An Integrated Approach for Modeling and Facilitating RFID-based Collaborative Logistics Processes

Yu Li Andreas Oberweis Huayu Zhang

Institute of Applied Informatics and Formal Description Methods (AIFB)
Karlsruhe Institute of Technology (KIT), 76128 Karlsruhe, Germany

{yu.li, andreas.oberweis, huayu.zhang}@kit.edu

ABSTRACT

Logistics processes, where RFID technologies are applied to improve process efficiency, and flexible organization of collaboration is required in the foreground, pose challenges to process modeling, analysis and design in Business Process Management. To meet these challenges, we propose in this paper an integrated approach which bundles XSLT nets, a novel variant of high-level Petri nets to model RFID-based logistics processes and to enable their automatic execution, as well as social networking coordinated by Community Processes to support the interconnection and web-based collaboration of logistics partners. The utility of the approach is demonstrated in a modeling use case.

Categories and Subject Descriptors

D.2.2 [**Software Engineering**]: Design Tools and Techniques – *Petri nets.*

General Terms

Design

Keywords

Logistics Processes, RFID, Social Networks, XSLT Nets, Community Processes

1. INTRODUCTION

RFID (Radio Frequency Identification) technologies are increasingly applied in logistics processes for identification and tracking of goods. Using RFID, enterprises are able to improve their performance and competitiveness due to the increased efficiency of information flow management that allows quicker reaction to real environmental changes. Besides, as logistics activities are usually performed across organizational boundaries (e.g., in third-party logistics), collaboration among different organizations is seen as a key differentiator in achieving integration and efficiency in logistics networks [1]. Logistics processes, in which RFID technologies are applied to improve process efficiency, and flexible organization of collaboration is required in the foreground, are referred to in this paper as *RFID-based collaborative logistics processes* (*RFID-CLPs*). To support the modeling and execution of RFID-CLPs, specific approaches are needed that should consider not only the manipulation of large amounts of RFID data while executing logistic activities, but also the organization of possible collaborations, even those that can be automatically triggered by transferring RFID data.

SAC'11, March 21-25, 2011, TaiChung, Taiwan.

In this paper we propose an integrated approach for modeling and facilitating RFID-CLPs, which bundles *XSLT nets* to model RFID-based logistics processes and social networking coordinated by *Community Processes* [2] to support interconnection of business partners and to facilitate web-based collaborations. As a variant of high-level Petri nets [3], XSLT nets are well suitable for modeling business processes with related XML data objects, and for transforming and manipulating these objects to enable automatic execution of process models.

Social networks are known as platforms that can stimulate the exchange and sharing of information among network members. Moreover, they can initiate collaborations through, e.g., searching for logistics partners. According to different development stages of social networks and results of the analysis of interpersonal relationships, activities and human resources can be coordinated in an easier and more targeted way in Community Processes for the initiation and execution of collaborations in social networks.

The communication and interaction between logistics processes and Community Processes are made possible in our approach by using message services, which enable loosely coupled, distributed and asynchronous data exchange between messaging peers. The main advantage of this approach is that the information flow in logistics processes can be effectively controlled in a coordinated collaboration context.

The rest of the paper is organized as follows: The next section surveys related work. In Section 3, RFID technologies are shortly introduced and RFID-CLPs are characterized. In Section 4, XSLT nets are proposed and elaborated. Section 5 explains how collaboration in RFID-CLPs is supported and coordinated based on Community Processes. Section 6 presents the concept of our integrated approach. A use case to which the approach is applied is described in Section 7. Section 8 concludes the paper with an outlook on future research.

2. RELATED WORK

As no approaches are proposed so far in the literature, which integrate the modeling of RFID-based logistics processes and the support for the coordination of web-based collaboration, existing work relevant to our approach can be differentiated into two categories: (1) Languages for modeling logistics processes and related data objects, and (2) approaches that support web-based collaboration in logistics processes.

Among logistics process modeling languages proposed in the literature, UML and web service-related languages (e.g., BPEL, XPDL, ebXML) also support modeling process data objects based on UML object diagram and XML, respectively. The problem of modeling with UML is that the integration of data model into process model is circuitous and no integrated data manipulation mechanism is provided. In web service-related languages, data objects (XML messages) can be modeled, processed and exchanged. Nevertheless, the absence or insufficiency of their formal foundations makes it impossible to automatically verify process models at design time.

As variants of high-level Petri nets with formal foundations, Predicate/Transition nets (Pr/T-nets) [4] and XML nets [5] integrate process modeling and the modeling of data objects stored in relational database and XML, respectively. However, complex structured data objects cannot be directly modeled with Pr/T nets. In XML nets, the expressiveness of XManiLa, a graphical language used to inscribe edges for data manipulation, is limited.

Approaches suggesting support for coordination of web-based collaborations in logistics processes can mainly be found in the area of e-Collaboration. [8] presents a survey of theoretical studies on the impact of e-Collaboration on supply chain management including descriptive frameworks, analytical models, empirical analysis, and case studies. Besides, there are many e-Collaboration tools such as E-Hub [9] and e-Marketplace [10] providing internet platforms for the communication among logistics organizations and the establishment of business relationships. Social software that promises inter- and intra-organizational alignments and information sharing in logistics processes is mentioned in [11]. However, the authors do not address concretely how social software, especially social networks, can be used to coordinate web-based collaborations. Especially, the support for RFID-triggered collaboration on web-platforms to improve logistics performance is not considered in the literature, which will be referred to in this paper.

3. RFID-BASED COLLABORATIVE LOGISTICS PROCESSES

In this section we shortly introduce RFID technologies and their usage in logistics processes. RFID is an automatic identification and data capture technology. An RFID system is composed of three elements: *transponder*, *reader*, and *middleware*. A transponder is an RFID tag (combined with an antenna) holding data that can be read and/or written. A reader (incl. antenna) communicates via radio waves with RFID tags and delivers information in digital format to a computer system (middleware) that bridges RFID hardware and enterprise applications. The communication between transponder and reader follows the requirements of the international standards EPC (Electronic Product Code) from EPCglobal. The EPC is a unique number used to identify a specific item that is stored on an RFID tag. To enable effective handling of massive amounts of EPC data among trading partners, EPCglobal has designed an EPC Network Architecture[1]. In this paper, we focus on the data exchange between the EPC Network system and enterprise applications where logistics processes are managed. The exchanged data including EPCs and configuration files for infrastructure components etc. can be described with the XML-based Physical Markup Language (PML)[2].

While adopting RFID technology in diverse areas of logistics processes such as outgoing/incoming goods control, warehousing, production, and distribution [12], the following challenges [13] which require careful attention must be met:

- Technological challenges: include high data volume and no capture of inter-object relationships [14], an entire RFID infrastructure [15], security and privacy implications [16], RFID-forced business process automation (business process automation based on RFID data processing).
- Organizational challenges: efficient and effective collaboration among logistics partners (e.g. Production-Distribution, Procurement-Production, and Distribution-Warehousing) upon RFID-based flow of information and goods.
- Economic challenges: Cost challenges, Patent challenges, ROI (Return on Investment) challenges, etc.

Our approach considers especially the technological and organizational challenges, i.e., RFID-forced business process automation and efficient and effective collaboration by integrating formal business process modeling language and social networking.

4. MODELING RFID-BASED LOGISTICS PROCESSES WITH XSLT NETS

PML that is introduced in the previous section can be used to describe observations of RFID readers as XML data objects that are exchanged between EPC Network and enterprise applications where information flow of logistics processes is managed. In order to efficiently manipulate these XML-described data objects and ultimately to achieve process automation and improvement, we propose XSLT nets that allow 1) graphical process modeling, 2) modeling of XML data objects of processes, 3) transformation and manipulation of XML data objects to enable automatic process execution, and 4) automated, static analysis of process models (check of validity and correctness) for process improvement.

XSLT nets represent a novel variant of high-level Petri nets. Preliminarily, a Petri net is a directed, connected and bipartite graph in which nodes represent places and transitions, and places can contain tokens. High-level Petri nets are, in contrast to elementary Petri nets, those variants of Petri nets whose tokens are typified and distinguishable. XSLT nets inherit basic modeling concepts including graphical notations from Petri nets such as places (depicted as circles), transitions (as squares), arcs, and tokens (as black dots). Tokens in XSLT nets are XML data objects (XML documents) that are modeled and typified by XML Schemas. To manipulate tokens and thus to enable object flow and process execution, Extensible Stylesheet Language Transformations (XSLT) is used to inscribe Petri net edges connecting places and transitions. In this section, we explain firstly XSLT edge inscription and then specify structure and dynamics of XSLT nets, based on which automated, static analysis of XSLT net models can be performed in order to preclude at design time model errors that may lead to economic loss in real execution.

In XSLT nets, depending on the direction of inscribed edges, XSLT edge inscriptions, or *Xinscriptions* for short, are divided into incoming and outgoing Xinscriptions. *Incoming Xinscriptions* inscribe edges leading from places to transitions and are used to read or (partially) delete XML documents from places, whereas *outgoing Xinscriptions* are assigned to edges leading from transitions to places and are used to insert XML elements or attributes into existing documents in places. Inserting a root element into an empty document implies the creation of a new document. The

---

[1] See http://www.epcglobalinc.org/standards for details.

[2] See PML Core Specification 1.0 available at http://xml.coverpages.org/PMLCoreSpec10.pdf.

read-, delete-, and insert-operations are realized through the execution of Xinscriptions based on XSLT transformations. In an XSLT transformation, an XSLT processor usually reads an XML source document and an XSLT stylesheet as input and generates a new target document in XML or another format (e.g., HTML, PDF, plain text, etc.) as output without changing the source document. While executing Xinscriptions, *variables* can be used as temporary storages for data that is read or deleted from source documents and then is to be inserted into target documents. These variables and their XML data values are stored in temporary documents called *variable documents*. In XSLT nets, the primary Petri net constructs are interpreted as follows:

- *Place and marking*: A place is viewed as a container for a set of XML documents (tokens) that conform either to an *upper* or to a *lower XML Schema*. Depending on direction of the Xinscription of the adjacent edge, the upper and the lower Schema can serve, respectively, either as *source* or as *target Schema* of the Xinscription (i.e., XML Schema for source or target documents in XSLT transformations). Accordingly, the marking, i.e., the set of tokens of the place, consists of two subsets of tokens that conform to the upper and the lower Schema, respectively.
- *Transition and transition inscription*: A transition represents an activity that executes operations (i.e., insert, delete, and read) on documents of its adjacent places. Each transition can be optionally inscribed by a logical expression, which is used to restrict execution conditions of document operations. Variables declared in corresponding incoming Xinscriptions can be used to formulate transition inscriptions.
- *Edge and edge inscription*: Depending on the direction, each edge is inscribed by an incoming or an outgoing Xinscription that specifies operations on the marking of the adjacent place.

Formally, we define (the structure) of XSLT nets as follows:

Definition 1 (XSLT net). An XSLT net is a tuple $XSLTN = (S, T, F, \Psi, \delta_{upper}, \delta_{lower}, \kappa, \tau, M_0)$, where:

1. (S, T, F) is a (Petri) net, where S is the finite set of places, T the finite set of transitions, and $F \subseteq (S \times T) \cup (T \times S)$ the set of edges.
2. $\Psi = (D, FU, PR)$ is a structure consisting of a non-empty, finite individual set D, a set FU of term functions defined on D, and a set PR of predicates defined on D (cf. corresponding definitions of Pr/T nets in [4]).
3. The functions $\delta_{upper}: S \to \mathbb{G}$ and $\delta_{lower}: S \to \mathbb{G}$ assign to each place $s \in S$ an upper and a lower Schema, respectively, where $\mathbb{G}$ is a set of XML Schemas.
4. The edge inscription function $\kappa: F \to \mathbb{X}$ assigns to each edge from F an Xinscription, where $\mathbb{X}$ is a finite set of Xinscriptions.
5. $\tau: T \to P_\Psi$ assigns to each transition $t \in T$ a transition inscription in form of a predicate logical expression that is built on the structure Ψ and the set of variables declared in Xinscriptions of all incoming edges of t, where P_Ψ is a set of such expressions.
6. $M: S \to 2^{\mathcal{T}_\Sigma}$ marks each place $s \in S$ with a set of XML trees (tree representation of XML documents), which again consists of two subsets of XML trees that respectively conform to the upper and the lower Schema, i.e., $\forall s \in S, M(s) = M_{upper}(s) \cup M_{lower}(s)$, where $M_{upper}: S \to 2^{\mathcal{T}_\Sigma}$ and $M_{lower}: S \to 2^{\mathcal{T}_\Sigma}$ are two mappings with $\forall s \in S: d \in M_{upper}(s) \Longrightarrow \delta_{upper}(s) \vdash d$ (read "document d is conform to / valid against Schema $\delta_{upper}(s)$") and $\forall s \in S: d' \in M_{lower}(s) \Longrightarrow \delta_{lower}(s) \vdash d'$. $\mathcal{T}_\Sigma$ denotes here the set of all XML trees over an alphabet Σ. M_0 is the initial marking. ■

The structure Ψ induces for each predicate logical expression $p \in P_\Psi$ and each instantiation β of variables a logical value with the function $\psi_\beta: P_\Psi \to \{0,1\}$ (cf. [4]).

The dynamics, i.e., the *transition occurrence rule* of XSLT nets is specified as follows: A transition in XSLT nets is *enabled* if the following conditions are satisfied:

- Each incoming or outgoing edge of the transition is inscribed by an Xinscription that is valid w.r.t. its source and target Schema, i.e., the upper and the lower (or the lower and the upper) Schema of the adjacent place.
- Each place in the pre-set of the transition (i.e., sources of edges leading to the transition) contains at least one non-empty XML document that is valid against the upper Schema of the place. In addition, if the target document of the corresponding transformation is not empty, it should not already be contained in the set that consists of all documents in the marking of the place except the source document.
- Each place in the post-set of the transition (i.e., targets of edges from the transition) contains at least one XML document that is valid against the lower Schema of the place. Note that the empty document is contained in every document set and is valid against the empty XML Schema. Additionally, the target document of the corresponding transformation should not be empty and not already be contained in the set that consists of all documents in the marking of the place except the source document.
- The transition inscription, if defined, must be evaluated to `true` for the given instantiation of variables used in the inscription.

If an enabled transition *occurs*, XML documents of pre-set places that are valid against corresponding upper Schemas are read or (partially) deleted. In the marking of post-set places, new XML documents are added, or new elements/attributes are inserted into existing documents that are valid against corresponding lower Schemas. Limited by the length of the paper, formal definitions w.r.t. validity of Xinscription and dynamics of XSLT nets will be presented in a separate paper. An example of XSLT nets will be given in the use case in Section 7.

5. COMMUNITY-BASED SUPPORT FOR COLLABORATION IN RFID-CLPS

As mentioned in Section 1, logistics processes are usually inter-organizational processes involving different logistics partners. Collaboration among logistics partners can be either vertical or horizontal. A vertical collaboration refers to a collaboration with customers, internally across functions, and with suppliers. In contrast, a horizontal collaboration includes collaboration with competitors, internally and with non-competitors, e.g. sharing manufacturing capacity [17]. In this paper, we focus on vertical collaboration, i.e., collaboration in procurement-production-distribution-marketing processes [18]. Issues related to (vertical) collaboration in logistics processes include the following [19]:

- Efficient communication and better understanding
- Improvement of internal and external trust
- Transparent information exchange (visibility of real demand)
- Realization of mutual benefits

Social networking is recognized to be an appropriate technology that helps to solve these issues due to the support for self-representation, contact establishment, and communication. In [2], a model called *Community Process* is introduced, which takes advantage of social networks to enhance both communication and coordination in collaborations.

A Community Process (CP) is a set of related activities of network members that are executed to achieve a collaboration output [2]. The activities of a Community Process differ between *Single Activities* (involving only one person) and *Collaborative Activities* (involving collaborators). By specifying Collaborative Activities, collaborations can be organized in ad-hoc situations, meaning that the manner of collaboration execution (e.g. job assignment, execution of Single Activities) is defined by individual peers. The modeling notation of Community Processes is derived from Petri nets by adding additional elements such as *F-*, *B-*, *C-block* representing abstract sub-process building blocks and *member* (with a name) for a social network member. Figure 1 shows an example of a simple Community Process model.

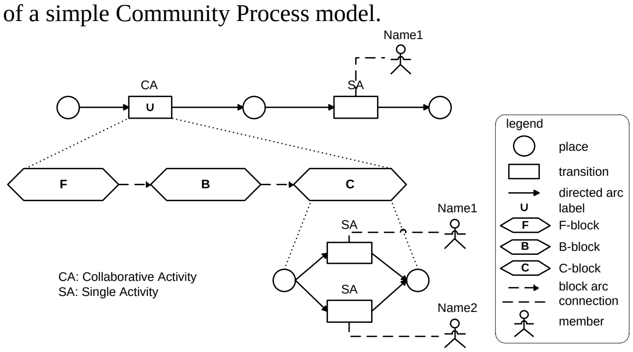

Figure 1. An example of a simple Community Process model

In the Community Process model, Collaborative Activities are modeled as transitions labeled/marked with “U” representing collaborative behavior that is refined by a sequential sequence of abstract sub-processes *Finding Partners* (F), *Building Relationship* (B), and *Collaboration Execution* (C). The first two sub-processes (F and B) focus on the preparation of collaboration, while the third sub-process (C) refers to the actual execution of assigned jobs. The whole Community Process is associated with a set of *Community Process Objects* that include e.g., *Community Users* (which describe network members) and *Community Contents* (that are data objects transferred from one activity to another). Based on user relationships (e.g., obtained through analyzing outgoing Emails or Chats [20]) that are stored and continuously updated in Community Users, social network structure can be created. Upon this structure, analysis methods can be applied to recommend contact persons or collaborators in one’s personal network while executing the sub-process F, or to suggest how to contact potential collaborators in sub-process B.

This model is used in our integrated approach, as shown in Section 6, to support collaborations among logistics partners who are expected to be members of a business social network. Furthermore, we present in the use case in Section 7 an RFID-enabled automatic collaboration initiation in logistics processes.

6. THE INTEGRATED APPROACH

In this section we elaborate our approach for modeling and facilitating RFID-CLPs, which integrates XSLT net-based business process modeling and community-based support for collaboration by using message service as connector. A *message service* is a middleware that enables loosely coupled, distributed, and asynchronous communication between messaging peers that are software components or applications. Messaging peers are connected by a messaging agent that provides facilities for creating, sending, receiving, and reading messages. Each peer can send messages to, and receive messages from any other peer without precise knowledge of each other (loose coupling). They only need to know the message format and the destination to which the message is sent. With the help of message queuing, message agent can temporarily store messages and forward them when the recipient is available (asynchronous communication). Implementations of message service include the Java Message Service (JMS)[3] that is a part of the Java Platform, Enterprise Edition.

Figure 2 illustrates the concept of our approach, in which Collaborative Activities of Community Processes are modeled as XSLT net transitions (labeled with “U” and named “collaborative transitions”) that represent activities for initiating collaboration in RFID-based logistics processes. The logistics processes run distributedly on servers A_1, …, A_n, whereas the F-, B-, and C-sub-process of Community Processes on servers C_1, C_2, and C_3. Note that it can hold that $A_i = A_j\ (i,j \in \{1,\dots,n\})$ and $C_i = C_j\ (i,j \in \{1,2,3\})$. The communication and interaction between logistics processes and Community Processes is realized by exchanging messages via a message service running on server B. If a collaborative transition in a logistics process occurs, XML data read or filtered by the incoming Xinscription is encapsulated in a message and sent via message service to the Community Process. Upon receipt of the message, the F-sub-process of the Community Process is triggered/enabled. During execution of the Community Process, the F-, B-, and C-sub-process can continue exchanging messages such as notifications, partner search results etc. with the collaborative transition. After the C-sub-process is finished, the XML-based result (e.g., collaboration agreement, contract etc.) is sent back as message to the collaborative transition and then is processed/manipulated by the outgoing Xinscription to generate or change XML documents in the post-set places of the collaborative transition so that the logistics process continues.

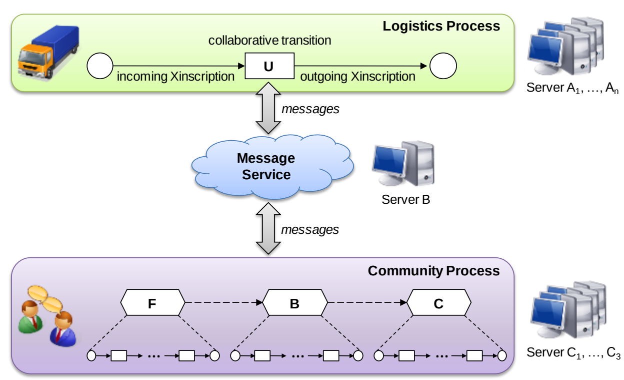

Figure 2. Concept of the integrated approach

As illustrated, message service is used to bridge logistics processes and Community Processes. In fact, it can also be used to enable communication between logistics transitions or sub-processes running on different servers (i.e. server A_1, …, A_n), or to loosely couple the F-, B-, and C-sub-process of distributed Community Processes.

The approach is integrative also in the sense that processes within the F-, B-, and C-block of Community Processes can be modeled

[3] http://java.sun.com/products/jms/.

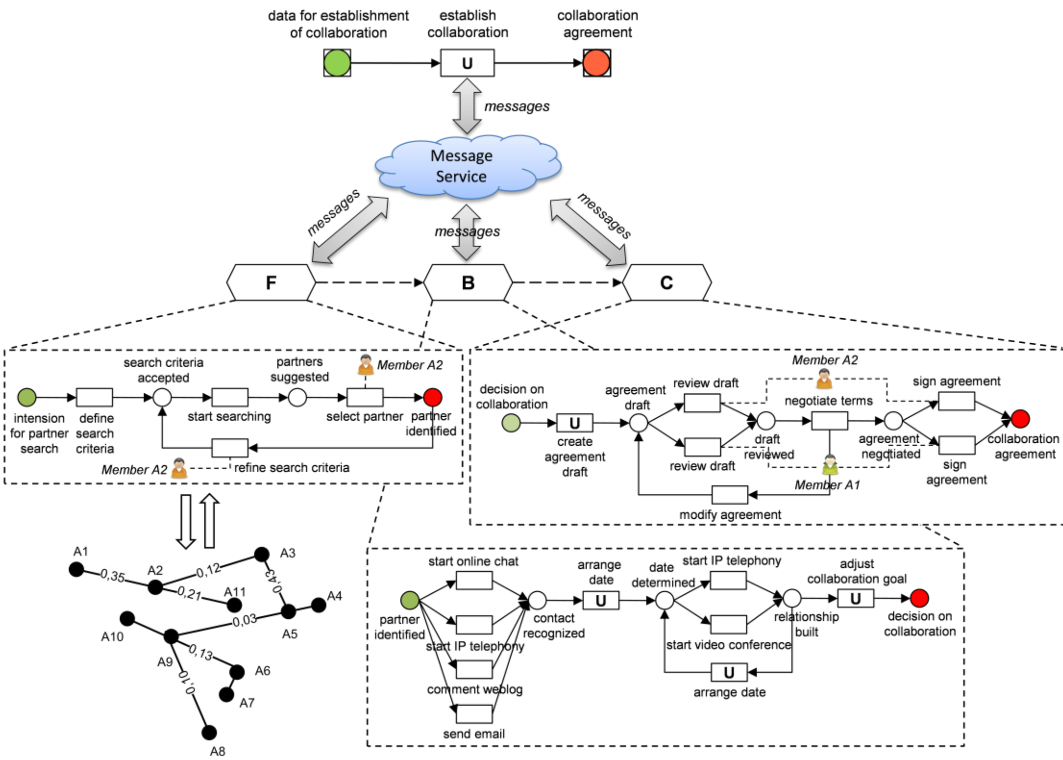

Figure 3. Community Process for supporting collaboration

using XSLT nets, where members correspond to roles that are assigned to XSLT net transitions. Data manipulated by Xinscriptions and flowing in these internal processes can be XML-based, social network-specific data like FOAF (Friend of a friend) ontology[4] in RDF (Resource Description Framework) or OWL (Web Ontology Language) format.

7. USE CASE

To demonstrate the usage of the approach described in the previous section, we model in this section a distribution logistics process in which manufacturers, a logistics service provider (LSP) and retailers are involved. Figure 3 shows the first-level process model of this use case.

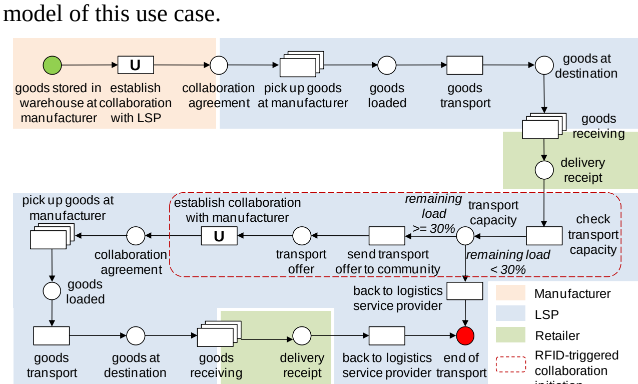

Figure 4. First-level model of an RFID-based collaborative logistics process

After having stored goods in his distribution warehouse, which should be transported to a domestic retailer, the manufacturer starts to establish collaboration relationship with an LSP by enabling the collaborative transition "establish collaboration with LSP", whose interaction with a Community Process running in a business social network is shown in Figure 4. The F-, B-, and C-blocks are concretized by XSLT net models that enable partner search, relationship building, and signing of collaboration agreement, respectively. In the F-block, according to search criteria provided by the manufacturer, *semantic matching* [21] of LSP profiles and *Social Network Analysis* (e.g., analysis of centrality, indegree/outdegree, transitivity, etc.) [22] upon the social network structure derived from Community Users (s. Sociogram in Fig. 4) are applied to generate a list of recommended partners, which can either be displayed on community platform, or sent via message service back to server of the manufacturer to enable some automatic partner selection procedure. The manufacturer can select one partner from the list, or repeatedly enable partner search with refined criteria until an appropriate partner is identified.

The logistics process continues upon receipt of the collaboration agreement, according to which the LSP partner who is in this use case assumed to have his own vehicles for transportation picks up goods at the manufacturer's warehouse, where RFID-based outgoing goods inspection is done. The loaded goods are then transported directly to the retailer's warehouse without cross-docking. The RFID-based incoming goods inspection made at the retailer's warehouse is modeled as a sub-process that refines the transition "goods receiving" (s. Figure 5).

[4] See FOAF Vocabulary Specification 0.97 available at http:// xmlns.com/foaf/spec/.

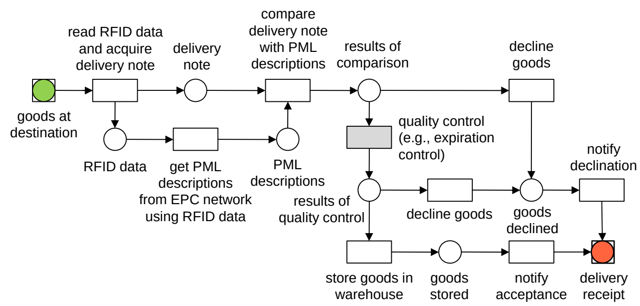

Figure 5. Subordinate process model for RFID-based incoming goods inspection

In this sub-process, RFID data read by RFID readers is used to acquire PML product descriptions from EPC network, which are then compared with the delivery note. Items whose PML descriptions differ from their corresponding entries in the delivery note are declined. The rest of items are sent to quality control such as expiration control. According to the quality control results, items are either declined, or accepted and stored into the retailer's warehouse. Finally, a receipt is issued by the retailer based on the declination or acceptance of items.

To demonstrate in detail how XSLT nets can be used to enable data flow in RFID-CLPs, we show in Figure 6 an XSLT net fragment for the expiration control, in which expiry date of goods are controlled based on PML data observed by an RFID reader. The PML Schema that typifies the observations is graphically represented by the upper Schema XS1 of the place "sensor observation". The activity "expiration control" checks the value of the XML element "expiry date" in the PML Schema to decide if an RFID-attached item will be added in a list of accepted items. If accepted (i.e., the expiry date is later than current date), the tag ID of the item will be extracted (read) from the source document of X1 (cf. code listing below) and stored as values of the variable `TagID` in the variable document `variables.xml`. Sequentially, the tag ID will be read by the outgoing Xinscription X2 (code is omitted here due to space limit) from the variable document and inserted into an existing document, meaning that the item is available for further quality control steps. As in this example tokens (PML observations) are deleted from the place "sensor observation", its lower Schema is empty and therefore not shown. In the place "list of accepted items", the upper and the lower Schema are identical in this case and both depicted by XS2.

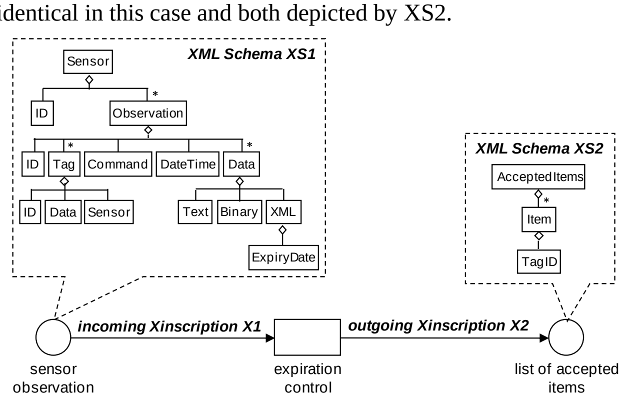

Figure 6. An XSLT net fragment for expiration control

Incoming Xinscription X1:

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="2.0" ...>
 <xsl:template match="/pmlcore:Sensor">
  <xsl:result-document href="variables.xml">
   <xsl:element name="values">
    <xsl:for-each select=
     "./pmlcore:Observation/pmlcore:Tag">
     <xsl:if test="./pmlcore:Data/pmlcore:XML/
      ExpiryDate/text() > current-date()">
      <xsl:element name="value">
       <xsl:attribute name="name">TagID
       </xsl:attribute> <xsl:copy-of
       select="./pmluid:ID/text()"/>
 ...
</xsl:stylesheet>
```

On the backhaul, the RFID reader installed on the truck observes current transport capacity. If the remaining load is lower than 30%, the trucker is instructed to directly drive back to the LSP. Otherwise, in order to improve vehicle utilization and minimize deadhead miles for economic and ecological reasons, an XML-based transport offer is generated by Xinscription on the basis of the PML observation and other related information such as current location, drive route, search radius etc. The offer for initiating collaboration is then sent automatically via the message service to the Community Process (s. Figure 4), in which a manufacturer whose goods need to be transported to a retailer will be searched for firstly. As one of the search criteria, the warehouse of the manufacturer and the retailer should be near to the drive route within the search radius. According to a list of recommended partners, the LSP contacts manufacturers until an appropriate one is found and an (oral) agreement is reached. The trucker then picks up goods at the manufacturer's warehouse and sent them to the retailer's warehouse. After the incoming goods inspection is done and the delivery receipt is issued, the trucker drives back to the LSP in this use case. The described process for minimizing deadhead miles can be repeated until the distance between the truck and the LSP is smaller than a certain number of kilometers. The whole logistics process terminates when the truck arrives at the LSP.

8. CONCLUSIONS

In this paper we have proposed an integrated approach for modeling and facilitating RFID-based collaborative logistics processes, in which XSLT nets are used to model and execute RFID-based logistics processes, while web-based collaborations are coordinated through Community Processes. The approach has the following advantages and mainly aims to support SMEs (Small and Medium Enterprises) where heterogeneous data and interaction patterns (e.g., communication channels) may obstruct their inter-organizational business operations.

- *Integration of process modeling and social networking*: By unifying XSLT nets for modeling business processes with related data objects, and Community Processes for coordinated, workflow-supported social networking, our approach provides powerful means to model RFID-based logistics processes and to enable collaboration among logistics partners.
- *Efficient data processing, presentation, sharing, and tracking*: By using XSLT for automated data transformation and message services for reliable data exchange, XML-based data such as business documents, RFID observations, social network ontologies etc. can be efficiently processed, presented in various formats (e.g. HTML, PDF, plain text etc.), and subsequently shared among logistics partners or social network members and tracked on different terminals (e.g. mobile devices).

- *Executability and verifiability*: Due to the executability and formal foundation of XSLT nets, models created using our approach can be validated through simulation, verified against model correctness (e.g., well-structuredness) [23], and executed by XSLT net-based workflow engine.
- *Flexibility and reusability through loose coupling*: By using message services to enable communication and interaction between logistics processes and Community Processes, among in-house processes of logistics partners, and among the F-, B, and C-blocks of Community Processes, the flexibility of collaborative logistics processes and the reusability of their components are increased because of loose coupling. Internal details of process components running on different servers can be hidden if interfaces (e.g. APIs) and message formats are unified.
- *Transparent and cost-saving establishment of collaboration relationship*: Interconnection of business partners and establishment of collaboration relationship through (RFID-triggered) social networking save cost and are more transparent in comparison with conventional mediation through third-party brokers. It is more transparent because communication behavior of partners and status of collaboration execution are made clear to all collaborators in the social network.

In the open source software toolset KIT-Horus [24], an RCP (Eclipse Rich Client Platform) based Java application aiming at supporting the development of process-oriented information systems, functionalities for modeling and simulating XSLT nets are already implemented as components (plug-ins). Currently, we are developing an XSLT net-based workflow engine and a social network platform on which Community Processes are supported. The presented approach is planned to be evaluated in industrial projects, not only in logistics branch, but also in other areas where business process modeling, data manipulation, and coordination of web-based collaboration need to be integratedly supported.

9. REFERENCES

[1] Power, D. J., Sohal, A. S., and Rahman, S-U. Critical success factors in agile supply chain management. *International Journal of Physical Distribution & Logistics Management, vol. 31, No. 4*, 2001, 247-65.

[2] Koschmider, A., Oberweis, A., and Zhang, H. Process-oriented coordination of collaborations in social networks. In *Proceedings of the 6th Int. Conf. on Web Information Systems and Technologies (WEBIST)*. INSTICC Press, 2010.

[3] Jensen, K., Rozenberg, G. (eds.) *High-level Petri nets: theory and application*. Springer, 1991.

[4] Genrich, H. J. Predicate/Transition Nets. In *Petri Nets: Applications and Relationship to other Models of Concurrency, Advances in Petri Nets 1986*. Springer, 1987.

[5] Lenz, K. and Oberweis, A. Inter-Organizational Business Process Management with XML Nets. In *Petri Net Technology for Communication Based Systems*. *LNCS, vol. 2472*, 243-263. Springer, 2003.

[6] Mevius, M. *Kennzahlenbasiertes Management von Geschäftsprozessen mit Petri-Netzen* (in German). Ph.D. Thesis, Verlag Dr. Hut, München, 2006.

[7] Kay, M. Comparing XSLT and XQuery. In *Proceedings of XTech 2005: XML, the Web and beyond*, 2005.

[8] Johnson, M. E. and Whang, S. E-Business and Supply Chain Management: An Overview and Framework. *Tuck School of Business Working Paper, No. 03-06*, 2002.

[9] Kathawala, Y., Abdou K., and von Franck, C. Supply chain/electronic hubs: a comparative analysis. *Benchmarking: An International Journal*, 450-470, 2002.

[10] Eng, T.-Y. The role of e-marketplaces in supply chain management. *Industrial Marketing Management, vol. 33, Issue 2*, 97-105, 2004.

[11] Awa, H. O. and Emecheta, B. C. Collaborative Supply Chain in the Digital Age: A Case Study of its Extent of Adoption by Indigenous Organizations in Building Inter-and Intra-firm Alignments. *Computer and Information Science, vol. 3, No. 1*, 2010.

[12] Zaboj, M. Using RFID in supply chain and retail store unit. *ZEMEDELSKA EKONOMI-KA, vol. 51*, 427-434, 2005.

[13] Wu, N. C., Nystrom, M. A., Lin, T. R., and Yu, H. C. Challenges to global RFID adoption. *Tech-novation, vol. 26, No.12*, 1317-1323, 2006.

[14] Cocci, R., Tran, T., Diao, Y., and Shenoy, P. J. Efficient Data Interpretation and Compression over RFID Streams. In *Proceedings of Int. Conf. on Data Engineering (ICDE'08)*, 1445-1447, 2008.

[15] Ornauer, J., Michielsen, E., and Rerisi, E. RFID Integration Services Markets, 2004 Q1. *Allied Business Intelligence*, 2004.

[16] Sarma, S. E., Weis, S. A., and Engels, D. W. RFID Systems and Security and Privacy Implications. In *Proceedings of the 4th Int. Workshop on Cryptographic Hardware and Embedded Systems (CHES 2002). LNCS, vol. 2523*, 454-469. Springer, London, 2002.

[17] Simatupang, T. M. and Sridharan, R. The collaborative supply chain. *International Journal of Logistics Management, vol. 13, No. 1*, 15-30, 2002.

[18] Fawcett, S. and Magnan, G. The Rhetoric and Reality of Supply Chain Integration. *International Journal of Physical Distribution & Logistics, 32(5)*, 339-361, 2002.

[19] Barratt, M. Understanding the meaning of collaboration in the supply chain. *Supply Chain Management, vol. 9, No. 1*, 30-42, 2004.

[20] Ehrlich, K., Lin, C. Y., and Griffiths-Fisher, V. Searching for experts in the enterprise: Combining text and social network analysis. In *Proceedings of Int. ACM Conf. on Supporting Group Work (GROUP'07)*, 117-126. ACM, 2007.

[21] Giunchiglia, F., Yatskevich, M., and Shvaiko, P. Semantic Matching: Algorithms and Implementation. *Journal on Data Semantics IX, vol. 4601/2007*, 1-38. Springer, 2007.

[22] Wasserman, S. and Faust, K. *Social network analysis: methods and applications*. Cambridge University Press, 1994.

[23] Sifakis, J. Structural properties of Petri nets. *Mathematical Foundations of Computer Science, LNCS, vol. 64*, 474-483. Springer, 1978.

[24] Klink, S., Li, Y., and Oberweis, A. INCOME2010 - a Toolset for Developing Process-Oriented Information Systems Based on Petri Nets. In *Proceedings of Int. Workshop on Petri Nets Tools and Applications (PNTAP 2008)*. ACM, 2008.