
Pablo David Villarreal
GIDSATD Research Group
Universidad Tecnológica Nacional - Facultad Regional Santa Fe
Lavaisse 610, Santa Fe, Argentina, 3000
pvillarr@frsf.utn.edu.ar

Enrique Salomone, Omar Chiotti
(1) GIDSATD Research Group
Universidad Tecnológica Nacional - Facultad Regional Santa Fe
Lavaisse 610, Santa Fe, Argentina, 3000
(2) INGAR – CONICET
Avellaneda 3657, Santa Fe, Argentina, 3000
{salomone,chiotti}@ceride.gov.ar

Abstract
Collaborative business models have been proposed to improve the performance of the organizations in the supply chain management. They are supported by Business-to-Business (B2B) E-Commerce relationships established between organizations. These models require business processes that span organizational boundaries and have to be jointly managed by the partner organizations. Furthermore, some collaborative models impose big challenges from the point of view of information systems: enterprise and systems autonomy, decentralized management, peer-to-peer interactions and negotiation capabilities. Workflow approaches were used to model interactions in the business process layer of B2B relationships. However, as we discuss in this paper, they present shortcomings to achieve these requirements.

To support collaborative models across B2B relationships, in this paper we present a B2B Interaction Protocol approach to model and manage interactions between organizations. These interactions represent public business processes that partner organizations agree to achieve a goal. The B2B interaction protocol approach enable to model and manage interactions between organizations in a high abstraction level. To do that, a graphical language is presented to model B2B interaction protocols, and the architecture of an Interaction Protocol Manager System is defined. Moreover, a comparison between workflow approaches and the B2B interaction protocol approach is described.

Keywords: Business Process, Business-to-Business, Interaction Protocol, Workflow
1 Introduction

The arrival of the Internet, Web technologies and distributed systems have enabled organizations to conduct businesses electronically originating what was later known as Business-to-Business (B2B) E-Commerce. Organizations can operate across organizational boundaries with the purpose of collaborating among them through B2B relationships. Several collaborative business models have been proposed to improve the performance of organizations in the domain of the supply chain management. Some of these models are: Collaborative Planning, Forecasting and Replenishment (CPFR) [8], Demand Activated Manufacturing Architecture (DAMA) [6] and Partner-to-Partner Collaborative Model [20]. These models require business processes that span organizational boundaries and have to be jointly managed by partner organizations. In addition, some models like the Partner-to-Partner Collaborative Model, present big challenges from the point of view of information systems:

- Each organization should be able to manage a B2B relationship with one partner independently of the other B2B relationships.
- The management of B2B relationships has to be jointly carried out by the parties in a decentralized way.
- Enterprises should be able to behave as autonomous entities, hiding their internal decisions, activities and processes.
- The above requirements bring about organizations to implement Peer-to-Peer (P2P) information systems to support the business process management in B2B relationships. P2P information systems consist of components allowing organizations systems to interact in a direct way without mediation of an independent third party system [19] [4].
- In some collaborative models for the supply chain management, the business process management among partners must include negotiation.

Management of B2B relationships implies to manage two types of business processes: processes that belong to the enterprise called intra-enterprise, private or internal business processes, and processes that belong to the two enterprises involved in a B2B relationship called B2B, inter-enterprise or public business processes. Public business processes have to be agreed by the partners, and have to be jointly defined, executed, monitored and controlled. Unlike the public processes, private processes are defined, executed, monitored and controlled by each enterprise in an autonomous way. The above requirements define a public process as actually implemented by private processes. Private Business Processes have been managed within the enterprises using traditional Workflow Management Systems (WMSs) [22], ERP systems or proprietary systems. A clear separation between private and public processes is key to provide the necessary isolation and abstraction between enterprise internal processes and processes across enterprises [3]. The challenge is how to manage public processes meeting the requirements imposed by a partner-to-partner collaborative model. In addition, it is necessary to develop mechanisms to integrate public processes with private processes within each enterprise.

Several approaches on inter-organizational workflow [1], cross-organizational workflow [10] and inter-enterprise business process management [7] have been proposed to manage public processes of a B2B relationship and their integration with private processes. However, these workflow approaches present shortcomings to achieve the capabilities of autonomy, decentralization, P2P interaction and negotiation. Therefore, with the objective of achieving the necessary capabilities of the public business process management in B2B relationships another approach is needed.

In this work, we present an approach based on B2B Interaction Protocols to manage interactions in B2B relationships. B2B Interaction Protocols are used to model and manage public business processes that represent the interactions among enterprises. B2B interaction protocols have the objective of modeling public processes in a high abstraction level, which can be understood by enterprises, people and systems. This paper is structured in the following way. Section 2 discusses workflow approaches used to manage business processes in B2B relationships, identifies their shortcomings and the most appropriate workflow approaches to manage B2B relationships. Section 3 defines the B2B Interaction Protocol approach, describes a modeling language and shows an example. Section 4 presents the main components involved in a generic architecture of an Interaction Protocol Management System. Section 5 compares the workflow approaches and the B2B interaction protocol approach. Section 6 presents conclusions and future work.


Workflow is the automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules [22]. A Workflow Management
System (WiMS) is a system which allows the definition, execution, monitoring and control of workflows. A WiMS has one or more workflow engines that interpret process definitions and automate the coordination and the control of tasks during the business process execution. A task is performed executing a program, enacting a human/machine action, or invoking another process (sub-process). Workflow is a key technology to manage business processes and it has been used with success by organizations to manage their private business processes. Hence, a natural approach to manage public business processes is using a WiMS. However, traditional WiMSs are not appropriate to manage public business processes, mainly because they are based on a centralized management of the business processes [7]. Therefore, a number of approaches have been proposed based on adaptations or extensions to the workflow theory to manage business processes that across organizations.

2.1. Approaches based on Workflow to Manage Public Business Processes

Three approaches were identified in [15] to define public business processes: *split and deploy*, *composition* and *subcontracting*. In addition, we can mention another approach which we call *private and public processes*. Following, these approaches are described.

2.1.1. Split and Deploy

In this case, a top-down approach is used to define a complete business process model, which specifies the activities and sub-processes to be carried out by the enterprises involved in a B2B relationship. The global business process is split in sub-processes and activities that are then deployed in each enterprise. In this way, the global business process definition encompasses the public process as well as the private processes of the enterprises. This definition has to be shared by organizations which have to use the same process definition model. This imposes organizations to use the same WiMS. This approach is used by distributed workflow systems (DWMs) [16], by example METEOR [17] and Mentor-Lite [21]. DWMs are not appropriate in the context of B2B relationships due to several issues [16] [3]. The main problem is that DWMs does not satisfy the autonomy requirements. On the one hand, the use of a global process definition does not hide the private processes of each organization. This is not desired by the enterprises due to competitive reasons. Furthermore, the global workflow definition assumes a tight coupling among the sub-workflows and any change in a private process would require revising the global workflow affecting all the enterprises involved. The tight coupling also causes that private processes executed by each enterprise have to be modified to accommodate to the global process. On the other hand, each enterprise may use different data and process representation models in its private processes and they may require accessing to private applications. However, DWMs impose to the organizations to deploy the same workflow engine for interpreting the global workflow definition. Hence, even though workflow engines are distributed, the control of the global workflow and its sub-workflows is still centralized by one WiMS.

2.1.2. Composition

In this case, a bottom-up approach is used to define a global business process: the public process. The global process is defined from segments (private activities and sub-processes) exposed and offered by each participant enterprise. This approach allows organizations to hide the internal details of their private business processes. Those segments are publicized in the global process definition through an interface that defines its purpose. Hence, the private processes are not completely hidden and there exists a coupling among the global process and its segments. Any change in a segment of an enterprise could cause an undesired execution in the global process. An advantage of this approach is that the enterprises would not require to change their private process to accommodate to the global process. This approach is useful to implement a virtual enterprise where each enterprise offers services (private processes) to create a virtual process that attend some shared objective. In a virtual enterprise the management is centralized by a third party (called a Trading community) or by an enterprise that coordinates the virtual enterprise. As example, this approach is used in WISE [14]. WISE provides a centralized WiMS that coordinates the execution of the global process invoking local workflows of each enterprise, which implement the services.

2.1.3. Subcontracting

In this case, a hierarchical process approach is used to define a main process, which is composed of activities that represent other processes. This is used when an enterprise has a process and wants to outsource some parts of the process. To do that, the enterprise hires the services of other enterprises that manage the outsourced sub-processes or activities. In this approach, the main process is defined by one enterprise. These outsourced parts of the main process are viewed by the main enterprise like black-box because their details are hidden. In this way, enterprises do not share a global process but they manage their processes autonomously. It is important to point out that the mentioned autonomy
is obtained by delegating the responsibility in the execution of the outsourced segments. Any failure in the execution can compromise the main process. Hence, the outsourcing contracts usually specify the quality of services required for the outsourced segments. As example, this approach has been used in CrossFlow [10].

2.1.4. Public and Private Processes

This approach attempts to manage business processes of a B2B relationship explicitly separating them between public and private processes. The organizations involved in a B2B relationship define and agree on a common public business process, which is used as a contract among organizations. The public business process contains activities that are not executable and are of interest to all parties. This means that the public process consist of abstract activities. Private processes define internal executable activities of the organizations that implement the activities of public processes. In this way, the definition of the public process has to be shared by the organizations. Furthermore, each organization define the private processes in an independent way and these definitions are hidden to others organizations.

From the public process, each organization knows what activities has to execute and what activities have to be executed by their partners. The advantage of this approach is that local changes on private processes do not have impact on the public process. Organizations can define and manage different public processes with each partner without changing definitions on their private processes. If an organization creates a new relationship with a partner using a new public process, the organization only needs to add the mechanism for integrating the public process with the private processes.

As an example of this approach we can mention the Public-to-Private approach to inter-organizational workflows [1]. This approach is based on the notion of inheritance where private processes are subclasses of the public process. The private processes are defined using transformation rules based on an ad-hoc notion of inheritance. Although each organization can define its private processes in any way that they want, the private processes are derived from the public process. This is not suitable in a B2B relationship based on P2P interactions where an enterprise wants to establish different relationships with each partner using different public processes. Each new public process will require the definition of the corresponding private processes each time. Nevertheless, this approach is based on notions of workflow verification that assure that private workflows do satisfy the public process.

Another example is the approach called inter-enterprise collaborative business process management [7]. In this case, organizations define a common collaborative process that is the public process. The process definition is shared by the organizations and it is based on a commonly agreed business protocol. Each execution of a public process does not consist of a unique process instance but of peer process instances run by Collaborative Processes Managers (CPMs) of participating parties. These peer instances share the same public process definition, but may have private processes that are also managed by the CPMs. The CPMs run peer instances independently and collaboratively and they interoperate through an inter-CPM communication protocol to synchronize the execution progress of the peer instances. The CPMs interoperate in a P2P way, which allows a decentralized management of the public process.

2.2. Suitable Workflow Approaches for the Business Process Management in B2B Relationships

According to the above described, Split and Deploy, composition and subcontracting are not appropriate to manage B2B relationships where the management of public business process has to be decentralized and the business process management systems of the participants have to interact in a P2P way. These approaches are more appropriate to manage business processes in B2B relationships based on e-marketplaces, where an architecture hub-and-spoke is used and the management is centralized. An e-marketplace can be implemented by a third party or by a dominant enterprise that controls the relationships of the partners. In this case, a centralized management allows e-marketplace to keep the visibility of the global processes.

Split and Deploy could be used if all partners use the same WfMS to manage public and private processes. Generally, this is not the case and this approach cannot be used because WfMSs of different vendors cannot interoperate due to the semantics of the control flow are different [13].

Composition does not require that partners use the same WfMS to manage their private processes. However, it requires that partners use the same WfMS to manage the public processes. Only one partner implements the WfMS that manages the public process and the other partners have to implement the components that allow to interoperate with the WfMS.

Subcontracting allows to enterprises have different WfMSs and could be implemented using traditional WfMSs. However, due to the hierarchical relationships among the processes obviously it is not appropriate to manage B2B relationships in a decentralized way using P2P systems.
In contrast, Public and Private allows the use of different WiMSs, although they have to interoperate using a particular communication protocol. This approach meets most of the requirements to implement decentralized B2B relationships. However, as we described in the next sections, the notion of workflows have some disadvantages that can be solved using interaction protocols.


Another approach to manage public business processes is the use of B2B Interaction Protocols. Enterprises are autonomous entities that have to interact to jointly manage public business processes. Therefore, a similitude between B2B enterprise interactions and software agents’ interactions can be made. Hence, we define B2B interaction protocols based on the agent interaction concepts used in the multi-agent systems area.


An interaction protocol describes a high-level communication pattern through an admissible sequence of messages between roles [2]. The main difference between B2B interaction protocol and workflow is that the former does not use the concept of activities and their interdependencies. A B2B interaction protocol is defined by the following elements:

- **Role.** A role represents the responsibility, in terms of a messages sequence, that an enterprise performs in a relationship.
- **Messages.** An interaction protocol specifies messages representing interactions. Messages contain a semantics that defines their type. The message types can represent business information, decision, proposal, acceptance, reject or acknowledgment. A message can be asynchronous or synchronous.
- **Control Flows.** An interaction protocol has two control flows. One represents the control flow of the messages, which defines the parallel or alternative messages of each interaction protocol step. The second represents the internal execution flow of a role that describes the different reactions of the role to the incoming messages.
- **Logical connectors.** To define the control flows an interaction protocol has to use connectors. These logical connectors are AND, XOR, and OR.
- **Deadline.** To represent that a message has to be received before a certain time, deadlines can be defined on messages. In this way, exceptions derived from lost messages or problems in the protocol execution can be handled.
- **Conditions.** They can be defined on messages to represent when a message can be sent.

3.2. B2B Interaction Protocol Modeling Languages

To represent and define B2B interaction protocols, a modeling language is required. Using a tool for specifying B2B interaction protocols in a graphical format enhances the design activity of business process. Therefore, a visual modeling language may be used to define the elements of the interaction protocols. Using again the analogy between B2B enterprise interactions and software agents’ interactions, we can use some proposals for modeling agent interactions from the multi-agent systems area. A proposal that we identified as appropriate to model interaction protocols is AUML (Agent Unified Modeling Language) [2]. In AUML, interactions among agents are modeled in Protocol Diagrams. It allows representing the elements that compose a B2B interaction protocol. Besides, it is based on the UML standard modeling language, which is a widely known language used in software engineering and has been also applied to model business processes. In this way, introducing some adaptations to the AUML protocol diagram, we define the B2B interaction protocol diagrams.

3.2.1. Elements of the B2B Interaction Protocol Diagrams

To accommodate agent interaction protocols to B2B interaction protocols, the form to describe the roles is: Enterprise / role (Figure 1). The role represents the responsibility of the enterprise in the interaction protocol, for example if it is a supplier or a customer. Role lifeline defines the time period during which the enterprise participates in the protocol. Lifelines are represented by dotted vertical lines. The lifeline may split up into two or more lifelines to show AND and OR parallelism and decisions, corresponding to branches on the incoming message flow. Figure 2(a) shows the graphical representation of the logical connectors. Logical connectors are also used to define parallelism and decisions in the message flow as Figure 2(b) shows. A thread of interaction, which is represented by a bar on the lifeline, shows the period during which the enterprise role is performing some private activities or processes to react to an incoming
message or to send a message. An asynchronous message is drawn as \(\rightarrow\). It represents the sending of the message without subordinated control. A synchronous message is drawn as \(\longrightarrow\). It represents that a role sends a message and then waits until an answer message is received and nothing else can be processed. A repetition of messages is represented by an arrow ending in a thread of interaction which is, according to the time axis, before or after the actual time point. Deadlines are represented by comments on the arrows and they indicate that the message has to be sent before a certain time. Also, conditions and guards can be added using brackets to define the condition when a message is sent.

![Figure 1. Roles of the Interaction Protocol](image)

**Figure 1. Roles of the Interaction Protocol**

![Figure 2. Logical Connectors](image)

**Figure 2. Logical Connectors**

### 3.3. An Example of a Public Business Process Model using an B2B Interaction Protocol

The Partner-to-Partner Collaborative Model [20] requires the decentralized management of the public business processes between enterprises and P2P interactions between their information systems. This model has been applied for managing supply chain processes through independent relationships between two enterprises. In each relationship, there exist only two enterprises that have to interact in a partner-to-partner way to execute collaborative business processes. These processes are the public processes.

An example of a public business process model using a B2B Interaction Protocol Diagram is shown in the Figure 3. This figure shows the B2B interaction protocol Collaborative Production and Capacity Planning, which is defined to represent the public process Collaborative Production and Capacity Planning of the Partner-to-Partner Collaborative Model. This public process has two roles, supplier and customer, that are performed by enterprise A and enterprise B respectively. The objective of this public process is that enters A and B agree on a mid-range supply plan of a certain product. Both enterprises are manufacturing enterprises. Hence, the exchanged product is for the supplier a final product and for the customer a raw material.

The process starts when the customer role sends two parallel messages with business information: `inform(SalesData)` and `inform(Mid-Range Requirement Plan)`. Once the supplier receives these messages, its private processes are invoked for generating the internal Aggregated Production Plan and then both plans are compared. The private processes are not defined by the interaction protocol because they are private aspects of the supplier. They occur during the time represented by the thread of interaction. After comparing the plans, the supplier has to decide whether to propose a mid-range supply plan or reject the customer plan because the customer requirements cannot be satisfied. This is defined with a logical connector XOR (exclusive or), which represents that only one of the two messages can be sent. This decision has to be done before the time indicated by a deadline, which is equal to two days after the date of the last received message.

The customer has two threads of interaction that represent the incoming messages. If the supplier proposes a plan, the customer executes its private processes and then the supply plan can be accepted, rejected, or an alternative plan can be
proposed by the customer. The last two alternative messages represent a repetition of messages. This means that the supplier again executes its private processes to determine whether to propose a plan or reject the customer plan. In the other case when the customer receives a message \textit{refuse (Unfeasible Mid-range Requirement plan)}, an alternative requirement plan can be proposed or the process can be ended without to achieve a consensus. In both cases, the customer has to respond before a deadline.

Finally, when the customer accepts the supply plan of the supplier, a message \textit{inform (End of Negotiation - Agreed Mid-Range Supply Plan)} is sent by the supplier to end the public process. This indicates that the process finishes and both partners have achieved a consensus in the \textit{Mid-Range Supply Plan}.

\begin{center}
\includegraphics[width=\textwidth]{B2B_interaction_protocol.png}
\end{center}

\textbf{Figure 3. B2B Interaction Protocol: Collaborative Production and Capacity Planning}


To execute B2B interaction protocols, enterprises have to use an Interaction Protocol Management System (IPMS) that allows the definition, execution, control and monitoring of the B2B interaction protocols. Each enterprise should execute an IPMS that interprets B2B interaction protocol definitions and executes the role that the enterprise has to perform in the B2B interaction protocols agreed with its partners. Figure 4 shows the main functional components that have to be included in an architecture of an IPMS. These components are described below.
Figure 4. Generic Architecture of an Interaction Protocol Management System (IPMS)

4.1. Definition and Analysis Tool for B2B Interaction Protocols

A functionality of the IPMS is the definition of interaction protocols. There are two requirements to support this definition. One requirement is the necessity of using languages to model B2B interaction protocols. Two types of languages are required: a graphical modeling language and a textual modeling language. The former has the objective of providing an intuitive semantics that allows business process designers to define and understand a public business process. In the above section, we described a graphical notation based on AUML protocol diagrams that is appropriate to represent public business processes through B2B interaction protocols.

The second language has the objective of supporting the exchange of interaction protocols definitions between enterprises. These definitions can be understood by the IPMSs of each enterprise. In addition, during the specification of the collaborative agreement, enterprises could compare interaction protocol definitions to determine if a matching between them exists. The AUML Protocol Diagram does not provide a textual language to represent interaction protocols. There is a proposal of a textual modeling language in AUML [11] to represent protocol diagrams, which should be studied to apply it to the B2B interaction protocol. In this paper, we do not focus on the definition of a textual modeling language for B2B interaction protocols.

Another requirement to define interaction protocols is the use of verification and validation mechanisms. To do that, a textual modeling language is also necessary. A possible approach is to derive an activity diagram for each role. Then this diagram can be verified and validated using approaches proposed for workflows. Furthermore, an analysis tool could be used to simulate the execution of interaction protocols.

4.2. B2B Interaction Protocol Manager

The B2B Interaction Protocol Manager (B2B IPM) executes, controls and monitors the role that enterprise performs in the B2B interaction protocols. This component uses a B2B Interaction Protocol Engine that interprets textual definitions of the interaction protocols and manages the instantiation of them. Interaction protocol definitions are retrieved from the Collaborative Agreements database. The B2B IPM controls the execution states of each interaction protocol instance keeping control information of each instance. When an interaction protocol instance requires to send a message, the B2B IPM previously has to invoke private WMSs or ERP systems through the Process Integration Manager. WMSs or ERP systems execute the private business process of the enterprise that generates or consumes the information of the interaction protocol messages. Once the private workflow was executed, the B2B Interaction Protocol Manager invokes the B2B Message Engine. It dispatches the messages on the network according to a B2B standard agreed with the partners. In the other case, when a message is received by the B2B Message Engine, it notifies to the B2B IPM, which determines if the message is waited by an interaction protocol instance. If it is right, through the Process Integration Manager, the B2B IPM invokes the private business process that requires the business document of the message.

To exchange messages of interaction protocols, the IPMSs should not use a proprietary low-level communication protocol because it requires that enterprises have to use the same IPMS. Therefore, enterprises should exchange these messages using a B2B standard. These standards have to be agreed between the partners and included in the collaborative agreement definition. In this way, a B2B interaction protocol can be used with two or more partners using different B2B standards according to they have agreed.

There are a large number of B2B standards that can be applied in different layers of a B2B framework [16]. Some of them can be only applied in the content layer (e.g., eCO, cXML) and other can be applied to all layers (e.g., ebXML, RosettaNet). These standards are also known as B2B protocol standards [5]. The term “B2B protocol” does not have the same mean that B2B Interaction Protocols proposed in this paper. These B2B standards can consist of the specification of the following elements [5]:

- **Public process definition**, which states the order in which the messages can be received or sent.
- **Message**, which contains documents and header information, like the sender, the recipient and the information describing the message type.
- **Exchange sequence**, which defines constraints on the messages, like time-outs, performance and if an acknowledgment has to be returned.
- **Document types**, which define the format of the documents involved in a message.
- **Document semantics**, which defines the legal values of a document element.
- **Packaging**, which defines how a message is packaged with header information specific.
- **Security**, which defines how a message has to be encrypted and the mechanism to be used for authentication.
- **Transport binding**, which defines how a message to be sent is encoded using a network protocol like HTTP/S or FTP.

Some B2B standards, like RosettaNet [18] and ebXML [9], define most of the above elements. RosettaNet does not completely define the message sequence that models the public process. Instead, RosettaNet provides Partner Interface Processes (PIPs), which are interfaces that represent an interaction point (a message) within a public process, although some PIPs require that when a message is sent, another message has to be received. Hence, RosettaNet PIPs define all the elements of a B2B standard, except a complete public process definition. In this way, the messages of a B2B interaction protocol can be implemented with RosettaNet. This means that each message of a B2B interaction protocol has to be exchanged according to the RosettaNet PIP standard specification.

In the same way, it is possible to use B2B interaction protocols with ebXML. ebXML allows to define the message sequence that models the public process. A graphical interaction protocol model can be translated in a textual model based on the ebXML standard to define public processes. If the partners agree on other B2B standards that do not define all the above elements, the partners have to jointly define these elements.

Therefore, the B2B message engine carries out all the actions required to exchange messages across the network. It has a manager for each element of the B2B standard, except for the public process element, which is managed by the B2B IPM using B2B interaction protocols.

4.4. Process Integration Manager

This component provides the mechanisms used by the enterprise to integrate public processes with private processes. In addition, this component uses APIs required to invoke private applications of the enterprise.

4.5. Collaborative Agreements Database

This component is a database that stores all the information about the collaborative agreements of the enterprise with its partners. It contains the textual definitions of the interaction protocols that the enterprise has agreed with its partners and also it maintains information about the B2B standards agreed to exchange messages.
4.6. Private WfMSs and ERP systems

Private WfMSs and ERP systems execute private processes that generate and require business information that the enterprise sends to or receives from the partners. This systems are external to the IPMS and are invoked by it when a message of a B2B interaction protocol instance has to be sent or is received.


The B2B interaction protocol approach presents several advantages that can be highlighted with respect to workflows approaches for the management of public business processes in B2B relationships.

Using a B2B interaction protocol approach, enterprises can jointly manage B2B interaction protocols without the use of the same IPMS type. This means that IPMSs can be independent and be built in different ways using different information system technology. The only requirement of the IPMSs is to use the same language to define B2B interaction protocols. In this way, a B2B interaction protocol can be interpreted by all IPMSs.

IPMSs can be independent because they interoperate using a B2B standard. In those cases where the B2B standard is not complete, they can implement an extension of a B2B standard in a particular way. In addition, IPMSs does not require to transfer instances because each IPMS knows the message that has to send or receive in each execution step according to the definition of the B2B interaction protocol. Instead, workflow approaches require that each enterprise uses the same WfMS. This is because different WfMSs cannot interoperate since each WfMS uses different semantics and structures to define the control flow of the workflow [13]. Therefore, workflow instances cannot be transferred from a WfMS to another one. A proposal to solve this problem with workflows is to use a proprietary communication protocol to synchronize instances among WfMSs [7]. However, this communication protocol is a low level protocol that has to be implemented by WfMSs of the enterprises.

Workflow approaches require enterprises to expose services or activities to their partners and also require defining the interdependencies and the control flow of activities. Although activities of public processes are abstract, they represent functionality that enterprise expose to its partners. B2B relationships require a mechanism to represent interactions among enterprises without representing the behavior and functionality that support the interactions. Therefore, B2B interaction protocol approach provides a greater autonomy because it does not require representing these aspects. In a B2B interaction protocol, the partners only know the messages that have to be exchanged in each execution stage of the interaction.

Collaborative models require representing negotiation in the public processes of a B2B relationship. B2B interaction protocols allow representing negotiation using a message semantics that expresses the purpose of the message. A message represents an intention, like acceptation, reject or proposal. In workflow approaches, these intentions can also be represented but it can be made using activities, which should not be showed in the public processes.

Finally, B2B interactions do not require representing the five perspectives that a workflow model has to support: functional, behavioral, informational, operational and organizational [12]. Functional perspective specifies the activities of the workflow. Behavioral perspective specifies the control flow of the workflow activities. Informational perspective specifies the information required in the execution of each activity and control rule. Operational perspective specifies the applications that perform each activity. Organizational perspective specifies who perform what activity and with what tools, using a definition of organizational roles. All these perspectives are important to support private processes but not to support public processes or interactions among enterprises. The organizational perspective is not significant in B2B relationships where the collaboration is partner-to-partner because of these relationships involve two enterprises. The operational perspective should not be supported in public processes. The knowledge of the private applications, databases and WfMS (those managing the private processes) that an enterprise uses to support public processes is not appropriate. The knowledge of these private systems by the enterprise partners reduces the enterprise autonomy. This perspective has to be supported by enterprises in a private way and not visible to its partners. In addition, every time activities are represented in a public process, a tight coupling among the partner systems is created. Therefore, the functional perspective is not required either in B2B interactions. Instead, a B2B interaction protocol approach includes only the required perspectives of a B2B interaction. A behavioral perspective is supported through the definition of a message control flow and an internal control flow of the roles. The informational perspective also is included through the definition of the message content. The organizational perspective is also included since B2B interaction protocol specifies the two roles involved in the B2B relationship.
6. Conclusions

Collaborative business models supported by B2B relationships impose four important requirements for the management of interactions between enterprises in the business process layer. These requirements are: autonomy, decentralization, peer-to-peer interaction and negotiation. They have to be supported by the information systems that manage the business processes involved in the B2B relationships.

Toward that direction, in this paper we have presented a B2B Interaction Protocol approach to model and manage interactions between enterprises. These interactions represent the public business processes that partners agree to achieve a goal. The B2B interaction protocol approach makes possible to model interactions between enterprises in a high-level abstraction. Therefore, for the modeling of the B2B interaction protocols we have described a graphical language based on the AUML diagram protocols. This language provides intuitive semantics to enhance the design activity and can represent all the elements of the B2B interaction protocols. Furthermore, we have proposed a generic architecture of an Interaction Protocol Management System with the required components to support the management of the B2B interaction protocols.

Several workflows approaches have been discussed. We have identified that split and deploy, composition and subcontracting are workflow approaches that are not suitable to manage public business processes in B2B relationships that support collaborative models. Public and Private is a workflow approach that meets some requirements imposed by the collaborative models. However, workflow approaches present shortcomings to meet these requirements as we have identified in the comparison and evaluation of the workflow approaches and the B2B interaction protocol approach. Instead, the B2B interaction protocol approach achieves all the requirements of the collaborative models.

Autonomy is fulfilled by B2B interaction protocols in several aspects. They do not model the functional and operational perspectives that are not required in this type of B2B relationships. Hence, they hide the internal activities, processes and the information systems that each enterprise uses to manage both private and public processes. Beside, according the architecture of an IPMS, partners can use different implementations of the IPMSs because they exchange messages across the network using a B2B standard. So, IPMSs does not require proprietary low-level communication protocols to interact between them and the partners can use independent IPMSs. Also, the architecture of an IPMS assures that the B2B interaction protocols are independent of the B2B standards. This allows enterprises may define the same B2B interaction protocols with some partners but using different B2B standards to exchange messages. All these features increase the autonomy of the enterprises and their information systems.

Decentralization is achieved because B2B interaction protocols require that enterprises manage in a separate and independent way the role that they are responsible in the interaction protocol. In this way, partners use independent IPMSs that execute their roles and manage in a decentralized way the B2B interaction protocol.

The peer-to-peer interaction between the information systems of the enterprises is a natural approach using B2B interaction protocols. It is not suitable to implement a B2B interaction protocol using a third party because the definition of the B2B interaction protocols only allows the representation of the roles involved in the B2B relationships.

Negotiation can be improved by the B2B interaction protocols because they allow enterprises define not only the parallel and alternative messages but also the intentions of the enterprises. Messages can represent acceptations, rejects, proposals, business information, decisions and acknowledgments. All these intentions are required to represent negotiations in public business processes.

Finally, B2B interaction protocols can be used as a contract between partners for defining collaborative agreements in the business process layer. We consider that the time spent in the design of collaborative agreements can be reduced using a B2B interaction protocol approach.

Future work consists of:

- Expand the elements of the B2B interaction protocol diagrams to support other features as exceptions and protocol dependence to model graphically B2B interaction protocols.
- Define the mechanisms to integrate B2B interaction protocols with private processes.

References


