Supporting the Diversity of B2B E-Contracting Processes

Samuil Angelov and Paul Grefen

ABSTRACT: Contracts specify the rights and obligations of the contracting parties. Information technologies can make e-contracting more efficient and effective. A process model for flexible B2B e-contracting is proposed that separates concerns by distinguishing function and communication perspectives of e-contracting processes supplemented with activity constraints. The model supports the elaboration of fully detailed e-contracting process specifications, coherent communication between contracting parties, and seamless coupling between the internal and external activities of a company.

KEY WORDS AND PHRASES: Business-to-business e-contracting, e-contracting process model, flexible e-contracting.

Business process modeling aims at improving the efficiency and effectiveness of business processes executed in a company. Software applications with different levels of complexity are provided to coordinate and automate intra-organizational processes. Until recently, software applications modeling and supporting business processes were intra-organizational and thus limited to the scope of one company. With the rapid development of information technology, the possibility for supporting cross-organizational business processes has emerged. Research and standardization efforts have concentrated on modeling and realizing cross-organizational processes by supporting information systems [1, 9, 17, 26].

Business-to-business (B2B) contracting encompasses a collection of coherent intra- and cross-organizational activities. To the extent allowed by the level of information technology, there have been many attempts to realize information systems that can support contracting processes [6, 7, 17, 18, 24]. Unfortunately, none of these systems has the flexibility to support the complete range of contracting processes. This is mainly because of their reliance on fixed, structured models of the contracting process for automated contracting support. Contracting processes are far too complex and variable to be addressed through traditional process specifications. Consequently existing systems only support context-specific contracting processes and provide little flexibility in business relations.

This paper defines a general model for B2B e-contracting that can specify diverse and flexible e-contracting processes. The proposed model has three elements: a function perspective, a communication perspective, and constraint operators. The function perspective provides descriptions of e-contracting activities at multiple levels of abstraction. It supports process designers in the selection and specification of contracting activities. The communication perspective only provides descriptions of communication activities. It is used to define the communication activities in the function perspective. It also supports the proper configuration of communication activities in contracting process specifications. The operators are used to define constraints on the execution

of contracting activities. The model facilitates the definition of e-contracting process specifications that provide the necessary foundation for flexible support of contracting relations by contracting information systems. The model can also be used to analyze existing e-contracting process specifications. The discussion in this paper is based on ideas presented in [2].

Case Presentation

The on-line edition of the Dutch newspaper *Telegraaf* (www.telegraaf.nl) sells advertising space on its network of electronic editions. According to M. Udo, its on-line sales manager, *Telegraaf* publishes ads in return for a corresponding payment personal communication, 2005]. Its clients are usually large or medium-sized companies that want to advertise their products. A contract between *Telegraaf* and the advertiser is established for each ad campaign.

The contracting process is not strictly defined and allows flexibility to each party. A client will perform a number of activities. Figure 1 lists 12 activities typically performed by clients. The UML activity diagram shows one of the possible control flows for these activities.

In parallel, *Telegraaf* also performs a number of activities. Figure 2 lists 13 activities typically performed by *Telegraaf.* Again, a possible control flow for these activities is shown in a UML activity diagram.

E-Contracting Process Model: Requirements Analysis

Contracting is a complex process. Depending on the context, contracting processes can vary significantly even when performed by the same company. Changes in the business and legal contexts can necessitate changes in a company's contracting process. The variability of contracting processes is a substantial problem for efforts to automate contracting. The e-contracting process specifications have to be flexible because of their variability, and this makes them highly complex. Moreover, the processes of the contracting parties have to be synchronized, and the parties have to have a common understanding of the processes performed. E-contracting aims at automating the traditional contracting process. This means that requirements on contract process specifications must also be applicable to e-contracting process specifications. In addition, e-contracting process specifications must be detailed, so that they can be automatically executed and managed by an information system.

Variability of Contracting Processes

The main aspects of variability of the contracting process are shown below.

- **Sequence of execution of activities:** The order in which contracting activities are executed depends on the company's preferences or the contracting context. An example from the *Telegraaf* case is the
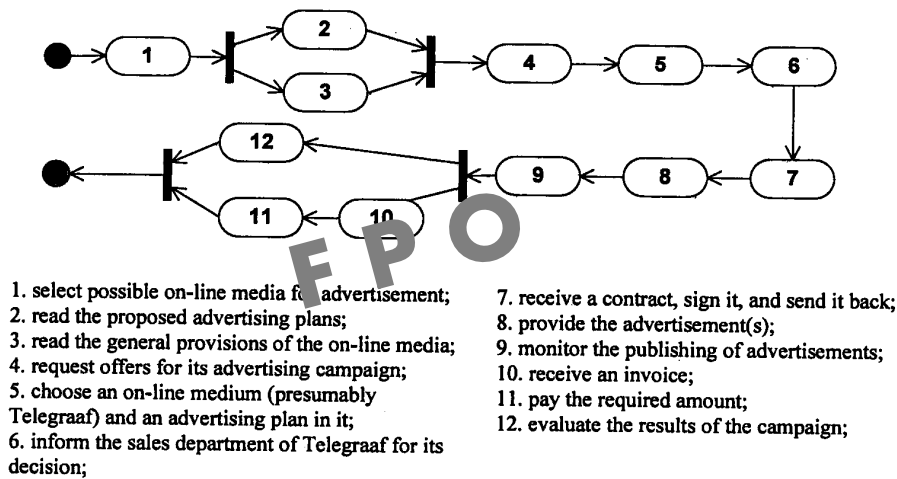
1. select possible on-line media for advertisement;
2. read the proposed advertising plans;
3. read the general provisions of the on-line media;
4. request offers for its advertising campaign;
5. choose an on-line medium (presumably Telegraaf) and an advertising plan in it;
6. inform the sales department of Telegraaf for its decision;

7. receive a contract, sign it, and send it back;
8. provide the advertisement(s);
9. monitor the publishing of advertisements;
10. receive an invoice;
11. pay the required amount;
12. evaluate the results of the campaign;

Figure 1. Client-Side Activities and Control Flow <<ALL FIGURES ARE FOR PLACEMENT ONLY (POOR SCANS OF A PHOTOCOPY) / SUPPLY DATA FILES FOR ALL 18 ILLUSTRATIONS>>
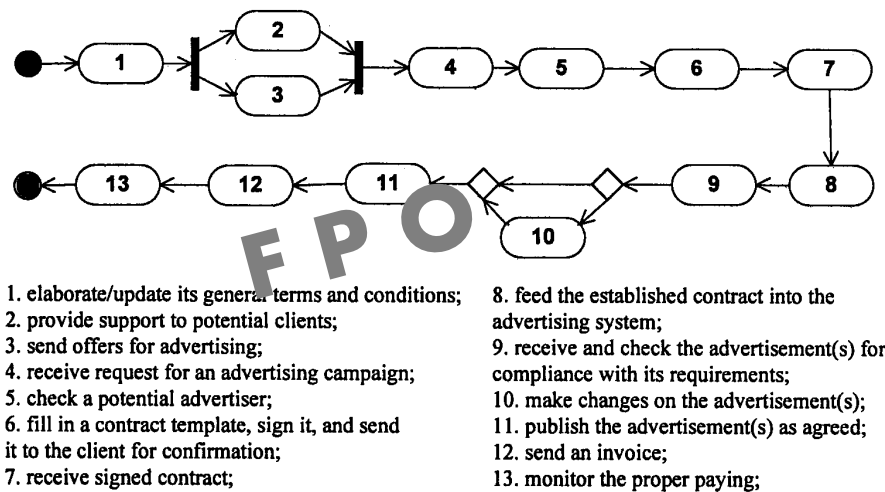

1. elaborate/update its general terms and conditions;
2. provide support to potential clients;
3. send offers for advertising;
4. receive request for an advertising campaign;
5. check a potential advertiser;
6. fill in a contract template, sign it, and send it to the client for confirmation;
7. receive signed contract;

8. feed the established contract into the advertising system;
9. receive and check the advertisement(s) for compliance with its requirements;
10. make changes on the advertisement(s);
11. publish the advertisement(s) as agreed;
12. send an invoice;
13. monitor the proper paying;

Figure 2. *Telegraaf*-Side Activities and Control Flow <<SUPPLY DATA FILE>>

"send invoice" activity, which can be executed immediately after the contract is established, during the advertising campaign, or after the campaign (*see Figure 2*). The decision when to perform the activity depends on the context (e.g., the level of trustworthiness of the client). As this demonstrates, there may be numerous modeling possibilities for the sequence in which activities are executed.

- **Optional execution of activities:** Many activities are executed only in certain contracting contexts. For example, activities related to finding a party may be skipped if the parties already know each other from previous contacts. Thus, the context may sometimes make it possible to remove certain contracting activities from the contracting process specification. Furthermore, the execution of many activities is determined during the execution of a contracting process. In the *Telegraaf* case, the "make changes on advertisements" activity is performed only if the advertisements do *not* comply with *Telegraaf*'s requirements *(see Figure 2)*.
- **Repetition of execution of activities:** Certain activities may be executed several times during an e-contracting process. For example, the "provide advertisement" activity must be executed at least once (*see Figure 1*), but if the advertising company revises the ad after submission, it will re-execute this activity in order to send the new version.

The variability of contracting processes means that it is not easy to define a detailed and complete e-processing contract model that will be applicable to every arbitrary party in every arbitrary context. The optional execution of activities indicates that some activities can easily be omitted from an e-contracting process model. Thus, explicit attention must be given to the *completeness* of an e-contracting process model. The variability of e-contracting processes indicates that two types of *flexibility* must be addressed in an e-contracting process model. First, the model must be flexible enough to support the definition of e-contracting processes in every possible business scenario. Second, companies must be able to design process specifications that are as open as possible to support the different scenarios in which they may have to establish trading relationships.

Coherence of E-Contracting Processes

A company's contracting process must be in coherence with its *external* and *internal* processes. Coherence with external processes means that every party should perform its e-contracting process in harmony with the process of its counter-party. For example, *Telegraaf* requires an advance payment from its clients. Processes will not be coherent if *Telegraaf* expects payment before the advertising campaign but the client expects to pay after it begins. Contracting is often associated with the performance of a set of communication activities aimed at establishing and enacting a contract. However, in practice, contracting involves a large number of internal activities that precede or follow communication activities. These internal activities are an important part of the contracting process but are often omitted from the specifications. Thus, "coherence with internal processes" means coherence between communication contracting activities and internal contracting activities.

Requirements on an E-Contracting Process Model

Several requirements for an e-contracting process model follow from the preceding observations:

- **Requirement 1 (Detailed):** An e-contracting process model should provide a possibility for constructing detailed e-contracting process specifications— specifications that enable automatic execution and management by information systems
- **Requirement 2 (Complete):** An e-contracting model should be complete, enabling companies to specify any domain- or company-specific contracting process.
- **Requirement 3 (Flexible):** An e-contracting model should support specification of contracting processes in all possible business scenarios.
- **Requirement 4 (Flexible specifications):** An e-contracting process model should allow the design of flexible e-contracting process specifications.
- **Requirement 5 (External coherence):** An e-contracting process model should support specification of coherent communications between contracting parties.
- **Requirement 6 (Internal coherence):** An e-contracting process model should guarantee seamless linking between internal and communication activities.

To summarize, an e-contracting process model should support the definition of detailed, complete, flexible, and (internally and externally) coherent e-contracting processes.

Possible Approaches

There are two existing approaches to the specification of e-contracting processes. The first approach specifies a structured, e-contracting process using traditional process-modeling techniques but cannot simultaneously satisfy Requirements 1, 2, 3, and 4 (as also in the context of negotiations between parties in [32]). Attempting to address these requirements will make the model uncontrollably complex and impossible to understand, and maintain. Consequently, contracting models based on traditional modeling techniques are either highly restrictive (and thus violate Requirements 2 and 4) or at a high level of abstraction (violating Requirements 1 and 4). Since flexibility and detail are a must for modeling e-contracting processes, this is unacceptable.

The second approach specifies parts of e-contracting processes in patterns. Each pattern contains a structured, abstracted specification of a process [4, 30]. A company instantiates predefined patterns that are applicable to its contracting context. In essence, this approach tries to resolve the violation of Requirements 1, 2, 3, and 4 in the first approach by providing structured specifications only of parts of the process. These parts are combined to form

the complete contracting process specification for the specific context. As a result, a company has the flexibility to compose contracting processes using "building blocks." This approach is taken, for example, in RosettaNet, as well as by van Dijk and FIPA [12, 31]. It has two major problems. The first pertains to the selection of "building blocks." Most patterns will cover one or more points of variability. Patterns have to be defined for all of the different scenarios that may take place over these points of variability. This results in a significant number of patterns. In order to create a contracting process based on patterns, one must have profound knowledge of the content of all the patterns and the relationships among them. This makes the selection and combination of patterns a complex task. The nature of patterns constitutes the second problem. Patterns only address common contracting activities (that is why they are called patterns) and omit contracting activities specific to a given company—thus violating Requirements 2 and 6.

This paper addresses the deficiencies of these two approaches by suggesting an approach based on the use of unstructured e-contracting process specifications. It goes a step further than the second approach discussed above by reducing the "building blocks" to single activities.

Main Elements of the E-Contracting Process Model

Two perspectives of the e-contracting process must be defined in order to provide a model that satisfies the requirements identified in the preceding section.

As stated in Requirement 2, special attention must be given to the completeness of the e-contracting process model. Completeness is attained by defining the function perspective. In this perspective, e-contracting activities are decomposed at different levels of abstraction. The function perspective provides a complete picture of the e-contracting activities to a certain level of decomposition. The level of decomposition in this paper suffices for modeling detailed e-contracting processes that can be automatically executed (see Requirement 1). The hierarchical presentation of the function perspective allows further decomposition. Issues specific to a particular business domain (e.g., the advertising domain) can be addressed in this way.

As stated in Requirements 5 and 6, an e-contracting model should guarantee coherence of cross-organizational activities as well as between cross- and intra-organizational activities. This is achieved by elaborating a communication perspective in addition to the function perspective. The communication perspective classifies communication activities and indicates their connection to internal activities. The communication perspective is a specialization tree. The construction of the communication perspective is based on speech-act theory (SAT) [3, 21, 29]. Using SAT, communications between parties can be coupled, and internal business processes can be coupled with the e-contracting communication activities to be modeled, thereby achieving coherent cross- and intra-organizational processes.

The activity leaves in the communication perspective are a subset of the activity leaves in the function perspective. The communication perspective

facilitates the process of defining coherent communication activities in the function perspective. It is used in constructing the last level of the function perspective where concrete communication activities are identified. The role of the communication perspective is not limited to the construction of the function perspective. It guides process designers in selecting communication activities from the function perspective. Also, as will be shown further on, any subsequent decomposition of leaves in the function perspective may uncover new communication activities. In these situations, the process designer uses the communication perspective to specialize the leaf activities of the function perspective to the emerging communication activities. For this reason, the communication perspective is an integral part of the proposed model.

As there are no relations defined between the activities from the function perspective, the model incorporates every possible e-contracting scenario (see Requirement 3). A set of constraint operators is provided to allow companies to model relations between activities. These operators enable companies to define constraints on activities that are applicable in their context. Constraints define basic relations between activities but leave freedom for innumerable execution paths, thus providing the flexibility required in e-contracting processes (see Requirement 4).

The two perspectives, in combination with the constraint operators, constitute a tool with which a business party can construct its own, flexible, complete, and detailed e-contracting process specifications. The model enables coherent communications between the parties.

E-Contracting Process Model

Our discussion of the e-contracting process model will begin with the function perspective, which is fundamental for the model. The communication perspective, described next, will be used in defining the model's lower levels of detail. The results from the communication perspective finalize the description of the function perspective. Last, the third element of the model is presented: the operators for definition of constraints.

Function Perspective

The function perspective is a combination of decomposition associations and a subtyping hierarchy. There are three levels of abstraction in the hierarchy: phase, abstract activity, and concrete activity.

Phase Level

An e-contracting process consists of a number of phases. Researchers have taken different views of these phases (e.g., [14, 22]). This paper adopts the model proposed by Gisler, Stanoevska-Slabeva, and Greunz model because it reflects the best existing business practice [14]. The model consists of four
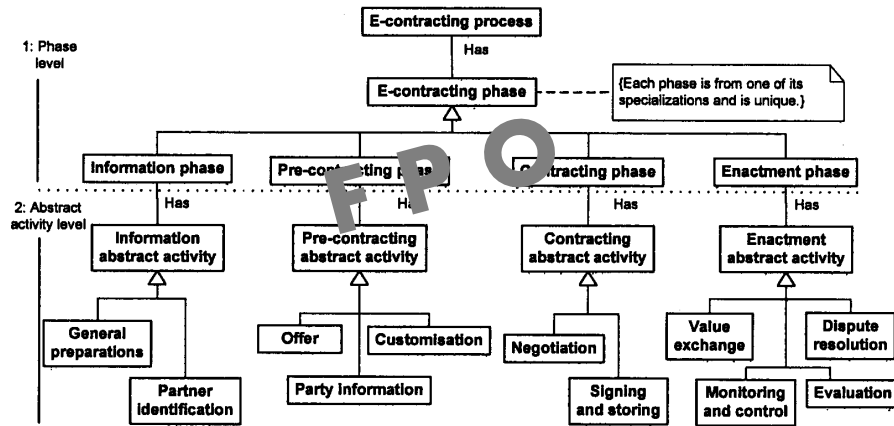
Figure 3. Function Perspective (levels 1 and 2) <<SUPPLY DATA FILE>>

phases: information, pre-contracting, contracting, and enactment (*see Figure 3*). In the information phase, general preparations are made, information is collected and distributed, and potential partners are identified. In the precontracting phase, activities for collecting additional information and for determining whether to initiate negotiations with the other party are performed. In the contracting phase, the contract is negotiated and established. During the enactment phase, activities required for the exchange of values and for controlling the behavior of the parties are performed. Contract performance is evaluated at the end of the enactment phase. This general model of an e-contracting process, unlike other models, pays explicit attention to the legal aspects and legal requirements of the e-contracting process. This leads to the identification of the precontracting phase, which contains the gradual transition from activities that use documents that are not legally binding (e.g., an "invitation to treat") to activities in the contracting phase that involve documents that are legally binding. For this reason, this general model was adopted at the highest level of abstraction of the function perspective.

The foremost industrial solutions, academic results, and standards in the field of e-contracting were consulted in the course of this work (e.g., [7, 9, 14, 19, 22, 24, 26, 27]). These efforts indicate certain requirements on the activities in the e-contracting process. For example, monitoring and control support during contract enactment are still not supported to their real potential, as already addressed in [7] and 24]. In decomposing the function perspective into second and third levels of detail, full consideration was given to the requirements on e-contracting processes already identified by industry and academia.

Abstract Activity Level

Each phase is decomposed to abstract activities. For example, the information phase is decomposed to information abstract activities (*see Figure 3*). Abstract activities, the second level of abstraction of the function perspective, are col-

lections of concrete activities (the third level of abstraction of the function perspective). An abstract activity is a set of concrete activities that together deliver a specific value to a party. Abstract activities are specialized according to the concrete activities performed in them (process aspect) and the data associated with these concrete activities (data aspect).

Specializations of information abstract activities. The information phase comprises two abstract activities: general preparations and partner identification.

- **General preparations:** At this initial stage, the parties prepare internally for the e-contracting process. They clarify their goals, limitations, and strategies, and define general contract provisions, contract templates, and the like.
- **Partner identification:** At this stage, the parties *disseminate/obtain* information to/about potential business partners. This information can be, for example, advertisements or requests for products/services. From a party's point of view, a successful result of this stage is a collection of companies that, based on the information obtained, match its search criteria.

Specializations of precontracting abstract activities. The precontracting phase comprises three abstract activities: offer, customization, and party information.

- **Offer:** In order to get provisional contract information, consumers can **request** offers from suppliers. An offer is a document that provides information on the operational aspects of the exchange, and is accompanied by the general provisions of the company.
- **Customization:** After an offer is exchanged between companies, a company might *request* whether customization of the offer is possible.
- **Party information:** More information about a company may be required at any stage after the identification of the matching companies. In this activity, parties **collect** additional information about other companies, such as evaluations of previous contract activities, official information on the company state, and digital certificates. This information can be collected directly from the company or through a trusted third party (TTP), such as a national chamber of commerce or a reputation rating center [20, 23].

Specializations of the contracting abstract activities. In the contracting phase, the major abstract activity is negotiation. If negotiations on a contract lead to an agreement, a contract is signed and stored.

- **Negotiation:** In the negotiation activity, a contract for the exchange of values is negotiated.
- **Signing and storing:** When a contract is agreed upon, the parties have to sign it. Signing requires an *exchange* of the signed contract between the parties. The parties store the signed contract internally.

They may store it externally as well. For example, they may **send** it to a third party (e.g., notary).

Specializations of enactment abstract activities. In the enactment phase, the exchange value, monitoring and control, dispute resolution, and evaluation abstract activities are identified.

- **Value exchange:** At this stage, the supplier and consumer *exchange* the agreed values (products/services/money). The exchange of values can be preceded or accompanied by exchange of the data required for the performance of these activities.
- **Monitoring and control:** A party wants to be constantly aware of the status of contract performance, and of its rights and pending obligations. This awareness helps it to comply with the contract and adapt its behavior to changes in the environment. That is why parties monitor and control the enactment process [15, 23, 33].
- **Dispute resolution:** Parties may consciously or unconsciously violate the agreement. When noncompliance with the contract is detected, the parties **exchange** information on the disputed topic. If the information exchange does not lead to a resolution of the dispute, they may *request* dispute resolution from an arbitrator or a court [23].
- **Evaluation:** In this activity, the parties evaluate the business relationship. For example, a company might evaluate the counter-party's level of compliance with the contract, the level of compliance of its business processes with the contract [16], and so on. This evaluation serves to improve future contract activities of the companies and, when externally **shared**, to give information to other companies about the contracting potential of the party. Trusted third parties (e.g., reputation rating centers) can collect and maintain externally shared evaluations.

Decomposition and specialization of abstract activities result in the third level of the function perspective—level of concrete activities (not shown in Figure 3). The explanations of the abstract activities used **boldface** to indicate that certain abstract activities involve communications between parties. Thus, decomposition of abstract activities into concrete activities requires explicit modeling of communication activities (i.e., activities that send or receive messages) between companies. As already explained, the communication perspective is used to specialize the communication activities in the third level of the function perspective.

Communication Perspective

The communication perspective facilitates the specialization of abstract activities in the function perspective to concrete cross-organizational activities. It guides companies in designing communication activities. It standardizes the specification of communication activities, thereby enabling the parties to attain

coherence in communications with one another. Finally, it supports companies in linking their communication and internal activities. As will be explained below, the communication perspective is based on speech-act theory.

Speech-Act Theory in E-Contracting

Speech-act theory is a theoretical approach researched and used by many cognitive sciences. Although it originated in nontechnical research domains [3, 29], recent research efforts have shown that SAT can be beneficial for the domain of information technology. Its potential to facilitate the construction of automated, coherent messaging between information systems has already been observed [13, 21]. As summarized by Kimbrough and Moore, SAT consists of three basic elements: the decomposition of speech acts, the F(P) framework, and the F framework [21].

Every speech act can be seen as a composition of four actions. The first action is the action of utterance (the *utterance act*)—the speaker's action of producing information. Next, the hearer has to infer the proposition uttered by the speaker (the *locutionary act*). Third is the hearer's inferring of the attitude the speaker put into the utterance act (the *illocutionary act*). Fourth and last is the *perlocutionary act*—activities performed by the hearer that result from the speaker's utterance act.

The F(P) framework in SAT expresses the fact that almost every illocutionary act "involves an expression by the speaker of a propositional attitude towards some (possibly complex) proposition" [21] (where F is the illocutionary force on the proposition P). This framework is mostly used as a foundation for the construction of complex expressions containing multiple propositions and illocutionary forces. As it is of no importance for the approach proposed in this paper, it will not be discussed any further.

During e-contracting, the parties communicate with other contracting parties and with parties not part of the contract relation (third parties). As the goal is specification of communications between parties, the employment of SAT as a theoretical base on which communications can be viewed is a logical step. The suitability of SAT as a base for automation of communications between information systems is treated at length by Kimbrough and Moore [21}. The decomposition of speech acts as defined in SAT is employed to identify speech acts that are part of an e-contracting process.

An e-contracting party can perform one or the other of two possible classes of utterance acts to initiate a communication with another party: it can either *request* information or produce information (i.e., *inform*) [25]. These communication-initiating activities are usually preceded by internal activities (e.g., the decision process to request certain information). In Figure 4, utterance acts are denoted in gray and internal activities in white.

A *request* act by a party (Party A) is processed and interpreted by its counterparty (Party B). In line with SAT, an utterance act is followed by locutionary and illocutionary acts. In Figure 4, these acts are called "*process and interpret request*." After message processing and interpretation, a party decides whether to react to the request and, if so, what the response will be (the perlocutionary act). In
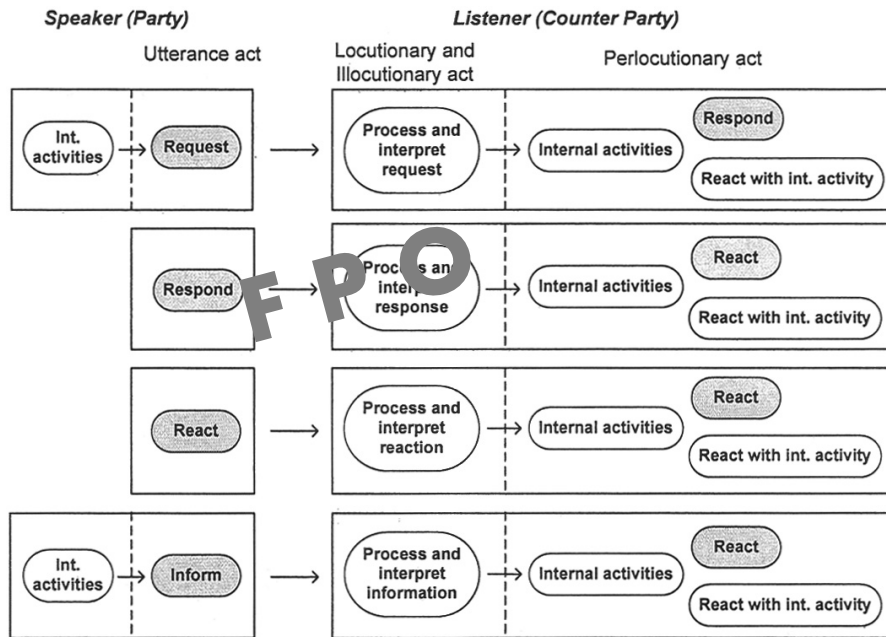
Figure 4. Speech Acts in E-Contracting <<SUPPLY DATA FILE>>

this step, a company will initially perform internal perlocutionary activities like "take decision" (whether to react/how to react) and "prepare response." As these internal activities may vary among companies, they are simply called "*internal activities*" in Figure 4. When a decision is taken to react, the company may react to the request internally (*react with an internal activity*), externally (with a *respond* utterance act), or both. When a perlocutionary act leads to an utterance act, a new speech act is initiated. The response is processed and interpreted by Party A, the receiver. Party A analyzes it and may react with another utterance act or only internally, within the boundaries of the company. When the reaction is an utterance act, the counter-party receives this reaction, processes and interprets it, and subsequently can again react with an utterance act or stop the communication process and react only internally. Thus, the request utterance act can be followed by a response utterance act and subsequently by an unlimited number of reaction utterance acts (reaction to the response, reaction to the reaction, etc.). Although one might argue that a response is a type of reaction, it is considered separately because it has its own specific semantics. A response utterance act is a reaction to a request for information, whereas a react utterance act is a reaction to received information. For this reason, the respond utterance act is considered to be a separate class.

The *inform* act is performed to deliver information to the counter-party. It is processed and interpreted by the counter-party. If necessary, the party will react internally or externally.

To summarize, two types of conversations can exist in an e-contracting scenario. The first is a request, followed by a response, followed by a reaction to the response (which can be followed by a reaction to the reaction, and so

on). The second is an inform act followed by a reaction (which can be followed by a reaction to the reaction, and so on). Thus, long conversations between parties are modeled in the form of a react-react loop of utterance acts (initiated by a request or an inform act). As will be discussed further on, certain rules govern communication activities (e.g., a request should always precede a response act).

Structure of Communication Perspective

All communication activities are specialized from a root speech activity. In Figure 5, the communication perspective is specialized into "external" activities (utterance acts) and "internal" activities (locutionary, illocutionary, and perlocutionary acts). As already discussed, an utterance act can be a *request, respond to request, react to sent data,* or *inform party* activity. This set of communication activity types covers all possible utterance acts in a communication process (other frameworks, e.g., [25], identify similar types of communication activities). Concrete utterance acts can be specialized from them.

To achieve coherence between intra- and cross-organizational activities, the internal activities related to the utterance acts have to be specialized next to the utterance acts. The addition of intra-organizational activities to the communication perspective links communication activities to the internal activities associated with them. This is in alignment with the SAT statement that an utterance act should not be separated from the activities performed on the listener's side.

The level of abstraction of the internal activities in Figure 5 suffices for the goals pursued in the communication perspective with respect to achieving coherence between internal and external activities. For this reason, no further specialization of the internal activities is performed. This is left to the companies, which can define these specializations in the way most appropriate for their business context. For coherent communication between parties, however, the utterance acts require further specialization.

Specializations of Utterance Acts

The cross-organizational communications that may take place in the negotiation abstract activity will now be identified. Speech-act theory is used to define a number of specializations of the utterance acts in the high-level view of the communication perspective. The goal is to give an example of the use of SAT and the communication perspective in the definition of concrete activities from the third level of the function perspective. The negotiation abstract activity was selected for its complexity, because its decomposition results in specialization of all utterance activities from the high-level view of the communication perspective. The decomposition of the other abstract activities from the function perspective is performed in an analogous way. The complete set of specializations of the utterance activities that results from the decomposition of all abstract activities is provided in the Appendix.
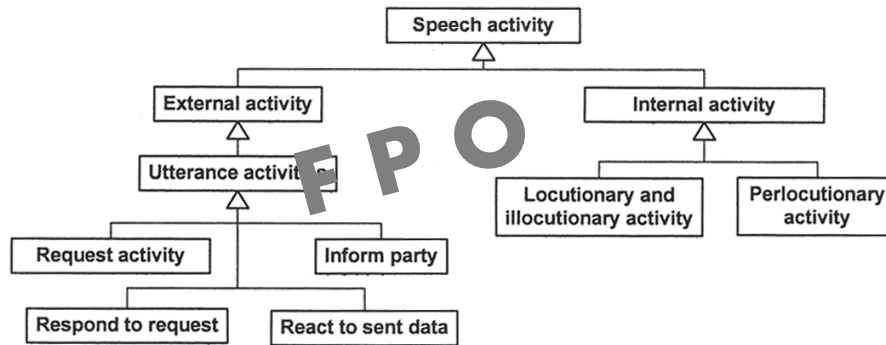
Figure 5. Communication Perspective (high-level view) <<SUPPLY DATA FILE>>

The parties in the negotiation abstract activity exchange contract offers (or elements of contract offers) until they reach an agreement on a contract or face irresolvable differences and end the negotiation process. Clearly, this activity includes a number of speech acts. The high-level view of the communication perspective is used to identify the utterance acts that may take place in this activity.

Communications can be initiated in both ways—a party may request its counter-party for a contract offer or may send a contract offer as an information activity. We start by identifying the activities in conversations started by a request for a contract offer. In this scenario, one of the parties initially *requests* a contract offer from the other. This gives the *request contract offer* specialization of the request utterance act (*see Figure 6*). In most situations, the requesting party will be the consumer. In some cases, however, the requesting party will be the supplier (e.g., when the consumer is a large company that dictates the contracting process).

As stated elsewhere, the requested party processes and interprets the request and may respond to the contract offer request (*see Figure 7*). A response can contain the contract offer, can decline the request, or can be of a complex nature (e.g., a commitment to provide a response later on, followed by the response as promised). The response is validated and evaluated (*locutionary and illocutionary activities*). A reaction to a contract offer may be elaborated and sent (*see Figure 8*). The reaction can state acceptance of the contract offer, rejection, termination of negotiations, a counter-contract offer, and so on. If a counter-offer is produced, the counter-party processes and interprets the counter-offer and can react on it. This can be repeated until a contract offer is accepted or is rejected with interruption of the negotiation process [14, 28].

A party can also directly *send a contract offer* without an initial request (*see Figure 9*). The received response/information message is *processed and interpreted* and a party may again react to the contract offer. A party needs an identical set of possible reactions to react to a message received either as a response or an inform activity. For example, the party will use an identical set of reactions to react to a *respond contract offer request* and a *send contract offer* activity,. For this reason, only one reaction to an *inform* and *response* message is defined.

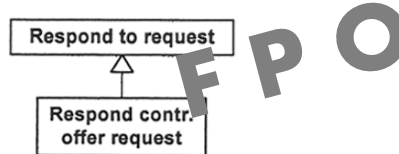**Figure 6. Specialization of Request Activity <<SUPPLY DATA FILE>>**

Figure 7. Specialization of Respond Activity <<SUPPLY DATA FILE>>

Figure 8. Specialization of React Activity <<SUPPLY DATA FILE>>

Figure 9. Specialization of Inform Activity <<SUPPLY DATA FILE>>

The other communication activities identified in the decomposition of abstract activities are defined in an analogous way (see Appendix). In general, internal activities are not modeled, because they are company-specific. The only perlocutionary activities modeled are either utterance acts or internal activities common to all companies.

The semantics of the contents of messages must be predefined in a message ontology and agreed upon by the communicating parties. For example, the semantics of a reaction to contract offer may be an acceptance, a rejection, a counter-offer, or a termination of negotiations. Ideas from other communication protocols can be used in defining the semantics of the message content. For example, the "Coordinator" system described by Flores et al. and the Library for Speech Acts define communication acts at a lower level of detail and implicitly reach a definition of the semantics of the content of certain messages (e.g., "agree," "refuse," "cancel," "acknowledge," "counter-offer") [11, 13].

SAT was used to identify and structure the communication acts in e-contracting. However, the application of SAT for modeling contractual communications between information systems has one more important implication. Modeling of speech acts that take place among information systems requires all direct and *indirect* speech acts to be modeled explicitly. In contracting, in addition to conveying a message by a speech act, a party may indirectly state its commitment to an activity in the contracting process or to the contract between the parties. A commitment expressed by a party is a legally binding statement and may influence the reaction of the listener. For example, a request for a customization process may mean that if the customization that is requested can be provided, the party commits to start contract negotiations. Similarly, acceptance of a contract offer usually means that the party commits to sign a contract based on the contract offer. This observation does not call for any changes in the structure of the communication perspective as defined earlier. It does, however, lead to the requirement that when a speech act implies commitment, the commitment is explicitly stated in the communication message and can be correctly interpreted by the message receiver. One possible way to address this requirement is by defining a parameter in a message that contains the commitment expressed by the message sender. The parameter will contain information indicating whether a party commits to the execution or nonexecution of a certain task of the e-contracting process or to a clause in the e-contract (or to the complete e-contract). The usage of common process and contract ontologies makes it possible to automate the illocutionary act of interpreting commitments. For operational reasons, a number of other parameters also have to be defined in messages. For example, parameters stating the process ontology that is used and the type of the communication act have to be defined. The results from [10] can be used as a starting point. As message structure and message semantics are beyond the scope of this paper, they will not be discussed any further.

Based on the results in this section (and the complete set of specializations of utterance activities listed in the Appendix), the abstract activities from the function perspective are decomposed and specialized to concrete activities. As stated previously, the cross-organizational activities in the function perspective are all leaf activities in the communication perspective.

Concrete Activity Level of Function Perspective

The decomposition and specialization to concrete activities of the abstract activities will now be explained. Further decomposition and specialization of concrete activities to lower levels of abstraction will also be discussed.

Abstract and Concrete Activities for Information Phase

As shown in Figure 10, the information phase has two abstract activities, *general preparations* and *partner identification* .
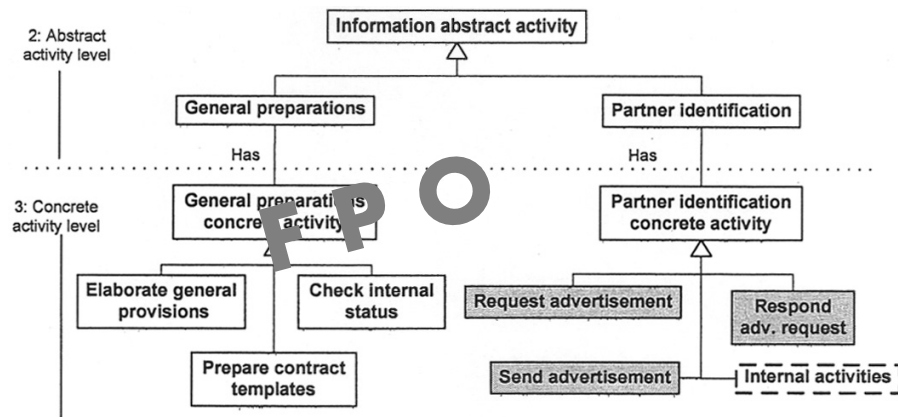
Figure 10. Abstract and Concrete Activities (Information Phase)
<<SUPPLY DATA FILE>>

The *general preparations* abstract activity comprises *elaboration of general provisions, preparation of contract templates,* and *check of the internal status* of the company. These are possible internal activities that a company would execute. They can be further decomposed and specialized to reflect in more detail the company's specific preparation activities.

For the *partner identification* concrete activity in the communication perspective (see Appendix), the identified activities are *send advertisements, request advertisements,* and *respond advertisement request* activities (utterance activities are represented in gray). In Figure 10, the locutionary, illocutionary, and internal activities to received advertisements are shown as a general activity termed *internal activities* because these are company-specific activities that must be specialized in the specific company context. A thick, dashed line is used to show that it is a general activity that aggregates several internal activities. The same notation is used in the following subsections.

Abstract and Concrete Activities for Precontracting Phase

The precontracting phase, as shown in Figure 11, has three abstract activities: *offer, party information,* and *customization.* These are all communication activities. In other words, they involve a number of utterance acts that may be accompanied by corresponding internal activities. Thus, results from the communication perspective are directly used in their decomposition (see Appendix).

For the *offer* concrete activity, using the communication perspective, the *request for offer* accompanied by the *respond* and *react* utterance acts and the *send offer* utterance act have been identified. Message are *processed and interpreted* by the receiving company, which may perform certain internal activities. Only a general activity that includes the *locutionary, illocutionary,* and *internal* activities performed after every utterance act is represented in Figure 11.
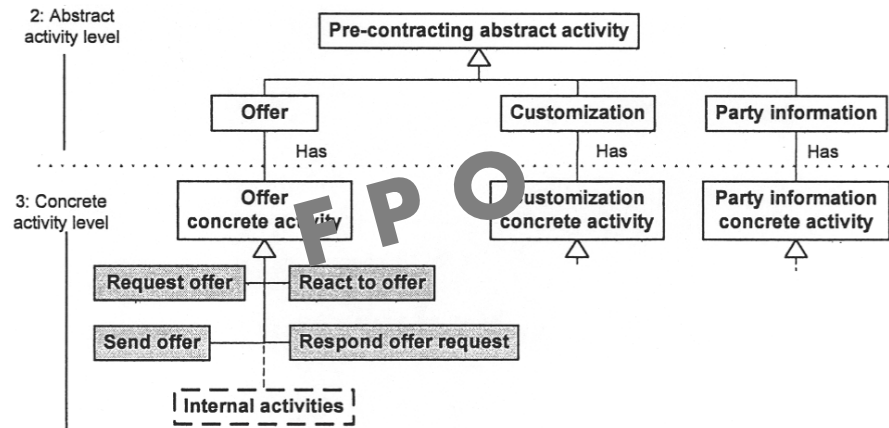
Figure 11. Abstract and Concrete Activities (Precontracting Phase)
<<SUPPLY DATA FILE>>

Companies must individually define their internal activities related to these utterance acts.

In order to simplify Figure 11, the specializations of the *customization* and *party information* concrete activities are not shown. These specializations are defined in the communication perspective and are listed in an analogous way.

Abstract and Concrete Activities for Contracting Phase

Negotiation activities are communication activities. The results from the communication perspective can be used (see Appendix). The identified activities are listed in Figure 12.

If a negotiation ends with an agreed by all parties contract, each company signs the contract. Signing is an internal activity that will take place within any company regardless of the context. That is why it is explicitly specified in the model. Contracts signed internally are sent for signing to the other parties. E-notaries may collect the signatures of the parties and certify the contract. Regardless of the signing protocol, each company receives a contract signed by all the parties. After receiving the final contract, companies perform at least two internal activities. First, they verify signatures, and then, if these are approved, they internally store the contract for future reference. Again, as these internal activities are common to all companies, they are explicitly specified in the model.

Abstract and Concrete Activities for Enactment Phase

The exchange of the agreed in the contract products for a corresponding reward can take place in various ways. For example, in the case of digital goods, the delivery of the goods may be a *communication* activity, because the goods can
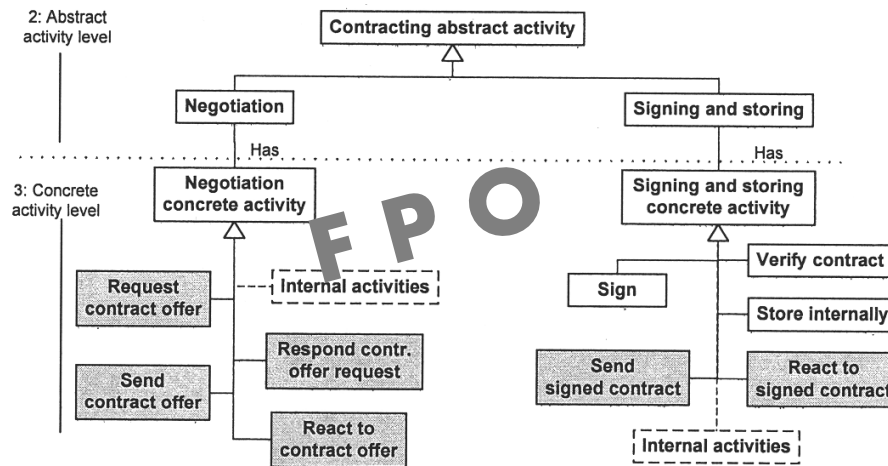
**Figure 12. Abstract and Concrete Activities (Contracting Phase)
<<SUPPLY DATA FILE>>**

be sent electronically to the buying party. For this reason, these activities are modeled at a high level of abstraction as *deliver service/product* and *deliver reward.* Companies must specialize these activities in their context. When these activities involve communication activities, the communication perspective is used for the specialization. During value exchange, parties can exchange enactment data. This communication is modeled via the communication perspective (see Appendix). The resulting concrete activities are shown in Figure 13.

In the *monitoring and control* concrete activity, a company monitors internally the status of the contract enactment. For monitoring of the status quo at the counter-party, it may request from the counter-party data on the enactment status (a communication activity modeled via the communication perspective). Based on the monitoring results, a party may impose *control* over the contract enactment. When it is agreed to allow external control (e.g., imposing control on cross-organizational workflows), the control activity becomes a communication activity and has to be specialized using the communication perspective. At any moment, parties can notify each other for certain events (e.g., successfully completed activities) by sending a notification message.

In case of contract noncompliance, a company sends information on the problem to its counter-party. It may request a satisfactory dispute resolution from the counter-party. If the response and subsequent reactions do not lead to a resolution of the problem, a company may request resolution from a third party (e.g., an arbitrator).

The evaluation internal activity results in an evaluation that is stored internally. Companies may share their evaluations with third parties.

Further Decomposition of Leaf Activities

The activities identified at the concrete activity level are leaves in the function perspective. This level of decomposition is sufficient for modeling standard
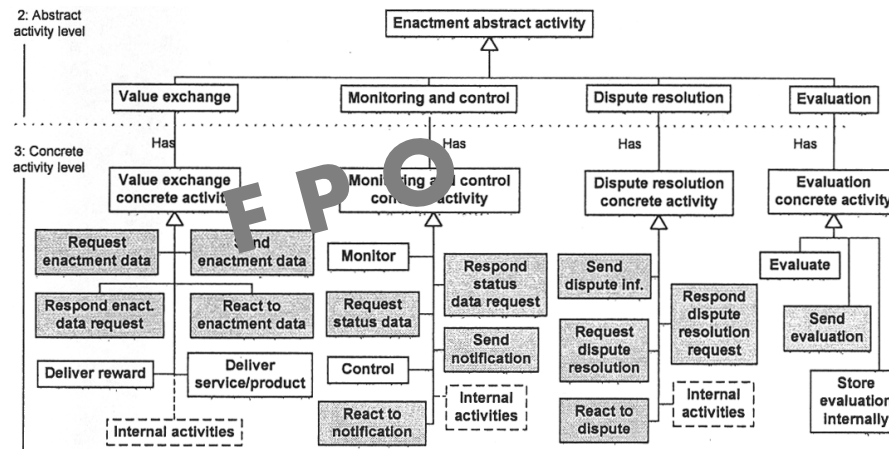
Figure 13. Abstract and Concrete Activities (Enactment Phase)
<<SUPPLY DATA FILE>>

e-contracting processes. As already mentioned, depending on the business situation, certain activities from the function perspective may have to be further decomposed and specialized. Thus, the defined activity tree can be decomposed to new levels of detail (e.g., domain level, company level, service level). Only leaf activities from the function perspective can be further decomposed and specialized. This follows from the fact that the function perspective represents a complete description of the e-contracting activities at different levels of detail. Each new decomposition and specialization gives a lower level of abstraction. When a concrete activity in the function perspective is specialized to a lower level of detail, and a new cross-organizational communication is identified, the communication perspective is used to define them.

Operators for Definition of Constraints

The operators that impose constraints on the execution of activities comprise the third element of the process model. The set of operators presented here is used only to illustrate the role of operators in the process model. Formalization of the operators and verification for their completeness will be the subject of future work.

Figure 14 defines a classification tree of the operators. There are two classes of operators: those that apply to a single activity (unary) and those that apply to two activities (binary). From the unary operators, only the existence operator REQUIRED is considered. A binary operator can define an order, parallelism, or existence constraint. The SEQUENCE operator is defined to express a required sequence of activities, the DURING operator to express parallelism constraints, and the IMPLY operator to define execution dependencies between activities. If additional types of constraints must be imposed, this set can be extended with other types of operators. The operators are defined below.
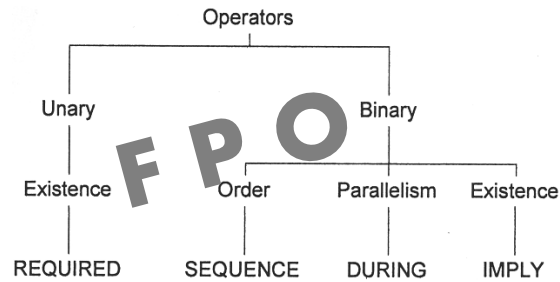
Figure 14. Operators for Definition of Constraints <<SUPPLY DATA FILE>>

- **REQUIRED (A_1)**: The REQUIRED operator indicates that the activity given as an argument must be executed for the successful enactment of an e-contracting process. *Example:* REQUIRED (Value exchange).
- **SEQUENCE (A_1,A_2)**: The SEQUENCE relation expresses that the execution of A_1 has to precede that of A_2. This relation does not require the execution of the activities. The SEQUENCE operator can be extended to define the sequence of more than two activities; for example, SEQUENCE ($A_1,A_2, \ldots ,A_n$). This extension is used only to reduce excessive writing. *Example:* SEQUENCE (Offer, Negotiation) shows that if the two activities are executed, the execution of the *Offer* activity has to precede the execution of the *Negotiation* activity.
- **IMPLY (A_2,A_1, cond)**: The IMPLY relation expresses that if activity A_2 is executed, then activity A_1 is executed as well. Or, in other words, A_2 implies A_1. A condition may indicate when the implication is valid. *Example:* IMPLY (Signing and storing, Negotiation) states that if a *Signing and storing* activity is executed, then this implies the execution of the *Negotiation* activity.
- **DURING (A_1,A_2)**: The DURING relation requires activity A_1 to be executed during the execution of A_2. The DURING operator can be extended for more than two activities. DURING (($A_1, \ldots ,A_n$), B_1) states that the activities $A_1, \ldots ,A_n$ have to be executed during the execution of activity B_1. *Example:* DURING (Monitoring and control, Value exchange) states that the activity *Monitoring and control* has to be executed during the execution of the *Value exchange* activity.

It is important to note that a number of constraints apply on all e-contracting scenarios. For example, the constraint IMPLY (Signing and storing, Negotiation) is independent of the business context. Similarly, a number of basic relations exist between speech acts. For example, a *request* activity is associated with a *processing and interpretation* activity, the *processing and interpretation* activity is related to *perlocutionary* activities, and so on. These constraints on communication activities apply on all e-contracting scenarios and consequently are an integral part of the model. For reasons of space, they are not presented in this paper.

Construction and Usage of E-Contracting Process Specification

The e-contracting process model supports companies in the specification of various, flexible e-contracting processes. As will be explained below, the proposed model can be applied for the elaboration and use of e-contracting process specifications in business relations.

Process Specification Construction

A party constructing an e-contracting process specification starts by identifying the activity leaves in the function perspective that will be included in the process definition. The phase and abstract activity levels of the function perspective are used to guide parties to the required branches of the concrete activity level. The communication perspective is used to facilitate the selection of concrete communication activities. The constraints defined over the communication activities are used in selecting the proper set of activities. The identification of leaf activities is a time-consuming process, but can be facilitated by previously defined specifications. This step may require additional decomposition of some of the leaf activities in the function perspective. When decomposition reveals cross-organizational activities, the communication perspective is used together with the defined constraints on communication activities. Companies make use of the set of operators defined in the model to set constraints on the identified activities. Then events—internal, external, or both—that trigger the identified leaf activities are specified [5].

In certain scenarios, companies can go one step further and define fully structured e-contracting process specifications. This is possible in contexts where the e-contracting process does not vary in the different business scenarios and all parties in the domain stick to a common, predefined e-contracting scenario. Another situation that allows usage of structured e-contracting process specifications is when the company has a significant influence over the trading relation and can dictate the contracting process. Structured process specifications can be defined by using the function and communication perspectives to identify the activities and then applying traditional process-modeling techniques, such as activity diagrams (part of UML) or Petri nets. Activity diagrams are used in the leading standardizing cross-organizational processes (e.g., ebXML and RosettaNet). Techniques from structured process modeling can also be used to specify a part of the e-contracting process. For example, if a company's internal activities do not require a high degree of flexibility, they can be specified using activity diagrams.

This set of steps for the construction of e-contracting process specifications will require adequate tool support. A software module for the construction of contracting process specifications is required to facilitate the identification of leaf activities and check process specifications. This module will be part of the design module of an e-contracting management system. It should be able to interface with the e-contracting management system (to export specifications).

Process Specification Usage

A company executing the information phase may find an advertisement that matches its requirements. The activities supported by the company and its matching party and the constraints applied on them are then compared, preferably in an automated manner. The result of this comparison is a statement indicating whether the process specifications of the two companies allow them to engage in a common e-contracting process (the common e-contracting process starts from the precontracting phase). However, it may be that the two specifications do not fully match or even have discrepancies. The differences between specifications may be in different sets of identified activities, at different levels of abstraction (e.g., a company may have specialized certain activities from the third level of the function perspective), different constraints applied on activities, and so on. In such cases, the process specifications have to be aligned (preferably in an automated manner). Different strategies can be applied for the alignment. For example, a company might accept the additional activities defined by the other party (union alignment); the other company might remove the additional defined activities (intersection alignment), and so on. The strategies will depend on the specific situation. The process of alignment is beyond the scope of this paper. Once the two process specifications are aligned, each company can start executing its part from the aligned e-contracting process specification.

Application of Model in the *Telegraaf* Case

As a practical example to complement the theoretical explanation in the preceding section, the proposed e-contracting process model will be used to specify part of the e-contracting process at *Telegraaf*. The activities performed by *Telegraaf* in its current contracting process are the foundation for the e-contracting process definition. The discussion will concentrate on part of the enactment phase because of the larger number of activities that may take place in this phase (*see Figure 13*) and consequently the greater complexity of creating a process specification.

Identification of Leaf activities in Enactment Phase

As was discussed above, the first step for a process designer is to identify the leaf activities from the function perspective performed by *Telegraaf* in the enactment phase. The abstract and concrete activities of the enactment phase in the function perspective are used at this stage. According to Figure 13, three abstract activities are part of the enactment phase: *Value exchange, Monitoring and control,* and *Evaluation.* Next, the designer investigates the concrete activities that may be performed by *Telegraaf* in the *Value exchange* abstract activity.

During the **value exchange**, the parties exchange the agreed values and the information required for the exchange. As explained earlier, *Telegraaf*

must publish the agreed number of impressions of advertisement(s) and receive the corresponding financial reward. Thus, the "Deliver service" and "Deliver reward" concrete activities (*see Figure 13*) must be included in the process specification by *Telegraaf*. For convenience, the process designer may rename the "Deliver service" concrete activity *Make impressions* and the "Deliver reward" concrete activity *Make payment*. The underlining of the *Make payment* activity indicates that it has to be performed by *Telegraaf*'s client. This activity is included in the process specification by *Telegraaf* in order to allow process alignment between *Telegraaf* and its clients. Thus, a client that wants to advertise at *Telegraaf* is required by *Telegraaf* to support this activity (and the constraints applied on it). According to Figure 13, data related to the exchange of values may be exchanged. In this business case, *Telegraaf* must receive and check the advertisement(s) for compliance with its requirements and send an invoice to the client. Thus, two conversations must be modeled. Both conversations follow the "inform" style. *Telegraaf* requires its client to send information (*Send advertisements*). *Telegraaf* will process and interpret the message (each received message must be processed and interpreted, but to make the discussion less complex, locutionary and illocutionary activities are omitted from the example process specification in this section). *Telegraaf* will perform the *Check advertisements* perlocutionary activity as a reaction to the customer's message. If necessary, it may also perform the *Change advertisements* internal activity. *Telegraaf* has to *react to the advertisements* (i.e., it may reject or accept the advertisement). Note that this activity is identified thanks to the communication perspective and was omitted in the initial presentation of the case earlier in the paper. *Telegraaf* will prepare an invoice as an internal activity that precedes the second conversation and will send the invoice (*Prepare invoice, Send invoice*). As *Telegraaf* does not anticipate any reaction to its invoice, it does not model a reaction activity to the *send invoice* activity. Finally, the following concrete activities are identified: *Send advertisements*; *Check advertisements*; *Change advertisements*; *React to advertisements*; *Make impressions*; *Prepare invoice*; *Send invoice*; *Make payment*.

This short example illustrates the usage of the function and communication perspectives. A process designer follows in a top-down manner the decomposition of each phase to abstract activities and the decomposition of each abstract activity to concrete activities to identify the activities applicable for the business context of the company. If a conversation must be defined, the process designer selects the communication style(s) that will be supported by the party, and defines the utterance, locutionary, illocutionary, and perlocutionary activities that the parties may perform. The process designer may define specializations of the leaf activities in the function perspective. For example, in the case of *Telegraaf*, the *Send enactment data* activity was specialized to *Send advertisements* and *Send invoice* activities. The other activities from the *Telegraaf* process are identified in an analogous way.

Definition of Constraints

The constraints listed below are defined on the concrete activities identified in the preceding subsection:

C1. SEQUENCE (<u>Send advertisements</u>, Check advertisements).
C2. IMPLY (<u>Send advertisements</u>, Check advertisements).
C3. SEQUENCE (Check advertisements, React to advertisements).
C4. IMPLY (Check advertisements, React to advertisements).
C5. SEQUENCE (Check advertisements, Change advertisements).
C6. IMPLY (React to advertisements, Make impressions, ads approved).
C7. SEQUENCE (Prepare invoice, Send invoice, <u>Make payment</u>).
C8. IMPLY (Prepare invoice, Send invoice).
C9. IMPLY (Send invoice, <u>Make payment</u>).
C10. REQUIRED (<u>Send advertisements</u>).

C1 states that activities *<u>Send advertisements</u>, Check advertisements* are executed in this sequence. C2 states that if a client sends an advertisement, *Telegraaf* must check it. C3 and C4 state that after checking the advertisements, *Telegraaf* must inform its client whether it accepts or rejects the advertising material. C5 states that if *Change advertisement* is to be executed, it is executed after the advertisements are checked. C6 states that if the advertisements are approved, advertising may start (as scheduled in the contract). Analogously, C7, C8, and C9 define the sequence of execution and implication of activities. C10 states that the client must send the advertisements for contract enactment to take place.

Constraints are defined analogously on the other activities from the process used by *Telegraaf*.

Discussion

The e-contracting process model used above to define a part of the e-contracting process specification of *Telegraaf*. The application of the model showed that the combination of communication and function perspectives facilitates the identification of concrete activities in an e-contracting process, minimizing the risk of omitting concrete activities from the process specification.

The model allows the definition of a complete e-contracting process of *Telegraaf* (related to Requirement 1). In this case, a number of utterance acts (e.g., the *Send enactment data* activity) from the function perspective had to be additionally specialized. Thus, as already discussed, the proposed process model can be further elaborated by providing lower levels of detail that reflect domain or company specifics (related to Requirement 2). The flexibility of the model allowed the definition of a process specification specifically for the context of *Telegraaf* (related to Requirement 3). The usage of constraints on activities instead of fully structured processes allowed the definition of a flexible e-contracting process, allowing *Telegraaf* to decide on the execution of certain activities or on the sequence of their execution depending on the context of a specific scenario (related to Requirement 4). For example, according to the process specification, *Telegraaf* may send the invoice at any point in time (prior to, during, or after the advertising campaign takes place). The model allows *Telegraaf* to define a set of required communication activities that a counter-party must support. The usage of the communication perspective

by both *Telegraaf* and its client is a foundation for coherent communications between them during e-contracting (related to Requirement 5). In addition, the usage of speech-act theory as a foundation of the communication perspective allowed seamless integration between the external and internal activities of *Telegraaf* (related to Requirement 6).

Thus, the example shows that the e-contracting process model can be applied for the definition of an e-contracting process in the case of *Telegraaf*. It shows that in this case the model addresses the six requirements defined earlier in the paper. The model will have to be applied for e-contracting process specification in different business domains to prove its applicability and compliance with the identified requirements.

Comparison with Related Work

The e-contracting process model will now be compared to analogous models in two established standardization frameworks, RosettaNet and ebXML [8, 26]. The discussion will also consider a specific proprietary development, Solution Composer, a business-modeling tool developed by SAP AG, a leading company in this domain, which provides an e-business platform that supports internal business processes and collaborations between companies.

RosettaNet

In RosettaNet, a three-level hierarchy of the activities of companies is defined [267]. The hierarchy is built by decomposing the domain of e-business supply-chain activities. The e-business supply-chain domain is divided into clusters, and the clusters are divided into segments, each of which contains one or more Partner Interface Processes (PIPs). A PIP is a leaf of the decomposition tree. The three levels of decomposition of the e-business supply-chain activities have roles similar to the three levels of abstraction in the function perspective of the proposed model—achieving completeness by gradually reducing abstraction. The level of abstraction of PIPs is comparable to the level of concrete activities in our function perspective. A PIP specifies the business roles for a given business process, the conversation activities between the roles, and the business documents exchanged by the partners. A PIP provides a conversation specification by defining the communication activities in a structured process. In RosettaNet, two communication patterns exist: request-response pairs of activities and notification activities. A PIP includes only one pattern. Thus, a party is not allowed to apply "push" behavior in a PIP that employs the request-response pattern. This limits the communication flexibility of parties. The communication pattern in the proposed model only applies constraint on the activities and gives maximum flexibility for communication to the contracting parties. Also in contrast to RosettaNet, it allows more complex conversations to be defined and performed in the form of a react-react pattern.

Unlike the approach proposed in this paper, RosettaNet gives no attention to internal business processes and their relation to cross-organizational

processes. As a result, it only defines communication activities, whereas our approach describes the complete e-contracting process.

RosettaNet provides a "semi-structured" process model. That is to say, communications between parties are specified as structured processes, but there are no constraints (or operators for definition of constraints) on the relations between PIPs. In some PIPs, relations to other PIPs are briefly mentioned in a free text format.

ebXML

In ebXML, the Business Process Specification Schema (BPSS) provides a framework for the specification of business processes [8]. A Collaboration Protocol Profile (CPP) defines the capabilities and requirements of a party in a possible e-commerce process. The CPP embeds or references the party's Business Process Specification. Two parties that want to establish business relations compare their CPPs. The intersection of the CPPs that is mutually agreed on defines the Collaboration Protocol Agreement (CPA) [9].

The business transaction choreography in BPSS defines the order in which transactions are executed. The rules that define the order of activity execution are based on activity diagram control flow concepts (e.g., transition guards). Business collaborations in BPSS may contain a set of business activities. A business activity can be a business transaction or a business collaboration. Transactions are atomic units that cannot be further decomposed. The decomposition of the business collaborations, similar to the approach in the function perspective, provides a way to reduce the complexity of process specifications. However, the use of activity diagram control flow concepts implies structured business processes, which limits the flexibility of parties. Our e-contracting process model allows the definition of flexible e-contracting process specifications. The communication pattern that guarantees coherent communication between parties in BPSS is simpler than in our approach. The only possible communication in a business transaction is through request-response pairs of activities. Responses in these pairs are optional. Thus, a request without a response in BPSS is equivalent to an inform activity in our communication pattern. A series of request-response pairs that simulate a two-way conversation between parties is equivalent to a request followed by a response followed by an arbitrary number of reactions. Although the communication pattern in BPSS is simpler in its definition, the requirement for each response activity to be preceded by a request activity can unnecessarily complicate conversations between parties. In contrast to ebXML, our proposed approach gives explicit attention to coupling between internal and external activities.

SAP Solution Composer

The composer tool provided by SAP (www.sap.com) is an example of a proprietary solution for business modeling. This tool is intended for the construction of process specifications. It does not prescribe a modeling method.

In addition to the modeling tool, SAP provides predefined industry-specific and cross-industry business process models that can be used as templates for the creation of concrete business processes. However, they are not flexible and imply specific choices. For example, in the Request for Quotation (RfQ) model, the consumer publishes a RfQ, evaluates the received responses, and sends notification to the supplier with the best offer. Since the model is predefined, it does not allow a request for customization or for more information to be sent to one or more of the suppliers in order to obtain a personalized quotation or information not given in the offer.

The set of predefined models is incomplete because contract negotiation and establishment are not part of the processes it describes. A taxonomy is used to structure the list of predefined business models. The criteria for the classification of the processes in the taxonomy are ambiguous and do not allow easy identification of the models. Relations between different process specifications are not elaborated.

Conclusions

Business-to-business contracting processes are strongly influenced by the business and legal contexts. In consequence, contracting processes are characterized as complex and variable. Contracting parties must adapt their contracting processes to the context.

This paper defines an e-contracting process model. The model is based on two perspectives and on a set of operators that define constraints. It allows the definition of complete, detailed, and flexible contracting process specifications, and facilitates the seamless integration of cross-organizational and internal contracting activities. The model's applicability and compliance with the requirements of an e-contracting process model was demonstrated by applying it to the e-contracting process for on-line advertising of the Dutch news medium *Telegraaf*. Specifications created with the model serve as a foundation for the automated management of diverse and flexible contracting relations. Flexible IT support of trading relations will lead to greater popularity and endorsement of electronic contracting by businesses that currently view it as a restrictive and inflexible solution.

The model proposed here can also be used to analyze existing contracting process specifications. Omissions and inconsistencies can be discovered, and existing process specifications can be improved.

The development of an e-contracting system that supports flexible e-contracting process specifications is a logical next step. Work on the design of a detailed, multi-level, reference architecture for e-contracting is currently being finalized.

REFERENCES

1. Alonso, G.; Fiedler, U.; Hagen, C.; Lazcano, A.; Schuldt, H.; and Weiler, N. WISE: Business to business ecommerce. In **<<name (ed.),>>** *Proceedings*

of the 9th International Workshop on Research Issues on Data Engineering: Information Technology for Virtual Enterprises. Los Alamitos, CA: IEEE Computer Society, pp. 132–139.

2. Angelov, S., and Grefen, P. Support for B2B e-contracting: The process perspective. In V. Marik, L.M. Camarinha-Matos, and H. Afsarmanesh (eds.), *Knowledge and Technology Integration in Production and Services: Balancing Knowledge in Product and Service Life Cycle.* <<**place:**>> Kluwer, 2002, pp. 87–96.

3. Austin, J. *How to Do Things with Words.* Cambridge: Harvard University Press, 1975.

4. Casati, F.; Castano, S.; Fugini, M.; Mirbel, I.; and Pernici, B. Using patterns to design rules in workflows. *IEEE Transactions on Software Engineering, 26,* 8, (2000), 760–785.

5. Chiu, D.K.W.; Cheung, S.C.; and Till, S. A three-layer architecture for e-contract enforcement in an e-service environment. In <<**name (ed.),**>> *Proceedings of the 36th Annual Hawaii International Conference on System Sciences.* Los Alamitos, CA: IEEE Computer Society, 2003, pp. 74–74.

6. Dan, A.; Dias, D.; Nguyen, T.; Sachs, M.; Shaikh, H.; King, R.; and Duri, S. The Coyote project: Framework for multi-party e-commerce. In <<**name (ed.),**>> *Proceedings of the Second European Conference on Research and Advanced Technology for Digital Libraries.* London<<**place ok?**>>: Springer-Verlag, 1998, pp. 873–889.

7. diCarta Inc. Enterprise contract management. 2002.

8. ebXML. ebXML Business process specification schema, v1.01 6. 2001.

9. ebXML. Technical architecture specification v1.0.4. 2001

10. FIPA. FIPA ACL message structure specification. SC00061G. Geneva: Foundation for Intelligent Physical Agents, 2002.

11. FIPA. FIPA communicative act library specification. SC00037J. Geneva: Foundation for Intelligent Physical Agents, 2002.

12. FIPA. FIPA contract Net interaction protocol specification. SC00030H. Geneva: Foundation for Intelligent Physical Agents, 2002.

13. Flores, F.; Graves, M.; Hartfield, B.; and Winograd, T. Computer systems and the design of organizational interaction. *ACM Transactions on Information Systems, 6,* 2, (1988), 153–172.

14. Gisler, M.; Stanoevska-Slabeva, K.; and Greunz, M. Legal aspects of electronic contracts. In H. Ludwig, Y. Hoffner, C. Bussler, and M. Bichler (eds.), *Proceedings of the CAISE*00 Workshop on Infrastructure for Dynamic Business-to-Business Service Outsourcing.* CEUR-WS.org, 2000.

15. Governatori, G. Representing business contracts in RuleML. *International Journal of Cooperative Information Systems, 14,* nos. 2–3 (2005), 181–216.

16. Governatori, G.; Milosevic, Z.; and Sadiq, S. Compliance checking between business processes and business contracts. In <<**name (ed.),**>> *Proceedings of the 10th IEEE International Enterprise Distributed Object Computing Conference.* Los Alamitos, CA: IEEE Computer Society, 2006, pp. 221–232.

17. Grefen, P.; Aberer, K.; Hoffner, Y.; and Ludwig, H. CrossFlow: Cross-organizational workflow management in dynamic virtual enterprises. *International Journal of Computer Systems Science and Engineering, 15,* 5 (2000), 277–290.

18. Griffel, F.; Boger, M.; Weinreich, H.; Lamersdorf, W.; and Merz, M. Electronic contracting with COSMOS—how to establish, negotiate and execute electronic contracts on the Internet. In <<**name (ed.),>>** *Proceedings of the 10th IEEE International Enterprise Distributed Object Computing Conference Proceedings of the 2nd International Workshop on Enterprise Distributed Object Computing.* <<**place: publisher,>**> 1998, pp. 46–55.

19. Jennings, N.R.; Faratin, P.; Norman, T.J.; O'Brien, P.; and Odgers, B. Autonomous agents for business process management. *International Journal of Applied Artificial Intelligence, 14*, 2 (2000), 145–189.

20. Keen, P.; Balance, C.; Chan, S.; and Schrump, S. *Electronic Commerce Relationships: Trust by Design.* Upper Saddle River, NJ: Prentice Hall, 2000.

21. Kimbrough, S., and Moore, S. On automated message processing in electronic commerce and work support systems: Speech act theory and expressive felicity. *ACM Transactions on Information Systems, 15*, 4 (1997), 321–367.

22. Milosevic, Z., and Bond, A. Electronic commerce on the Internet: What is still missing? In <<**name (ed.),>>***Proceedings of the 10th IEEE International Enterprise Distributed Object Computing Conference: Proceedings of the 5th Annual Conference of the Internet Society.* Los Alamitos, CA: IEEE Computer Society, 1995. <<**pp.?>>**

23. Milosevic, Z.; Jøsang, A.; Dimitrakos, T.; and Patton, M.A. Discretionary enforcement of electronic contracts. In <**name (ed.),**>*Proceedings of the 10th IEEE International Enterprise Distributed Object Computing Conference: Proceedings of the 6th International Enterprise Distributed Object Computing Conference.* Los Alamitos, CA: IEEE Computer Society, 2002, pp. 39–50.

24. Oracle. *Oracle Enterprise Contract Management: The Roadmap to Success.* 2001. **<<is this a book? If so give place of publication and name of publisher**. **If not, other means of accessing>>**

25. Putman, J. *Architecting with RM-ODP.* Upper Saddle River, NJ: Prentice Hall, 2001.

26. RosettaNet. RosettaNet Implementation Framework: Core Specification (RNIF 02). 2001.

27. SAP. Solution Composer quick guide. 2004.

28. Schoop, M., and Quix, C. Towards effective negotiation support in electronic marketplaces. In <<**name (ed.),>>** *Proceedings of 10th Annual Workshop on Information Technologies & Systems.* **<<place: publisher>>,** 2000. pp. 1–6.

29. Searle, J. *Speech Acts: An Essay in the Philosophy of Language*. Cambridge: Cambridge University Press, 1969.

30. Tut, M., and Edmond, D. The use of patterns in service composition. In <<**name (ed.),>>** *CAiSE '02/ WES '02: Revised Papers from the International Workshop on Web Services, E-Business, and the Semantic Web*. London<<place **ok?>**>: Springer-Verlag, 2002, pp. 28–40.

31. van Dijk, A., The contracting agent: Concepts and architecture of a generic software component for electronic business based on outsourcing of work. Ph.D. dissertation, Eindhoven University of Technology, 2001.

32. Weigand, H.; Schoop, M.; de Moor, A.; and Dignum, F. B2B negotiation support: The need for a communication perspective. *Group Decision and Negotiation, 12*, 1 (2003), 3–29.

33. Xu, L., Monitoring multi-party contracts for e-business. Ph.D. dissertation, University of Tilburg, 2004.

Appendix

The complete set of specializations of utterance activities is presented below. These specializations were defined by identifying cross-organizational communications in the abstract activities (second level of detail of the function perspective) and decomposing them using the SAT framework.

The request activities are shown in Figure 15. In addition to requests to counter-parties, companies can also send requests to third parties (e.g., external brokers). Request activities that can be targeted to third parties are represented with a dotted pattern.

For each request, there is a corresponding response. Figure 16 presents the specializations of the respond utterance act.

Figure 17 presents the specializations of the inform activity.

Figure 18 shows specializations of the react activity.

As is apparent from the list of react activities, a number of inform and respond activities do not have a corresponding react activity (e.g., the *send advertisement* inform activity). The reason for this is that these inform and respond activities cannot invoke an external react activity, but would rather have an internal reaction or an external reaction in the form of the utterance
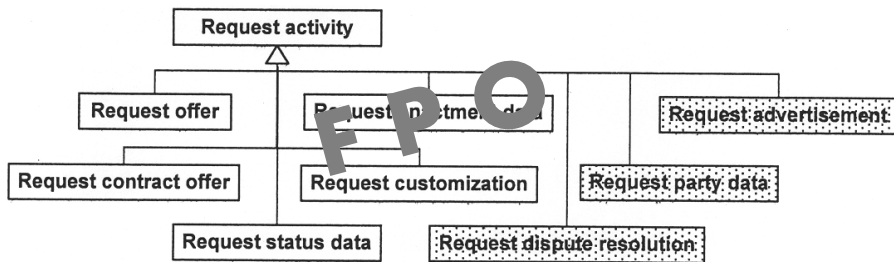

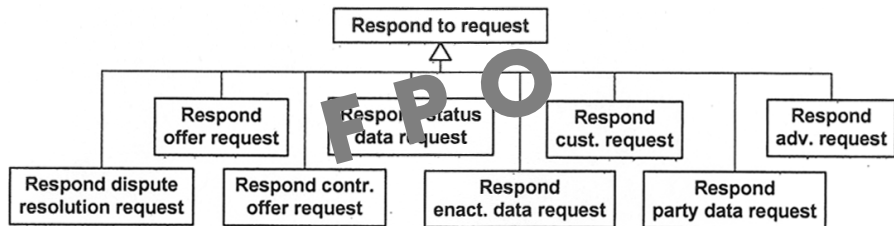**Figure 15. Communication Perspective: Specializations of Request Utterance Activity <<SUPPLY DATA FILE>>**

Figure 16. Communication Perspective: Specializations of Respond Utterance Activity <<SUPPLY DATA FILE>>
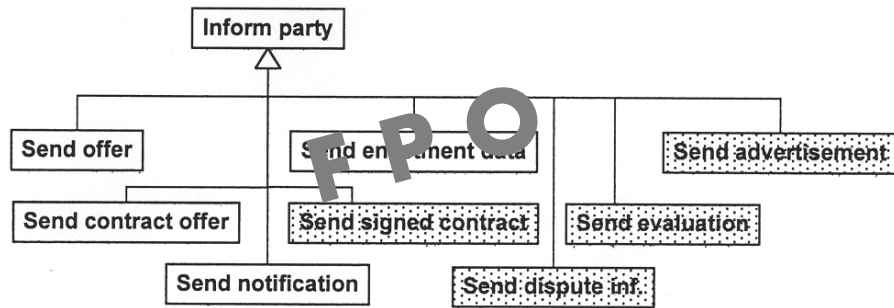
Figure 17. Communication Perspective: Specializations of Inform Utterance Activity <<SUPPLY DATA FILE>>
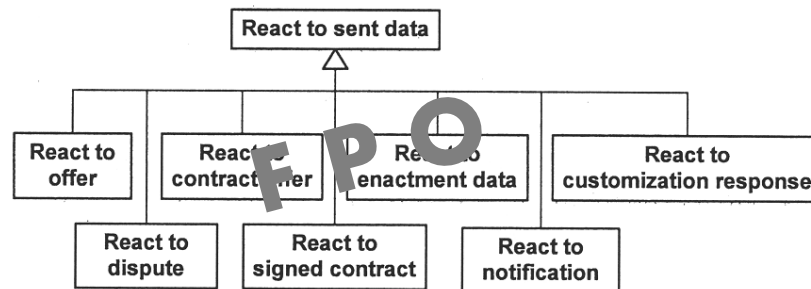

Figure 18. Communication Perspective: Specializations of React Utterance Activity <<SUPPLY DATA FILE>>

act initiating a new conversation. For example, the *send advertisement* activity can only trigger certain internal processes or the start of an offer exchange.

SAMUIL ANGELOV (s.angelov@tue.nl) is an assistant professor in the Department of Technology Management at Eindhoven University of Technology, from which he received his Ph.D. in February 2006. In the period from 1998 to 2003, he worked for the Bulgarian Academy of Sciences, Sofia University, and Twente University, and was involved in the European INCO COPERNICUS project "MALL 2000." During his Ph.D. studies, he participated in the SOBI project of the Telematica Institute and focused on the design of information systems for highly automated electronic contracting. He is a member of SOCOLNET. His research interests are in the field of enterprise and information system modeling and focus on e-commerce and e-contracting.

PAUL GREFEN (p.w.p.j.grefen@tue.nl) has been a full professor in the Department of Technology Management at Eindhoven University of Technology since 2003, chairing its Information Systems subdepartment and leading the ICT Architectures group. He received his Ph.D. in 1992 from the University of Twente, where he held assistant and associate professor positions in the Computer Science Department until early 2003. He was a visiting researcher at Stanford University in 1994. He was involved in the WIDE ESPRIT project, which focused on advanced database support for workflow management systems, the CrossFlow IST project, which aimed at cross-organizational workflow support for dynamic virtual enterprises, and the CrossWork IST project, focusing on advanced process support in the automotive industry. He is currently in-

volved in the XTraConServe NWO project working on contracted transactional services. He has served on the program committees of many international conferences, was the senior editor of a book on the WIDE project, and has published a book on workflow management. He is a member of the editorial boards of the *International Journal of Cooperative Information Systems* and the *International Journal of Service Oriented Computing and Applications*, and of the management board of the Beta Research School. His current research interests include architectural design of complex information systems, interorganizational workflow management, high-level transaction management, and contract support in e-business.