Three-Level Process Specification for Dynamic Service Outsourcing: From Petri Nets to ebXML and WFPDL

Paul Grefen and Samuil Angelov

Computer Science Department, University of Twente
P.O. Box 217, 7500 AE Enschede, Netherlands
www.cs.utwente.nl/~{grefen,sangelov}
{grefen,sangelov}@cs.utwente.nl

Abstract. Service outsourcing is the business paradigm, in which an organization has part of its business process performed by a service provider. In dynamic markets, service providers are selected on the fly during process enactment. The cooperation between the parties is specified in a dynamically made electronic contract. This contract includes a process specification that is tailored towards service matchmaking and cross-organizational process enactment and hence has to conform to market and specification standards. Process enactment, however, relies on intra-organizational process specifications that have to comply with the infrastructure available in an organization. In this paper, we present a three-level process specification framework for dynamic contract-based service outsourcing. This framework relates the two process specification levels through a third, conceptual level. Petri nets are used for conceptual process specification, whereas the external and internal levels are based on the ebXML standard respectively workflow management technology. We show how the framework can be placed in the context of infrastructures for cross-organizational process support.

1 Introduction

Nowadays, many organizations focus on specific parts of their business process and rely on partners in a market to perform the additional parts of the process required to reach their business goals. A common business paradigm is that of service outsourcing, in which an organization focuses on its core business process (the central part of its primary process) and has secondary process parts enacted on its behalf by service provider organizations. In this paradigm, the outsourcing organization is referred to as service consumer. The details of service outsourcing are specified in a contract between both parties.

Service outsourcing can be found in many market segments. An example is an insurance company that focuses on the core insurance process and outsources secondary activities like customer call handling, and insurance claim loss assessment. Other ex-

H. Ehrig et al. (Eds.): Petri Net Technology ... , LNCS 2472, pp. 264-280, 2003.

amples can be found in the logistics domain, where companies outsource their logistic subprocesses to specialist organizations in this field. We will elaborate an example from this domain in the sequel of this paper.

The combination of service consumer and service provider can be seen as a virtual enterprise that presents itself to a third party (for example a customer) as a single entity. Traditionally, these virtual enterprises have a more or less stable character over time. In modern dynamic e-commerce markets, however, players in a market and general market conditions change that fast, that a more dynamic approach is required to service outsourcing to create or retain a competitive position. This means that in service outsourcing, service consumers dynamically determine which service providers to use in the enactment of their business processes.

Dynamic service outsourcing implies a number of steps: services have to be identified and defined, compatible business partners have to be found in an efficient way, contracts have to be made dynamically, process enactment infrastructures have to be set up and coupled, and the process has to be actually enacted. Requirements to speed and costs dictate that all of this takes place electronically, preferably in a fully automated fashion.

To enable this, clear specifications are necessary that describe business processes in the context of the above-mentioned steps in service outsourcing. Given the fact that these steps are quite diverse in nature, a single process specification will not be adequate for all these activities. Hence, we propose a multi-level process specification approach to be used in dynamic service outsourcing. In this approach, each level has its own specification characteristics and related specification techniques. The level of process specification aimed at collaboration has to be embedded in a framework for electronic contracts to support the proper specification of the business relationship between partners in an outsourcing relationship. As such, the electronic contracts are the specification of the dynamic virtual enterprise, in which the process specification provides the operational interaction specification.

The multi-level framework we propose is inspired by the ANSI-SPARC model [18] that is well known in the database community for describing three levels of data management. The general approach to service outsourcing used in this paper is inspired by the approach taken in the CrossFlow project [12, 14].

1.1 Structure of this Paper

In Section 2, we first present our approach in terms of a three-level process model, explaining what the global purpose of each of the levels is. In Sections 3 to 5, we discuss each of the three levels and the mappings between them in more detail. At the central level, Petri Nets are used as specification technique. At the two other levels, we use UML activity diagrams in the context of ebXML respectively a workflow process definition language of a commercial workflow management system. The levels are illustrated by means of a running example from the telecom and logistics area. We place the process model in the context of process support infrastructures in Section 6. We end the paper with conclusions. Main purpose of this paper is to paint the overall picture of our approach to multi-level process support for service outsourcing and show the position of various process specification techniques in this approach.

2 A Three-Level Process Model

In this section, we introduce our three-level process specification framework for dynamic service outsourcing. We first explain the need for a separate conceptual process specification level. Then, we present the three levels of the framework and their relationships.

2.1 The Need for a Conceptual Specification Level

In service outsourcing, we require process specifications that specify all necessary details of process and data structures with respect to these services and their enactment. These specifications exist on two levels from an operational point of view.

Firstly, we require process specifications that can be shared between multiple organizations (service consumer and service providers). These specifications should allow for service brokering between partners when setting up a virtual enterprise on the one hand, and service enactment in a running virtual enterprise on the other hand. Brokering is used to relate service consumers and providers, either through simple matching of specification characteristics or through more advanced matchmaking. In service enactment, services are actually executed in an inter-organizational setting. Interoperability is the main characteristic of this specification level. The nature of this level is often imposed by market standards. If an organization operates in multiple markets, it may be confronted with multiple standards.

Secondly, we require process specifications at the implementation level of the process enactment systems employed by each of the organizations cooperating in a virtual enterprise. Executability is the main characteristic of this specification level. The nature of this level is often imposed by current process and data management systems – be it of an up-to-date or legacy nature. Again, an organization having a heterogeneous infrastructure may be confronted with multiple specification types here.

None of the above two levels is focused on conceptual process specification, however, that is independent from both practical collaboration and implementation aspects. A conceptual specification is required, though, for three main reasons:

1. A conceptual specification is necessary to guard the quality of processes through proper design, analysis, and validation of process structures. Certainly, in the field of e-business where automated processes are the backbone of an organization, quality of these processes is of the highest importance.
2. A conceptual specification is used as a ‘semantic bridge' between the standards used at the interoperability level on the one hand and the implementation level on the other hand. This bridge provides a separation of concerns with respect to specific details of semantics at the two levels.
3. A conceptual specification is also required as a unifying level to bridge multiple specification types at both interoperability and implementation levels. As such, it is a basis for portability of process specifications across multiple markets with diverse standards on the one hand and multiple process and data management systems with diverse technologies on the other hand.

2.2 The Three-Level Framework

For the reasons discussed above, we introduce a third, conceptual process specification level between the two levels discussed above. Consequently, we arrive at the following three-level process specification model:

External Level: The external level is geared towards communicating a process specification between different organizations. It can be considered a projection of the conceptual level, where projection uses hiding and translation operations.

Conceptual Level: The conceptual level is the centerpiece of process specification. It is independent from external use and internal implementation. It is used for conceptual reasoning about the process, e.g. for design and analysis purposes. The conceptual level is a combination of abstraction and aggregation of the internal level.

Internal Level: The internal level is geared towards enactment of processes in the context of a specific organization, e.g., by means of workflow management systems. The internal level is a mapping of the conceptual level, where mapping is a combination of translation (specialization for a specific platform) and refinement.

The relation between the three process levels in service outsourcing is illustrated in Figure 1. In this figure, we see that two organizations can each have local conceptual and internal process specifications, but that they share a common external specification. This common specification can be based on a market standard, can be exported by one organization and imported by the other, or can be constructed in a process of negotiation between the two organizations.

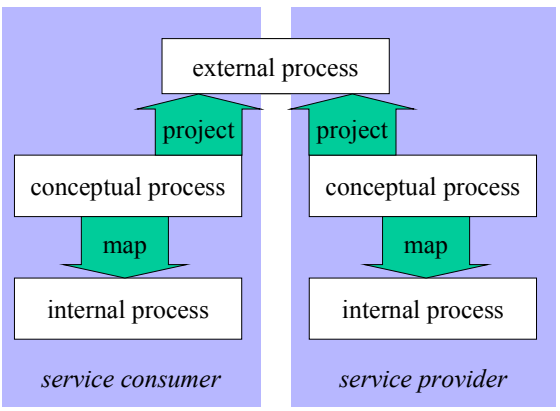

Fig. 1. Three-level framework

Note that the three above levels coincide more or less with the well-known ANSI-SPARC three-level model in the database community [18, 8]. The ANSI-SPARC model distinguishes between external, conceptual and internal levels for data management. Aims of the conceptual and internal level are similar to that in our approach – except of course for the data versus process management perspective. Main difference is the use of the external level. In the ANSI-SPARC model, the external level contains multiple views on a database that cater for different user groups in an organi-

zation. In our model, the external level contains multiple ‘views' on a process to cater for different organizations in an electronic market.

In the sections below, we discuss the levels of the framework in more detail. We start with the conceptual level, as this level is the pivotal point in our framework. Next, we move ‘up' to the external level, paying attention to the nature of this level and the projection from the conceptual level. After that, we move our attention ‘down' to the internal level and to its mapping from the conceptual level.

3 The Conceptual Level

The conceptual process specification level is the center point in our framework for cross-organizational process support. It is used for the design, analysis and verification of cross-organizational processes. In this section, we first discuss the nature of this process specification level. We illustrate this level with an example process specified in Petri Nets. Next, we pay attention to design and analysis of conceptual process specifications.

3.1 The Nature of the Conceptual Process Specification Level

The conceptual process exhibited by an organization to the outside world is usually less detailed than the implementation of the same process actually enacted by the organization. Consequently, several levels of detail exist in a conceptual process specification. These levels of detail correspond to process aggregation levels. The mapping between these aggregation levels is dealt with by a process refinement hierarchy.

As the conceptual specification level is the design and analysis level of our process specification framework, we place the process refinement levels that have a conceptual meaning within this level. Refinement that is related to mapping to process enactment infrastructures is dealt with in the mapping between conceptual and internal process specification – see Section 5.2. This ensures a separation of concerns between the ‘intention' of a process and its implementation on a specific platform.

To properly support design and analysis tasks at the conceptual level, an adequate process specification technique has to be selected. As we deal with discrete business processes, we require a discrete process modeling technique. As unambiguous reasoning about process specifications should be possible at this level, we require a specification technique with a formal background. Clearly, the chosen technique should be mappable to both the external and the internal process specification levels.

Variants on Petri Nets are widely accepted in the business process reengineering and workflow management domains (see for example [1, 2]), both from a research and from an industrial perspective. Their token-based semantics are well suitable to model properties of business processes. Also, many tools exist for design and analysis of Petri Nets. Therefore, we use Petri Nets as the specification technique for the conceptual specification level in this paper. We illustrate this below by means of an example process.

3.2 An Example Conceptual Process Specification

Our example is based on a real-world logistics scenario in the telecom industry developed in the CrossFlow project [6, 12]. In this scenario, a telecom operator sells mobile phones (GSMs) and network subscriptions for these phones. The subprocess of delivering the phones to clients is dynamically outsourced to a logistics company. The example process at the telecom operator is specified as a Petri Net at the conceptual level in Figure 2.

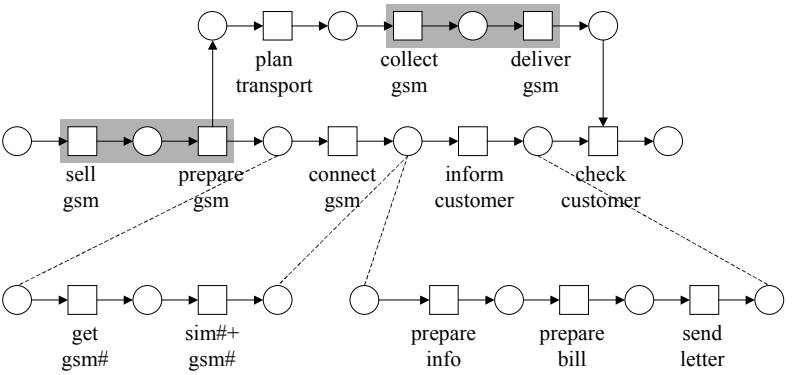

Fig. 2. Example conceptual process, levels 1 (top) and 2 (bottom)

In the upper part of the figure, we see the level 1 conceptual process model. The telecom operator sells a GSM, using telephone or web communication with its client. It then prepares the GSM for transport. After that, two parallel process branches are executed. In-house, the GSM is connected to the network and the customer is informed by mail about the connection, the personal identification code and the subscription. The process of physically delivering the GSM to the customer is outsourced to a logistics operator, who performs the delivery in three sequential steps[1]. After completion of the two parallel branches, the telecom operator checks up on the customer by phone.

Apart from the control flow of the process, certain execution characteristics are required, e.g. with respect to transactional properties or timing of processes. Two examples are drawn as gray boxes in Figure 2 (following the idea of spheres from [5]). The box around ‘sell gsm' and ‘prepare gsm' indicates a transactional property: these two activities should be executed atomically. The box around ‘collect gsm' and ‘deliver gsm' indicates a timing property: these two activities should be completed within a certain timeframe (i.e., after the phone has been picked up at the telecom company, it should be delivered at the client in say three days). Note that in practice, more execution characteristics will be specified.

In the lower part of Figure 2, we see (part of) the refinement of the level 1 into the level 2 process model. Activity ‘connect gsm' has been refined into two activities for retrieving a GSM phone number respectively coupling this in the administration to the

[1] Note that we have used a simple example outsourced process for reasons of brevity and clarity. The outsourced process in [6] contains more than 10 steps and alternative process flows.

SIM number of the phone. Activity ‘inform customer' has been refined into three activities for preparing customer information, preparing a bill and sending a letter to the customer including information and bill.

3.3 Designing, Analyzing and Verifying Conceptual Process Specifications

Design of processes takes place at the conceptual level, preferably in a top-down fashion. In a service outsourcing scenario, this means that at the highest aggregation level of the conceptual level, the conceptual interaction between service consumer and service provider are determined.

At this level, the service consumer specifies its conceptual cross-organizational business process in conceptual terms. This process contains activities that are enacted in-house, which are refined in the conceptual level if necessary. It also contains activities to outsource, which are not further refined, but enacted by a service provider.

The service provider also specifies the service it offers in conceptual terms at this level. The service specification may be extended with additional activities that are internal to the service provider, for example administrative activities. All activities can be refined within the conceptual process specification level.

Refinement of processes at the conceptual level should stop at the level where independence from execution infrastructures cannot be guaranteed. This means, for example, that process properties of specific workflow management systems should not be reflected at the conceptual level.

Summarizing the above, there is a highest process aggregation level at the conceptual specification level – determined by collaboration characteristics – and a lowest aggregation level – determined by platform independence. We call these levels the conceptual upper and lower bound. This is illustrated in Figure 3, where we see three conceptual refinement levels (C1 to C3) between external (E) and internal (I1 and I2) levels.

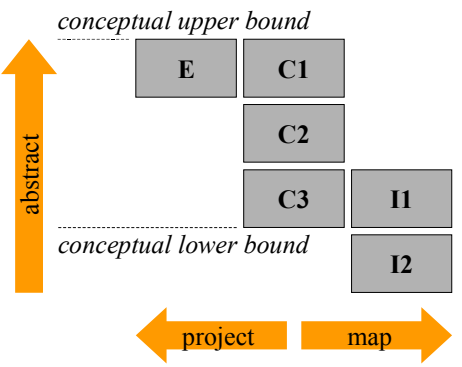

Fig. 3. Conceptual upper and lower bound

The nature of the conceptual process specification allows for analysis and verification of processes. Analysis may, for example, focus on transactional characteristics of business processes [5]. Transactional characteristics determine for example which parts of a process must be executed in an atomic fashion and under which conditions

access to shared data is possible. Analysis may also have a more quantitative character, for example by using simulations of process runs. Verification techniques may be used to prove certain characteristics of processes, e.g., reachability or termination characteristics. Analysis and verification lead to the assessment of quality characteristics of services that may be required as attributes at the external process specification level.

4 The External Level

Interoperability of processes is the main focus of the external level. Below, we discuss the nature of external process specifications and their mapping from the conceptual level. We illustrate this level with an example. Next, we discuss the role of external process specifications in electronic contracts. We end this section with the brief discussion of an example contract model that includes explicit process specification.

4.1 The Nature of External Process Specifications

Process specifications at the external level aim at defining interoperation (or collaboration) of two organizations. More concrete, these specifications are used for two purposes:

Service Matchmaking: The external process specification is used in matchmaking between service consumers and service providers, i.e., in setting up a virtual enterprise. The external process required by a consumer should match with an external process offered by a provider in order to link their processes.

Cross-Organizational Process Control: The external process specification is used for controlling the cross-organizational service enactment, i.e., it is the basis for cross-organizational process monitoring and control.

Translation of the conceptual process specification to the external level is necessary to conform to market standards – this to allow interoperability between partners. Where the conceptual process specification uses a formal process model – as explained above – the external model is based on a process specification standard that is common within a certain market. In this context, we find XML-based standards like ebXML [7] or WSDL [22]. In this paper, we assume ebXML as the basis for the external level. Consequently, we use UML activity diagrams as process specification technique (see also Section 6.2) – we illustrate this in the sequel. The semantics of activity diagrams in workflow specification is discussed in [9].

A process specification at the external level can be a high-level abstraction of a complex process that on the conceptual level is considerably refined into separate subprocesses. Still the process specifications at the external level should not be too general as they may turn the process of service outsourcing in a black box process, thereby not allowing the fine-grained cooperation that is required in dynamic virtual enterprises. Clearly, this is closely related to the discussion of the conceptual upper bound in the previous section.

The external level includes service execution characteristics that define how the service is executed. These characteristics can be of different natures. An important class of characteristics is the specification of the transactional behavior of service execution, determining for example how failures and concurrent access to shared resources are handled. A second class is the specification of monitoring and control points in the service process and available control primitives – this to specify how a service consumer can observe and influence the enactment of an outsourced service. Quality of service attributes, e.g. execution times or success probability, form another important class.

4.2 Mapping the Conceptual Level to the External Level

The external specification is a projection of the conceptual process specification, where projection is a combination of hiding and translating.

Parts of the conceptual process can be hidden in the external process because they are not relevant for a cooperation partner. A service consumer might hide everything of its conceptual process but the subprocess to be outsourced. A service provider will usually show most of its conceptual process, but might hide some administrative tasks at the end of a service specification. Process steps may also be renamed from conceptual to external level, as internal names may not coincide with standard names in a market.

Translation is necessary to transform primitives from the conceptual process specification technique into the process specification standard used at the external level. Also, execution quality characteristics determined at the conceptual level have to be translated into service attributes at the external level.

4.3 Example External Process Specification

The top part of Figure 4 shows the external process based on the conceptual process discussed before. As indicated before, we base the external level specification on ebXML. Consequently, we use UML activity diagrams as the graphical process specification technique (see also Section 6.2).

We show how the top-level conceptual process is projected onto the external level. Only the activities to be outsourced are represented at the external level in the external representation language. We see that steps in the process are renamed. Where the conceptual specification uses telecom-specific names, the external specification uses names generally used in the logistics industry: ‘collect gsm' is for example renamed into ‘collect parcel'.

The mapping between Petri Net and activity diagram is straightforward in this simple case: each transition in the Petri Net is mapped onto an activity in the activity diagram. In more complex cases, the mapping may be less trivial. In [10], a comparison is made between the two techniques, elaborating the differences between them in the context of workflow management.

Enactment characteristics specified in the conceptual process specification are mapped to separate enactment clauses at the external level (shown in the upper right corner in the figure). Both process specification and enactment clauses must be integrated into one transferable process specification.

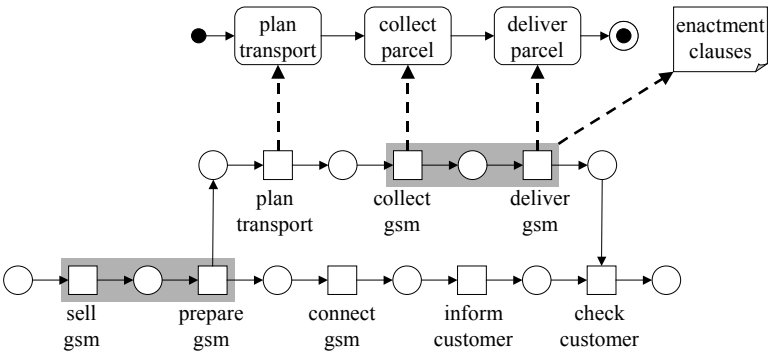

Fig. 4. Example external process (top), projected from conceptual (bottom)

4.4 External Process Specifications in Electronic Contracts

Electronic contracts form the basis for the automated formation of dynamic virtual enterprises. One of the most important aspects of this formation is the alignment of the business processes of the cooperating organizations. For this reason, the external process specification is a key element in electronic contracts in the domain of service outsourcing [3, 15].

On the external level, there are two types of process specifications that are relevant to electronic contracts: establishing contracts between service consumer and provider and enacting contracts based on established contracts. In this paper, we focus on the specification of processes related to contract enactment, i.e., processes for enacting outsourced services. For reasons of completeness, however, we briefly describe both types of processes below.

The first type of process specification describes the establishment of an electronic contract, i.e., the process of negotiation and exchanging preliminary data between two organizations. This process is specified in an electronic offer and subsequently – if necessary – in an electronic contract. The reason to include this process specification in a contract is for example to achieve contract reusability. The established contract can be stored and subsequently, when new business relations between the parties appear, the preliminary process can be repeated in an automated manner and will lead to a new instance of the contract. This kind of process specification can also be reflected at the conceptual and internal specification levels.

The second type of process specification concerns a specification of the outsourced service. This specifies the process execution chronology and components, and supports the cross-organizational process monitoring and control. Specification of these processes at the right level of granularity at the external level is of high importance for the proper execution of the services accompanying contract enactment, e.g., contract monitoring, contract enforcement, etc. The granularity is determined in the design process at the conceptual level, as described in Section 3. The design determines which process specifics can remain hidden from the contractual relation between a service consumer and a provider. These observations for the process specifications embedded in the electronic contract are an underpinning for the need of a conceptual level that will play a mediating role and will allow independence between the internal and external levels.

4.5 Example Contract Model

In Figure 5, the top level of the CrossFlow contract model [15] is shown as an example contract structure. This model consists of five main entity clusters: a concept space that defines concepts relevant in the contract, a workflow schema that describes a service process, enactment clauses that specify how to execute the workflow schema, usage clauses that specify how to establish a contract, and a natural language description of the contract for human interpretation. The concept space is essential for specifying the context of the service outsourcing, e.g., to identify the parties involved in service outsourcing. The various clusters are interlinked to describe relations between them.

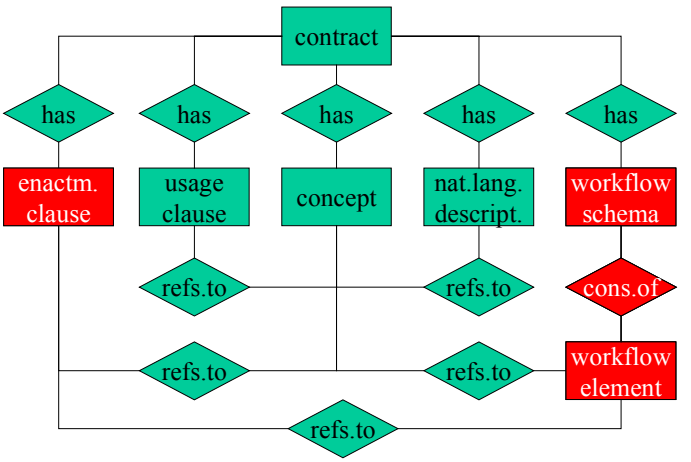

Fig. 5. CrossFlow contract model

The workflow schema and enactment clauses coincide with the external process specification as proposed in this paper. The workflow schema describes the service process in an abstract process specification language, consisting of activities and transitions. This abstract language is easily mappable to UML activity diagrams as used in this paper. Enactment clauses come in different ‘flavors' to describe specific classes of execution characteristics, for example transactional behavior and quality of service parameters.

5 The Internal Level

The internal process specification is used to have local parts of cross-organizational processes enacted by process support systems. Below, we first describe the nature of internal process specifications. Next, we discuss the mapping of specifications from the conceptual level to the internal level. Finally, we show an example internal process specification based on our running example.

5.1 The Nature of Internal Process Specifications

Internal process specifications are used to describe enactment processes by organizations participating in cross-organizational processes.

Workflow management systems are a general infrastructure for the automated support of business processes enactment. Often, these systems are separate entities in an information system infrastructure; sometimes they are embedded in other systems, like ERP systems. Hence, the internal process specification level can be based on a workflow process definition language (WFPDL). WFPDLs are discrete in nature and often use Petri Net based process models, so they match the requirements as discussed at the conceptual level. They usually contain primitives allowing easy specification of various process constructs, for example various forms of process splits and joins. We show an example WFPDL in the sequel of this section.

The internal process specification contains a mapping of the activities in the process to the enactment resources available in the organization. These resources include the employees in the organization and the back-end information systems employed in the business processes. WFPDLs contain primitives that cater for this specification.

In practice, a WFPDL is usually WFMS-specific: each WFMS has its own WFPDL. Most of the languages, however, conform to standard concepts [20] defined by the Workflow Management Coalition [21].

5.2 Mapping the Conceptual Level to the Internal Level

Abstract process models defined at the conceptual level have to be mapped to workflow models in a WFPDL. This requires a syntactical translation, but also the addition of resource specifications as discussed above. Given the global similarity of the natures of process primitives in conceptual specification language and WFPDL, process translation is in most practical cases not problematic. In specific cases, however, the precise semantics of constructs at both levels have to be analyzed to obtain a correct translation. Clearly, quality characteristics identified at the conceptual level have to be taken into account in the translation – such that guarantees specified in contracts at the external level are in fact delivered at the internal level.

In some cases, the process aggregation level available at the internal level is lower than that of the lowest refinement level of the conceptual level. The conceptual level may, for example, contain multi-user tasks, whereas the workflow system employed at the internal level only allows single-user tasks. In this case, an additional refinement step has to take place to go from conceptual to executable internal level (as illustrated in Figure 3 where level I1 is refined into I2). Note that refinement only takes place here if technology limitations dictate this. Refinement from a functional point of view takes place at the conceptual level. We further illustrate this below.

5.3 Example Internal Process Specification

In Figure 6, we show level 1 of the internal process specification in the graphical version of the WFPDL of the FORO workflow management system [11]. The graphical representations of most other WFPDLs are more or less comparable – the reason for

choosing the FORO notation is that it makes some details explicit (as explained below).

We see that the conceptual process specification can be mapped almost directly onto the internal specification. Start nodes, end nodes, splits and joins in the process are, however, modeled explicitly in the FORO FWPDL. Also, the difference between atomic tasks and subprocesses is explicitly modeled (note that tasks ‘connect gsm' and ‘inform customer' are represented by a different symbol than the other tasks; this symbol indicates subprocesses that are exploded at a lower level). For reasons of brevity, we do not illustrate how specifications of resource mappings (roles in the process) are added to the internal process specification.

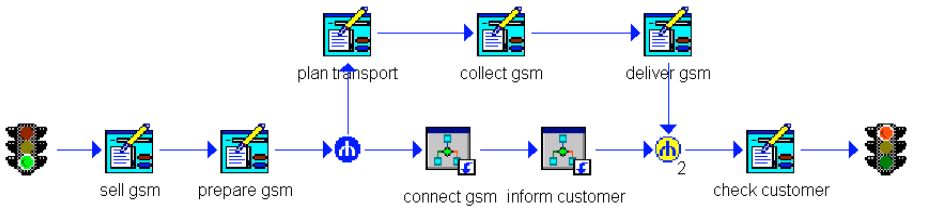

Fig. 6. Example internal process in FORO WFPDL (level 1)

Note that the above example mapping from conceptual to internal specification is rather straightforward. Depending on the functionality of the process enactment platform, the mapping may be more complex. If the process enactment platform would not support nested processes (subprocesses), for example, the nested structure of the conceptual specification as shown in Figure 2 would need to be ‘flattened' into a single level. If the process enactment platform would not allow access to multiple back-end systems in one task, the task ‘sim#+gsm#' at the second conceptual level would have to be split into two tasks ‘get sim#' and ‘enter sim#'. The resulting process specification at the internal level is shown in Figure 7. Note that this level is a refinement of levels 1 and 2 of the conceptual level. This refinement is made at the internal level, as it is driven by platform specifics, not by conceptual process functionality. As such, the distinction between refinement at conceptual and internal process levels caters for both a separation of concerns and portability of conceptual process specifications across implementation platforms.

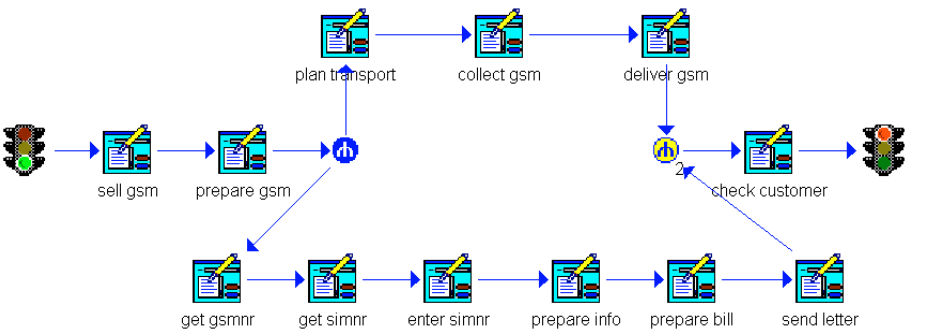

Fig. 7. Example internal process specification for limited platform

6 Process Support Aspects

In this section, we focus on more practical aspects of the proposed three-level modeling approach by placing it in the context of two example process support environments – note that there are more relevant environments. To take the research point of view, we first place the approach in the context of the CrossFlow architecture. To take the industrial standardization point of view, we place the approach in the context of the ebXML architecture. As our approach leads to the existence of multiple interrelated process specifications per process, storage of process specifications is an important practical aspect. Therefore, we end this section by relating storage aspects to our framework.

6.1 CrossFlow Architecture

In the CrossFlow project, concepts and technology for workflow support in dynamic virtual enterprises have been developed [12, 23]. In the context of this project, the formation of virtual enterprises is based on dynamic service outsourcing, as advocated in this paper as well.

Service offerings and service requests are specified in electronic contract templates [15], which are matched by a service matchmaker. An established electronic contract is the basis for the dynamic generation of a service enactment infrastructure [13], based on workflow management technology.

In the CrossFlow approach, external process specification is part of contract specification. For this purpose, the contract model contains a process specification submodel, based on the standard of the Workflow Management Coalition. The process specification submodel is augmented with an extensible set of enactment clauses, which specify service enactment characteristics, as discussed in Section 4.1. Apart from this, the contract model includes usage clauses, describing aspects of establishing contracts – this is a first step towards specification of establishment processes as discussed in Section 4.4. The contract model is reflected in a dedicated, XML-based contract specification language [15].

The internal process specification in CrossFlow is formed by the workflow specification language of the workflow management system, in the case of the CrossFlow prototype IBM MQSeries Workflow [16].

In the CrossFlow approach, no separate conceptual level for process specification is used. An internal enactment specification is used for a direct mapping of external onto internal level [13]. This means that conceptual process design and analysis are not integrated into the CrossFlow approach. This can be illustrated by the support for advanced cross-organizational transaction management in CrossFlow [19]. This functionality is supported both at the external and internal levels, but without conceptual analysis facilities integrated in the model. In complex situations, however, this analysis is desirable to determine the precise semantic effects of transaction management.

6.2 ebXML Architecture

ebXML is an initiative governed by UN/CEFACT and OASIS, joined by major companies and standardizing consortiums, that aims to provide a framework for establishment of business relations and subsequent execution of business transactions [7].

ebXML provides a process modeling technique based on the UN/CEFACT Modeling Methodology (UMM) that utilizes UML. The modeling technique employs UML activity diagrams and sequence diagrams in both a graphical and a textual, XML-based format. In Section 4.3, we have shown an example activity diagram.

The process modeling and specification stage in the ebXML framework relates to the conceptual level of our three-level process model. However, ebXML does not differentiate process specifications at the conceptual level from process specification at the external level. As these two levels are mixed up and there is no explicit attention for refinement of processes in the approach, detailed mappings to internal level process specifications may be hard to obtain.

As a result of process modeling in the ebXML approach, Collaboration Protocol Profiles (CPPs) are created at the service consumer and service provider sides that reflect the process specification at the external level of both parties. The CPP includes specifications of the supported processes, transaction chronology descriptions, etc. and is the contract offer specification described in Section 4.3. On the base of the CPPs of both parties, a Collaboration Protocol Agreement (CPA) is created. The CPA is the electronic contract that specifies the agreed processes that will be performed, their chronology and possible interactions between parties.

The existence of ebXML as a global standard specification allows achieving the interoperability required on the external level. ebXML does not cover in its specification process specifications at the internal level. Other standardizing efforts exist with comparable goals as ebXML, for example WSDL [22] and BizTalk [4].

6.3 Process Specification Storage

The proposed approach to service outsourcing leads to complex process specifications at three levels, with possibly one-to-many mappings between the levels. Consequently, an infrastructure for specification storage and management is needed, paying due attention to consistency requirements between specifications and version management of specifications. Storage of all specifications in central process specification database is an approach to match these requirements. This database will have a semi-structured data format and might be supported by a database management system with an XML front end.

As discussed in this paper, specifications at the external level are usually stated in a language based on XML, e.g., ebXML. Mapping to XML infrastructures is thus trivial at this level. At the conceptual level, this requires a mapping of the used specification technique, for example Petri Nets, into XML. At a number of places, XML representations of Petri nets are currently being developed. An example is the Petri Net Markup Language [17]. At the internal level, an XML representation of a WFPDL is required. The application of XML for workflow management currently finds some application in Wf-XML standard of the Workflow Management Coalition [21], although this language does not cover a complete WFPDL. Commercial workflow

management systems, e.g. IBM's MQSeries Workflow [16], are being developed towards XML support.

7 Conclusions

The use of contract-based dynamic service outsourcing opens ways to efficient handling of fine-grained cross-organizational processes. Efficient means that contractual outsourcing can take place in a fast and cheap way – quite different from the traditional paper-based situation. This paradigm allows the creation and dismantling of short-term virtual enterprises, thereby supporting highly dynamic cooperation in fast-changing markets. This can even enable completely new business models in situations where formal service outsourcing is required but traditional ways to do so are too costly or slow.

The proposed three-level approach to business process specification provides a clear separation of concerns in business process support, thereby increasing quality, flexibility and reusability of process specifications in cross-organizational settings. Having a conceptual process specification as a center point and external and internal specifications as derivatives, provides a separation of concern in process design on the one hand and portability over both market standards and implementation platforms on the other hand. The separation of concerns is becoming increasingly important, as the complexity of automated cross-organizational processes grows through the advent of e-business and e-commerce. Portability in both dimensions is required to deal with fast changes in this domain, both with respect to business and technical aspects.

We have shown how Petri Nets can be used as an adequate specification technique for the conceptual process specification level. In this paper, we have illustrated the external and internal specification levels by means of UML activity diagrams in the context of ebXML respectively a specific workflow process definition language.

A spectrum of work is being performed at the University of Twente that is related to the approach presented in this paper. Support for e-contract handling is being researched, using the three-level framework for contract specification (both data and process aspects). Research into flexible architectures for cross-organizational process support is being performed, both on the conceptual and technical architecture levels. Models and architecture for transactional cross-organizational workflow processes are a further topic of research.

Acknowledgements

All former CrossFlow project members are acknowledged for their role in the CrossFlow project, on which part of the ideas presented in this paper are based. Juliane Dehnert is thanked for her feedback on the draft version of this paper.

References

[1] W. van der Aalst; *The Application of Petri Nets to Workflow Management*; Journal of Circuits, Systems and Computers, Vol. 8, No. 1, 1998; pp. 21-66.

[2] N. Adam, V. Atluri, W. Huang; *Modeling and Analysis of Workflows using Petri Nets*; Journal of Intelligent Information Systems, Vol. 10. No. 2, 1998; pp. 131-158.

[3] S. Angelov, P. Grefen; *B2B eContract Handling - A Survey of Projects, Papers and Standards*; CTIT Technical Report 01-21; University of Twente, 2001.

[4] *BizTalk Framework*; http://www.biztalk.org.

[5] W. Derks, J. Dehnert, P. Grefen, W. Jonker; *Customized Atomicity Specification for Transactional Worflows*; Procs. 3rd Int. Symp. on Cooperative Database Systems for Advanced Applications; Beijing, China, 2001; pp. 155-164.

[6] M. Duitshof; *Logistics Prototype Deployment Report*; CrossFlow Project Deliverable D13; KPN Research, The Netherlands, 2000 (available via http://www.crossflow.org).

[7] ebXML Technical Architecture Specification v1.0.4; http://www.ebxml.org.

[8] R. Elmasri, S.B. Navathe; *Fundamentals of Database Systems*; Benjamin/Cummings, 1994.

[9] R. Eshuis, R. Wieringa; *A Formal Semantics for UML Activity Diagrams - Formalising Workflow Models*; CTIT Technical Report 01-04; University of Twente, 2001.

[10] R. Eshuis, R. Wieringa; *A Comparison of Petri Net and Activity Diagram Variants*; Procs. 2nd Int. Coll. on Petri Net Technologies for Modelling Communication Based Systems; Berlin, 2001; pp. 93-104.

[11] *FORO Workflow Management System*; http://dis.sema.es/projects/FORO/foro.html; Sema Group, Spain.

[12] P. Grefen, K. Aberer, Y. Hoffner, H. Ludwig; *CrossFlow: Cross-Organizational Workflow Management in Dynamic Virtual Enterprises*; Int. Journ. of Computer Systems Science & Engineering, Vol. 15, No. 5, 2000; pp. 277-290.

[13] Y. Hoffner, H. Ludwig, C. Gülcü, P. Grefen; *Architecture for Cross-Organisational Business Processes*; Procs. 2nd Int. Worksh. on Advanced Issues of E-Commerce and Web-Based Information Systems; Milpitas, CA, USA, 2000; pp. 2-11.

[14] Y. Hoffner, S. Field, P. Grefen, H. Ludwig; *Contract Driven Creation and Operation of Virtual Enterprises*; Computer Networks, Vol. 37, No. 2; Elsevier, 2001; pp. 111-136.

[15] M. Koetsier, P. Grefen, J. Vonk; *Contracts for Cross-Organizational Workflow Management*; Procs. 1st Int. Conf. on Electronic Commerce and Web Technologies; London, UK, 2000; pp. 110-121.

[16] *IBM MQSeries Workflow*; http://www-4.ibm.com/software/ts/mqseries/workflow/; IBM.

[17] *Petri Net Markup Language*; http://www.informatik.hu-berlin.de/top/pnml/.

[18] D. Tsichritzis, A. Klug; *The ANSI/X3/SPARC DBMS Framework*; AFIPS Press, 1978.

[19] J. Vonk, W. Derks, P. Grefen, M. Koetsier; *Cross-Organizational Transaction Support for Virtual Enterprises*; Procs. 5th Int. Conf. on Cooperative Information Systems; Eilat, Israel, 2000; pp. 323-334.

[20] Workflow Management Coalition; Workflow Standard Interface 1: Process Definition Interchange Process Model; Doc. Nr. WfMC TC-1016-P; 1998.

[21] Workflow Management Coalition; http://www.wfmc.org.

[22] W3C; *Web Services Description Language (WSDL) 1.1*; http://www.w3.org/TR/wsdl.

[23] *CrossFlow Project*; http://www.crossflow.org; CrossFlow Consortium.