Development of Innovative Services Enhancing Interoperability in Cross-Organizational Business Processes

Stefan Huber, Cyril Carrez, Hannes Suttner

HAL Id: hal-01572104
https://inria.hal.science/hal-01572104

Submitted on 4 Aug 2017

Development of Innovative Services Enhancing Interoperability in Cross-organizational Business Processes

Stefan Huber[1], Cyril Carrez[2], Hannes Suttner[1]

[1] Siemens AG, Vienna, Austria
[2] SINTEF ICT, Oslo,Norway
{stefan.a.huber, hannes.suttner}@siemens.com
Cyril.Carrez@sintef.no

Abstract. This paper presents the vision and ongoing results of the COIN (FP7-216256) European project for the development of collaborative and cross-organizational business process interoperability. Services for selectively publishing parts of private business processes are developed which enable the creation of a cross-organizational business process (CBP). Gap detection services provide analysis of the CBP with regards to interoperability gaps like potential deadlocks.

1 Introduction

COIN (FP7-216256) is an integrated project in the European Commission Seventh Framework Program. According the COIN vision [1] by 2020 enterprise collaboration and interoperability services will become an invisible, pervasive and self-adaptive knowledge and business utility at disposal of the European networked enterprises from any industrial sector and domain in order to rapidly set-up, efficiently manage and effectively operate different forms of business collaborations, from the most traditional supply chains to the most advanced and dynamic business ecosystems.

The mission of the Coin IP is to study, design, develop and prototype an open, self-adaptive, generic ICT integrated solution to support the above 2020 vision. COIN is starting from existing research results in the field of Enterprise Collaboration and Enterprise Interoperability.

In this paper we present the vision and ongoing results for the definition and prototypic development of cross-organizational business process interoperability (CBPip) services in the course of the COIN project. The main objective of these services is to ensure a successful business process collaboration of participating enterprises. We concentrate on two main streams: rule-based transformation of private business processes into public processes, and detection of business process interoperability problems at design-time level. The transformation helps an enterprise to extract from its

private processes the parts that can be of interest when collaborating with a partner. The detection of interoperability problems helps the enterprises to design the cross-organizational business process and detect some problems at design time. Our previous work [4] is extended by providing more flexible transformations using business rules, and by analyzing the CBP for interoperability gaps.

This paper is structured as follows. The general approach of the CBPip services is depicted in section 2. The paper then presents the two services developed in this context: a service providing the private-to-public transformation of the involved business processes (section 3), and a verification service for finding gaps in the cross-organizational business process (section 4). Section 5 describes related work. Conclusions and future work are presented in section 6.

2 General Approach

In the scope of COIN it is important to distinguish "collaboration" from "interoperability".

According to COIN [1], *Enterprise Collaboration* comes from a business perspective and identifies the process of enterprises - mainly SMEs - to set-up and manage cross-enterprise win-win business relations in response to business opportunities. *Enterprise Interoperability* originates by the ICT world and identifies a capability of enterprise software and applications to exchange information and to mutually understand the information exchanged at the level of data, applications, processes and enterprise models involved. Enterprise Interoperability (EI) services provide functionality for applying IT solutions that overcome interoperability problems between two or more enterprises, and thus enabling them to set-up and run collaborations.

In this paper, we define *business interoperability* as the capability of two or more systems to cooperate using exchanged information, and an *interoperability gap* is a situation when interoperating business processes do not deliver the expected results while each of the business processes does. To be able to cooperate, a network of organizations needs to first define a Cross-Organizational Business Process and analyze it to find interoperability gaps.

Whereas other COIN services deal on collaboration issues, the services described in this paper are focused on interoperability issues only.

2.1 Creating a Cross Organizational Business Process (CBP)

Before showing how to create a CBP we first define some terms:

Process:
   A Process is any activity performed within a company or organization.
Cross-Organizational Business Process (CBP):
   A CBP describes a process involving two or more organizations.
Private Process:
   A Private Process is an internal process of a partner, and its functionality is invisible to any external entities. Making elements of a private process visible is the precondition of its "interoperability".

Public Process (View):

A Public Process is a published process. It is a specific view on a Private Process of a partner. Via the Public Process, the collaborating partners may invoke specific functionality of the corresponding internal process.

Private2Public transformation

A Private2Public transformation is a model-to-model transformation, which generates public views for specific partners. It is based on Visibility Rules.

Visibility Rules:

Visibility rules are business rules that define which process elements are shown to which partner. They are defined in SBVR (Semantics of Business Vocabulary and Business Rules [5]).
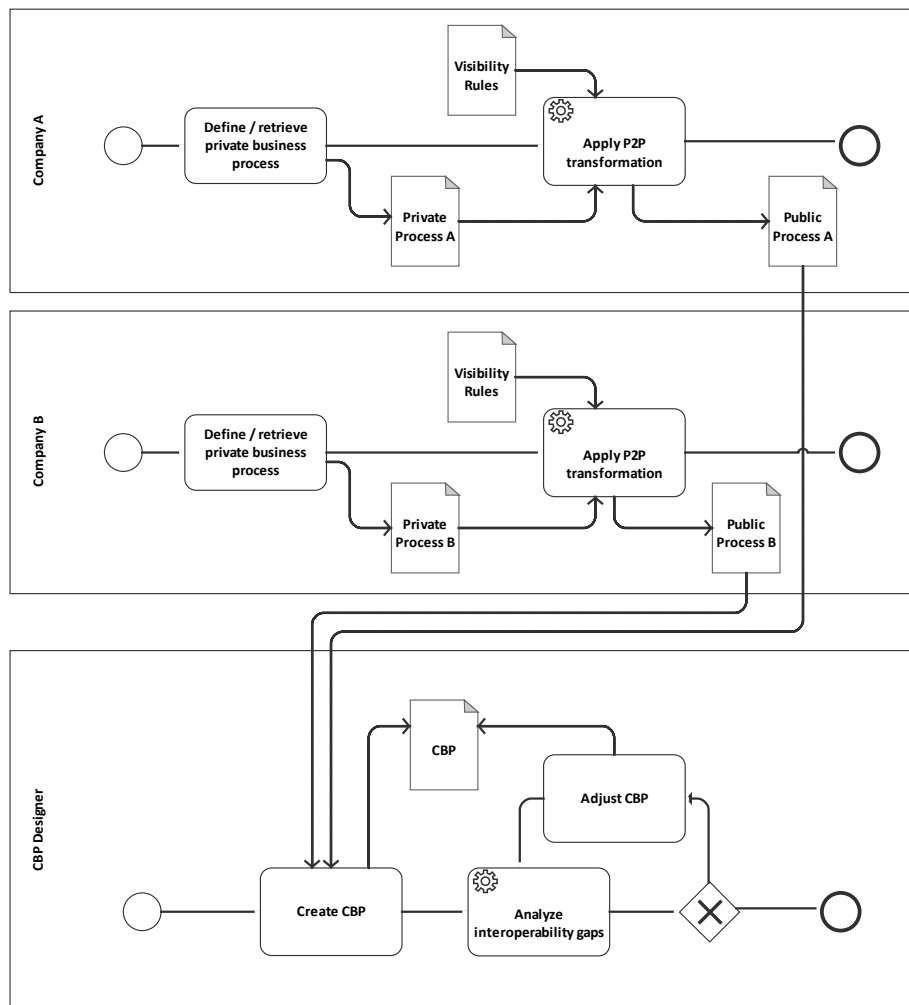
Fig. 1. Creating and Analyzing the Cross-Organizational Business Process (CBP).

Figure 1 shows the workflow for creating the CBP. The starting point is the business process model of a specific company, e.g. "Private Process A" for company A. The company A has already defined a set of "Visibility Rules" which specify the visibility rights to different elements of Private Process A, applied to different kinds of partners. The rules are applied to Private Process A for the company B, hence generating the Public Process A that can be used to model the CBP with company B. Similary, the Public Process B is generated. Under these prerequisites, the CBP can be modelled by using elements of the public processes of the involved companies.

2.2 Analyzing a CBP

Once the CBP has been specified, it needs to be analyzed for interoperability gaps.

A *Cross-Organizational Business Process Interoperability Gap* (CBPip gap) is a condition when two or more collaborating public processes don't deliver the expected business results, although they may work faultlessly under other conditions. In other words, an existing CBPip gap is a condition through which the CBP workflow process execution leads to unsuccessful business collaboration. Gaps can be deadlocks (i.e. the CBPip process possibly will not terminate) or interface mismatch (i.e. the formats of a sent document does not match the format being expected by the receiving organization).

In this paper, we address the potential business process collaboration problems at a technical level, and recognize them as CBPip gaps. Once the CBP is modelled, the CBPip gap detection is invoked and parses the CBP model to detect the CBPip gap patterns. The gap patterns are conceptually defined in classes and types. On detection, the known reason or an assumption of the CBPip gap will be displayed.

The following sections describe the Private2Public transformation and CBPip gaps.

3 Private-to-Public Transformation

This section presents the transformation from Private Processes to Public Processes and how Visibility Rules are used in such a transformation.

The aim of the Private2Public transformation is to produce a public process of a private process of a given company. This public process will hide all the private and critical data of the private process, resulting in a simplified process. Instead of annotating all activities and data with a public/private tag, we aim for the use of a business rule language in order to describe the visibility of different process elements of a private process. We chose SBVR (Semantics of Business Vocabulary and Business Rules [5]) for describing visibility rules. As a standard, SBVR offers a strong basis; it also allows a representation of rules in Structured English, and hence is more readable and usable for Business Analysts who will produce Visibility Rules. The main drawback of SBVR is the complexity of its meta-model and the production of a suitable vocabulary; however those drawbacks are taken care of by the technology experts who will develop tools to handle the complexity of the meta-model (end-users will

see only the SBVR rules in Structured English) and who will create the vocabulary with the help of business experts.

3.1 Principles of the Private to Public Transformation

The SBVR for CBP allows describing how a public process should be produced from a private process, by way of business rules. This is achieved through three principles (Figure 2):

- The **SBVR rules are linked to specific elements of Processes**, so the transformation service knows which elements are described in the rules. This is done by two ways: First, the elements in a Process (Activities, Data, …) can be annotated with an SBVR vocabulary which describes the purpose of those elements. However, as this may clutter the diagrams, a second possibility is to use the names of the elements in a Process (e.g.: an activity involving some Invoice contains the string "invoice" in its name).
- **A Vocabulary (in SBVR) which describes the semantic of Process elements, as well as the roles of partners in a CBP**. The vocabulary defines noun-concepts related to the modelling of Process elements (Activity, Data) or to their semantics (Sales Department, Customer, etc.), as well as verb-concepts to specify associations ("an activity *manipulates* some invoice"[1]). Notably a verb-concept "*is visible to*" associates elements in Processes with partners in the CBP.
- **Business rules which specify what should be public**. By default, nothing is shown to partners. The rules are then based on the RBAC principle (Rule-Based Access Control [6]). The business rules are based on the verb-concept "*is visible to*", and tell which element in the process should be visible to which kind of partner. For instance, "An activity that *is about* management *is visible to* some strategy department".

---

[1] In this paper we show the Visibility Rules in Structured English, with four font styles for term, Name, *verb* and keyword, as documented in the SBVR specification [5, pp237-238].
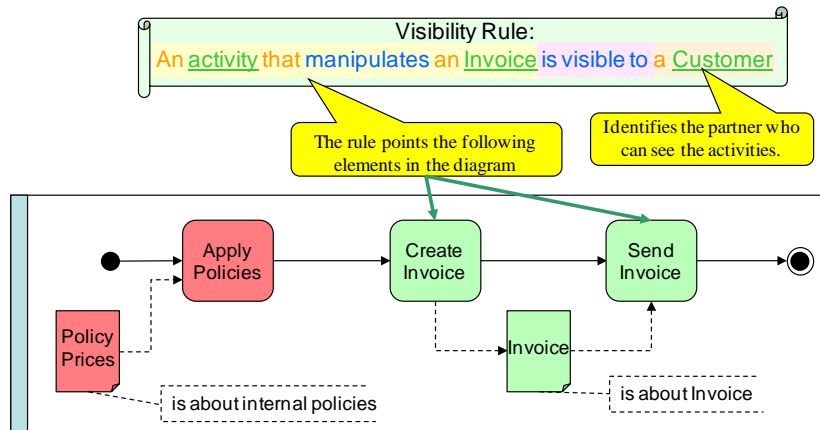
Fig. 2. Principles of visibility rules.

Those principles are illustrated in Figure 2. The private process applies pricing policies in order to create and then send an invoice. The last two activities handle some data which is annotated with the SBVR expression "*is about* Invoice". A Visibility Rule can then use this information and specify: "An activity that *manipulates* an Invoice *is visible to* a Customer", which means that the activities "Create Invoice" and "Send Invoice", which send and receive some data that is about Invoice, are concerned by this rule and will be made visible to a Customer. The other activity "Apply Policies" and the critical data "Policy Prices" will be removed in the public view.

3.2 SBVR Vocabulary and rules

In order to build the SBVR Vocabulary, we studied the COIN use case scenarios of healthcare and aeronautics clusters, and business examples which came along with the Enhydra Shark process execution environment[2]. This resulted in a vocabulary summarized in Figure 3. Facts and concepts related to process models identify abstract concepts that represent elements in BPMN, as well as the most important fact-types for visibility rules, "*is visible to*". Those terms are specialized with concepts related to roles, activities, control flow and data.

---

[2] http://www.together.at/prod/workflow/tws

Fact and concepts related to the process models

process element
activity
    General Concept: process element
data
    General Concept: process element
control flow
    General Concept: process element
role

processElement *is visible to* role

Fact and concepts related to the activity & its behaviour

activity behaviour
validation, management
    General Concept: activity behaviour
datakind

activity *is about* activityBehaviour

activity *creates* datakind
activity *checks* dataKind
activity *manipulates* datakind

Fact and concepts related to the roles (will view the public process)

department, supplier, customer
    General Concept: role
sales department, strategy department, …
    General Concept: department

Fact and concepts related to the control flow

public activity
    General Concept: activity
control flow *follows* public activity
control flow *leads to* public activity

Fact and concepts related to the data

data *is about* datakind
product, shipment, critical_data, exchanged document, …
    General Concept: data kind
expense report, invoice , …   travel expenses
    General Concept: exchanged document   General Concept: expense report
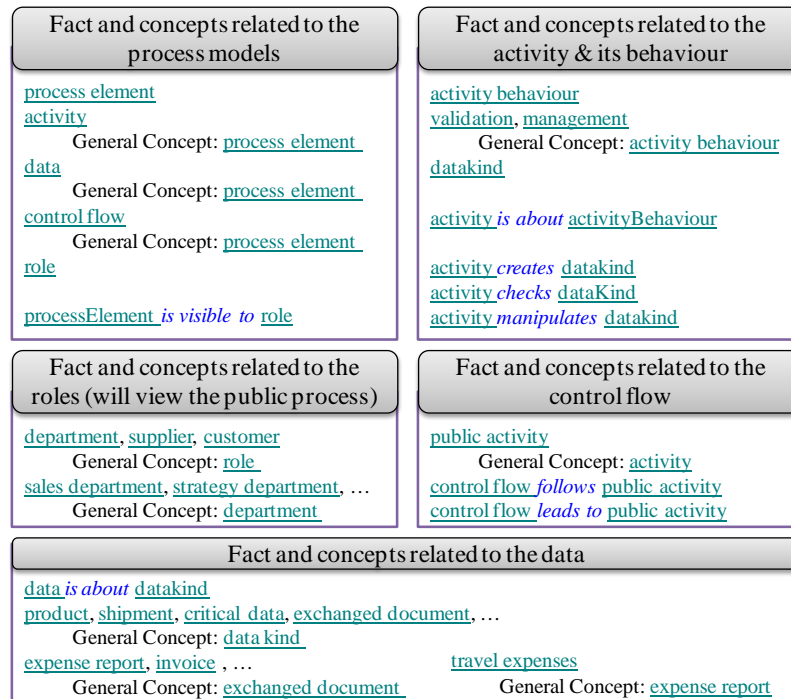
Fig. 3. Fact types and main concepts in the CBP Vocabulary.

3.3 Visibility rules

Visibility Rules specify which elements in the model of the process are to be shown in the public view. They are specified as *Advices of Possibilty* in SBVR[3], and are based on the fact type "*is visible to*", which relates a Process Element with some Role. Visibility rules have two parts: the first one introduces the Process Element (Activity, Data, ControlFlow) in the private process; this Process Element is further characterized ("Activity *is about* management") in order to address a specific element in the private process. The second part of the rule specifies who is able to see the Process Element at stake ("*is visible to* some Customer").

   Here are some examples of Visibility Rules:

   It is possible that an activity that *is about* validation *is visible to* some strategy department.

   It is possible that an activity that *manipulates* some feedback *is visible to* some customer.

---

3 SBVR specifies *Rules*, which remove some freedom, and *Advices* which add some freedom. The transformation is such that everything is private (removes all freedom), and the Visibility Rule tells what should be shown (add some freedom), the choice of *Advice* was preferred.

It is possible that a data that *is about* some reclamation *is visible to* some sales department.

It is possible that a sequence flow that *follows* a public activity *is visible to* some role.

3.4 Principles of the Transformation

The principle of the transformation is shown in Figure 4: each public activity will be put in its own workflow, to allow an easy composition of the public views in the CBP. This is illustrated in the top part of the figure, where the activities "Request Invoice" and "Send Invoice" are public[4], and the private process is transformed into two Workflows, one for each activity. The lower part of the figure presents a special case in the transformation. In case the private process presents a sequence of public activities, then those activities are put in the same workflow. The sequence must be linear, meaning that no gateways (fork / join) between the activities are permitted, and only one transition is allowed between the activities. For instance, if "Order" and "Request Invoice" are public activities, they will be put in a single workflow, as shown in the figure.
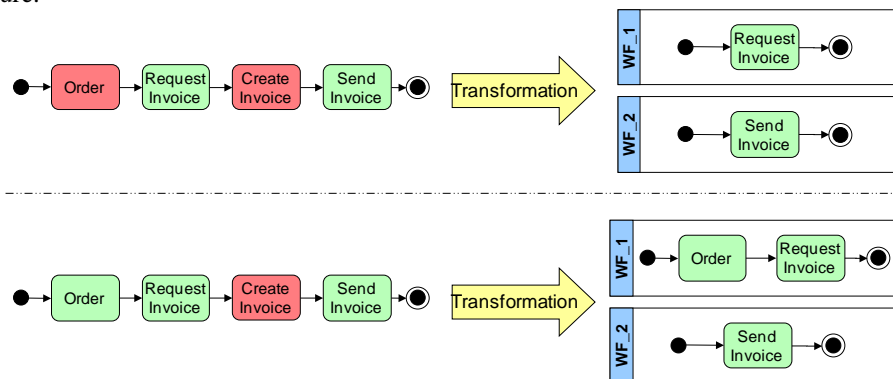

Fig. 4. Results of the transformation.

4 Interoperability Gap Detection

Once the CBP has been created by using elements of the Public Processes of the involved organizations, the CBP can be analyzed for interoperability gaps.

Business process collaboration is seen as given, i.e. already agreed business collaboration between the partners of a specific CBP is a pre-condition for the CBPip. Although gaps on business collaboration level are not explicit subject of our consid-

---

[4] For the sake of clarity we removed the Visibility Rules and colored the activities instead (red/dark grey=private, green/light grey=public).

erations, we allocate them a unique gap class. CBPip considers gaps during design-time. Gaps at execution time (run time) are not considered in this paper; those gaps may arise during the CBP model execution as a result of common IP based network connectivity problems. Note that design-time is a phase where the process will be considered during its specification, e.g. during modelling in a business process modelling tool. Contrarily, run-time is a phase where a process or the automated parts of a process will be considered during its execution on a process execution engine.

We first give a short example of a CBPip Gap before describing the classification of those gaps. The detection service is presented in the end.

4.1 Description of an Exemplary CBPip Gap

As an introductory example a company A models a private business process $BP_{Travel\ reporting}$. Company A publishes activity "Create expense report" as a public process $VP_{Report\ Creation}$ (see upper left part of Figure 5). Company B does very much the like and publishes $VP_{Report\ Confirmation}$ (see upper right part of Figure 5). Both public processes are then used as parts of a Cross-organizational Business Process $CBP_{Travelmanagement}$ (grey activities in the bottom part of Figure 5).

Each company uses its own set of rules to transform (parts of) the private business process into public processes. In the example above, the transformation of activity "Create expense report" to $VP_{Report\ creation}$ works correctly. At company B the used rule-set implies that condition 2, which is attached to the activity "Confirm expenses", will not be published (take notice of the yellow note and the red highlighted arrow in Figure 5).

This transformation is also valid, but in the composed $CBP_{Travelmanagement}$ the missing condition might lead to a multiple source deadlock: when the upper part of the $CBP_{Travelmanagement}$ starts and follows the activity workflow, it might happen that this stream finishes before a "Send Report" message (from the activity $VP_{Report\ creation}$ in the lower part of the $CBP_{Travelmanagement}$) arrives at activity $VP_{Report\ confirmation}$. As a consequence this results in the following deadlock situation: when the lower part of the $CBP_{Travelmanagement}$ starts and reaches the activity $VP_{Report\ creation}$ the process will – due to the condition 1 "Do not continue before expenses are confirmed" – hold and wait until a "Confirm Report" message arrives. Because of the fact that the upper part of the $CBP_{Travelmanagement}$ has already finished, this waiting position will never be left and the final result is a multiple source deadlock, i.e. parts of the $CBP_{Travelmanagement}$ will never be executed (dashed activity in the $CBP_{Travelmanagement}$), and the process itself will starve and never finish.
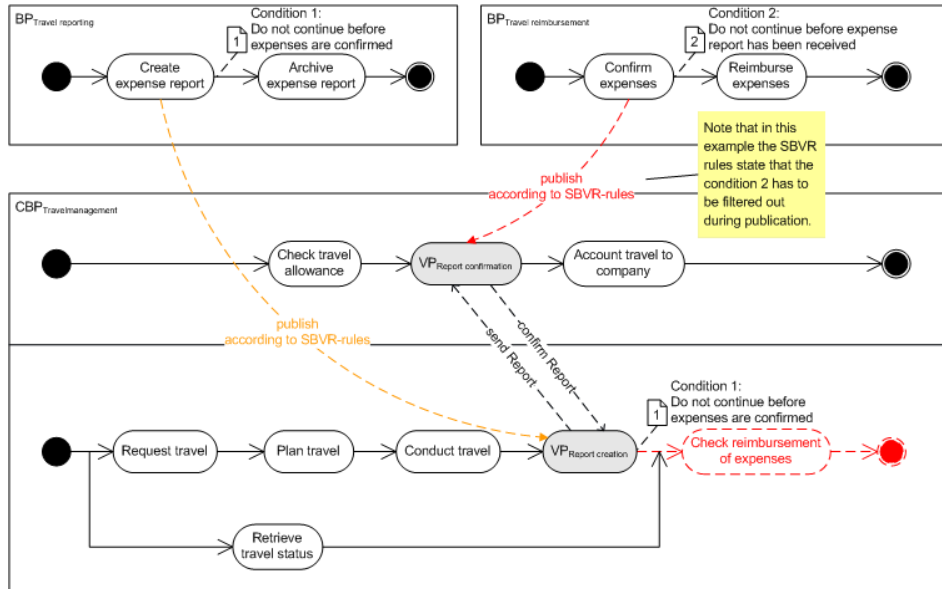
Fig. 5. Example: Erroneous Transformation leading to an Interoperability Gap
(Multiple Source Deadlock).

The example above illustrates that although the transformation of each private process works correctly, interoperability gaps in a resulting CBP might occur. The challenge of scenarios as such lies in detecting interoperability gaps between partners and improving the support to bridge the gaps.

4.2 Classification of CBPip Gaps

The gap classes that have been identified so far on Business Level are:
- Deadlocks
- Interface Mismatches

Although this list is far from being exhaustive, the classification of CBPip-Gaps on Business Level, as well as the first set of identified CBPip Gap class types is depicted in Figure 6. Each class is further detailed hereafter.
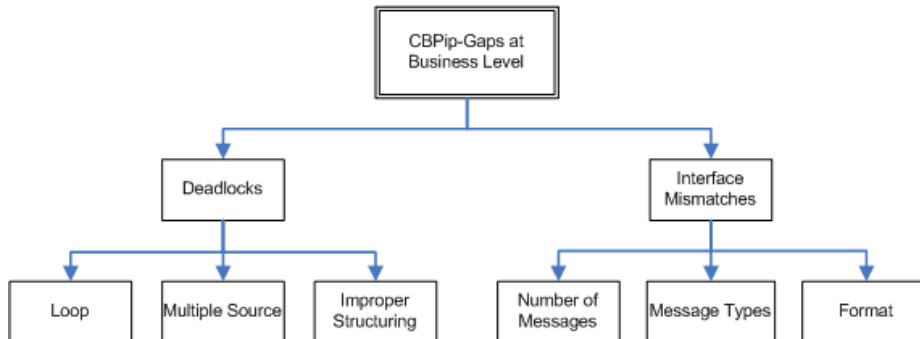
Fig. 6. Classification of CBPip Gaps.

Deadlocks

As stated in [7]: *A deadlock in a process model is given if a certain instance of the model (but not necessarily all) cannot continue working, while it has not yet reached its end.* The research by [8] also points out that CBPip deadlocks can be clustered in structural types such as:
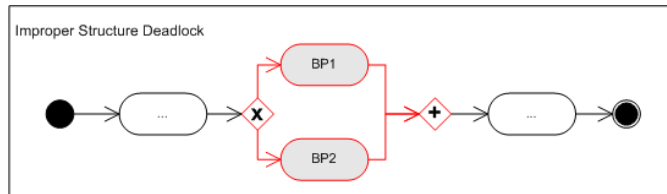
- Loop Deadlock
- Multiple Source Deadlock and
- Improper Structuring Deadlock

A **Loop Deadlock** occurs when there is an execution path from the output of an AND join back to a subset of its input points. If this path contains an XOR-split, deadlock occurs only when the branch leading to the loop is chosen. In case there is a path that does not contain XOR-splits deadlock occurrence is certain. See Figure 8 for an example.

A **Multiple Source Deadlock** occurs when an AND-join has input points which are at some point in the process up-stream originate from two different sources. Assuming that none of the source nodes is the AND-Join itself, we can see that the multiple-source pattern can occur (distinctly from other pattern) only when the process structure is one of the following:

- One of the two sources is an XOR-split.
- The process has multiple start points that are later on synchronized. In case of models specified in BPMN, multiple starts are permissible. Actually, multiple start points resemble an AND-split between the start events, hence we can deduce that there is reachability between two or more sources (start events) to the AND-join node. See Figure 8 for an example.

An **Improper Structuring Deadlock** occurs, when an AND-join receives input that early started from an XOR-split. See Figure 7 for an example.

Fig. 7. Exemplary Improper Structure Deadlock at cross-organizational business process CBPc integrating public processes BP1 and BP2.
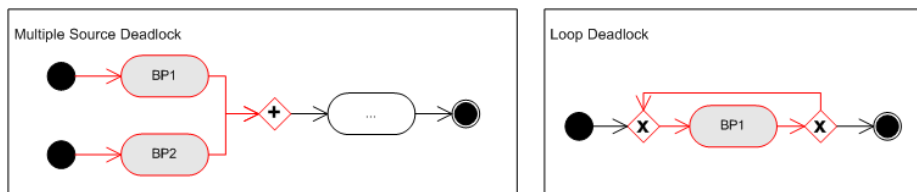
Fig. 8. Exemplary Multiple Source Deadlock and Loop Deadlock.

Detection of deadlocks is done via path analysis considering flows and gateways in the BPMN model. [7] proposes queries to detect such deadlocks. Because of the prototypic nature of the gap detection service a straightforward approach based on those queries was chosen.

Interface Mismatches

An interface mismatch in a CBP model is a serious impediment that prevents two separately-modelled business processes of successful interoperation due to different design assumptions.

A main difficulty of interoperable CBPs interacting with one another in ways not necessarily foreseen is to have interfaces that are generic enough so that they ensure easy and dynamic CBP interface adaption. It addresses a sub-problem of CBP mediation that arises when the interface that a CBP provides does not match the interface that it is expected to provide in a given interaction.

CBPip interface mismatches can be clustered in structural types such as:

- Number of Messages Interface Mismatch and
- Message Types Interface Mismatch

A **Number of Messages Mismatch** occurs when the number of messages provided differs from the number of messages expected. This gap applies to both directions:

- The business process provider provides a number of messages – the business process consumer expects a different amount of messages.
- The business process consumer provides a number of messages – the business process provider expects a different amount of messages.

A **Message Types Mismatch** occurs when the message format provided differs from the message format expected (e.g. the business process provider provides an e-mail – the business process consumer expects an SMS). This gap again applies to the consumption as well as to the provision scenario, like in the prior gap.

A **Format Mismatch** occurs when a message flow involves exchange of data or documents across organizations and the sending organization uses a different format than the receiving organization expects for this message flow.

4.3 CBPip Gap Detection Service

In general the process in providing innovative services that allow to bridge interoperability gaps falls into two steps: The first is to detect the CBPip Gaps, while the second is to eliminate CBPip Gaps – or at least to propose primitive courses of action to surmount the CBPip Gaps. Since each gap class has its own characteristics it was decided to follow a step by step strategy and start with the investigation on the basis of CBPip Gaps of type Deadlock.

The principal idea to detect deadlocks is to analyze the xPDL representation of a CBP model and to try to detect the deadlocks within the model. The approach described in [8] is adapted by COIN providing a web service relying on a set of so called deadlock patterns whose occurrence in process models usually leads to deadlocks. The WebService takes as request parameters the xPDL representation of a CBP model, and the name of the workflow to be verified. It delivers the number of detected deadlocks in the workflow with a short description for each deadlock as well as a suggestion for the deadlock elimination. The suggestion represents a primitive solution to bridge each interoperability gap.

5 Related Work

As a predecessor of COIN, the ATHENA IP Project [2] proposed a CBP framework, with three levels of modelling: private, public ("view") and abstract process modelling. Public and private processes are linked together at companies' sites, and are executable in a BPEL engine, in order to achieve collaborative processes between enterprises. Finally, the abstract process is an abstract view of the public process with only input/output operations. Abstract processes are not executable (or not meant to be), and are composed in the CBP. The CBP is also not executable, but can be used to monitor the choreography of public processes.

The business rule based private-to-public transformation is a very significant improvement against the private-to-public transformation concept as described in ATHENA. We are using SBVR (Semantic Business Vocabulary Rules) [5] to define business collaboration rules. Note also that we can produce several public views from the private process, by changing only the Visibility Rules, as they specify what process elements are visible to which partner. Visibility Rules are specified in a separate file, and can apply to any number of private processes. The idea of annotating business process element to provide semantics has also been used by Rospocher et al. [8]: the authors provide mechanisms for semantically annotate business processes, using text-annotations (like we do) and ontologies.

Concerning deadlock detection in SOA protocols, several approaches exist. Seguel et al. [9] for instance describe approaches for creating protocol adaptors for service

component integration. This work also covers some deadlock situations, but is merely based on SOA protocol and not at BPMN cross-organizational business process aspects. Other work relating to gap detection exists based on Petri Nets. Dijkman et al. [10] try to map BPMN on Petri Nets (by the way finding some deficiencies in the BPMN definition) and then perform static checks on the resulting Petri Net like finding dead tasks or deadlocks. Anyway, this work does not put a focus on issues concerning mismatches occurring in cross-organizational business processes. Bonet et al. [11] do not consider BPMN directly, but provide some interesting static analysis approaches on Petri Nets. In this work focus is more on performance issues, additionally covering some deadlock situations.

6 Conclusions and Future Work

We described a way of creating and analyzing a cross-organizational business process consisting of a model transformation of private to public business processes, creating a cross-organizational business process (CBP) by using the elements of the published process elements and analyzing the resulting CBP for interoperability gaps.

An eclectic diversification of interoperability gaps is currently not available in standard literature. Interoperability problems have been marginally addressed e.g. in the ATHENA project [2] or by Awad and Puhlmann [7]. But neither business process interoperability gaps within the enterprise, nor gaps of cross-organizational business processes have been the subject of a thorough analysis.

As the automated gap detection is based on the information being captured in BPMN and xPDL models, the degree automated gap detection is limited to information being captured in those models. Other gaps, like gaps in the quality of service required/offered in interactions between organizations cannot be detected in such models.

Future work will focus on extending the CBPip gap categorization and classification and improving the implementation of the business interoperability services. The number of identified CBPip Gap classes is far from being exhaustive. There is still room left to proceed with the initiated research to detect more gap problems, classify the gaps and describe them in more detail. Whereas currently all gaps are considered of same priority, defining different priorities for different gap kinds can also be part of future work.

Aiming at (semi-)automated resolution of CBP interoperability gaps is also a topic for further research, especially of the gap classes "interface mismatch" where a combination of other COIN services concerning ontology reconciliation can be leveraged.

The vocabulary for visibility rules we presented in this article is only a first version. We are currently working on an extension for health-care. The visibility rules we have presented should also not be overlapping, which is a shortcoming of the current implementation. We are planning to enhance the visibility rules with priorities (e.g. in case of conflicts, some rules will have precedence). Finally, as SBVR and ontologies are based on the Common Logic [12], it would be interesting to look for possible synergies with similar approaches for semantically annotating business processes using ontologies, like the shared workspace presented in [8].

7 References

1   COIN, "COIN Home Page", COIN IP. http://www.coin-ip.eu/
2   ATHENA, Specification of a Cross-Organisational Business Process Model, Deliverable D.A2.2, Version 1.0, June 2005
3   BPMN12, Business Process Management Notation, version 1.2., OMG, http://www.omg.org/spec/BPMN/1.2/PDF/
4   Carrez C., Del Grosso E., Karacan O., Taglino F.: Towards Cross-organisational Innovative Business Process Interoperability Services. 2nd IFIP WG5.8 International Workshop on Enterprise Interoperability (IWEI09). Valencia (Spain). LNBIP 38, Springer, 2009
5   OMG, Semantics of Business Vocabulary and Business Rules, version 1.0, http://www.omg.org/spec/SBVR/1.0/
6   Role Based Access Control, ANSI/INCITS 359-2004, Feb. 2004.
7   Ahmed Awad and Frank Puhlmann: Structural Detection of Deadlocks in Business Process Models, BIS 2008, LNBIP 7, pp.239–250, Springer-Verlag Berlin Heidelberg 2008.
8   M. Rospocher, C. Di Fransescomarino, C. Ghidini: Collaborative Specification of Semantically Annotated Business Processes. In BPM 2009 Workshops, LNBIP 43, Springer-Verlag, 2010.
9   R. Seguel, R. Eshuis, P. Grefen: An Overview on Protocol Adaptors for Service Component Integration, Beta Research School Working Paper Series 265, Eindhoven University of Technology, December 2008
10   Remco M. Dijkman, Marlon Dumas, and Chun Ouyang : Formal Semantics and Analysis of BPMN Process Models using Petri Nets, http://eprints.qut.edu.au/7115/1/7115.pdf, 2010
11   Pere Bonet, Catalina M. Llado, Ramon Puigjaner: PIPE v2.5: a Petri Net Tool for Performance Modeling. In 23rd Latin American Conference on Informatics, October 2007.
12   ISO/IEC 24707:2007 - Information technology — Common Logic (CL): a framework for a family of logic-based languages. 2007.