Value-driven coordination process design using physical delivery models*

Roel Wieringa[1], Vincent Pijpers[2], Lianne Bodenstaff[1], Jaap Gordijn[2]

[1] University of Twente, Enschede, The Netherlands,
{roelw | bodenstaffl}@ewi.utwente.nl
[2] Free University. Amsterdam, The Netherlands,
{pijpersv | gordijn}@few.vu.nl

Abstract. Current e-business technology enables the execution of increasingly complex coordination processes that link IT services of different companies. Successful design of cross-organizational coordination processes requires the mutual alignment of the coordination process with a commercial business case. There is however a large conceptual gap between a commercial business case and a coordination process. The business case is stated in terms of commercial transactions, but the coordination process consists of sequences, choices and iterations of actions of people and machines that are absent from a business case model; also, the cardinality of the connections and the frequency and duration of activities are different in both models. This paper proposes a coordination process design method that focusses on the the shared physical world underlying the business case and coordination process. In this physical world, physical deliveries take place that realize commercial transactions and that must be coordinated by a coordination process. Physical delivery models allow us to identify the relevant cardinality, frequency and duration properties so that we can design the coordination process to respect these properties. In the case studies we have done so far, a physical delivery model is the greatest common denominator that we needed to verify consistency between a business case and a coordination process model.

1 Introduction

Current e-business technology enables the execution of increasingly complex cross-organizational business processes that link IT services provided by different companies. The complexity of these networks makes it important to make a business case that shows for each partner that it is economically sustainable to participate in the network, and to make this case before operational details of the coordination infrastructure are designed. A coordination process need be designed only if the business case is positive for each of the partners. But it is hard to design a coordination process based on a multi-party business case only, because the conceptual gap between the two is very large [1]. Where a

* Research supported by NWO project 638.003.407

business case makes an estimation of numbers of commercial transactions and their monetary value over a period of time, a coordination process consist of operational activities performed at particular times satisfying specific cardinality, frequency and duration constraints. But despite this conceptual gap, the coordination process must be designed to be consistent with the business case, and this alignment must be maintained during the entire period of cooperation between the businesses.

Recent proposals to design coordination processes based on business case models [2,3,4] recommend that the process designer analyze a business case model to identify transfers of ownership before designing the coordination activities needed to implement a business case model. The reasoning is that the coordination process must realize the ownership transfers involved in commercial transactions, and that it therefore pays off to analyze ownership transfer first. However, the concept of ownership is very complex. Asking a process designer to analyze it before designing a coordination process does not simplify the process design task. Moreover, important process design information such as cardinality, frequency and duration properties of the coordination process are not uncovered this way. In this paper we take a less complex route, that nevertheless yields more information for the process designer. We start from the observation that the business case and coordination models are views on one shared physical world, where physical deliveries take place (section 5). Each commercial transaction is realized in physical deliveries, and the coordination process must coordinate these deliveries. Physical delivery modeling makes it easier to think about cardinality, frequency and duration properties of deliveries, and this in turn makes it relatively simple to design coordination processes that realize these deliveries (section 6). Our case studies provide support to the hypothesis that physical delivery models provide a shared semantic structure for proving mutual consistency between business case and coordination models (section 7).

In section 2 we describe our running example and in sections 3 and 6 we present a business case model and coordination model of this example, respectively. There are several notations for business case modeling, such as the e^3-*value* notation [5] used in this paper, REA [6] and BMO [7]. Our argument does not depend on the notation used as long as the business case model contains estimations of the value and number of commercial transactions between the business partners over a period of time. We explain the e^3-*value* notation in section 3. Well-known notations for coordination modeling are Petri Nets [8], BPMN [9] and UML activity diagrams. In this paper we use UML activity diagrams, because we want to show who is waiting for which activity, and which objects are passed around.

2 Running Example

Our running example concerns electricity distribution in the Netherlands: Electricity suppliers provide electricity to consumers by obtaining it from producers and having a distributor deliver it at the consumer's home. Consumers pay for electricity as well as for the use of the distribution network, and see this in their bill, where distribution is charged explicitly. However, they do all payments to the supplier, which then forwards payment for the distribution to the distributor. In the Netherlands there is one electricity distribution network per geographical region, but there are several suppliers and producers the consumer can choose from.

3 Value modeling

An e^3-*value* model of a networked business case consists of a diagram, called a *value model,* that represents the businesses participating in the network for a period of time, what they exchange of value in this time, plus a set of estimations of the number and value of the exchanges, that allows us to calculate the net present value (NPV) (see e.g. [10] for an explanation of the NPV concept) of the revenue generated by these exchanges for each business. We call the time period represented by the value model the *time extent* of the model, or *extent* for short. The value model allows NPV estimations of revenue for the time extent of the model only. It is possible to consider a sequence of value models over subsequent time extents, where for example the first model represents initial investments and start-up and the second model represents exploitation of this initial investment.

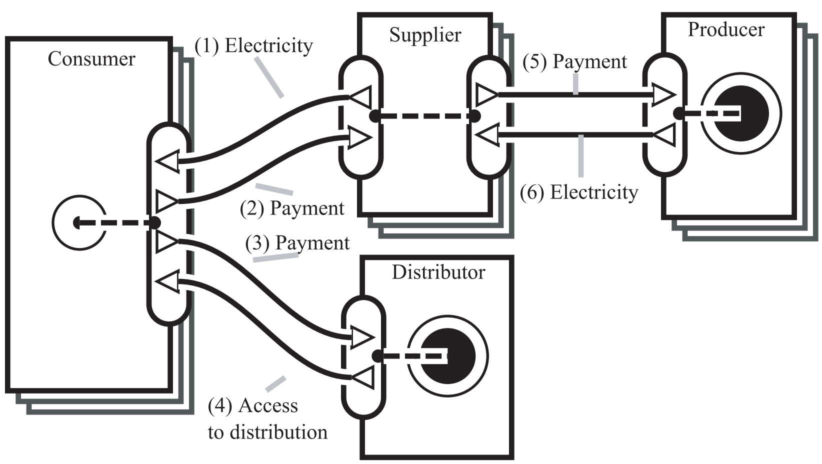

Fig. 1. Value model for electricity delivery. The value transfers have been numbered for ease of reference.

Figure 1 shows a value model of electricity distribution. Rectangles represent *economic actors* that can have needs and can offer something in value in return, such as money, to meet those needs. Actors in a value model can be businesses or consumers. Stacked rectangles represent *market segments,* which are sets of economic actors with the same needs.

Actors are connected by *value transfers.* If the source or destination of a value transfer is a market segment, this means that all actors in this segment can produce or consume this transfer. Value transfers are transfers of so-called *value objects.* A value object can be a good, service, data, experience, or anything else of value to the receiver. What are value objects depends on the specific actors in case. Music may be of value to one actor and be a nuisance to another. The arrow may even be reversed in different value models involving the same value object. For example, a hotel may offer a wireless access provider the opportunity to provide wireless access to its guests—a captive market, which is of value to the access provider, who therefore pays the hotel for this. However, another hotel and another access provider, or the same two actors in another time period, may view this differently: The access provider takes care of a service regarded by all guests as normal infrastructure, and therefore the hotel pays the provider for this. What, in a given time extent, is of value to whom, is depends on the needs and desires of those actors in that time period.

Value transfers enter or leave an actor through a *value interface,* which is a unit of atomicity: In the time extent of the business case, when one transfer occurs, the other occurs too in that same period. e^3*-value* does not allow us to say *when* these transfers occur, only *that* they occur in the time extent of the business case. This is sufficient for the purpose of business case modeling, where we only need to estimate how often commercial transactions occur in the time extent of a business case, not when they occur.

Actors can have needs, represented by a dot placed inside the actor rectangle. An actor can have this need any number of times in the time extent of the model. When placed inside a market segment, the dot means that any actor in this segment can have this need. The commercial transactions required to meet a consumer need are linked by a dashed line called a *dependency path* to the consumer need. A bull's eye indicates the model boundary, i.e. further transactions such as those with suppliers down the value chain are not considered in this business case. This means that the business case developer does not view these transactions as having any impact on this business case. In general, a dependency path is an acyclic graph with and-or nodes. We will illustrate and-nodes in the health insurance example.

We believe that each conceptual model should have a single purpose and should contain all and only the information needed to fulfill this purpose. The purpose of an e^3*-value* model is to make a business case, and the graphical value model of figure 1 and the supporting computational techniques (not treated here) are exactly what is needed to do the NPV estimation of the revenue generated for each of the business actors in the case. The value model shows what is happening commercially, but does not tell us how this is done operationally. We use coordination models for that.

4 Cross-organizational case coordination

In this section we use a variation of activity diagrams that retains the root of these diagrams in statecharts [11]. As pointed out earlier, we choose this statechart-like variation of activity diagrams because it makes explicit when an actor is performing activity and when it is waiting for someone else to do something. Figure 2 contains the legend.

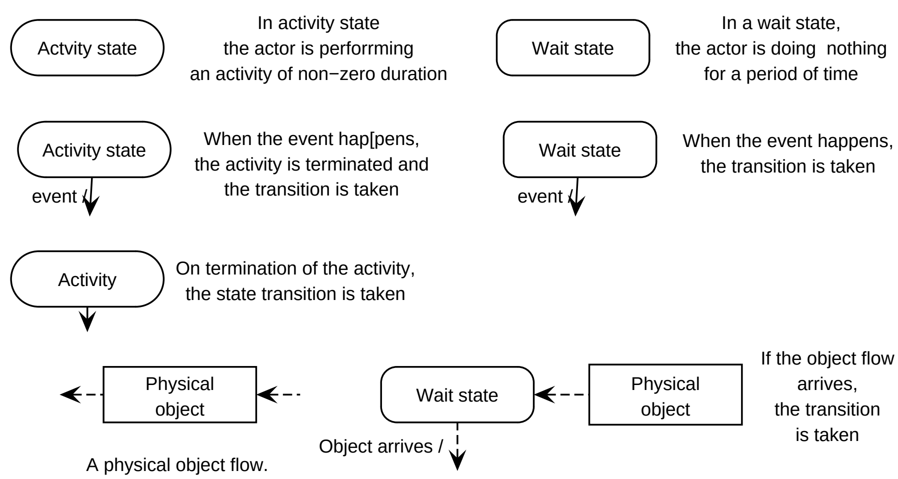

Fig. 2. Legend for the version of activity diagrams used in this paper.

Figure 3 shows a coordination process involving four named actors, a consumer c, supplier s, producer p and distributor d (in the lower half of the diagram). The process runs over a period of a year, i.e. the time extent of our business case. We indicated the parts of the coordination process that correspond to commercial transactions by dashed rounded rectangles. Payments 2 and 3 involve an monthly interaction between supplier and consumer and a further yearly interaction (one in this example) between the supplier and distributor. Payment 5 consists of a quarterly interaction between supplier and distributor. Delivery of electricity (value transfer (1)) and access to the distribution network (service provision(6)) are started in parallel to the payment procedures and stop at the end of the contract period (1 year). The case handled in this coordination process is the consumer need for electricity during one year.

The coordination model has no owner. No single actor is responsible for its execution. It is the joint responsibility of all actors that it is executed, and if each actor does what promised to do, then the coordination process occurs. The process in figure 3 makes the assumption that all actors are trustworthy, i.e. do what they should do. This is a simplifying assumption that we return to later.

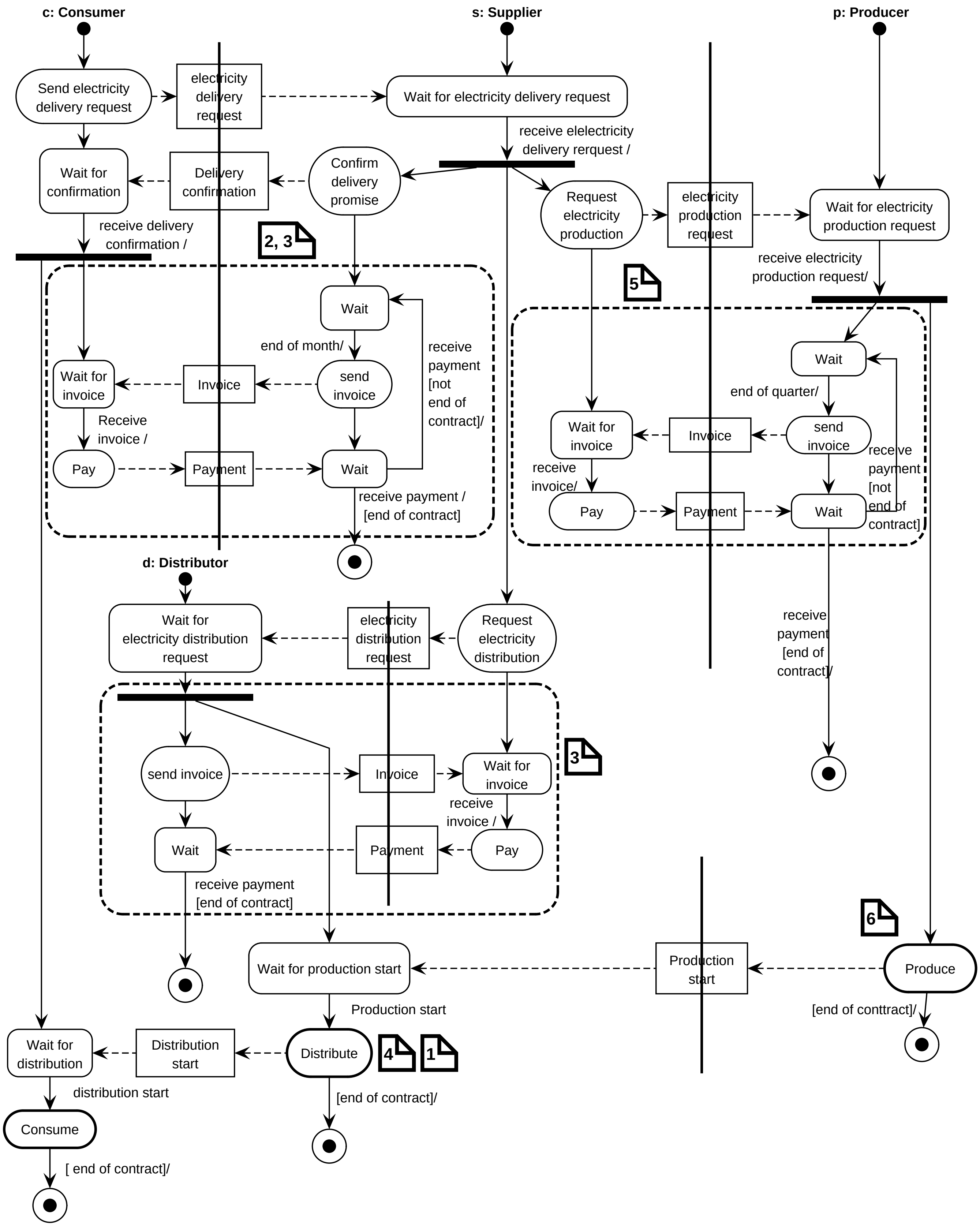

Fig. 3. Coordination process for the electricity delivery scenario of figure 4. The comment signs refer to value transfers.

First, we turn to the central topic of this paper, which is how we can design a coordination process starting from a business case model.

5 Physical delivery modeling

We claim that coordination modeling is facilitated by making a physical delivery model first, because the business case and coordination model are both views of a network of physical deliveries. Ultimately, every business case describes an economic view on the physical world, and every process model describes events in the physical world. Software is a state of hardware, and data is a state of the physical world too. Money is physical too, realized by means paper, metal, or computer hardware as digital money. Human services such as coaching, teaching, consultancy, providing help through a help desk, providing financial services, etc. consist of physical processes too, including people moving to places, advice being produced in the form of sounds, reports being written, and money being passed around or stored. Each of these processes has a physical realization, even though we usually talk about them as disembodied entities with certain financial, logical or semantic properties. But if there would be no physical events, none of these disembodied entities and events would exist.

Commercial transactions are usually accompanied by a system of legal rights and obligations, permissions and prohibitions that are treated by stakeholders as non-physical. However, creation of these norms must be done by a physical process (e.g. signing a document); we maintain physical evidence of the existence of these norms in the form of physical documents; and the norms govern physically observable behavior of people and goods in the real world, such as buying and selling, listening to music or watching a movie. The non-physicality of these legal norms is a useful fiction that allows us to abstract from complex physical processes.

Views such as a value model or a coordination process model represent part of the shared conceptual models by which people organize themselves. But to understand the relation between two models that are so fundamentally different, we must understand what each of these models means in terms of a shared physical world. We do this by means of physical delivery models, that will allow us to understand which physical activities realize commercial transactions and therefore need to be coordinated by a coordination process.

5.1 Physical objects and their delivery

To help identifying the physical deliveries that realize commercial transactions, we distinguish the following kinds of physical objects.

- *Discrete goods* are identifyable entities that can form a set. Examples are physical products such as chairs and cars, but also inert physical entities such as houses and airport runways. The distinguishing feature is that when two physical goods are put together, they form a set of two elements.

- *Cumulative goods* do not preserve their distinctness when put together. When you add water to water, the result is still water. Ultimately, cumulative goods consist of atoms that form a set, but at the level of phenomena that interests us for physical delivery modeling, we will be talking about cumulative goods such as water or electricity. Where discrete goods can be measured by counting them, cumulative goods must be measured by choosing a measurement unit.

The distinction is not a metaphysical statement about the ultimate structure of the physical world but a hint what to look for when modeling physical deliveries. It is not important to allocate a physical object to exactly one of these two categories: What is important is to identify the physical objects to be coordinated in realizing commercial transactions. Cumulative goods can always be parceled into discrete units. For example, teaching is cumulative, but we usually parcel it into numbered lessons, which can be viewed as discrete goods.

Two special cases that are important in e-business are sense experiences and lexical goods. *Sense experiences* are a particular kind of cumulative goods, namely those that stimulate our senses, such as sound, sight, smell etc. For example, on-line music provides a sense experience to a user. *Lexical objects* are a particular kind of discrete goods, that have a meaning for stakeholders. Lexical objects are physical information carriers such as paper documents, states of a computer (digital documents), traffic signs, money coins, paper money, digital money, etc.

5.2 Delivery scenarios

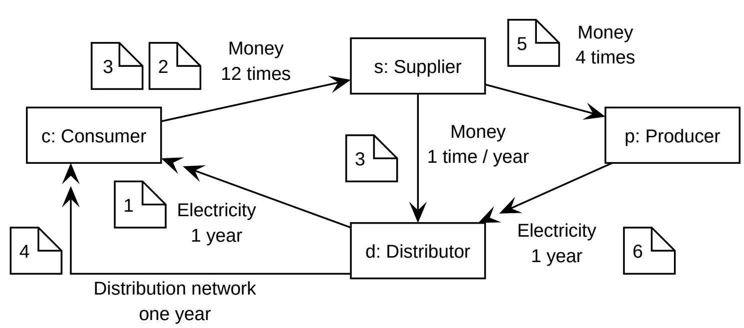

Fig. 4. Physical delivery scenario of an electricity consumer need. The numbers inside comment boxes refer to value exchanges in figure 1.

Figure 4 shows a delivery model for the business case of figure 1. Rectangles represent actors, which must be the same actors as those of the value model. We treat actor names as type names, and individuals can be represented by declaring proper names for them. So in figure 4 we represent four individuals named c, s, d and p. If there would have been, say, two producers p1 and p2, then we could have represented these as two distinct rectangles in the diagram, labeled by p1 and p2. And if we would have wanted to represent a set of producers of arbitrary size, then we could have represented the Producer type as a rectangle without a proper name for any individual producer.

Figure 4 actually represents one delivery *scenario*. It shows what deliveries must occur to satisfy the need of consumer c in the time extent of the business case. This is the delivery scenario corresponding to the coordination scenario of figure 3, which shows how this consumer need is handled.

Another consumer could receive electricity from two producers, one producing "green" electricity and one producing "dirty" electricity, which would represented in that delivery scenario by means of two producer boxes. Alternatively, we could take the point of view of a particular supplier and show the network of deliveries that this supplier is involved in. In the Netherlands, this supplier would interact with many consumers and producers and with one distributor. Each of these scenarios has important consequences for the internal business processes performed by the actors.

In our delivery scenarios we will always take the point of one consumer, and more in particular of one consumer need, because the business network exists to satisfy this need. Our value and coordination models show the commercial and operational feasibility of meeting this need. The delivery scenario of a consumer need shows the deliveries in the network required to satisfy that need.

The delivery scenario abstracts from much of the infrastructure used to realize the business case. For example, banks used to make payments are not represented in figure 4. The reason is that a delivery model contains the same actors as a business case model; and the business case of this example ignores the payment infrastructure because the cost of using it does not impact the business case.

5.3 Types of deliveries

Deliveries are represented by arrows pointing from provider to receiver. A physical delivery starts at the point in time when the receiver is able to physically handle the delivered object, which we will call *access*. This can happen in two ways: The physical object moves to the receiver, or the receiver moves to the physical object. The money and electricity transfers in figure 4 are of the first kind, because these are transported to the receiver. Delivery of the distribution network is of the second kind. The consumer probably obtained access to the electricity network because he/she moved into a house that is connected to the network. Connection of a network to a house and movement of the consumer into this house is out of scope of the business case and therefore out of scope of physical delivery modeling.

We distinguish *time-continuous* from *time-discrete* delivery, represented by double-headed and single-headed arrows, respectively. Time-continuous delivery takes place over a period of time and time-discrete delivery takes place at one instant of time of zero duration. All deliveries stop at some time, because our value models, and therefore their corresponding delivery models, have a finite time extent. But time-continuous deliveries stop some time after they are started, whereas time-discrete deliveries stop at the point in time when they are started. Labeling deliveries as time-continuous or time-discrete is choosing a granularity of time, because what is taken to be time-discrete at one level of granularity is a period at a lower level. The distinction is important because coordination for these two kinds of deliveries must guarantee different timing properties.

Deliveries are named after the physical entity that the receiver receives access to. This still leaves the modeler a choice what the physical entity received actually is: energy, electrons, atoms, metal, paper, coins, bank notes, money, a payment? All of these can be regarded as names of physical entities, albeit increasingly abstract names. Choosing an abstraction level is unavoidable in any modeling activity, but this is not an arbitrary choice. To choose a name, we first rule out names of which the meaning depends on the value or process models, because the delivery model should only represent basic information that is needed to understand these model. If we would now include in the delivery models information that can only be understood by first understanding a value- or process model, we would have introduced a vicious circularity. This rules out "payment" as a delivery name, for this is defined in terms of a commercial transaction in the value model.

Second, among the remaining possibilities we choose an abstraction that tells us what stakeholders want to be able to *observe* in any realization of the business case model. In all cases, energy, electrons and atoms are being delivered, but this is not what stakeholders want to observe. The supplier wants to observe a delivery of money from the consumer, and so this is the name of the delivery. In another case, such as one in which coin collectors are described, we could have selected "coins" as name for a delivery because this is what the collector wants to observe. They want to observe this because this is the physical event that they are willing to provide a reciprocal value object for, as indicated in the value model.

Because receivers are paying (in the form of some value object sacrificed by the receiver) for the delivery, they usually not only want to observe but also to measure the delivery. In the electricity example, the amounts of money transferred and of electricity delivered are what stakeholders want to measure when the value model is realized. They want to measure these phenomena because this is what the business case rests on; Participants can verify whether the business case is satisfied by observing these deliveries. The commercial transactions counted in the value model have a particular value that can then be observed at the delivery points in a delivery model, i.e. at the points where deliveries enter the receiver.

So the arrows in a delivery model represent observations, and possibly measurements, that stakeholders want to make in physical reality. Figure 4 tells us that in the time extent of the model, they expect to make 12 observations of money transfer from consumer c to supplier s, 1 year of electricity delivery of the distributor to the consumer, etc. The physical network is actually fitted with measurement instruments (electricity meters, information systems) that make and record these observations.

5.4 Frequency and duration properties

For one scenario, we can represent *frequency and duration* information. For time-discrete delivery we can state *how many times* delivery occurs in the time-extent of the business case, and for time-continuous delivery we can state *how long* a delivery takes place and *how many times* this takes place. Frequency and duration information provides us with an important guide for process design later. For example, the delivery scenario of figure 4 shows that the supplier collects consumer payments made monthly, and forwards them to the distributor 4 times a year. This requires therefore the distributor to maintain stores (databases) of information about payments and to execute a distributor process which processes batches of consumer payments.

5.5 Cardinality properties

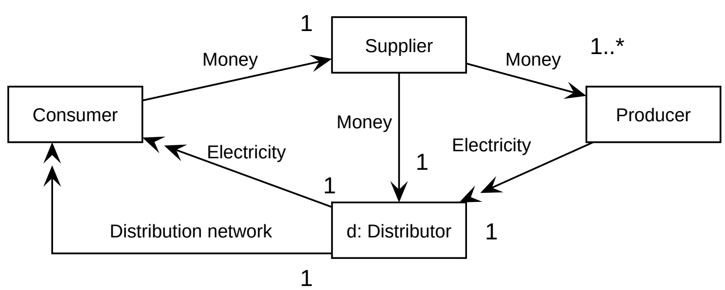

Fig. 5. Model of delivery cardinalities. See text for the semantics of cardinalities.

Instead of modeling a scenario from the point of view of one actor we can represent cardinality properties of deliveries in a viewpoint-independent model. Figure 5 shows a delivery model that does *not* take the point of view of any actor. All nodes represent types, except the distributor node because in our example there exists only one distributor. The diagram represents cardinality properties in the same way as is done in ER models [11]. The cardinalities in the diagram have the following meaning;

- Each consumer can transfer money to exactly one supplier. (Different consumers may transfer money to different suppliers.) Each supplier may receive money from any number of consumers.

- Each supplier can transfer money to at least one producer and each producer can receive money from any number of suppliers.
- Each consumer can receive electricity from exactly one distributor, each supplier can transfer money to exactly one distributor, and each producer supplies electricity to exactly one distributor. This is the meaning of the cardinality 1 at the distributor side of the connections to Distributor.
- Because we have named the distributor d, the model also says that all consumers receive electricity from the same distributor, all suppliers transfer money to the same distributor, that this is the same distributor that all consumers receive electricity from, etc.

5.6 Semantic relation between value–, delivery– and coordination models

The physical delivery models provide a physical meaning to the commercial transactions identified in a value model and the activities represented in a coordination model. This is an informal meaning relation that the analyst can nevertheless make as precise as needed. The physical meanings of the commercial transactions in the electricity value model are listed in figure 6. These definitions do not follow from the models. Rather, the models allow the analyst to provide these definitions and use them to achieve a shared understanding with all stakeholders.

- In commercial transaction (1, 2) of figure 1 the supplier ensures that the consumer gets electricity and receives payment for this from the consumer. As shown in figures 3, 4 and 5, this is physically realized by the consumer transfering money to the supplier 12 times in the business case time extent (one year), and by the distributor delivering electricity to the consumer during this period.
- In commercial transaction (3, 4), the distributor makes available its distribution infrastructure to the consumer and the consumer pays for this. This is a time-continuous delivery of a discrete good, the distribution network. Payment by the consumer goes in two deliveries via the supplier. The supplier sends money to the distributor once a year.
- In commercial transaction (5, 6) the producer sells electricity to the supplier against a payment. Since the supplier only buys the electricity in order to provide it to the consumer, the producer delivers the electricity to the distributor, which can then pass it on to the consumer.

Fig. 6. Informal semantics of commercial transactions in terms of physical delivery models.

6 Coordination modeling

A coordination scenario is designed by first following the dependency path of a consumer need in the value model, and modeling the delivery scenario required to realize these transactions, including frequency and duration constraints. This in turn is then used to create a coordination scenario such as shown in figure 3, taking into account the meaning of each value transfer in terms of physical deliveries. The coordination model scenario gives more detailed, operational information than the delivery scenario and shows that commercial transactions occur according to well-known patterns, such as delivery of a service after a request for service, and payment after reception of an invoice. The coordination scenario operationalizes deliveries in terms of a number of objects that cross the boundaries between actors in figure 3, that realize the deliveries of the delivery scenario of figure 4. The coordination process satisfies the frequency and duration requirements of the delivery scenario have been satisfied by loops (for payment in installments) and by start and stop actions (for time-continuous delivery).

As pointed out before, a coordination process is not owned by any actor; it is a joint responsibility to execute it, and it is actually executed by each actor performing its own business processes. Moreover, other scenarios may be needed to serve other consumers. A consumer who wants electricity from two producers (a "green" and a "dirty" one) will have a different delivery scenario and therefore a different coordination scenario, etc. Because these coordination scenarios are performed by every business actor performing its own business processes, the coordination scenarios are actually coordination *requirements* on these internal business processes. And the viewpoint-independent cardinality model of figure 5 imposes additional requirements, concerning the number of partners that an actor should be able to interface to. We summarize the coordination requirements requirements for each actor in figure 7, where the frequency and duration requirements of the delivery scenario are stated as examples. Note that the mismatches in cardinality of received and provided deliveries indicate a requirement to maintain buffers and do batch processing, for example for the supplier.

7 Discussion and further work

To summarize, we start from a situation where a business case has been made that is positive for each partner, for example in the form of an e^3*-value* model. Before operationalizing this in terms of coordination scenarios, we propose modeling delivery scenarios first. We have given the following guidelines for delivery modeling:

- Identify discrete and cumulative physical objects delivered at business actors. Some of these objects may be experiences or lexical objects.
- Delivery is the provision of access and this may involve the physical object moving to the receiver or the receiver moving to the delivered object.
- Distinguish time-continuous from time-discrete deliveries.

- Each consumer needs to be able to pay one supplier (not necessarily the same supplier for all consumers) by transferring money 12 times. Scenario example: Consumer c must pay supplier s in 12 installments per year. Each consumer must be be able to access the distribution network and receive electricity from distributor d for a year.
- Each supplier must be able to receive money from any number of consumers as payment for electricity delivered and for the access to the distribution network, and forward the distribution fee to the distributor d. Scenario example: supplier s must be able to receive money payments from consumer c 12 times a year and forward the fee due to d once a year. Each supplier must also be able to pay for electricity bought from one or more producers. For example supplier s must pay producer p 4 times a year.
- Each producer must be able to receive money payments for electricity from any number of suppliers. Scenario example: Producer p must be able to receive payment for electricity 4 times a year. Each producer must also be able to provide electricity to distributor d. For example, producer p must be able to provide electricity to d for 1 year.
- The distributor must be able to supply electricity to any number of consumers through its distribution network, provided by any number of producers. It must be able to receive payments from suppliers for electricity delivered. Scenario example: It must be able to receive a yearly payment from supplier s for electricity delivered to a client of s.

Fig. 7. Coordination requirements for the business actors.

- Consider physical objects of which the names can be understood without having to understand the value– and coordination models first.
- Taking the view of one particular consumer need, model the delivery scenario for this need and identify duration and frequency of each delivery in this scenario.
- Consider deliveries that participants want to be able to observe if they want to verify that the business case is satisfied.
- Taking a type-level view not associated to any particular actor, identify cardinality properties of deliveries.

In general, delivery and coordination modeling may reveal services that parties were not aware of. In our example, the delivery models show that the supplier is handling payments for the distributor and may require the distributor to pay for this service. This would lead to a change in the business case, hence in the e^3-*value* model, and hence in the delivery and coordination scenarios. We have illustrated this earlier [16].

We have shown that delivery modeling can be used in the electricity delivery example, but can it be used in other examples? So far, we have done three other cases with delivery modeling: a health care insurance case, international trade with a bill of lading, and handling landing and docking at an international airport. These cases have a very diverse mix of commercial services and products exchanged among partners, and in all cases delivery modeling simplified coordination process design. Further case studies must indicate the limitations on applicability of delivery modeling.

There is some resemblance of delivery models to Jackson's problem frames [12]. Both kinds of models structure the world into domains (business actors in delivery models) that share phenomena at their interfaces (physically observable deliveries in delivery models). However, where problem frames are used to show how a machine inserted in a domain causes satisfaction of requirements, delivery models are used to show what observable deliveries must occur to satisfy a business case.

Zlatev, Wombacher [13] and Bodenstaff [14] defined reduced models as semantic structure in which to interpret e^3-*value* models and coordination models to prove that they are consistent. In the cases we have done so far, their reduced models turned out to be our physical delivery scenario models, without the cardinality, frequency and duration constraints. We think the reason for this is that reduced models represent the information shared by business case and coordination models, and that this shared information is exactly the physical deliveries that realize the value model and that must be coordinated by the coordination process. Further work is needed to show conclusively that reduced models are physical delivery models.

In all our models we assume that all actors will do what they promise to do, e.g. will perform their business processes according to coordination requirements. In practice this is not a realistic assumption [15,16]. Actors may be unreliable because they intend to get a free ride from others, or because they lack the resources to live up to their promises. Either way, we must assess the risk of actors not satisfying the coordination requirements. The less trust we have in an actor, the higher we will assess the risk of doing business with this actor. For every trust assumption that we drop, we may have to add new business actors (e.g. trusted third parties) to the business case, or new activities (e.g. control mechanisms) to the coordination process, that will reduce the risk of the cooperation [17]. In our future research we will investigate what role delivery modeling can play in assessing trust assumptions and designing risk mitigation mechanisms.

References

1. Gordijn, J., Akkermans, J., Vliet, J.v.: Business modelling is not process modelling. In: Conceptual Modeling for E-Business and the Web, Springer (2000) 40–51 LNCS 1921.
2. Andersson, B., Bergholz, M., Grégoire, B., Johannesson, P., Schmitt, M., Zdravkovic, J.: From business to process models —a chaining methodology. In Latour, T., Petit, M., eds.: Proceedings of the CAiSE'06 Workshops and Doctoral Consortium, Namur University Press (2006) 211–218
3. Pijpers, V., Gordijn, J.: Bridging business value models and process models in aviation value webs via possession rights. In: 40th Hawaii International International Conference on Systems Science (HICSS-40 2007). (2007) 175

4. Weigand, H., Johannesson, P., Andersson, B., Bergholtz, M., Edirisuriya, A., Ilayperuma, T.: Value modeling and the transformation from value model to process model. In Doumeingts, G., Muller, J., Morel, G., Vallespir, B., eds.: Enterprise Interoperability: New Challenges and Approaches. Springer (2007) 1–10
5. Gordijn, J., Akkermans, H.: Value-based requirements engineering: exploring innovative e-commerce ideas. Requirements Engineering **8** (2003) 114–134
6. Geerts, G., McCarthy, W.E.: An accounting object infrastructure for knowledge-based enterprise models. IEEE Intelligent Systems and Their Applications (1999) 89–94
7. Osterwalder, A.: The Business Model Ontology - a proposition in a design science approach. PhD thesis, University of Lausanne, Lausanne, Switzerland (2004)
8. Jensen, K.: Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use. Springer (1997) Three Volumes.
9. OMG: Business Process Modeling Notation (BPMN) Specification (2006) www.bpmn.org.
10. Horngren, C.T., Foster, G.: Cost Accounting: A Managerial Emphasis, sixth edition. Prentice-Hall, Englewood Cliffs, NJ (1987)
11. Wieringa, R.: Design Methods for Reactive Systems: Yourdon, Statemate and the UML. Morgan Kaufmann (2003)
12. Jackson, M.: Problem Frames: Analysing and Structuring Software Development Problems. Addison-Wesley (2000)
13. Zlatev, Z., Wombacher, A.: Consistency between e3-value models and activity diagrams in a multi-perspective development method. In: CoopIS, DOA, and ODBASE: OTM Confederated International Conferences: On the Move to Meaningful Internet Systems, Springer (2005) LNCS 3760.
14. Bodenstaff, L., Wombacher, A., Reichert, M.U., Wieringa, R.J.: Monitoring collaboration from a value perspective. In Chang, E., Hussain, F.K., eds.: 2007 Inaugural IEEE International Conference on Digital Ecosystems and Technologies, Cairns, Australia. Volume 1., IEEE Computer Society Press (2007) 134–140
15. Bergholtz, M., Jayaweera, P., Johannesson, P., Wohed, P.: A pattern and dependency based approach to the design of process models. In: Conceptual Modeling (ER 2004), Springer (2004) 724–739 LNCS 3288.
16. Wieringa, R., Gordijn, J.: Value-oriented design of correct service coordination processes: Correctness and trust. In: 20th ACM Symposium on Applied Computing, ACM Press (2005) 1320–1327
17. Kartseva, V., Hulstijn, F., Gordijn, J., Tan, Y.H.: Modelling value-based inter-organizational controls in healthcare regulations. In Suomi, R., Cabral, R., Hampe, J.F., Heikkila, A., Jarvelainen, J., Koskivaara, E., eds.: Proceedings of the 6th IFIP conference on e-Commerce, e-Business, and e-Government (I3E'06), Springer (2006) 278–291 IFIP International Federation for Information Processing.