

# Aligning Collaborative Business Processes—An Organization-Oriented Perspective

Xiaohui Zhao, Chengfei Liu, Yun Yang, and Wasim Sadiq

**Abstract**—Business collaboration encompasses the coordination of information flows among organizations as well as the composition of their business processes toward mutual benefits. While integrating business processes of different organizations seamlessly, it brings great challenges to keep participating organizations as autonomous entities. To address this issue, we propose a new perspective on modeling collaborative business processes with a novel concept called relative workflow (RWF). With its visibility-control mechanism, the RWF model defines what a participating organization can perceive in collaboration and thereby allows each organization to customize its own collaboration process and behaviors. In this paper, we present a formal definition of RWFs and related algorithms for generating RWFs. A prototype is implemented on the Web service platform for the proof-of-concept purpose.

**Index Terms**—Collaborative business process, relative workflow (RWF), visibility control.

## I. INTRODUCTION

THE great surge of business globalization inspires organizations to be more ambitious of process collaboration by means of coupling their business processes with those of their trading partners. Undoubtedly, such cross-organizational business process integration enables the synergy of organizations, and therefore, the organizations can stay more competitive in the global market [1]–[5]. To enable such process collaboration, research efforts have been put on improving current workflow technologies for supporting collaborative business processes [6]–[9].

Recently, many approaches like markets of resources [10], electronic institutions [11], etc., have created a complex environment for business collaborations. In such an environment, workflow technology is widely deployed to model and automate business functions and collaborations. Traditional interorganizational workflow approaches mainly focus on modeling workflows from a *public* view [12], [13]. In this public view, a third-party designer or the main contractor (focal organization)

of a virtual enterprise often takes the most responsibilities of defining an interorganizational business process, such as designing the collaboration pattern and arranging collaboration behaviors. These approaches work well with the assumption that there exists a third-party designer or a main contractor that can see a certain level of details of all participating organizations. However, in many practical cases, this assumption is apparently overoptimistic. For the organizations that participated in the same collaborative business process, the relationship between each pair of organizations could be different. As such, in contrast to what public-view approaches assume, the visibility between participating organizations is *relative* rather than *absolute*. In another aspect, the predominant view of a third-party designer or a main contractor may put participating organizations in a passive position. This violates the fact that each participating organization acts as an autonomous entity, which naturally desires to define the business collaboration toward its own business objectives and benefits. Moreover, for the applications where partnerships change frequently and dynamically, the prefixed business collaboration in the public-view approaches is hardly applicable.

In the aspect of implementation technologies, a service-oriented architecture is widely adopted in enterprise collaborative environment, and the interfaces of operational components help hide the implementation details. Yet, such invocation interfaces are mainly function oriented and, therefore, cannot convey necessary process information for workflow tracking and process composition.

Aiming to facilitate organizational autonomy, business privacy, and collaboration flexibility in the process-oriented environment, we propose a new perspective on modeling business collaborations with a novel concept called *relative workflow* (RWF). In the RWF context, a collaborative business process is represented as a series of RWF processes, each of which is defined from the perspective of an individual participating organization. This allows each organization, as an autonomous entity, to design its own collaboration structure and behaviors. The public-view workflow design can be distributed into multiple one-party-oriented RWF process design. Different visibility constraints can then be defined for different organizations to reflect the fine granularity of visibility control between participating organizations.

The remainder of this paper is organized as follows. With a motivating example, in Section II, we analyze the business collaboration requirements that are not well supported by current approaches. Section III presents the formal framework of the RWF model, and Section IV addresses the procedure and the related algorithms for generating RWF processes. Section V justifies the RWF model in terms of information sufficiency and necessity. Section VI analyzes how to apply the RWF

Manuscript received June 30, 2008; revised December 10, 2008. First published September 25, 2009; current version published October 16, 2009. This work was supported in part by the Australian Research Council and in part by SAP Research under Grant LP0669660. This paper was recommended by Associate Editor M. Mora.

X. Zhao and C. Liu are with the Web and Data Engineering Research Group, Centre for Complex Software Systems and Services, Faculty of Information and Communication Technologies, Swinburne University of Technology, Melbourne, Vic. 3122, Australia (e-mail: xzhao@groupwise.swin.edu.au; cliu@groupwise.swin.edu.au).

Y. Yang is with the Faculty of Information and Communication Technologies, Swinburne University of Technology, Melbourne, Vic. 3122, Australia (e-mail: yyang@groupwise.swin.edu.au).

W. Sadiq is with the SAP Research Centre, Brisbane QLD 4000, Australia. Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TSMCA.2009.2027130

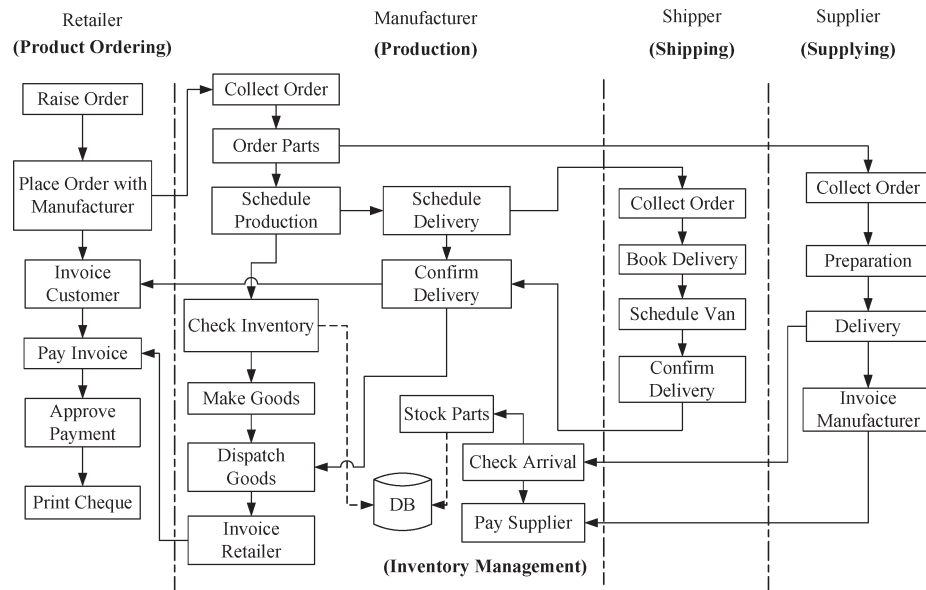


Fig. 1. Interorganizational workflow process example (modified from [18]).

framework in different collaboration schemes and highlights its appealing features for supporting dynamic collaborations. A Web-service-based prototype implementation is presented in Section VII. Section VIII discusses about the advantages and tradeoffs of the RWF approach. Finally, Section IX concludes this paper and outlines the future work.

## II. REQUIREMENT ANALYSIS WITH MOTIVATING EXAMPLE

Traditional interorganizational workflow design approaches streamline the participated business processes of different organizations into a public business process. As discussed earlier, this procedure has the following problems.

The first problem is the weak autonomy of organizations participating in the collaboration. Most interorganizational workflow modeling approaches assign a third-party designer or a main contractor to determine the collaboration choreography and orchestration of all participating organizations. However, this modeling scheme places the other minor participating organizations in a relatively passive position or as a mere member of the poll queue [14]. Due to this reason, these organizations behave in the collaboration as passively as a worker does in a pipeline workshop. Nevertheless, driven by global business expansion, organizations seek higher process customizability to adapt to local markets and national regulations. Consequently, organizations are challenged to strike the appropriate balance between centralized corporate standards and the autonomy needed to serve local markets. Furthermore, these facts result in an organization's inherent desire for the autonomy in choosing its own partner organizations, defining collaborative business processes by itself according to its business objectives, benefits, etc. Moreover, an appropriate third-party designer or a main contractor is not always obtainable, particularly in a loosely coupled collaboration environment.

The second problem is the coarse granularity of openness. This issue has been identified as early as in the 1980s [15]. Yet, most existing interorganizational-business-process model-

ing approaches compulsorily adopt a common business process, and all participating organizations share this public business process to conduct the collaboration. However, this public-business-process-based collaboration inevitably results in the discourse of excessive information or the insufficiency of necessary collaboration information [16]. In the former, some private business information may be unwillingly disclosed to an organization with a distant partnership. In the latter, business processes belonging to involved organizations cannot be integrated seamlessly. A customizable visibility control over interorganizational business processes is therefore on urgent demand to balance the information openness and privacy prevention elaborately, as well as to guarantee the maximal autonomy of participating organizations.

The third problem is the poor flexibility and adaptability of predetermined collaborative business processes. To adapt to the turbulent and rapidly changing environment, organizations have to modify their business processes. Consequently, a collaborative business process may transform in an *ad hoc* manner, according to the market requirements and changing partnerships [17]. Thus, the support for the reconfigurability of collaborative business processes is expected with strong demands.

Fig. 1 shows a collaboration scenario among four organizations. In this scenario, all the participating organizations have the knowledge of the whole collaboration process, which is predetermined and defined by a third-party designer or a main contractor such as the manufacturer. Once the collaborative business process has been defined, each participating organization acts passively and loses, more or less, its autonomy. It will be difficult for an organization to change its collaboration pattern or behaviors, for instance, to start a new partnership or to terminate an existing partnership. Moreover, the knowledge of the whole collaboration process gives no chance to define a close or distant partnership between participating organizations. For example, from Fig. 1, we can clearly see that the views from a retailer and a manufacturer on the collaborative process are different. While a manufacturer has a close partnership with all other participating organizations, a retailer, however,

only has a close partnership with a manufacturer via a proper source/supply contract. A retailer may not need to know and, actually, should not know the manufacturer's partnerships, for example, the partnership with a supplier. At the same time, a retailer may need to have some knowledge about a shipper of the manufacturer, so that the retailer can track the delivery of goods in transit. In case the manufacturer intends to change suppliers or shippers, it is also expected to modify the collaborative business process on the fly. Reluctantly, the public-view approaches act awkwardly to support these requirements.

To address these issues, we believe that business collaboration should be decided from the view of each individual organization. This implies that each organization defines its collaboration structure and behaviors by following corresponding contracts with proper partner organizations, and it can also change them later by updating existing contracts or signing new contracts. In this way, each organization is empowered with the authority to design its collaboration in a proactive mode. In addition, the views from different organizations may be different due to the partnerships and privacy reasons.

The proposed RWF model caters for these issues with an organization-oriented perspective. The offered support on autonomy, openness, and flexibility by the RWF model empowers organizations with more privacy control, partner selection freedom, and convenience in collaborations. These features also benefit the dynamic reconfiguration of the collaboration network, as only the perceivable changes need to be updated in the relative perspective, and thereby, the workflow updating load can be reduced when the network changes.

The research reported in this paper is based on a preliminary version of our RWF approach [19], with significant improvement and extension on the model justification, lifecycle from contracting to RWF generation, and the facilitating system design.

### III. RWF PROCESSES

In our context, a collaborative business process consists of several intraorganizational business processes of participating organizations together with the interactions between them. We call these intraorganizational business processes as local workflow processes.

**Definition 1 (Local Workflow Process):** A local workflow process  $lp$  is defined as a directed graph  $(\mathcal{T}, \mathcal{R})$ , where 1)  $\mathcal{T}$  is the set of nodes representing the set of tasks and 2)  $\mathcal{R} \subseteq \mathcal{T} \times \mathcal{T}$  is the set of arcs representing the execution sequence.

**Definition 2 (Organization):** An organization  $g$  is a participant in the business collaboration. An organization owns a set of local workflow processes  $\{lp^1, lp^2, \dots, lp^m\}$ . An individual local workflow process  $lp^i$  of  $g$  is denoted as  $g.lp^i$ ,  $1 \leq i \leq m$ , and  $m$  is the number of  $g$ 's local workflow processes.

In a loosely coupled collaboration environment, each organization expects to protect the critical or private information of its business processes from disclosing to other organizations. According to the two most important behaviors in the context of collaborative business processes, i.e., workflow tracking and workflow interaction, we define the following three values for task visibility as listed in Table I.

Due to the high diversity of business collaborations, these three values can never cover all visibility scenarios. This paper aims to provide a fundamental visibility-control mechanism

TABLE I  
VISIBILITY VALUES

Visibility value	Explanation
Invisible	A task is said invisible to an external organisation, if it is hidden from that organisation.
Trackable	A task is said trackable to an external organisation, if that organisation is allowed to trace the execution status of the task.
Contactable	A task is said contactable to an external organisation, if the task is trackable to that organisation and the task is also allowed to send/receive messages to/from that organisation for the purpose of business interaction.

with these three values, while this visibility value table is open for future extension.

**Definition 3 (Visibility Constraint):** A visibility constraint  $vc$  is defined as a tuple  $(t, v)$ , where  $t$  denotes a workflow task and  $v \in \{\text{Invisible}, \text{Trackable}, \text{Contactable}\}$ . A set of visibility constraints  $\mathcal{VC}$  defined on a workflow process  $lp$  is represented as a set  $\{(t, v) | t \in lp.\mathcal{T} \text{ and } v \in \{\text{Invisible}, \text{Trackable}, \text{Contactable}\}\}$ .

**Example 1:** Based on Fig. 1, two sets of visibility constraints are given as follows:

$$\begin{aligned} \mathcal{VC}_1 = \{ & (\text{'Raise Order'}, \text{Invisible}), (\text{'Place Order with Manufacturer'}, \text{Contactable}), (\text{'Invoice Customer'}, \text{Contactable}), (\text{'Pay Invoice'}, \text{Contactable}), (\text{'Approve Payment'}, \text{Invisible}), (\text{'Print Cheque'}, \text{Invisible}) \}. \\ \mathcal{VC}_2 = \{ & (\text{'Collect Order'}, \text{Contactable}), (\text{'Order Parts'}, \text{Invisible}), (\text{'Schedule Production'}, \text{Trackable}), (\text{'Schedule Delivery'}, \text{Trackable}), (\text{'Confirm Delivery'}, \text{Contactable}), (\text{'Check Inventory'}, \text{Invisible}), (\text{'Make Goods'}, \text{Trackable}), (\text{'Dispatch Goods'}, \text{Trackable}), (\text{'Invoice Retailer'}, \text{Contactable}) \}. \end{aligned}$$

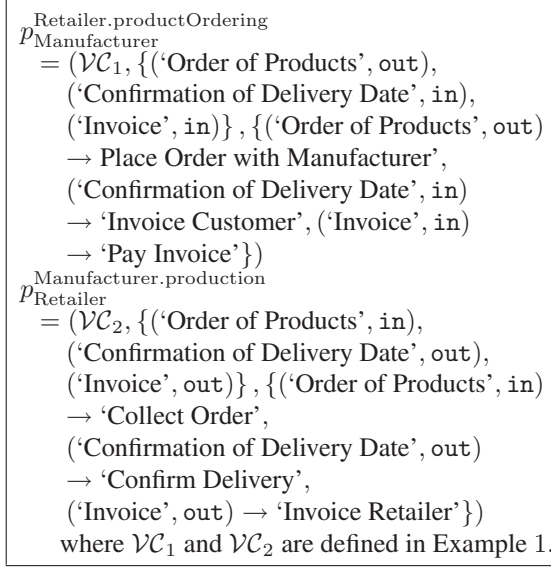
These two sets are defined on the "Product Ordering" and "Production" processes, respectively.

**Definition 4 (Perception):** A perception  $p_{g_0}^{g_1.lp}$  of an organization  $g_1$ 's local workflow process  $lp$  from another organization  $g_0$  is defined as tuple  $(\mathcal{VC}, \mathcal{MD}, f)$ , where  $\mathcal{VC}$ ,  $\mathcal{MD}$ , and  $f$  are defined as follows.

- 1)  $\mathcal{VC}$  is a set of visibility constraints defined on  $g_1.lp$ .
- 2)  $\mathcal{MD} \subseteq \mathcal{M} \times \{\text{in}, \text{out}\}$  is a set of the message descriptions that contains the message bodies and their passing directions.  $\mathcal{M}$  is the set of message bodies. The passing direction is defined from the perspective of  $lp$ 's host organization of, i.e.,  $g_1$ .
- 3)  $f: \mathcal{MD} \rightarrow g_1.lp_{g_0}.\mathcal{T}$  is a bijection from  $\mathcal{MD}$  to  $g_1.lp_{g_0}.\mathcal{T}$ , and  $g_1.lp_{g_0}$  is the *perceivable workflow process* of  $g_1.lp$  from  $g_0$ . Here, a perceivable workflow process represents the perceivable form of a local workflow process for a partner organization. The generation of  $g_1.lp_{g_0}$  from  $g_1.lp$  will be discussed in the next section.

**Example 2:** Based on the aforementioned motivating example, we give the perception of the retailer's Product Ordering process from the manufacturer and the perception of the manufacturer's Production process from the retailer as shown at the next page.





**Definition 5 (RWF Process):** An RWF process  $rp$  perceivable from an organization  $g_0$  is defined as a directed graph  $(\mathcal{T}, \mathcal{R})$ , where  $\mathcal{T}$  and  $\mathcal{R}$  are defined as follows.

- 1)  $\mathcal{T}$  is the set of local and perceivable tasks, which is a union of  $\mathcal{T}_L$  and  $\mathcal{T}_P$ .
  - a)  $\mathcal{T}_L = \bigcup_k g_0.lp^k.\mathcal{T}$  is the union of the task sets of  $g_0$ 's local workflow processes. Here,  $1 \leq k \leq m_0$  and  $m_0$  is the number of  $g_0$ 's involved local workflow processes.
  - b)  $\mathcal{T}_P = \bigcup_{i,j} g_i.lp_{g_0}^j.\mathcal{T}$  is the union of the task sets of the workflow processes perceivable from  $g_0$ . Here,  $1 \leq i \leq n$  and  $1 \leq j \leq m_i$ , while  $n$  is the number of  $g_0$ 's partner organizations and  $m_i$  is the number of  $g_i$ 's involved perceivable workflow processes for  $g_0$ .
- 2)  $\mathcal{R}$  is the set of arcs perceivable from  $g_0$ , which is a union of the following four parts, where  $i, j$ , and  $k$  are the same as in the definition of  $\mathcal{T}$ .
  - a)  $\mathcal{R}_L = \bigcup_k g_0.lp^k.\mathcal{R}$  is the union of the arc sets of  $g_0$ 's local workflow processes.
  - b)  $\mathcal{R}_P = \bigcup_{i,j} g_i.lp_{g_0}^j.\mathcal{R}$  is the union of the arc sets of the workflow processes perceivable from  $g_0$ .
  - c)  $\mathcal{L}_{\text{intra}}$  is the set of intraorganizational messaging links that connect tasks belonging to different local workflow processes and is defined on  $\bigcup_{i,j} (g_0.lp^i.\mathcal{T} \times g_0.lp^j.\mathcal{T})$ , where  $i \neq j$ .
  - d)  $\mathcal{L}_{\text{inter}}$  is the set of interorganizational messaging links that connect tasks between a local workflow process and a perceivable workflow process and is defined on  $\bigcup_{i,j,k} (g_0.lp^k.\mathcal{T} \times g_i.lp_{g_0}^j.\mathcal{T} \cup g_i.lp_{g_0}^j.\mathcal{T} \times g_0.lp^k.\mathcal{T})$ .

Fig. 2 shows how the components of the RWF model are related across organizations. An RWF process consists of local workflow processes, perceivable workflow processes, and the messaging links between workflow processes. A perception defines the visibility constraints for the tasks of a local workflow process. According to a given perception, a perceivable workflow process can be derived out from a local workflow process by composing the invisible tasks to visible ones. The perception

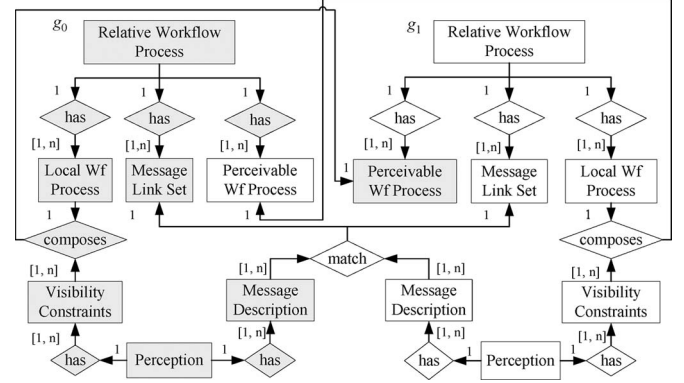


Fig. 2. RWF model.

also defines the message descriptions including message body and messaging directions for the involved messaging links. By matching the compatible message descriptions, two workflow processes can be connected together.

It is a necessary procedure for an organization  $g_0$  to define the perceptions on local workflow processes for its partner organizations, i.e.,  $g_1, g_2, \dots, g_n$ , before generating RWF processes. In Fig. 2, we only show one partner organization  $g_1$  for illustration. This procedure includes defining visibility constraints, messaging links, and mapping functions. Once the perceptions on local workflow processes are defined, an RWF process can be generated by other two steps: composing tasks and assembling RWF processes.

The purpose of composing tasks is to hide private tasks of local workflow processes. We choose to merge invisible tasks with the contactable or trackable tasks into composed tasks. According to the perceptions defined from  $g_1$ , a local workflow process of  $g_1$  after this step becomes a perceivable workflow process for  $g_0$ .

An organization may assemble RWF processes by linking its local workflow processes and the perceivable workflow processes from partner organizations via messaging links. As shown in Fig. 2, an RWF process of  $g_0$  consists of  $g_0$ 's local workflow processes, the perceivable workflow processes from  $g_1$ , and the messaging links obtained by matching the message descriptions defined in perceptions.

The details are discussed in the following section.

#### IV. GENERATING RWF PROCESSES

##### A. Defining Perceptions

The raw information for defining perceptions is from commercial contracts between collaborating organizations [20], [21]. Griffel *et al.* [22] have proposed a classical contract model in the Common Open Service Market for Small-to-medium sized enterprise (SME) (COSMOS) project, which classifies a contract into four major parts of *Who*, *What*, *How*, and *Legal Clauses*, as shown in Fig. 3. In this paper, we employ this contract model to help analyze and derive the visibility constraints for perceptions.

As the core part of this contract model, the *How* part defines the execution details for the obligations defined in the *What* and *Legal Clauses* parts. The *business interactions* of the *execution* component details how the parties defined in the *Who* part should interact with each other to fulfil the

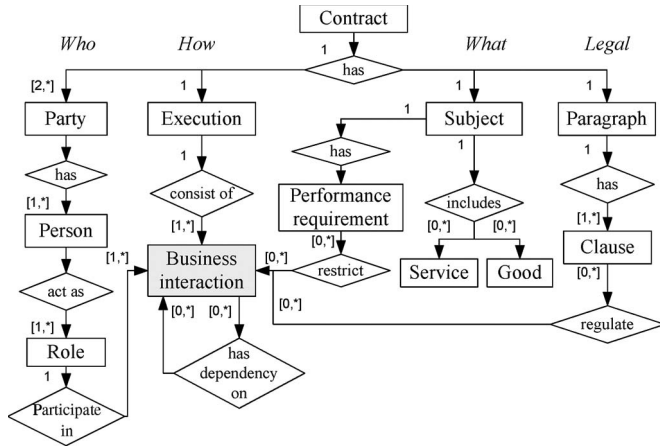


Fig. 3. Simplified contract model (modified from [23]).

collaboration. At the process level, each business interaction is supported by one or more workflow tasks of the involved workflow processes. Between these business interactions, there may exist *dependencies*, such as status dependencies or tracking requirements, etc., and these dependencies may further complicate the correlations between the supporting workflow tasks. In the contracting process, we call the organization that issues a contract a *host organization* and the responding organizations *partner organizations*.

Different from a contract, a perception is defined from the perspective of one organization toward the local workflow processes of its partner organizations. For the collaboration between organization  $g_0$  and partner organizations  $g_1, \dots, g_n$ , two sets of such perceptions are required to represent the collaboration, where  $\mathcal{PS}_1$  is the set of the perceptions defined on  $g_0$ 's participated local workflow processes  $g_0.lp_1, \dots, g_0.lp_{m_0}$  from  $g_1, \dots, g_n$ , i.e.,  $\{p_{g_0.lp_1}^{g_1}, \dots, p_{g_0.lp_{m_0}}^{g_1}, \dots, p_{g_0.lp_1}^{g_n}, \dots, p_{g_0.lp_{m_0}}^{g_n}\}$ , and  $\mathcal{PS}_2$  is the set of the perceptions defined on all participated local workflow processes of  $g_1, \dots, g_n$  from  $g_0$ , i.e.,  $\{p_{g_1.lp_1}^{g_0}, \dots, p_{g_1.lp_{m_1}}^{g_0}, \dots, p_{g_n.lp_1}^{g_0}, \dots, p_{g_n.lp_{m_n}}^{g_0}\}$ .

Thus, each participated workflow process in the contracted collaboration is assigned with a proper perception. To derive these perceptions, we need to analyze the business interactions defined in related contracts by recognizing necessary interorganizational messages and setting up visibility constraints for workflow tasks, etc. Algorithm 1 details the steps how  $g_0$ 's partner organization  $g_1$  generates a perception  $p$  of  $g_1$ 's local workflow process  $lp$  for  $g_0$ , according to the business interactions defined in contract  $c$ .

#### Algorithm 1. Generating perceptions

##### Input:

- $c$ —a contract signed by two organizations  $g_0$  and  $g_1$
- $g_0$ —the host organization
- $g_1$ —the partner organization
- $lp$ —an involved local workflow process of  $g_1$

##### Output:

$p$ —the generated perception of  $g_1$  from  $g_0$

1.  $p.VC = \emptyset$ ;  $p.MD = \emptyset$ ;  $p.f = \emptyset$ ;
2. **for each** task  $t \in lp$
3.  $p.VC = p.VC \cup \{(t, \text{invisible})\}$ ;

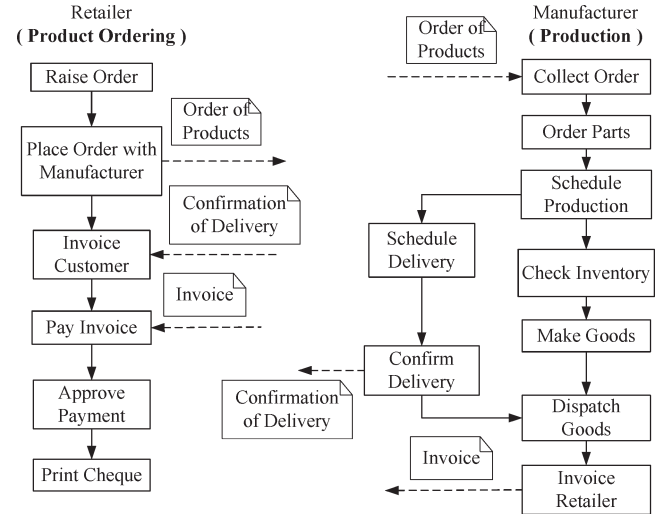


Fig. 4. Local workflow processes.

4. **for each** business interaction  $bi$  defined in contract  $c$
5.   **for each** task  $t \in lp$
6.     **if** task  $t$  provides necessary functions for  $bi$  **then**
7.       **if**  $\exists(t, \text{invisible}) \in p.VC$  **then**
8.          $p.VC = p.VC \cup \{(t, \text{contactable})\} \setminus \{(t, \text{invisible})\}$ ;
9.          $mdSet = \{\text{the message descriptions to be used by } t \text{ to support } bi\}$
10.        **for each** message  $md \in mdSet$
11.          $p.MD = p.MD \cup \{md\}$ ;
12.          $p.f = p.f \cup \{(md, t)\}$ ;
13.        **end for**
14.     **end if**
15. **for each** business interaction  $bi$  defined in contract  $c$
16.   **for each** task  $t \in lp$
17.     **if**  $bi$  has status dependence with  $t$  **then**
18.        **if**  $\exists(t, \text{invisible}) \in p.VC$  **then**
19.          $p.VC = p.VC \cup \{(t, \text{trackable})\} \setminus \{(t, \text{invisible})\}$ ;

This algorithm first sets all tasks invisible in lines 2–3. Lines 4–14 set the tasks that are directly involved in business interactions to be contactable and create the corresponding message descriptions for these contactable tasks. Lines 15–19 set trackable tasks according to the status dependence between tasks. The status dependence between business interaction  $bi$  and task  $t$  denotes that  $bi$  relies on the execution status of  $t$ .

Fig. 4 shows generated message descriptions, represented by the dashed arrows, for the Product Ordering process and the Production process mentioned in Section II. To represent the business interaction between these two processes, we define perception  $p_{\text{Manufacturer.productOrdering}}^{\text{Retailer.productOrdering}}$  of the retailer's Product Ordering process from the manufacturer and perception  $p_{\text{Retailer}}^{\text{Manufacturer.production}}$  of the manufacturer's Production process from the retailer, respectively. The visibility constraints of these two perceptions are already given in Example 1.

#### B. Composing Tasks

In this step, a local workflow process needs to hide its invisible tasks by composing them with proper contactable or trackable tasks and thereby attain the corresponding perceivable

workflow process. For an invisible task  $t$  and a contactable task  $t'$ , the composition conducts by following these rules: 1)  $t$  is to be hidden into  $t'$ , if  $t'$  is after  $t$  and  $t'$  has an outgoing message, and 2)  $t$  is to be hidden into  $t'$ , if  $t'$  is before  $t$  and  $t'$  has an incoming message.

For simplicity of discussion, we only consider composing one local workflow process  $lp$  of organization  $g_0$  from another organization  $g_1$ . We conduct a preprocessing on all split/join structures of  $lp$  such that, for all those branches consisting of only invisible tasks, a dummy task is created to delegate these branches. The algorithm for composing tasks is given as follows.

### Algorithm 2. Task Composition

#### Input:

$lp$ — $g_1.lp$ , organization  $g_1$ 's local workflow process  $lp$  before composition

$p$ — $p_{g_0}^{g_1.lp}$ , the perception of  $g_1$ 's  $lp$  from  $g_0$

#### Output:

$lp'$ — $g_1.lp_{g_0}$ , the perceivable workflow process composed from  $lp$  for  $g_0$ , according to  $p_{g_0}^{g_1.lp}$

1.  $lp' = lp$ ;
2.  $\mathcal{VT} = \{\text{all the visible tasks of } lp, \text{ defined in } p\}$ ;
3. **while**  $(\exists t, t' \in (lp'.\mathcal{T} - \mathcal{VT}))((t, t') \in lp'.\mathcal{R}) \wedge seq(t) \wedge seq(t')$   
 // here,  $seq(t) = (indegree(t)=1 \wedge outdegree(t)=1)$ 
  4.  $t^\circ = t + t'$ ;
  5.  $lp'.\mathcal{T} = lp'.\mathcal{T} \cup \{t^\circ\} \setminus \{t, t'\}$ ;
  6.  $lp'.\mathcal{R} = lp'.\mathcal{R} \setminus \{(t, t')\}$ ;
  7. **replace**  $t, t'$  in  $lp'.\mathcal{R}$  with  $t^\circ$ ;
8. **end while**
9. **while**  $(\exists t \in \mathcal{VT}(p'.f^{-1}(t) = (m, in) \wedge outdegree(t)=1) \wedge (\exists t' \in (lp'.\mathcal{T} - \mathcal{VT}))((t, t') \in lp'.\mathcal{R} \wedge indegree(t')=1))$ 
  10.  $t^\circ = t + t'$ ;
  11.  $\mathcal{VT} = \mathcal{VT} \cup \{t^\circ\} \setminus \{t\}$ ;
  12.  $lp'.\mathcal{T} = lp'.\mathcal{T} \cup \{t^\circ\} \setminus \{t', t\}$ ;
  13.  $lp'.\mathcal{R} = lp'.\mathcal{R} \setminus \{(t, t')\}$ ;
  14. **replace**  $t, t'$  in  $lp'.\mathcal{R}$  with  $t^\circ$ ;
15. **end while**
16. **while**  $(\exists t \in \mathcal{VT}(p'.f^{-1}(t) = (m, out) \wedge indegree(t)=1) \wedge (\exists t' \in (lp'.\mathcal{T} - \mathcal{VT}))((t, t') \in lp'.\mathcal{R} \wedge outdegree(t')=1))$ 
  17.  $t^\circ = t + t'$ ;
  18.  $\mathcal{VT} = \mathcal{VT} \cup \{t^\circ\} \setminus \{t\}$ ;
  19.  $lp'.\mathcal{T} = lp'.\mathcal{T} \cup \{t^\circ\} \setminus \{t', t\}$ ;
  20.  $lp'.\mathcal{R} = lp'.\mathcal{R} \setminus \{(t, t')\}$ ;
  21. **replace**  $t, t'$  in  $lp'.\mathcal{R}$  with  $t^\circ$ ;
22. **end while**

Lines 3–8 compose each pair of neighboring sequential invisible tasks into one invisible task. Lines 9–15 downward compose invisible tasks with incoming interaction tasks, and lines 16–22 upward compose invisible tasks with outgoing interaction tasks.

Fig. 5 shows the results of task composition: (a) is the perceivable Product Ordering process of the retailer from the manufacturer, and (b) is the perceivable Production process of the manufacturer from the retailer, where the dashed rectangles denote invisible tasks.

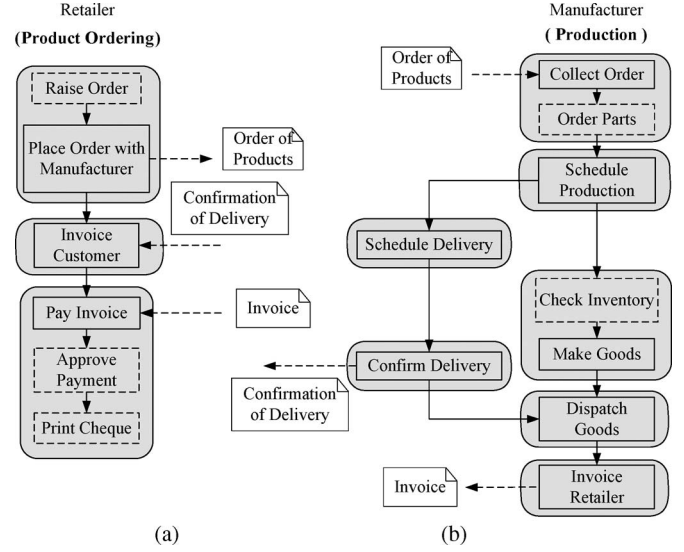


Fig. 5. Perceivable workflow processes.

### C. Assembling RWF Processes

In this step, related local workflow processes and perceivable workflow processes are connected together by linking the corresponding interaction operations. Algorithm 3 illustrates the linking procedure and the steps of matching message descriptions. For simplicity of discussion, we only consider matching one local workflow process  $lp$  of partner organization  $g_1$  from host organization  $g_0$  in the given algorithm.

By saying that one message description  $md_1$  matches another message description  $md_2$  in Algorithm 3, we mean that they have the same message, and one has passing direction “in” while the other has “out.” With the set  $\mathcal{L}$  of generated messaging links, we can assemble RWF processes

### Algorithm 3. Local Workflow Process Matching

#### Input:

$lp' = g_0.lp_{g_1}$ —the perceivable workflow process composed from  $g_0$ 's local workflow process  $lp$ .

$p = p_{g_0}^{g_1.lp}$ —the perception of  $g_0$ 's  $lp$  from  $g_1$

$ps = \{p_{g_0}^{g_1.lp^1}, \dots, p_{g_0}^{g_1.lp^{1n_1}}\}$ —the set of perceptions defined on  $g_1$ 's perceivable workflow processes from  $g_0$

#### Output:

$\mathcal{L}$ —the set of generated messaging links

1.  $\mathcal{L} = \emptyset$ ;
2. **for each**  $t \in lp'.\mathcal{T}$
3. **if**  $\exists md(p.f(md) = t)$  **then**
4.  $md_1 = p.f^{-1}(t)$ ;
5. **for each**  $p^\circ \in ps$
6. **for each**  $md_2 \in p^\circ.MD$
7. **if**  $md_1$  **matches**  $md_2$  **then**  $\mathcal{L} = \mathcal{L} \cup \{(t, p^\circ.f(md_2), md_1)\}$ ;
8. **end if**

Fig. 6(a) and (b) shows the RWF processes from the retailer's and the manufacturer's views, respectively, where the dashed connecting arrows denote the generated messaging links. This figure shows that, for the same collaborative business process,



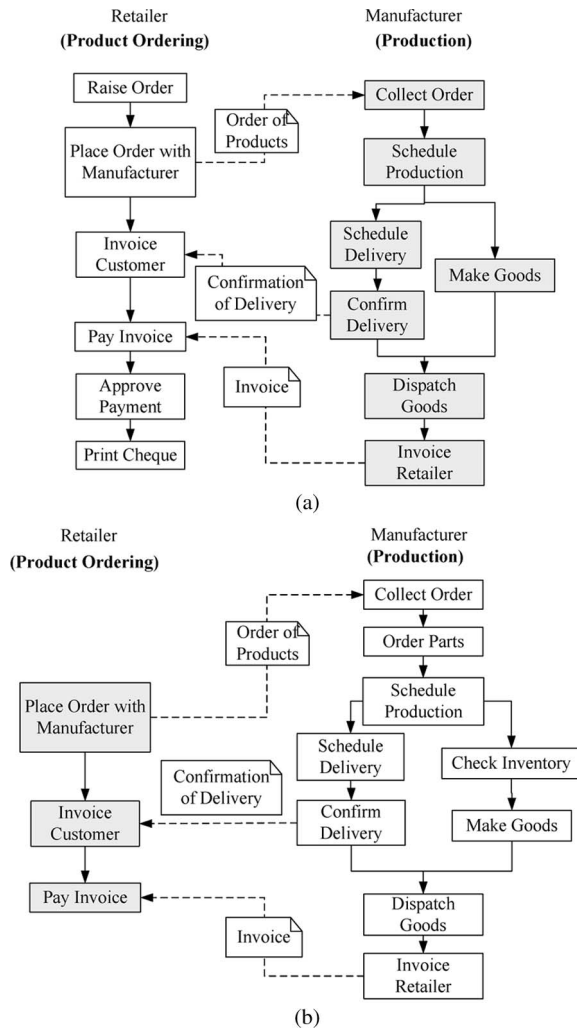


Fig. 6. RWF processes.

different participating organizations may have different views. This reflects the *relativity* characteristic of the model.

When generating RWF processes, a local workflow process can be a composite process that contains multiple sublocal workflow processes to cater for the complexity of business process realizations. Furthermore, an existing RWF process for an organization can also be treated as a local workflow process in a larger collaboration with different partners to reflect the hierarchy of collaborations. Such cases are quite common in service-outsourcing scenarios.

## V. MODEL JUSTIFICATION

In this section, we justify the RWF model from the aspects of information sufficiency and necessity. From the organization-oriented perspective, we define the information sufficiency and necessity for RWFs in terms of their partial views over a public-view collaborative business process. Theoretically, the following two properties describe the information necessity and sufficiency, respectively.

**Property 1 (Necessity):** An RWF process contains necessary information for the host organization to accomplish its responsibilities in the participated business collaboration.

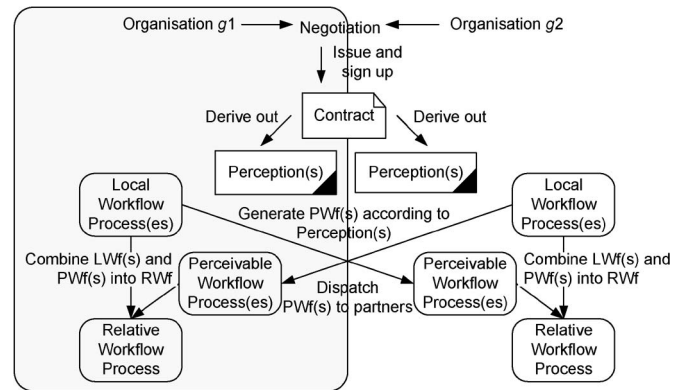


Fig. 7. Contracting process for traditional collaboration.

**Property 2 (Sufficiency):** A collaborative business process can be sufficiently represented by a finite number of RWF processes defined for participating organizations.

The detailed proofs for these two properties are given in the Appendix.

## VI. APPLICATION SCENARIOS

### A. Traditional Collaboration Scheme

The RWF approach fully supports the traditional collaboration scheme. As shown in Fig. 7, the traditional scheme starts the collaboration from contracting, where multiple organizations issue and agree with a contract. From the contract, organizations will derive the perceptions for their participated local workflow processes using Algorithm 1. Thereafter, according to these perceptions, these organizations will create corresponding perceivable workflow processes as stated in Algorithm 2 and dispatch the perceivable workflow processes to partner organizations. Upon receiving these perceivable workflow processes, an organization can connect them with its local workflow process and create the RWF process for the collaboration using Algorithm 3.

### B. "Browse and Pick"

As discussed in Section V, a collaborative business process modeled by public-view approaches can be equivalently substituted by a series of RWF processes. Moreover, our RWF approach also supports some applications that the public-view approaches can hardly cope with. One example is transient supply chains. In nowadays e-business, buyer, supplier, seller, and distributor organizations can exchange their trading information and find trading partners over e-marketplaces or other information portals. These sorts of collaborations are most likely to be dynamic and temporary. The partnership is usually decided by means of price matching, bidding, or auctions, and it terminates as soon as the trading accomplishes. As discussed earlier, our RWF approach fits well into such dynamic collaboration. Another example is a virtual organization alliance which consists of multiple SMEs. The SMEs join the virtual community to share business services from each other. Each organization in such an open alliance is aware of the advertised services and also needs to publish its business services to the community. Such a dual-awareness requirement can be well supported by the visibility-control-based perceptions. In

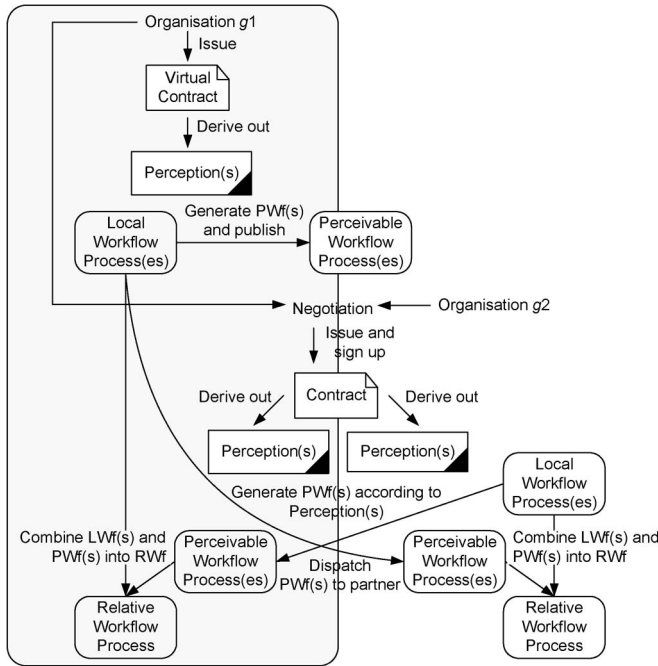


Fig. 8. Contracting process for browse-and-pick collaboration.

addition, the browse-and-pick building mechanism of RWFs can well serve this kind of alliances.

Different from traditional collaborations, organizations in this collaboration mode mainly follow an open contracting mechanism. As shown in Fig. 8, in this contracting mechanism, the host organization, for example,  $g_1$ , first lists the basic supply-and-demand requirements in a virtual contract. This virtual contract is issued for all potential partner organizations rather than a concrete organization. According to this virtual contract,  $g_1$  can derive out the corresponding perception and even generate the perceivable workflow process(es) for its involved local workflow process(es). These perceivable workflow processes are then released to public for advertising purpose, and therefore, they are the same to all potential partner organizations. Interested organizations, for example,  $g_2$  and others, may check the supply-and-demand information and the perceivable workflow processes before they decide whether to establish the collaboration. Once decided, the interested organizations can negotiate with the host organization and sign up a concrete contract. The remaining process is the same to the one for the traditional collaboration.

## VII. PROTOTYPE IMPLEMENTATION

### A. System Overview

In regard to enterprise system development, Web services have become a popular implementation platform [24]. To demonstrate the ideas discussed in this paper, a prototype base has been implemented on Sun Microsystems' Java Web Service Application Programming Interface (API) stack. This newly rearchitected API stack comprises Java API for XML Web Services [25], Java Architecture for XML Binding 2.0 [26], and SOAP with Attachments API for Java 1.3 [27]. This API stack represents a logical rearchitecture of Web service functionality in the open-source Java Enterprise Edition-compliant applica-

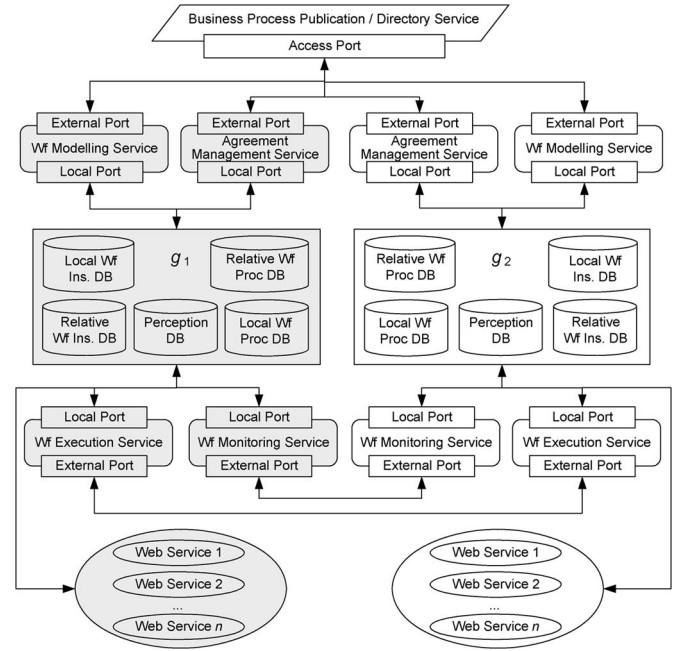


Fig. 9. Architecture of the proposed business process management system.

tion server, instead of the original XML Remote Procedure Call-oriented APIs.

Based on this new API stack, we have developed the business process management system architecture as shown in Fig. 9. This business process management system implements our RWF methodology with the benefits from a Web service's inherent advantages for distributed computing. For better integration and interoperability, Business Process Execution Language for Web Services (WS-BPEL) is deployed as the default business-process-definition language.

Fig. 9 shows two business process management systems belonging to collaborating organizations. An independent *Business Process Directory Service* stores all the business processes published by organizations in a *perceivable workflow process database* to provide a common directory for all organizations. Organizations can browse these published business processes and select appropriate ones to create collaborative business processes.

The business process management system of each organization consists of four administrative components, viz., the *Agreement Management Service*, the *Workflow Modeling Service*, the *Workflow Execution Service*, and the *Workflow Monitor Service*. Each administrative component has a local operation port and an external operation port, which are accessible to intraorganizational components and databases and partner organizations' components and databases, respectively. In addition, several databases are deployed for storing workflow processes, instances, and perceptions. The Web services shown in big ovals in Fig. 9 work as independent functional components, which collectively provide business functions under the invocation of *Workflow Execution Services* and in harmony with the underlying collaborative business processes.

The whole lifecycle of an RWF process through these four administrative components goes as follows. First, the *Agreement Management Service* wraps a local workflow process into a series of perceivable workflow processes for different partner organizations. Thereafter, the *Workflow Modeling Service*



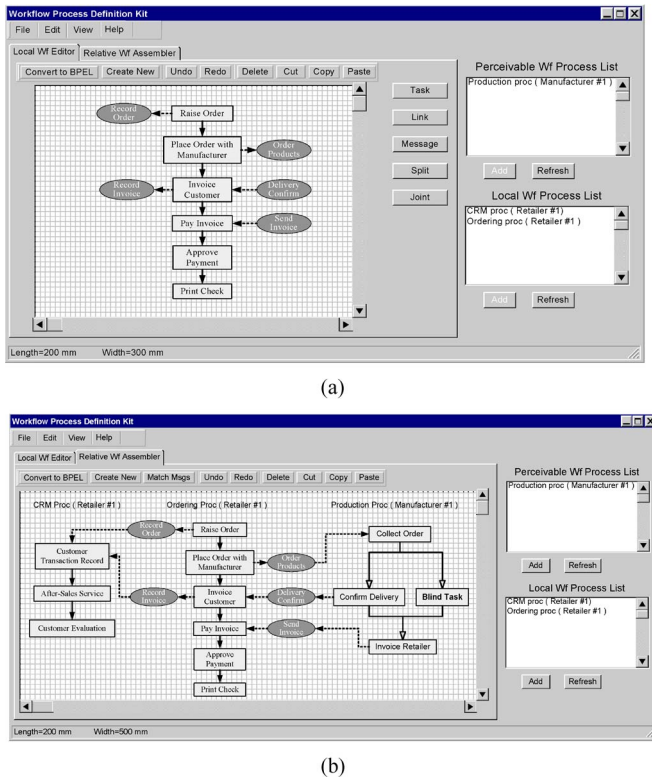


Fig. 10. User interfaces of workflow modeling tool. (a) User interface for local workflow process modeling. (b) User interface for RWF process modeling.

generates RWF processes with the perceivable workflow processes from partner organizations. Finally, RWF processes will be executed by the *Workflow Execution Services* of the host organization and the partner organizations. The monitoring over the execution will be handled by the *Workflow Monitor Services* of involved organizations.

### B. Prototype Components

1) *Agreement Management Service*: This component handles the documentation for collaboration preparation. Exactly, it is responsible for defining perceptions and generating perceivable workflow processes for partner organizations. This component maintains a list of partner organizations and creates proper perceptions on local workflow processes for these partner organizations. With these perceptions, it can wrap local workflow processes into perceivable workflow processes. Finally, these perceivable workflow processes will be published to the *Business Process Directory Service*.

2) *Workflow Modeling Service*: This component provides a specialized graphic modeling tool, as shown in Fig. 10, for users to define local workflow processes and store the created workflow processes in the local workflow process database. This tool also supports the downloading of perceivable workflow processes from the Business Process Directory Service to help users assemble RWF processes.

3) *Workflow Execution Service*: This component is used to coordinate and enact the execution of business process instances. This component caters the instance level management of both local workflow processes and RWF processes.

Fig. 11 shows the inner structure of the Workflow Execution Service. The *workflow manager* is in charge of navigating the

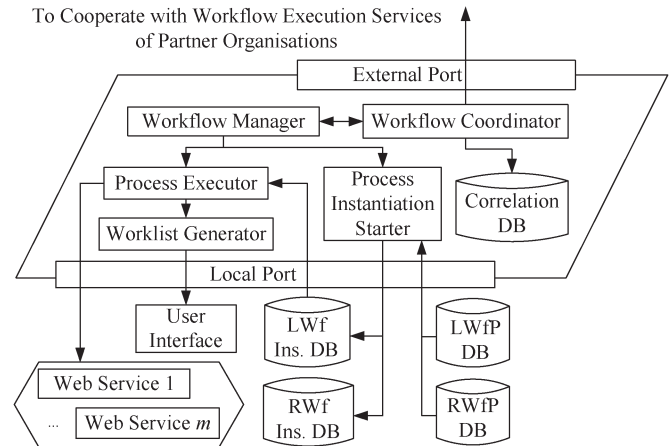


Fig. 11. Workflow Engine Service.

execution of workflow instances in general. It can initiate workflow instances via the *process instantiation starter*, dispatch intraorganizational and interorganizational interactions to the *process executor* and the *workflow coordinator* for execution, respectively, and use the *worklist generator* to assign tasks to human staff.

As to local workflow processes, this component works as a traditional workflow engine, responsible for creating workflow instances, navigating workflow instance execution, and controlling their interaction with workflow participants and applications.

In the context of RWFs, an RWF process contains both the local part, i.e., the included local workflow processes, and the foreign part, i.e., the included perceivable workflow processes. Therefore, the execution of an RWF instance requires the cooperation of *Workflow Execution Services* across organizational boundaries. Here, we use the RWF process shown in Fig. 6 to demonstrate such cooperation. First, we suppose that the retailer's workflow manager is at the time of executing task "place order with manufacturer." As this task initiates an interaction with the manufacturer, the workflow manager contacts the workflow coordinator with the purchase order as well as the instance identification (ID) of the contextual RWF instance. In fact, the instance ID of the contextual RWF instance corresponds to the instance ID of the product ordering process included in this RWF process. When notified, the workflow coordinator first checks the correlation database for preexisting correlations. As this is the first interaction with the manufacturer, there is no preexisting correlation. Thus, the retailer's workflow coordinator contacts the manufacturer's workflow coordinator with the purchase order. The manufacturer's workflow coordinator redirects the received order to the process executor via the workflow manager. When the process executor dispatches this order to a specific production process instance, the manufacturer's workflow coordinator creates a new correlation between the production process instance and the product ordering process instance in its correlation database. Thereafter, it synchronizes this correlation to the retailer's correlation database by notifying the retailer's workflow coordinator. Finally, the retailer's workflow manager continues to the next task, i.e., "invoice customer," while the manufacturer's workflow manager may keep collecting orders or go on to the next task, i.e., "order parts." As such, an interorganizational interaction

is completed by the cooperation of the two organizations' Workflow Execution Services. Details of recognizing instance correlations can be found in [28].

4) *Workflow Monitor Service*: This component handles both intraorganizational and interorganizational workflow tracking. The interorganizational workflow tracking starts from an RWF instance and then extends to the correlated workflow instances. Therefore, the interorganizational workflow tracking proceeds by propagating status inquiries through related organizations. Details of this tracking mechanism can be found in [29].

## VIII. RELATED WORK AND DISCUSSION

During the last years, various efforts have been devoted to developing business-to-business applications. In this section, we are to briefly review some approaches for collaborative business processes.

The ebXML [30] consortium defined a comprehensive set of specifications for XML document exchange among trading partners for the purpose of preserving electronic data interchange's substantial investments in business processes. However, a business process defined in ebXML mainly concentrated on the exchange of business documents rather than control and data flows. The Partner Interface Process [31] blueprints by RosettaNet specified interactions using Unified Modeling Language (UML) activity diagrams for the Business Operational View and UML sequence diagrams for the Functional Service View in addition to Document Type Definitions for data exchange. However, RosettaNet was primarily focusing on electronic markets with long-lasting prespecified relationships between parties with one party (such as the market maker), imposing rigid business rules.

Works on workflow/process views were related to ours [32], [33]. In regard to structural consistency during the process transformation, Liu and Shen [34] proposed an order-preserving approach for deriving a structurally consistent process view from a base process. In their approach, membership, atomicity, and order preservation rules were used to regulate the transformation. Recently, Eshuis and Grefen [35] formalized the operations of task aggregation and process customization, and they also proposed a series of construction rules for validating the structural consistency. Our approach focused more on the perception relations of organizations in the collaboration. Compared with their work, our approach emphasized the influences toward process structures resulted from the visibility restriction and the derivation of perceptions from business contracts.

To support process privacy and interoperability, many works targeted at applying workflow/process views in the interorganizational collaboration environment. Van der Aalst and Weske [36] proposed a "top-down" workflow modeling scheme in their public-to-private approach. Organizations first agreed on a public workflow, and thereafter, each organization refined the part where it was involved in and finally generated its private workflow. This work reflected a primitive idea of workflow views. In [37], Schulz and Orlowska focused on the cross-organizational interactions and proposed to deploy coalition workflows to compose private workflows and workflow views together to enable interoperability. Dustdar *et al.* [38] extracted an abstract workflow view to describe the choreography of a

collaboration scenario and compose individual workflows into a collaborative business process. By deploying workflow views at the workflow interconnection and cooperation stages, their approach allowed partial visibility of workflows and resources. In supplement to these works, our approach further emphasized the different perceptions of participating organizations. As to the best of our knowledge, our RWF research explicitly investigated the relativity of organizations' perceptions for the first time.

Preuner and Schrefl [39] investigated the generation of collaborative business processes from the perspective of service composition. In their approach, services were modeled as complex processes. Their approach distinguished observable and invocable activities and thereby supported the process tracking and execution in the collaboration environment. Their work also discussed the correctness criteria for composition and automatic composition operations on the basis of behavior diagrams. In comparison, our RWF framework supported collaborations with emphasis on the perception control during the collaboration lifecycle from contracting to workflow modeling and execution.

This paper proposed an RWF model, which observed a collaborative business process from the perspective of individual organizations, to support organizational autonomy and privacy protection. Table II listed the comparisons between our model and other workflow view approaches in terms of collaborative business process construction, visibility derivation, task-hiding techniques, etc.

Our RWF model followed a bottom-up assembling scheme and, therefore, could build a collaborative business process in a browse-and-pick mode. In this mode, an individual organization was allowed to choose partner organizations actively and assemble proper "off-the-shelf" perceivable workflow processes from partner organizations with its own workflow processes into an RWF process. This collaboration mode brought the following appealing features.

- 1) *High autonomy in collaborations*. The RWF model treated each organization as an autonomous entity and empowered it with the full authorization of defining the collaboration structure and behaviors. With this mechanism, organizations were never forced to adapt to the restrictions or irrationalities from a third-party designer or a main contractor. Thus, each organization owned high autonomy in handling its business collaboration.
- 2) *Information protection*. The visibility-control mechanism prevented private information disclosure at both task and process levels. A participating organization was allowed to tune the openness granularity of its internal business processes to different organizations according to different partnerships. Therefore, this mechanism guaranteed the organization's privacy protection, as well as secured the necessary openness.
- 3) *Flexible collaborations*. The browse-and-pick modeling mode freed organizations from the process inflexibility caused by the predefined collaborative business processes. Organizations were allowed to change partner organizations, modify collaboration behaviors, insert or remove proper business processes to or from the collaborative business process, etc. All of these customizations could be done in an *ad hoc* manner.

TABLE II  
COMPARISONS BETWEEN RWF AND OTHER APPROACHES

	CBP construction	Visibility derivation	Org-oriented	Task hiding techniques	Means of supporting cross-org collaboration
Relative WF	bottom-up	contract originated	explicitly org-oriented	composing tasks	via messaging links
P2P	top-down	not mentioned	implicitly	public workflow mapping	via public workflows
Grefen	not mentioned	not mentioned	implicitly	task aggregation / process customisation	via CrossFlow platform
Liu, Shen	bottom-up	not mentioned	Role-oriented	compound task	not mentioned
Schulz	not mentioned	not mentioned	implicitly	task aggregation	via coalition workflow
Issam Dustdar	not mentioned	not mentioned	implicitly	public process mapping	via cooperative processes and public processes
Preuner, Schrefl	Bottom-up	not mentioned	implicitly	Composition and abstraction	Via composite services

- 4) *Advanced information-hiding mechanism.* The proposed constraint-based visibility-control mechanism provided an adjustable information-hiding solution. This visibility-control mechanism well distinguished the diverse partnerships and authority levels between collaborating organizations. Other research works, such as workflow view [40] and public-to-private [36] approaches, attempted to support information hiding with partial workflow views and private processes, respectively. However, these artifacts only provided a primitive information-hiding solution. Neither of them took into account the relations between the process visibility and the organization partnerships and therefore failed to reflect the diversity of partnerships or authority levels.

However, the migration to RWF management may bring some tradeoffs, which can be potential limitations. Some tradeoffs and deducted limitations are summarized as follows, although they may be outweighed by many advantages offered by our methodology.

- 1) In the RWF context, different organizations deploy different collaborative business processes for the same collaboration. The inconsistency between these collaborative business processes inevitably results in the complex coordination between participating organizations. In a practical application environment, this may require extra functionalities for process storage and coordination. In the proposed system architecture, special components such as the RWF process database, the perceivable workflow database, and the local workflow process database are designed for business-process-definition storage. Other components, such as the workflow coordinator and the correlation database of Workflow Engine Service, are dedicatedly designed for process coordination.
- 2) The free assembling mechanism for RWF process generation relies on a publishing component and a discovery component for accessing perceivable workflow processes. In the proposed system architecture, the component Business Process Directory Service is dedicatedly designed to fulfil this function.
- 3) The extraction of visibility constraints from commercial contracts assumes that all contracts conform to the COSMOS format. This assumption may not stand in some cases, since many contracts follow different templates. Therefore, the automatic conversion algorithm is only applicable in limited situations. Instead, extra human

efforts may be required for interpreting the contracts to perceptions.

## IX. CONCLUSION AND FUTURE WORK

This paper has presented a new approach on aligning collaborative business processes with an RWF model. In this approach, each organization acts as an autonomous entity with the full control of choosing its partner organizations and defining its collaboration structure and behaviors. Instead of defining a common collaborative business process for all collaboration participants, each participating organization may define its RWF processes from its own perspective. Associated with an RWF process, a set of visibility constraints is defined to adjust the granularity of process visibility. In this paper, both the formal framework of the RWF model and the architecture of the facilitating system have been presented. We are to further this work by establishing a set of rules for verifying and validating the conformity of RWF processes in terms of process choreography.

On the other hand, the advantages of the organization-oriented view methodology are achieved at the cost of some compromises. Extra attention should be paid to these tradeoffs in future research to counteract and minimize these limitations.

## APPENDIX I (PROOF OF PROPERTIES)

### A. Property 1 (Necessity)

An RWF process contains necessary information for the host organization to accomplish its responsibilities in the participated business collaboration.

*Proof:* According to the contract model introduced in Section III, the responsibilities of an organization in the participated collaborative business process are defined in the *What* and *Legal Clause* parts of contracts. Furthermore, the *How* part describes the execution details for the content defined in *What* and *Legal Clause* using business interactions. These business interactions are thereafter converted into messaging interactions between the tasks that are set “contactable” in proper perceptions. With the perceptions defined for a specific organization, this organization can see all the contactable tasks of its partner organizations. The perceptions also provide necessary interface specifications, such as the message descriptions combined with the interfaces. The RWF process generated from these perceptions inherits all these pieces of information. Therefore, such an RWF process includes the necessary information for the host organization to fulfil its responsibilities in the collaboration. ■



## B. Property 2 (Sufficiency)

A collaborative business process can be sufficiently represented by a finite number of RWF processes defined for participating organizations.

*Proof:* This property emphasizes that a collaborative business process in a public view can be covered by a group of RWF processes, although each of these RWF processes only represents a partial view over the whole collaboration. To prove this, we first represent the structure of a public-view collaborative business process, for example,  $cbp$ , as a graph  $(\mathcal{N}, \mathcal{A})$ , where set  $\mathcal{N}$  denotes the set of involved workflow tasks and set  $\mathcal{A}$  denotes the set of all links.

In addition,  $\mathcal{A}$  contains two kinds of links, viz., a set of intraprocess links, for example,  $\mathcal{A}_{intra}$ , and a set of interprocess links, for example,  $\mathcal{A}_{inter}$ .

Based on the definition of an RWF process in Section III, each RWF process can also be represented as a graph  $(\mathcal{T}, \mathcal{R})$ , where  $\mathcal{T} = \mathcal{T}_L \cup \mathcal{T}_p$  and  $\mathcal{R} = \mathcal{R}_L \cup \mathcal{R}_p \cup \mathcal{L}_{intra} \cup \mathcal{L}_{inter}$ .

The tasks of local workflow processes are totally visible to the host organization, and therefore, all the tasks of a collaborative business process  $cbp$  can be obtained from the tasks of local workflow processes belonging to a group of RWF processes. This can be formalized as follows:

$$(\forall cbp) \exists RWF \left( cbp.\mathcal{N} \subseteq \bigcup_{rwf \in RWF} rwf.\mathcal{T}_L \right). \quad (1)$$

Here, set  $RWF$  denotes a set of RWF processes.

Given that the tasks of local workflow processes are all available, the intraprocess links between these tasks are also obtainable from these RWF processes, due to the definition of intraprocess links, i.e.,  $\mathcal{R}_L \subseteq \mathcal{T}_L \times \mathcal{T}_L$ . Here, we formalize this finding as the following:

$$(\forall cbp) \exists RWF \left( cbp.\mathcal{A}_{intra} \subseteq \bigcup_{rwf \in RWF} rwf.\mathcal{R}_L \right). \quad (2)$$

Regarding a specific RWF process, for example,  $rwf$ , of organization  $g$ , it includes the set of inter process links connecting a task of a perceivable workflow process and a task of a local workflow process. This means that set  $rwf.\mathcal{L}_{inter}$  includes the links that connect the tasks of  $g$ 's local workflow processes to the tasks of workflow processes belonging to  $g$ 's neighboring organizations in a public view. As such, a finite number of RWF processes, at most all the RWF processes of all participating organizations, will definitely cover the links between two workflow processes belonging to different organizations in a collaborative business process. Therefore, we can formalize this finding as the following:

$$(\forall cbp) RWF \left( cbp.\mathcal{A}_{inter} \subseteq \bigcup_{rwf \in RWF} rwf.\mathcal{L}_{inter} \right). \quad (3)$$

Based on (1)–(3), we can finally draw the following conclusion:

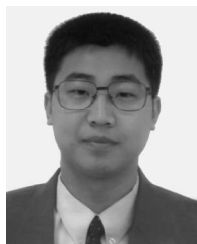
$$(\forall cbp) \exists RWF \left( cbp.\mathcal{N} \subseteq \bigcup_{rwf \in RWF} rwf.\mathcal{T}_L \right) \wedge \left( cbp.\mathcal{A} \subseteq \bigcup_{rwf \in RWF} (rwf.\mathcal{L}_{inter} \cup rwf.\mathcal{R}_L) \right).$$

This shows that Property 2 stands. ■

## REFERENCES

- [1] H. Osterle, E. Fleisch, and R. Alt, *Business Networking—Shaping Collaboration Between Enterprises*. New York: Springer-Verlag, 2001.
- [2] C. Bussler, *B2B Integration*. New York: Springer-Verlag, 2003.
- [3] H. Smith and P. Fingar, *Business Process Management—The Third Wave*. Tampa, FL: Meghan-Kiffer, 2003.
- [4] M. Weske, W. M. P. van der Aalst, and H. M. W. Verbeek, “Advances in business process management,” *Data Knowl. Eng.*, vol. 50, no. 1, pp. 1–8, Jul. 2004.
- [5] J. Krogstie, A. L. Opdahl, and S. Brinkkemper, *Conceptual Modelling in Information Systems Engineering*. New York: Springer-Verlag, 2007.
- [6] P. Grefen, K. Aberer, H. Ludwig, and Y. Hoffner, “CrossFlow: Cross-organizational workflow management for service outsourcing in dynamic virtual enterprises,” *Data Eng.*, vol. 24, no. 1, pp. 52–57, 2001.
- [7] I. Wetzel and R. Klischewski, “Serviceflow beyond workflow? IT support for managing inter-organizational service processes,” *Inf. Syst.*, vol. 29, no. 2, pp. 127–145, Apr. 2004.
- [8] C. Hsu, D. M. Levermore, C. D. Carothers, and G. Babin, “Enterprise collaboration: On-demand information exchange using enterprise databases, wireless sensor networks, and RFID systems,” *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 37, no. 4, pp. 519–532, Jul. 2007.
- [9] C. Liu, Q. Li, and X. Zhao, “Challenges and opportunities in collaborative business process management,” *Inf. Syst. Front.*, vol. 11, no. 2, pp. 201–209, Jul. 2009.
- [10] M. M. Cunha and G. D. Putnik, “Market of resources for agile/virtual enterprise integration,” in *Encyclopedia of Information Science and Technology*, vol. IV, M. Khosrow-Pour, Ed. Hershey, PA: Idea Group, 2005, pp. 1891–1898.
- [11] J. K. Debenham, “Interacting with electronic institutions,” in *Proc. 15th Int. Conf. Database Expert Syst. Appl.*, Zaragoza, Spain, 2004, pp. 181–190.
- [12] E. A. Stohr and J. L. Zhao, “Workflow automation: Overview and research issues,” *Inf. Syst. Front.*, vol. 3, no. 3, pp. 281–296, Sep. 2001.
- [13] A. Dogac and I. Cingil, “A survey and comparison of business-to-business e-commerce frameworks,” *ACM SIGecom Exchanges*, vol. 2, no. 2, pp. 14–25, Jun. 2001.
- [14] T. Weitzel, “Process governance and optimization for IT reliant business processes: An empirical analysis of financial processes in Germany's fortune 1,000 non-banks,” in *Proc. 39th Hawaii Int. Int. Conf. Syst. Sci.*, Kauai, HI, 2006, CD-ROM.
- [15] G. L. Shostack, “Designing services that deliver,” *Harvard Bus. Rev.*, vol. 62, no. 1, pp. 133–139, Jan./Feb. 1984.
- [16] S. K. Katsikas, J. Lopez, and G. Pernul, “Trust, privacy and security in e-business: Requirements and solutions,” in *Proc. Panhellenic Conf. Informatics*, Volos, Greece, 2005, pp. 548–558.
- [17] D. Chen and N. Daclin, “Barriers driven methodology for enterprise interoperability,” in *Proc. 8th IFIP Working Conf. Virtual Enterprises*, Guimarães, Portugal, 2007, pp. 453–460.
- [18] M. Anderson and R. Allen, “Workflow interoperability—Enabling e-commerce,” Workflow Management Coalition, Cohasset, MA, White paper, 1999.
- [19] X. Zhao, C. Liu, and Y. Yang, “An organisational perspective on collaborative business processes,” in *Proc. Int. Conf. Bus. Process Manag.*, Nancy, France, 2005, pp. 17–31.
- [20] M. Rouached, O. Perrin, and C. Godart, “A contract layered architecture for regulating cross-organisational business processes,” in *Proc. 3rd Int. Conf. Bus. Process Manag.*, Nancy, France, 2005, pp. 410–415.
- [21] A. Berry and Z. Milosevic, “Extending choreography with business contract constraints,” *Int. J. Coop. Inf. Syst.*, vol. 14, no. 2/3, pp. 131–179, 2005.
- [22] M. Merz, F. Griffel, M. T. Tu, S. Müller-Wilken, H. Weinreich, M. Boger, and W. Lamersdorf, “Supporting electronic commerce transactions with contracting services,” *Int. J. Coop. Inf. Syst.*, vol. 7, no. 4, pp. 249–274, 1998.
- [23] F. Griffel, M. Boger, H. Weinreich, W. Lamersdorf, and M. Merz, “Electronic contracting with COSMOS—How to establish, negotiate and execute electronic contracts on the Internet,” in *Proc. 2nd Int. Enterprise Distrib. Object Comput. Workshop*, 1998, pp. 46–55.
- [24] J. Zhang, C. K. Chang, L.-J. Zhang, and P. C. K. Hung, “Toward a service-oriented development through a case study,” *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 37, no. 6, pp. 955–969, Nov. 2007.
- [25] JAX-WS, Java API for XML Web Services, 2006.
- [26] JAXB, Java Architecture for XML Binding, 2006.
- [27] SAAJ, SOAP With Attachments API for Java, 2003.
- [28] X. Zhao, C. Liu, Y. Yang, and W. Sadiq, “Handling instance correspondence in inter-organisational workflows,” in *Proc. 19th Int. Conf. Adv. Inf. Syst. Eng.*, Trondheim, Norway, 2007, pp. 51–65.

- [29] X. Zhao and C. Liu, "Tracking over collaborative business processes," in *Proc. Int. Conf. Bus. Process Manag.*, Vienna, Austria, 2006, pp. 33–48.
- [30] ebXML, 2002. [Online]. Available: <http://ebxml.org>
- [31] RosettaNet, 2001. [Online]. Available: <http://www.rosettanet.org/>
- [32] X. Zhao, C. Liu, W. Sadiq, and M. Kowalkiewicz, "Process view derivation and composition in a dynamic collaboration environment," in *Proc. Int. Conf. Coop. Inf. Syst.*, Monterrey, Mexico, 2008, pp. 82–99.
- [33] X. Zhao, C. Liu, W. Sadiq, M. Kowalkiewicz, and S. Yongchareon, "WS-BPEL business process abstraction and concretisation," in *Proc. Int. Conf. Database Syst. Adv. Appl.*, Brisbane, Australia, 2009, pp. 555–569.
- [34] D. R. Liu and M. Shen, "Workflow modeling for virtual processes: An order-preserving process-view approach," *Inf. Syst.*, vol. 28, no. 6, pp. 505–532, Sep. 2003.
- [35] R. Eshuis and P. Grefen, "Constructing customized process views," *Data Knowl. Eng.*, vol. 64, no. 2, pp. 419–438, Feb. 2008.
- [36] W. M. P. van der Aalst and M. Weske, "The P2P approach to inter-organizational workflows," in *Proc. Int. Conf. Adv. Inf. Syst. Eng.*, 2001, pp. 140–156.
- [37] K. A. Schulz and M. E. Orlowska, "Facilitating cross-organisational workflows with a workflow view approach," *Data Knowl. Eng.*, vol. 51, no. 1, pp. 109–147, Oct. 2004.
- [38] I. Chebbi, S. Dustdar, and S. Tata, "The view-based approach to dynamic inter-organizational workflow cooperation," *Data Knowl. Eng.*, vol. 56, no. 2, pp. 139–173, Feb. 2006.
- [39] G. Preuner and M. Schrefl, "Requester-centered composition of business processes from internal and external services," *Data Knowl. Eng.*, vol. 52, no. 1, pp. 121–155, Jan. 2005.
- [40] D. K. W. Chiu, K. Karlapalem, Q. Li, and E. Kafeza, "Workflow view based e-contracts in a cross-organizational e-services environment," *Distrib. Parallel Databases*, vol. 12, no. 2/3, pp. 193–216, Sep.–Nov. 2002.



**Xiaohui Zhao** received the B.Eng. and M.Eng. degrees in computer science from the Harbin Institute of Technology, Harbin, China, in 2001 and 2003, respectively, and the Ph.D. degree in computer science from the Swinburne University of Technology, Melbourne, Australia, in 2007.

He is currently a Research Fellow with the Web and Data Engineering Research Group, Centre for Complex Software Systems and Services, Faculty of Information and Communication Technologies, Swinburne University of Technology, Melbourne,

Australia. He has published around 20 peer-reviewed papers in international journals and conferences. His research interests include business process management, pervasive computing, service-oriented computing, etc.

Dr. Zhao has organized a series of workshops and conferences.



**Chengfei Liu** received the B.S., M.S., and Ph.D. degrees in computer science from Nanjing University, Nanjing, China, in 1983, 1985, and 1988, respectively.

He is currently a Full Professor and the Head of the Web and Data Engineering Research Group, Centre for Complex Software Systems and Services, Faculty of Information and Communication Technologies, Swinburne University of Technology, Melbourne, Australia. Prior to joining Swinburne in 2004, he taught at the University of South Australia,

Adelaide, and the University of Technology Sydney, Sydney, Australia, and was a Research Scientist with the Cooperative Research Centre for Distributed Systems Technology (DSTC), Brisbane, Australia. He also held visiting positions at the Chinese University of Hong Kong, Shatin, Hong Kong, the University of Aizu, Aizuwakamatsu, Japan, and the IBM Silicon Valley Laboratory, San Jose, CA. He has published more than 100 peer-reviewed papers in various journals and conference proceedings and has served on technical program committees and organizing committees of about 60 international conferences or workshops in the areas of database systems, Web information systems, and workflow systems. He was a Guest Coeditor for two special issues for the *Journal of Database Management* and the *Information Systems Frontiers*.

Dr. Liu is a recipient of the Vice Chancellor's Award for Research Excellence at Swinburne University of Technology in 2007.



**Yun Yang** was born in Shanghai, China. He received the M.Eng. degree in computer science from The University of Science and Technology of China, Hefei, China, in 1987 and the Ph.D. degree in computer science from The University of Queensland, Brisbane, Australia, in 1992.

He is currently a Full Professor and the Associate Dean (Research) of the Faculty of Information and Communication Technologies, Swinburne University of Technology, Melbourne, Australia. Prior to joining Swinburne University of Technology as an

Associate Professor in late 1999, he was a Lecturer and Senior Lecturer with Deakin University, Geelong, Australia, during 1996–1999. Before that, he was a Research Scientist with the Cooperative Research Centre for Distributed Systems Technology (DSTC) during 1993–1996. He was also with Beihang University, Beijing, China, during 1987–1988. He has published more than 160 papers on journals and refereed conferences. His research interests include software engineering, peer-to-peer, grid, and cloud computing-based workflow systems, service-oriented computing, Internet computing applications, and computer-supported cooperative work.



**Wasim Sadiq** received the Ph.D. degree in computer science from the University of Queensland, Brisbane, Australia, in the area of conceptual modeling and verification of workflows.

He is the Deputy Research Director of SAP Research Centre, Brisbane, Australia. He is a core team member of the SAP Research Portfolio Office responsible for architecting the global research strategy and portfolio. He has over 22 years of research and development experience in the areas of enterprise applications, business process management,

workflow technology, service-oriented architectures, database management systems, distributed systems, and e-learning. He has published and presented several research papers in leading international conferences and journals and has filed more than 20 patent applications. He has led several research projects collaborating with academic and industry partners in Australia, Europe, and the U.S. He has been actively contributing in the internal strategic research and development of next-generation SAP enterprise applications. He is proficient in a variety of software development technologies and environments.