of organizing work in a business context, the consent on what the starting point in the investigation of the intended business should be is no longer unanimous.

According to the traditional definitions, a business process consists of a number of tasks designed to produce a product or service and is meant to cross functional boundaries in that it may involve members of different departments [5]. This point of view has spurred a fruitful line of research, now labeled as activity-centric, and a standard notation, BPMN [2].

In recent years, the notion of PAIS (Process-Aware Information System) [6], which advocates a tighter integration between the areas of information systems and business processes, has brought about a shift of interest from the activity-centric perspective to the artifact-centric one. The latter emphasizes the identification of the key business entities (called artifacts) and of their life cycles, which show how the artifacts evolve over time through the execution of business operations (a.k.a. tasks).

The analysis (reported in [4]) of the operations of a global financing division resulted in a high-level model consisting of 3 major artifact types, whose life cycles include 18 states and approximately 65 tasks. The major benefit is the right level of granularity, which facilitates communication among the stakeholders and helps them focus on the primary purposes of the business.

H. Decker et al. (Eds.): DEXA 2013, Part I, LNCS 8055, pp. 209–223, 2013.

The major criticism raised against the activity-centric approach is the emphasis placed on the tasks and the control-flow elements, while the business entities are not considered as first-class citizens. As a matter of fact, the data flow in the activity-centric approach is based on process variables and there is no automatic mapping between business entities and process variables. Moreover, this perspective seems to be more suitable for automated processes than for human-centric ones, as it lacks an adequate representation of the situations in which different courses of action are possible and the choices depend on human decisions. The artifact-centric approach has the potential for coping with such issues owing to the emphasis placed on the business entities on which human decisions are grounded.

A critical aspect in the artifact-centric approach is the handling of tasks (called spanning tasks) operating on two or more artifacts in that some form of coordination of their life cycles is needed. This paper grounds the coordination of the life cycles on the notion of correlated transitions, i.e., transitions that belong to different life cycles and must be performed jointly. Three kinds of correlation are examined: they are referred to as generative, selective and direct correlation.

A new notation called Acta (Artifacts, Correlations and TAsks) is also presented. Acta models are made up of two components, the structural model and the dynamic one. The structural model basically shows the artifacts in terms of their properties and the dynamic model is the collection of their life cycles.

Another important point is the impact that the artifact-centric approach has on the structure of work lists. This paper presents a solution in which the work lists are organized on the basis of the artifacts their owners are in charge of.

This paper is organized as follows. Section 2 is an overview of Acta and of coordination issues; section 3 describes two motivating examples concerning a purchase requisition process and a negotiation one. Section 4 illustrates structural models and section 5 analyzes the correlated transitions needed in the dynamic models of the examples. Section 6 presents the structure of work lists, section 7 discusses the related work and section 8 provides the conclusion of the paper.

2 Acta Models and Coordination Issues

Acta models are made up of two components, the structural model and the dynamic one. The structural model shows the entities involved in a business process, in terms of their types, associations and attributes. Emphasis is placed on mandatory relationships and associative attributes, which play an important part in the specification of the tasks, as will be illustrated in the next sections.

The dynamic model of a business process is the collection of the life cycles of the artifact types needed. A life cycle defines the path to be followed over time by any artifact of the type under consideration from the initial state to a final one, and it is represented as a state-transition diagram. The states represent the stages in the path and the transitions cause the artifact to move from the current state to the next one. In this paper, transitions denote human tasks, i.e., actions to be carried out by persons who participate in the process by playing specific roles. Transitions may also be associated with automatic tasks.

A major issue in the artifact-centric perspective is the coordination of the life cycles of the business entities involved. Coordination in Acta is achieved through correlated transitions, i.e., transitions that belong to different life cycles and must be performed jointly.

A group of correlated transitions is the result of a unit of work, called spanning task, which affects artifacts of different types. Three kinds of correlation are examined in this paper: generative, selective and direct correlation. A short account is given in this section, while a more detailed description is provided in the next sections with the help of the motivating examples.

In a group of correlated transitions, one is the master transition and the others are the subordinate transitions; the spanning task is associated with the master transition. The artifact acted on by the master transition is called master artifact and the artifacts affected by the subordinate transitions are called subordinate artifacts.

When the effect of a spanning task is the generation of new artifacts, generative correlation takes place. The correlated transitions are the master transition, i.e., the one affecting the master artifact, and the initial transitions in the life cycles of the new artifacts. Generation acts are usually remembered through associations between the master artifact and the subordinate ones. For example, an account manager who is not able to directly fulfill a purchase requisition coming from a customer generates a number of requests for quote for different suppliers: the spanning task enterRequestsQ operates on a purchase requisition (the master artifact) so as to produce new requests for quote (the subordinate artifacts). The purchase requisition changes state and the requests for quote enter the initial state; in addition, the requests for quote are associated with the purchase requisition.

Selective correlation means that correlation stems from a selection of artifacts. For example, the account manager selects one request for quote from among those that have been fulfilled by the suppliers so as to fulfill the purchase requisition. Both the request for quote and the purchase requisition change state; the selection is remembered by means of a new association established between the artifacts. Selective correlation may be additive or constitutive. In the first case, a new association is established between artifacts, such as the purchase requisition and the request for quote, already connected. In the second case, the artifacts become connected for the first time; an example is a spanning task that enables a broker to match a request and an offer, and the result is a connection between two artifacts that before were unrelated.

Direct correlation means that a spanning task can also bring about a change of state for the artifacts that are not input artifacts but are associated with the input ones. For example, a confirmed purchase requisition makes the account manager produce a confirmed request for quote: the request for quote is the one associated with the input purchase requisition.

Acta models are meant to be conceptual models: tasks specify their intended effects by means of post-conditions, while constraints are expressed by pre-conditions and selections by selection rules.

3 Description of the Examples

This section presents a short description of two motivating examples, whose models will be illustrated in the next sections. The description focuses on the most important aspects and disregards the details, such as the attributes of the entities.

3.1 Purchase Requisition Process

The process enables a selling organization, referred to as the seller, to carry out commercial transactions with its partners (customers and suppliers). Three roles are involved, account manager, customer and supplier. Account manager is an internal role of the seller while the other roles designate external users acting on behalf of the partners. The process runs on a platform that enables all the users to operate on the same information system.

Customers enter purchase requisitions to get the prices for the goods they need. The purchase requisitions are handled by the account managers. The process assumes that each customer is served by one specific account manager, who may serve several customers. While processing a purchase requisition, an account manager has three options: they may fulfill it, reject it or involve a number of suppliers.

In the third case, they enter a number of requests for quote, each one directed to a different supplier; this is a case of generative correlation. Suppliers may fulfill a request for quote or may reject it. The account manager may select one request for quote from among those fulfilled (additive selective correlation) so as to fulfill the corresponding purchase requisition, or they may give up and reject the purchase requisition. In addition, they must withdraw all the requests for quote fulfilled except for the one selected, if any.

The customer may confirm a fulfilled purchase requisition or may withdraw it. If the purchase requisition is based on a request for quote, the account manager is in charge of confirming or withdrawing it, respectively; this is a case of direct correlation.

The details regarding the attributes of the entities are ignored; however, the general meaning of the above-mentioned actions can be inferred from the verbs used. Therefore, fulfilling a request implies providing the information required, e.g., the prices of the goods needed by a customer; confirming (or withdrawing) a request means that the requester is satisfied (or dissatisfied) with the information provided by the recipient of the request.

3.2 Negotiation Process

The process enables brokers to manage sales campaigns. After a broker has started a campaign, sellers may enter offers of products and buyers may enter requests for products. The broker may match one or more requests and one or more offers to produce a transaction; the matching rule is not considered. This is a case of constitutive selective correlation: the selection is made from among peer entities and there are no direct connections between them.

The broker may reject requests and offers at their discretion. After some time, the campaign is closed by the broker.

4 Structural Models

Structural models show the entities involved in business processes, in terms of their types, associations and attributes. They extend UML class models with the purpose of emphasizing mandatory relationships and associative attributes, which play an important part in the specification of the tasks, as will be illustrated in the next section.

Relationships are connections between pairs of types; they represent the associations that may exist between the instances of the types involved. Cardinalities place constraints on the number of connections between instances. The standard cardinality is many-to-many and indicates that any entity of one type may be connected to a number of entities of the other type; this number is not predetermined and may be zero as well.

Associations may be mandatory on one side and optional on the other; in this case, the relationship is shown as an oriented link whose origin is the type for which the association is mandatory.

Mandatory relationships are important in that they determine which associations have to be set when a new entity is generated. Depending on the multiplicity of the relationship, the newly generated entity will be connected to one or more entities of the destination type. On the destination side of a mandatory relationship, the default multiplicity is one; on the source side it is 0 to many.

The structural model of the purchase requisition process is shown in Fig.1a. Types Customer, Supplier and AccountMgr represent the participants in the process in terms of the roles they play; such types are referred to as role types. Types PurchaseR and RequestQ represent purchase requisitions and requests for quote, respectively.

Relationships imply attributes called associative attributes in the entities involved. Such attributes refer to single entities or collections of entities depending on the cardinalities of the corresponding relationships. The names of the associative attributes may be omitted and in such cases they take the name of the type they refer to, in the singular form or in the plural one (with the initial in lower case) depending on the cardinality of the relationship. The name of a relationship is obtained from the names of the types involved (if there are no other relationships between the same pair of types) or from the names of the corresponding associative attributes (either explicit or implicit) in alphabetic order. For example, the name of the relationship between AccountMgr and Customer is accountMgr-customers or AccountMgr-Customer.

The meaning of the mandatory relationships appearing in Fig.1a is as follows. Purchase requisitions are generated by customers, and then a relationship (Customer-PurchaseR) is needed to identify the generator of each purchase requisition. Relationship AccountMgr-Customer connects a customer with the account manager who will take care of their purchase requisitions.

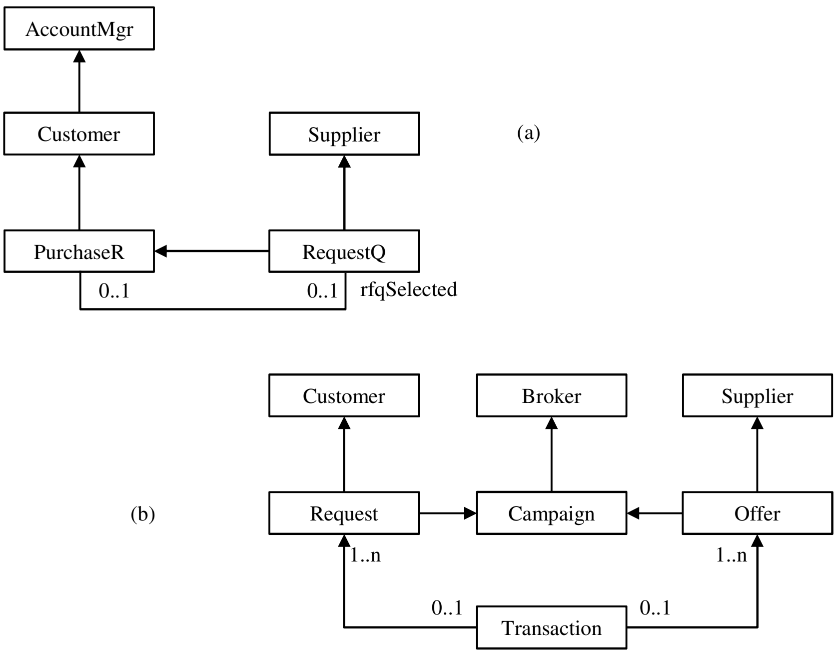


Fig. 1. The structural models of the processes: purchase requisition (a) and negotiation (b)

Requests for quote are complements of purchase requisitions; then when a new request for quote is generated, it is associated with the purchase requisition it is a complement of, and it is also connected to the supplier entity representing the recipient of the request for quote. The relationships involved are purchaseR-requestsQ and supplier-requestsQ, respectively; term requestsQ is the implicit associative attribute on the RequestQ side. Relationship purchaseR-rfqSelected associates a purchase requisition with the request for quote that has been selected from among those fulfilled (if any).

Associative attributes are essential for navigational purposes as will be illustrated in the next section. For example, if pr denotes a certain purchase requisition, navigational expression "pr.customer.accountMgr" returns the AccountMgr entity related to the Customer entity associated with the entity denoted by pr. The syntax of navigational expressions in Acta is based on a simplified version of OCL [16].

In the negotiation process, whose structural model is shown in Fig.1b, types Broker, Customer and Supplier represent the participants. Campaigns are managed by brokers and such associations are represented by relationship Campaign-Broker. Transactions are made up of a number of requests and a number of offers; however, since requests and offers may be rejected, the cardinality of the relationships Request-Transaction and Offer-Transaction is 0 or 1 on the transaction side.

5 Dynamic Models

This section presents the dynamic models of the motivating examples with the purpose of illustrating the three kinds of correlation introduced in section 2. The dynamic model of a business process is the collection of the life cycles of the artifact types needed; for convenience they are shown in the same figure. The dynamic models of the purchase requisition process and of the negotiation process are shown in Fig.2 and in Fig.3, respectively. The following subsections provide the general features of the Acta notation, illustrate initial tasks and post-conditions, and present the examples of the correlations introduced in section 2.

5.1 General Features of the Acta Notation

Tasks are associated with transitions. Human tasks show the role required, or its acronym, before the task name, e.g. “AM: enterRequestsQ”. Automatic tasks have no role indications. Spanning tasks appear in two or more life cycles and group a number of correlated transitions. The names of such transitions are identical; however, the master transition can be distinguished from the subordinate ones in that its name appears in bold while the other names are shown in italics. Moreover, role names only appear on master transitions.

The process model specifies the effects and the constraints of the tasks by means of post-conditions and pre-conditions, respectively; they are based on the structure of the entities provided by the information model and are expressed with a simplified form of OCL [16].

A human task is assigned to one role, but several participants may play the same role; it is then necessary to determine whether any participant playing the role required is entitled to perform the task or a specific participant is needed. The performer is generic in the first case, and specific in the second one. The performer of task enterPurchaseR in Fig.2 is generic, as indicated by the qualifier (any) following the role acronym. The performers of all the other human tasks are specific. The specificity is determined by a connection, either direct or indirect, between the role type and the artifact type. The account manager in charge of a given purchase requisition is not a generic one, but the one associated with the customer who issued the purchase requisition. In general, a rule is needed to specify the desired connection; however, in simple situations like those addressed in this paper, there is no need to express these rules explicitly. The simplification is due to the assumption that the connections between roles and artifact types are based on chains of mandatory relationships; what is more, such chains are assumed to be unique. As a matter of fact, on the basis of the information model shown in Fig.1a, there is only one such path from type PurchaseR to type AccountMgr and it consists of types PurchaseR, Customer and AccountMgr.

If a final state, i.e., a state having no output transitions, implies a notification for a specific participant, the role name (or acronym) appears after the state name. For example, when a purchase requisition enters state rejected (shown in Fig.2), a notification has to be sent to the appropriate customer; the same rules introduced for the identification of the task performers apply in this case as well.

5.2 Initial Tasks and Post-Conditions

The purchase requisition process consists of the purchase requisition (PurchaseR) life cycle and the request for quote (RequestQ) one.

Purchase requisitions are entered by customers when they want to. An initial task is then needed: it is named enterPurchaseR and is associated with the initial transition, i.e., the one entering the initial state and having no source icon. Its post-condition "new PurchaseR" appears in the task description section below the life cycle. The "new" operator asserts that a number of new entities of the type specified will exist after the execution of the task. If the multiplicity is 1, it is omitted; multiplicity n (cf. task enterRequestsQ) means that the number is decided by the performer of the task.

Due to mandatory relationship Customer-PurchaseR, the newly generated artifact needs to be connected with a customer entity. The partners of mandatory relationships are automatically searched for among the entities forming the context of the task. In general, the context of a task includes the current artifact, the entity representing the performer of the task (referred to as the performer entity) and the entities selected by the performer. The context of task enterPurchaseR consists of the performer entity only, which, however, fits the requirements of the relationship in that it is a customer entity; therefore, the newly generated purchase requisition will be connected to this entity.

5.3 Generative Correlation

A purchase requisition in the initial state is handled by an account manager who may reject or fulfill it, or enter a number of requests for quote directed to suitable suppliers.

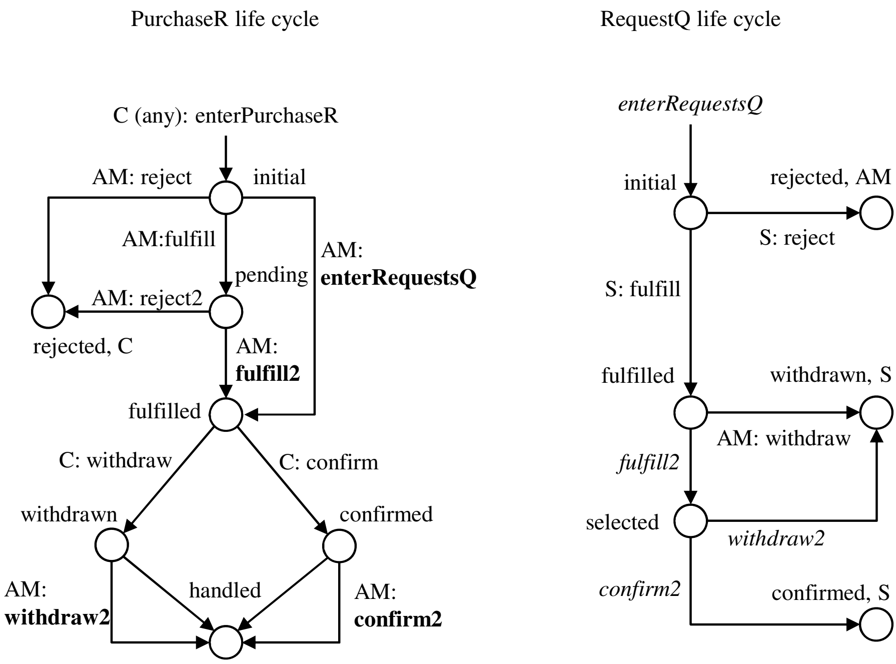
Task enterRequestsQ performs a generative correlation. This effect is implied by post-condition "new n RequestQ". The mandatory relationships related to type RequestQ require a new request for quote to be connected to two entities, i.e., a purchase requisition and a supplier entity. The former entity is matched by the current artifact but for the latter entity the context of the task provides no match; in such cases, it is up to the performer to choose a suitable entity from among those available. The selection may be subjected to the constraints indicated in the requirements, if any.

At the end of the task, the purchase requisition is moved in the pending state.

5.4 Additive Selective Correlation

A newly generated request for quote may be fulfilled or rejected by the supplier. When the purchase requisition is in the pending state, the account manager may reject it with task reject2 or they may fulfill it with task fulfill2 provided that a request for quote is chosen from among those fulfilled by the suppliers.

Task fulfill2 carries out an additive selective correlation in that the quote to be chosen is already connected with the purchase requisition. The master transition is in the PurchaseR life cycle.



Abbreviations for roles
C = Customer; AM = AccountMgr.
Tasks
enterPurchaseR:: post: new PurchaseR.
enterRequestsQ:: post: new n RequestQ.
fulfill2:: with requestQ in purchaseR.requestsQ; post: purchaseR.rfqSelected == requestQ.
confirm2:: pre: purchaseR.rfqSelected != null; with requestQ as purchaseR.rfqSelected.
withdraw2:: pre: purchaseR.rfqSelected != null; with requestQ as purchaseR.rfqSelected.

Abbreviations for roles
S = Supplier; AM = AccountMgr.

Fig. 2. The dynamic model of the purchase requisition process

The description of task fulfill2 includes two parts, i.e., the selection rule and the post-condition. The selection rule “with requestQ in purchaseR.requestsQ”, introduced by keyword with, indicates that one entity of type RequestQ is needed and it is to be selected from among those associated with the input purchase requisition. As a general rule, the entities selected by the performer become part of the context of the task. The state of the requests for quote is not explicitly indicated as it can be found in their life cycle; in fact, it is the input state (fulfilled) of the subordinate transition fulfill2. The selection basis, which is the collection of the entities from among which the choice has to take place, may be empty and in this case the spanning task is not enabled.

The post-condition “purchaseR.rfqSelected == requestQ” indicates that the choice is recorded by means of a new association based on relationship purchaseR-rfqSelected. Since requestQ is the term used in the selection rule to denote the request for quote chosen by the performer, the associative attribute rfqSelected will refer to that request for quote.

A request for quote in state fulfilled may be withdrawn by the account manager; the reason is to prevent the requests for quote not selected from remaining blocked in this state.

5.5 Direct Correlation

The customer may confirm or withdraw a fulfilled purchase requisition, whose state becomes confirmed or withdrawn, respectively.

If the purchase requisition is based on a request for quote, the account manager is in charge of confirming or withdrawing it with tasks confirm2 or withdraw2, respectively. These tasks must be performed only if there is a request for quote associated with the input purchase requisition, i.e., if associative attribute rfqSelected is not null. The conditional nature of the tasks is indicated by the pre-condition "purchaseR.rfqSelected != null". If the pre-condition is false the alternative transitions without labels (which are automatic transitions) are carried out so as to bring the purchase requisition to final state "handled".

Tasks confirm2 and withdraw2 provide an example of direct correlation: if they are enabled, they also act on the request for quote which is obtained from the input purchase requisition through the associative attribute rfqSelected, as expressed by the selection rule "with requestQ as purchaseR.rfqSelected". The request for quote is then moved to state confirmed or to state withdrawn.

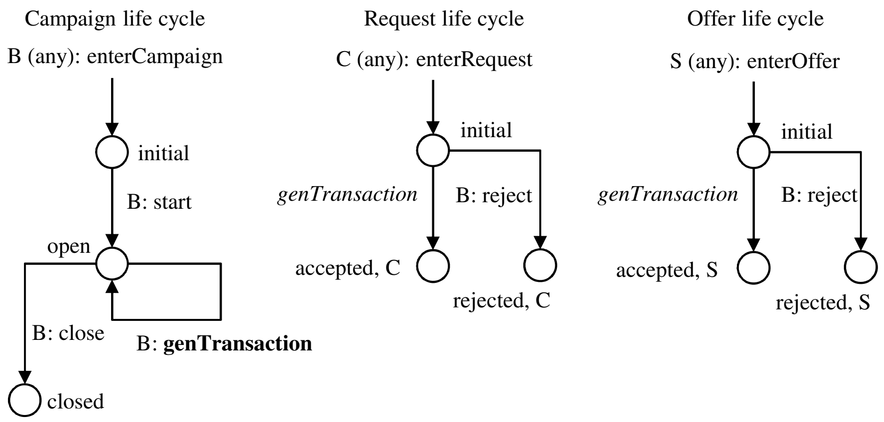
5.6 Constitutive Selective Correlation

The negotiation process whose dynamic model and information one are shown in Fig.3 and in Fig.1b, respectively, presents an example of constitutive selective correlation.

The process consists of the Campaign, Offer and Request life cycles. When a campaign is open, suppliers may enter offers and customers may enter requests. They do so with the initial tasks enterOffer and enterRequest, respectively.

Generating a new offer or a new request implies submitting it to an open campaign; therefore the generation of these artifacts takes place within a context, which is determined by the choice of an open campaign, this choice being made by the performer of the task. For this reason, the descriptions of the tasks include the selection rule "with campaign (lifecycleState == open), which means that their performers have to select an open campaign as a contextual entity for the newly generated artifact. The term "lifecycleState" denotes a system attribute that provides the name of the current state in the life cycle of the artifact under consideration.

The broker can reject offers and requests or combine them into transactions. Task genTransaction carries out a constitutive selective correlation in that no previous connections exist between the artifacts combined in the newly generated transaction. The requirements specify no matching rule for the selection of offers and requests; if, instead, one is given, it will be expressed as a pre-condition of the task.



Abbreviations for roles
B = Broker.
Tasks
genTransaction::
with offers in campaign.offers,
with requests in
campaign.requests;
post: new Transaction.

Abbreviations for roles
B = Broker, C = Customer.
Tasks
enterRequest::
with campaign (lifecycleState
== open); post: new Request.

Abbreviations for roles
B = Broker, S = Supplier.
Tasks
enterOffer::
with campaign (lifecycleState
== open); post: new Offer.

Fig. 3. The dynamic model of the negotiation process

6 Artifact-Centric Work Lists

Business processes are means to organize work and their models should then show what the units of work are and to which roles they are entrusted. During their execution, the units of work are assigned to the appropriate participants through their work lists.

However, the organization of the work lists is not independent of the approach adopted for the representation of the business processes. With activity-centric notations, such as BPMN, the entries of the work lists draw on the tasks defined in the process models and then their labels follow the pattern “task-name info” where info stands for information taken from the task parameters. By clicking on an entry, a participant may perform the corresponding task through the graphical interface provided by the implementation.

In artifact-centric notations, instead, the focus is on the artifacts and then it is natural to structure the work lists on the basis of the artifacts that their owners are in charge of. This section discusses the issue with reference to the examples illustrated in the previous sections.

The content of work lists is a kind of viewpoint that the participants in the process are provided with on the artifacts they are in charge of. The viewpoint includes the options currently available and it changes on the basis of the decisions taken by the participants. One of the challenges is the identification of the most expressive technique to help participants work in this way. The Acta approach suggests replacing textual entries with a more structured representation stressing the distinction between master artifacts and subordinate ones.

An example is given in Fig.4 with reference to the purchase requisition process. A work list for a certain account manager is presented; Fig.4a shows the content of the work list before task fulfill2 is performed and Fig.4b shows it immediately after.

The work list is organized in two columns, the left one referring to master artifacts and the right one to subordinate artifacts. The master column includes two purchase requisitions, pr2 in the initial state and pr1 in the pending state; pr1 and pr2 represent identifiers. Each entry contains three major fields: the identifier, the life cycle state and the options available. Artifact pr2 is in the initial state and hence there are no subordinate items; the performer may select one option out of three. On the contrary, purchase requisition pr1 has three subordinate requests for quote, i.e., rfq1 and rfq3 in state fulfilled and rfq2 in state rejected. For the last one, there is no option available, while the others may be rejected or selected.

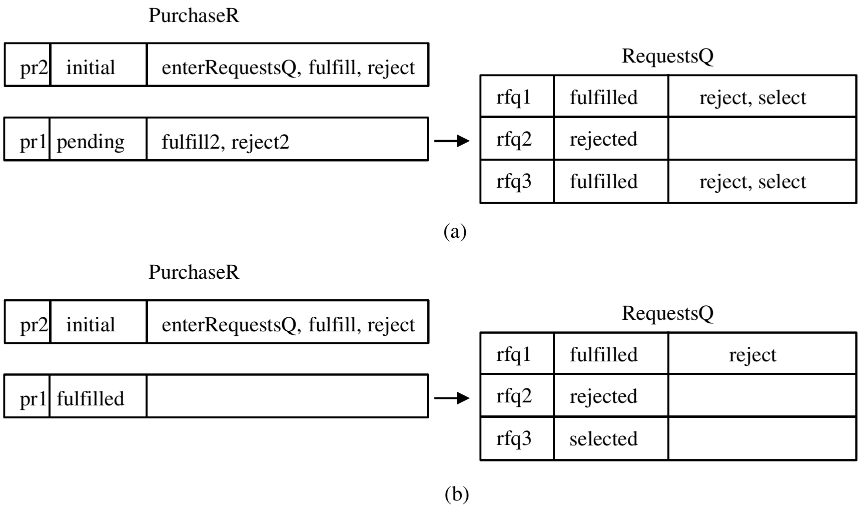


Fig. 4. An example of work list before (a) and after (b) fulfill2 is performed

The select option appears when an artifact, such as rfq1 or rfq3, may be involved in a selective correlation through a subordinate transition. Since a subordinate transition can only take place in conjunction with the corresponding master transition, the select option enables the performer to mark a subordinate artifact before carrying out the spanning task. As a matter of fact, if the account manager wants to fulfill purchase requisition pr1 with the help of request for quote rfq3, first they select rfq3 and then they choose the fulfill2 option on pr1. At the end of the task, the state of pr1 is fulfilled and no options are available because this state is handled by the customer who issued the purchase requisition. The request for quote selected, i.e., rfq3, is in the selected state and no options are available because the output transitions of this state are directly correlated with their master transitions; when the master artifact is acted on, the subordinate artifacts are acted on as well. For the request for quote rfq1, the select option is no longer available as the master artifact is no longer in the pending state.

7 Related Work

According to Sanz [17], the roots of the artifact-centric perspective can be found in past research on entity-based dynamic modeling, but only in recent years, the core ideas have permeated the discipline of Business Process Management.

Term artifact has been introduced in [15] to designate a concrete and self-describing chunk of information that business people use to run a business. The artifact types and their life cycles come from experience and show how the actual entities evolve over time: the business activities, which are responsible for the state transitions, are introduced in a subsequent step of analysis along with the business rules governing their execution.

In the case-handling approach [1], a process is meant to take care of a specific entity type (e.g., an insurance claim), called the process case: the purpose is to improve the flexibility of the control flow as the process evolution depends on the state of the case and not only on the tasks performed [10].

The BALSA framework [7] builds on the notion of artifact and adds services, which encapsulate units of work acting on one or more artifacts, and associations, which specify various kinds of constraints for the services.

In the artifact-centric approach, there are three major issues to cope with, i.e., structure, dynamics and coordination. Structure is about the properties (attributes and associations) of the artifacts involved, dynamics encompass the artifact life cycles and coordination is concerned with the synchronization of the life cycles. Such issues are dealt with in various ways, ranging from separate models to holistic ones.

The Guard-Stage-Milestone (GSM) approach [8] is a holistic technique: the major building blocks are the artifacts, which contain informational aspects (attributes and associations), life cycles and coordination items (events and rules). The drawback is the difficulty of understanding the propagation of the events between the life cycles. For example, an activity performed on an entity can produce an event that is targeted at another entity and triggers an activity affecting this entity. The second activity then turns out to be correlated with the first one.

In ArtiNet [9] and Chant [3], the life cycles are integrated in one model and coordination is obtained with transitions operating on two or more artifacts. Models are monolithic in that they are based on Petri nets where places represent artifact states and tokens denote artifacts.

In other approaches, where the relationships between artifact types are explicitly defined, coordination takes advantage of them, in particular of hierarchical relationships. Hierarchical structures, such as those related to physical systems, are addressed by COREPRO [14], which provides specific means to achieve mutual synchronization between the state of a compound object and those of its components. The approach presented in [12] is aimed at automatically generating a process model from the entity life cycles provided that the synchronization points are manually identified beforehand.

The PHILharmonicFlows approach [11] is based on micro processes and macro processes; the former define the life cycles of the artifacts and the latter provide a coordination mechanism consisting of macro steps and macro transitions. A macro step is associated with an artifact type and a particular state of its life cycle; at run time, it refers to the artifacts being in that state. A macro transition activates an output macro step only when the artifacts collected in the input macro steps satisfy certain conditions, which are related to the structure of the artifacts (in particular to the associations). In Acta, coordination is carried out by the spanning tasks, which are included in the life cycles and then no separate coordination model is needed.

With the Proclets framework [13], life cycles can be defined in separate building blocks, called Proclets, equipped with ports through which they can send and receive messages. The drawback is to address coordination with notions, such as messages and send/receive operations, which are too close to the programming domain.

8 Conclusion and Future Work

The artifact-centric approach is a promising viewpoint on business processes in that it places emphasis on the life cycles of the major business entities involved. However, it must face a number of challenges, such as the coordination of the life cycles and the handling of tasks spanning two or more life cycles.

This paper has analyzed these challenges on the basis of a new notation named Acta and with the help of two motivating examples. The main contribution of this paper is the notion of correlated transitions along with their classification in three major kinds, i.e., generative, selective and direct correlations. This notion is grounded on the distinction between master transitions and subordinate ones; the corresponding artifacts are called master artifacts and subordinate ones, respectively. Tasks spanning two or more life cycles determine a group of correlated transitions operating on one master artifact and on a number of subordinate artifacts. A spanning task is defined in the life cycle of the master artifact and its name appears in the labels of the subordinate transitions.

The artifact-centric approach has greater potential than the activity-centric one for coping with human-centric processes in which different courses of action are possible and the choices depend on human decisions. For this reason, it is essential to work out new structures for the work lists. This paper has illustrated one in which the artifacts are shown along with the options compatible with their states and the spanning tasks take advantage of a visual representation emphasizing the connections between the master artifacts and the subordinate ones.

The Acta notation is a proof of concept and current work is devoted to the definition of a suitable support environment.

Acknowledgements. The author wishes to thank the anonymous reviewers for their helpful comments.

References

1. van der Aalst, W.M.P., Weske, M., Grünbauer, D.: Case handling: a new paradigm for business process support. Data & Knowledge Engineering 53, 129–162 (2005)
2. BPMN, Business Process Model and Notation, V.2.0, http://www.omg.org/spec/BPMN/ (retrieved May 13, 2013)
3. Bruno, G.: Combining information and activities in business processes. In: Liddle, S.W., Schewe, K.-D., Tjoa, A.M., Zhou, X. (eds.) DEXA 2012, Part II. LNCS, vol. 7447, pp. 481–488. Springer, Heidelberg (2012)
4. Chao, T., et al.: Artifact-Based Transformation of IBM Global Financing. In: Dayal, U., Eder, J., Koehler, J., Reijers, H.A. (eds.) BPM 2009. LNCS, vol. 5701, pp. 261–277. Springer, Heidelberg (2009)
5. Davenport, T.H., Short, J.E.: The new industrial engineering: information technology and business process redesign. Sloan Management Review, 11–27 (Summer 1990)
6. Dumas, M., van der Aalst, W.M.P., ter Hofstede, A.H.M.: Process-Aware Information Systems: bridging people and software through process technology. Wiley, New York (2005)
7. Hull, R.: Artifact-centric business process models: Brief survey of research results and challenges. In: Meersman, R., Tari, Z. (eds.) OTM 2008, Part II. LNCS, vol. 5332, pp. 1152–1163. Springer, Heidelberg (2008)
8. Hull, R., et al.: Introducing the Guard-Stage-Milestone approach for Specifying Business Entity Lifecycles. In: Bravetti, M., Bultan, T. (eds.) WS-FM 2010. LNCS, vol. 6551, pp. 1–24. Springer, Heidelberg (2011)
9. Kucukoguz, E., Su, J.: On lifecycle constraints of artifact-centric workflows. In: Bravetti, M., Bultan, T. (eds.) WS-FM 2010. LNCS, vol. 6551, pp. 71–85. Springer, Heidelberg (2011)
10. Künzle, V., Reichert, M.: Towards object-aware process management systems: Issues, challenges, benefits. In: Halpin, T., Krogstie, J., Nurcan, S., Proper, E., Schmidt, R., Soffer, P., Ukor, R. (eds.) BPMDS 2009 and EMMSAD 2009. LNBIP, vol. 29, pp. 197–210. Springer, Heidelberg (2009)
11. Künzle, V., Reichert, M.: PHILharmonicFlows: towards a framework for object-aware process management. Journal of Software Maintenance and Evolution: Research and Practice 23(4), 205–244 (2011)
12. Küster, J.M., Ryndina, K., Gall, H.C.: Consistency of business process models and object life cycles. In: Kühne, T. (ed.) MoDELS 2006. LNCS, vol. 4364, pp. 80–90. Springer, Heidelberg (2007)
13. Mans, R.S., et al.: Proclets in healthcare. Journal of Biomedical Informatics 43(4), 632–649 (2010)
14. Müller, D., Reichert, M., Herbst, J.: Data-driven modeling and coordination of large process structures. In: Meersman, R., Tari, Z. (eds.) OTM 2007, Part I. LNCS, vol. 4803, pp. 131–149. Springer, Heidelberg (2007)
15. Nigam, A., Caswell, N.S.: Business artifacts: an approach to operational specification. IBM Systems Journal 42(3), 428–445 (2003)
16. OCL, Object Constraint Language, V.2.3.1, http://www.omg.org/spec/OCL/ (retrieved May 13, 2013)
17. Sanz, J.L.C.: Entity-centric operations modeling for business process management - A multidisciplinary review of the state-of-the-art. In: 6th IEEE International Symposium on Service Oriented System Engineering, pp. 152–163. IEEE Press, New York (2011)