

Discovering Activity-Performer Affiliation Knowledge on ICN-based Workflow Models*

KWANGHOON PIO KIM

*Collaboration Technology Research Laboratory
Department of Computer Science
Kyonggi University
Kyonggi-do, 443-760 Korea*

Technology-supported social networks have been penetrating many aspects of our lives from friendships/blogging sites to working organizations. Particularly, individuals as employees of companies have started to adopt a sort of organizational knowledge based on their operational technologies. This paper, so, focuses on a special type of organizational social network knowledge acquired from deploying workflow technologies, which is dubbed ‘*activity-performer affiliation knowledge*.’ That is, the paper theoretically derives a series of concepts and algorithms not only for representing and discovering the knowledge but also for analyzing the discovered knowledge. These theoretical concepts and related algorithms are based upon the methodology of information control net workflow models, and the discovered knowledge eventually represents involvements and participation relationships between a group of performers and a group of activities in workflow models. Finally, we summarily describe the implications of the activity-performer affiliation knowledge, and how much it is worth discovering them in workflow-driven organizations and enterprises producing massively parallel interactions and large-scaled operational data collections.

Keywords: information control net, workflow model, workflow-supported affiliation networking knowledge, activity-performer affiliation, bipartite graph and matrix, density and degree-centrality analysis

1. INTRODUCTION

In general, a workflow management system consists of two components|modeling component and enacting component. The modeling component allows a modeler to define, analyze and maintain workflow¹ models by using all of the workflow entities that are necessary to describe work procedures, and the enacting component supports users to play essential roles of invoking, executing and monitoring instances of the workflow model defined by the modeling component. Particularly, the logical foundation of the workflow management system is based upon the modeling component which is called workflow model. Until now, many workflow models have been proposed in the workflow literature, and almost all commonly employ the five essential entity-types, such as activity, role, actor, repository and application entity-types, to represent organizational works and their procedural collaborations. These entity-types eventually become reflecting the typical people-oriented organizational perspectives like behavioral, social, infor-

Received May 31, 2011; accepted March 31, 2012.

Communicated by Francisco J. García-Peñalvo, Ricardo Colomo-Palacios and Jane Yung-Jen Hsu.

* This work (Grants No. 00047962) was supported by Business for Cooperative R&D between Industry, Academy, and Research Institute funded by the Korea Small and Medium Business Administration in 2011.

¹ In terms of the terminological usage, the term, workflow, can be interchangeably used with the term, business process. We prefer the former to the latter in this paper.

mational, collaborative, and historical perspectives, onto workflow models. Therefore, we can conclude that these workflow management systems are “human conceptual systems” that must be not only designed, deployed, and understood within their social and organizational contexts, but also conveyed, facilitated, and navigated in large-scaled operational knowledge collections.

More recently, individuals as employees and their companies have started to adopt the concept of social networks both intra-organizationally [20-22] and inter-organizationally [23, 24]. The technology to support such social networking, so, is diverse ranging from the standard desktop to immerse virtual environments, and other applications and services. Likewise, the workflow literature starts being interested in “social networking.” It begins from the strong belief that social relationships and collaborative behaviors among employees who are involved in enacting the specific workflow models affect the overall performance and being crowned with great successes in the real businesses and the working productivity as well. Consequently, applying the concept of social network and its analysis methods to workflows, which is termed work-flow-supported social network, has been emerging in the literature. Workflow-supported affiliation network, which is the main issue of this paper, is a special type of workflow-supported social networks; activity-performer affiliation knowledge is an essential knowledge that can be discovered from workflow-supported affiliation networks, which is also called workflow membership network and represents the involvement of a set of performers in a set of activities in a workflow model. In a workflow model, performers (or actors) are linked through their joint participation in activities or by their common relationship in organizations. Conversely, workflow activities are connected to the extent that they have performers in common. Eventually, through the activity-performer affiliation knowledge, it is possible to visualize how performers and activities are simultaneously interrelated in a workflow model.

There, of course, might exist two main research issues in workflow-supported affiliation networks. One is a knowledge *discovery* issue, the other has something to do with a knowledge *rediscovery* issue. The latter is concerned with mining workflow-supported affiliation networking knowledge from workflow enactment event logs; the former is to discover workflow-supported affiliation networking knowledge through exploring a certain type of associations among the entity-types of workflow models, such as activity-performer association, activity-application association, activity-role association, role-performer association, and model-performer association. More specifically, the paper would differentiate the former from the latter, and be narrow scoping to the activity-performer entity-types of associations. In other words, the paper tries to discover a defined activity-performer affiliation networking knowledge (workflow build-time aspect) embedded in a workflow model, and it, so, gives a series of concepts and algorithms for discovering activity-performer affiliation networking knowledge from an ICN-based workflow model.

In terms of making up the paper, the next section gives the technological backgrounds, mainly focusing on the ICN-based workflow model and its perspectives. And the next consecutive section describes the details of activity-performer affiliation networking knowledge, like its representation, discovery, and analytics. Finally, we give a summary with a brief description of its related works and conclusions including future works.

2. BACKGROUNDS: ICN-BASED WORKFLOW MODEL

This section shortly introduces the basic concept of ICN-based workflow model [1-3, 17] as a technological background. In describing the ICN-based workflow model, we start from defining a workflow meta-model [17] that is theoretical basis of the ICN-based workflow model, and next we introduce the graphical notations and their formal representations of the model.

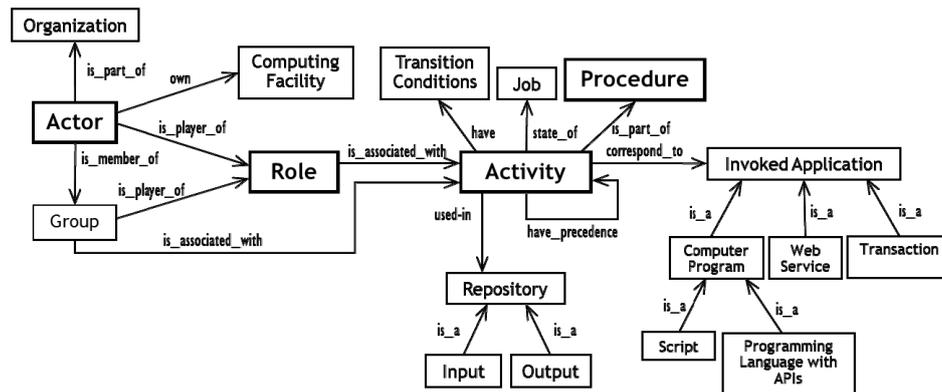


Fig. 1. The workflow meta-model.

2.1 The Workflow Meta-Model

In describing an ICN-based workflow model, we would use the basic workflow terminology, such as workflow procedure, activity, job, workcase, role, actor/group, and invoked application including web services. These terms become the primitive entity types to be composed into an ICN-based workflow model, and also they have appropriate relationships with each other, as shown in Fig. 1. The followings are the basic definitions of the primitive entity types:

- A **workflow procedure** is defined by a predefined or intended set of tasks or steps, called activities, and their temporal ordering of executions. A workflow management system helps to organize, control, and execute such defined workflow procedures. Conclusively, a workflow procedure can be described by a temporal order of the associated activities through the combinations of sequential logics, conjunctive logics (after activity A, do activities B and C), disjunctive logics (after activity A, do activity B or C), and loop logics.
- An **activity** is a conceptual entity of the basic unit of work (task or step), and the activities in a workflow procedure have precedence relationships, each other, in terms of their execution sequences. Also, the activity can be precisely specified by one of the three entity types|compound activity, elementary activity and gateway activity. The compound activity represents an activity containing another workflow procedure, which is called subworkflow. The elementary activity is an activity that can be realized by a computer program, such as application program, transaction, script, or web ser-

vice. And, the gateway activities imply those activities that are used to control the execution sequences of elementary/compound activities. The types of gateway activities consist of conjunctive gateway (after activity A, do activities B and C), disjunctive gateway (after activity A, do activity B or C), and loop gateway. Particularly, both the disjunctive gateway and the loop gateway need to be set some specific **transition conditions** in order to select one of the possible transition pathes during the execution time. The transition condition itself can be defined by using the input/output relevant data on the **repository**. Additionally, each activity has to be associated with a real performer, such as organizational staff (role, participant) and system, who possesses all ownerships over that activity.

- A **role**, as a logical unit of the organizational structure, is a named designator for one or more participants, which conveniently acts as the basis for participating works, skills, access controls, execution controls, authority, and responsibility over the associated activity.
- An **actor** is a person, program, or entity that can fulfill roles to execute, to be responsible for, or to be associated in some way with activities and workflow procedures.
- Multiple instances of a workflow procedure may be in various stages of execution. Thus, the workflow procedure can be considered as a class (in object oriented terminology), and each execution, called a **workcase**, can be considered an instance. A workcase is thus defined as the locus of control for a particular execution of a workflow procedure.
- An **invoked application program** that automatically performs the associated activity, or provides automated assistance within hybrid activities are called **scripts**. If an activity is executed in automatic or hybrid mode, this means that whole/part of the invoked application program associated with the activity is automatically launched by a workflow enactment service.
- Finally, a **repository** is a set of input and output relevant data of an activity. Eventually, the repository provides a communication channel between the workflow enactment domain and the invoked application programs domain. That is, the input and the output repositories are used to realizing the input parameters and the output parameters of the associated invoked application program, respectively.

2.2 Information Control Net

An ICN-based workflow model can be defined by capturing the notations of workflow procedures, activities and their control precedence, invoked applications, roles, actors, and input/output repositories, as explained in the previous section of the workflow meta-model. In this section, we define the basic concept of workflow model with respect to the formal and graphical descriptions of ICN-based workflow model. The following **Definition 1** is a formal definition of ICN-based workflow model, and its functional components to be used for retrieving workflow-related information, such as activity precedence (control flow), activity-role association, activity-relevant data association (data flow), activity-invoked application association, activity-transition condition association, and role-actor association information. Based upon these types of information, it is possible to retrieve several types of derived workflow-related information like activity-actor association, relevant data-invoked application association, role complexity, ac-

tor complexity information, and so forth.

Definition 1 Information Control Net (ICN) for formally defining workflow model. A basic ICN is 9-tuple $\Gamma = (\delta, \rho, \gamma, \lambda, \varepsilon, \pi, \kappa, \mathbf{I}, \mathbf{O})$ over a set of \mathbf{A} activities (including a set of group activities), a set of $\mathbf{E} \subseteq (\mathbf{A} \times \mathbf{A})$ edges (pairs of activities), a set \mathbf{T} of transition conditions, a set \mathbf{R} of repositories, a set of \mathbf{G} of invoked application programs, a set of \mathbf{P} of roles, and a set of \mathbf{C} of actors (including a set of actor groups), where $\wp(\mathbf{A})$ represents a power set of the activity set, \mathbf{A} :

- \mathbf{I} is a finite set of initial input repositories, assumed to be loaded with information by some external process before execution of the ICN;
- \mathbf{O} is a finite set of final output repositories, perhaps containing information used by some external process after execution of the ICN;
- $\delta = \delta_i \cup \delta_o$
where $\delta_o: \mathbf{A} \rightarrow \wp(\mathbf{R})$ is a multi-valued mapping function from an activity to its sets of (immediate) successors, and $\delta_i: \mathbf{A} \rightarrow \wp(\mathbf{R})$ is a multi-valued mapping function from an activity to its sets of (immediate) predecessors;
- $\rho = \rho_i \cup \rho_o$
where $\rho_o: \mathbf{A} \rightarrow \wp(\mathbf{A})$ is a single-valued mapping function from an activity to its set of output repositories, and $\rho_i: \mathbf{A} \rightarrow \wp(\mathbf{A})$ is a single-valued mapping function from an activity to its set of input repositories;
- $\gamma = \gamma_i \cup \gamma_o$
where $\gamma_o: \mathbf{R} \rightarrow \wp(\mathbf{A})$ is a single-valued mapping function from a repository to its set of out-degree activities, and $\gamma_i: \mathbf{R} \rightarrow \wp(\mathbf{A})$ is a single-valued mapping function from a repository to its set of indegree activities;
- $\lambda = \lambda_a \cup \lambda_g$
where $\lambda_g: \mathbf{A} \rightarrow \mathbf{G}$ is a single-valued mapping function from an activity to its invoked application program, and $\lambda_a: \mathbf{G} \rightarrow \wp(\mathbf{A})$ is a single-valued mapping function from an invoked application program to its set of associated activities;
- $\varepsilon = \varepsilon_a \cup \varepsilon_p$
where $\varepsilon_p: \mathbf{A} \rightarrow \mathbf{P}$ is a single-valued mapping function from an activity to a role, and $\varepsilon_a: \mathbf{P} \rightarrow \wp(\mathbf{A})$ is a single-valued mapping function from a role to its set of associated activities;
- $\pi = \pi_p \cup \pi_c$
where $\pi_c: \mathbf{P} \rightarrow \wp(\mathbf{C})$ is a single-valued mapping function from a role to its set of associated actors, and $\pi_p: \mathbf{C} \rightarrow \wp(\mathbf{P})$ is a single-valued mapping function from an actor to its set of associated roles;
- $\kappa = \kappa_i \cup \kappa_o$
where $\kappa_i: \mathbf{E} \rightarrow \wp(\mathbf{T})$ is a single-valued mapping function from an edge to a set of control-transition conditions; and $\kappa_o: \mathbf{T} \rightarrow \wp(\mathbf{E})$ is a single-valued mapping function from a control-transition condition to a set of edges.

3. DISCOVERING ACTIVITY-PERFORMER AFFILIATION KNOWLEDGE

This section starts from introducing the basic concept of activity-performer affiliation

networking knowledge, and its graphical and formal representations. Next, it devises a knowledge discovering algorithm and a bipartite matrix generation algorithm to discover activity-performer affiliation networking knowledge from an ICN-based work-flow model, and to analyze the discovered activity-performer affiliation networking knowledge, respectively.

3.1 Activity-Performer Affiliation Knowledge

As stated in the previous section, workflow models employ five essential entity-types, such as activity, role, actor, repository and application entity-types, to represent organizational works and their procedural collaborations. A certain pattern of associations and affiliations among these entity-types eventually can be discovered to reveal people-oriented organizational perspectives, like behavioral, social, informational, collaborative, and historical perspectives, on the underlying workflow models and their enactment systems. This paper is interested in discovering the social perspective embedded in a workflow model. The social perspective on a workflow model can be revealed in a form of either social networking knowledge or affiliation networking knowledge. So, we would define in this paper that the workflow-supported social networking knowledge implies dyadic network of linkages among performers, and that the workflow-supported affiliation networking knowledge implies non-dyadic network of collectivities linked through multiple memberships of performers.

The paper, particularly, focuses on the affiliation networking knowledge formed by two key elements, as shown in Fig. 2, a set of performers and a collection of subsets of

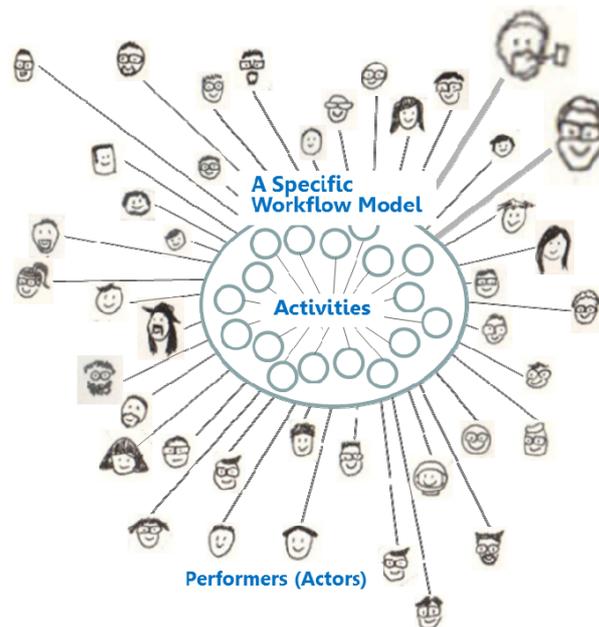


Fig. 2. The workflow-supported (activity-performer) affiliation knowledge.

performers (called activities) in a workflow model. An affiliation network to be discovered from a workflow model is non-dyadic because the affiliation relation relates each performer to a subset of activities, and relates each activity to a subset of performers. From the figure, it might be quite in the nature of things to raise a question as followings:

- Which performers are linked to each other as members of collectivities (activities), and which collectivities are linked to each other through shared performers in the specific workflow procedure?

Conclusively, the answer for the question ought to be able to convey a wide variety of very valuable and meaningful insights to the organizational knowledge, which is the primary rationale in the concept of discovering workflow-supported affiliation networking knowledge. In other words, from the discovered organizational knowledge of activity-performer affiliations, it is possible to visualize how actors and activities are simultaneously interrelated (involvement and participation) in the specific workflow model. The main purpose of this paper is to theoretically develop a means for discovering and visualizing performer-activity affiliation knowledge from an ICN-based workflow model.

3.2 Knowledge Representation: Activity-Performer Affiliation Network Model

In order to represent the workflow-supported activity-performer affiliation knowledge, the paper newly defines a graphical (Bipartite Graph) and formal representation model, which is dubbed activity-performer affiliation network model. An activity-performer affiliation network model, which is abbreviated as APANM, consists of two types of nodes – a set of performers and a set of activities – and a set of relations between nodal types. Thus, an activity-performer affiliation network is a two-mode network, through which it used to accomplish the following dual objectives:

- to uncover the relational structures of workflow-performers through their joint involvement in activities, and
- to reveal the relational structures of workflow-activities through their joint participation of common performers.

Definition 2: Activity-performer Affiliation Network Model An activity-performer affiliation network model is formally defined as $\Lambda = (\sigma, \psi, \mathbf{S})$, over a set \mathbf{C} of performers (actors), a set \mathbf{A} of activities, a set \mathbf{V} of weight-values, a set $\mathbf{E}_p \subseteq (\mathbf{C} \times \mathbf{A})$ of edges (pairs of performers and activities), and a set $\mathbf{E}_a \subseteq (\mathbf{A} \times \mathbf{C})$ of edges (pairs of activities and performers), where, $\wp(\mathbf{A})$ represents a power set of the activity set, \mathbf{A} :

- \mathbf{S} is a finite set of work-sharing actors or groups of some external activity-performer affiliation network models;
- $\sigma = \sigma_p \cup \sigma_v$ /* Involvement Knowledge */
where, $\sigma_p: \mathbf{C} \rightarrow \wp(\mathbf{A})$ is a single-valued mapping function from a performer to its set of involved activities; $\sigma_v: \mathbf{E}_p \rightarrow \mathbf{V}$ is a single-valued mapping function from an edge ($\in \mathbf{E}_p$) to its weight-value;
- $\psi = \psi_a \cup \psi_v$ /* Participation Knowledge */

where, $\psi_a: \mathbf{A} \rightarrow \wp(\mathbf{C})$ is a single-valued mapping function from an activity to a set of participated performers; and $\psi_v: \mathbf{E}_a \rightarrow \mathbf{V}$ is a single-valued function from an edge ($\in \mathbf{E}_a$) to its weight-value.

Additionally, those relational structures can be weighed to measure the extent of their strengths by assigning a value to each of relations between nodal types. Therefore, there are two types of activity-performer affiliation networks – binary activity-performer affiliation network and valued activity-performer affiliation network. In the binary activity-performer affiliation network, its value (0 or 1) implies a binary relationship of involvement (or participation), while values in the valued activity-performer affiliation network may represent various implications according to their application domains; typical examples of values might be stochastic (or probabilistic) values, strengths, and frequencies. The formal knowledge representation of activity-performer affiliation network model is defined in Definition 2.

And the graphical knowledge representation is depicted by an affiliation graph. So, an activity-performer affiliation network's graphical model consists of two types of graphical nodes – a set of performers (shaped in hexagon) and a set of workflow activities (shaped in circle) – and a set of non-directed edges between two nodal types, which means that a workflow affiliation network is a non-directed graph. That is, in an activity-performer affiliation graph, non-directed lines connect performers aligned on one side of the diagram to the workflow activities aligned on the other side. Importantly, an activity-performer affiliation graph does not permit lines among the performers nor among the workflow activities. Therefore, an activity-performer affiliation graph with 'g' performers and 'h' workflow activities can be transformed into a matrix with 2-dimension of 'g × h'.

3.3 Knowledge Discovery: Activity-Performer Affiliation Knowledge Discovering Algorithm

At this moment, it is important to emphasize that activity-performer affiliation networking knowledge would not be modeled or designed but be automatically discovered from workflow procedures. So, this paper devises an algorithmic discovery methodology to discover activity-performer affiliation knowledge, represented by an activity-performer affiliation network model, by exploring the internal social perspectives – ε_p (activity-role mapping information) and π_c (role-actor mapping information) – of an ICN-based workflow model. Likewise, we have to remind that it shouldn't be differentiated the single-actor binding activity type from the group-actor binding activity (real-time groupware activity) type, where a group of actors is simultaneously assigned to cooperatively perform a single activity; almost all current available workflow models do not support such a real-time groupware activity type. However, as a future work, we need to cope with these social relationships caused from the group-actor binding activities in discovering activity-performer affiliation knowledge. The following is the algorithm to automatically discover an activity-performer affiliation network model from an ICN-based workflow model:

Activity-performer Affiliation Knowledge Discovering Algorithm:**Input** An ICN, $\Gamma = (\delta, \rho, \gamma, \lambda, \varepsilon, \pi, \kappa, \mathbf{I}, \mathbf{O})$;**Output** A Binary Activity-performer Affiliation Network Model, $A = (\sigma, \psi, \mathbf{S})$;

```

Begin Procedure
  For ( $\forall \phi \in \mathbf{C}$  (performer-set)) Do /*  $\sigma = \sigma_p \cup \sigma_v$ : Involvement Knowledge */
    Begin
      Add all members of  $(\varepsilon_a(\pi_p(\phi)))$  To  $\sigma_p(\phi)$ ;
      Add "weight-value  $\leftarrow 1$ " To  $\sigma_v$ (all edges of  $(\phi, \sigma_p(\phi))$ );
    End
  For ( $\forall \alpha \in \mathbf{A}$  (activity-set)) Do /*  $\psi = \psi_a \cup \psi_v$ : Participation Knowledge */
    Begin
      Add all members of  $(\pi_e(\varepsilon_p(\alpha)))$  To  $\psi_a(\alpha)$ ;
      Add "weight-value  $\leftarrow 1$ " To  $\psi_v$ (all edges of  $(\alpha, \psi_a(\alpha))$ );
    End
End Procedure

```

As stated in the previous subsection, there are two kinds of activity-performer affiliation network models; one is the binary, the other is the valued. The current knowledge discovering algorithm shows only discovering a binary activity-performer affiliation network model, because any weighted relationships or any meaningful semantics except existence relationships are not applied to the involvement and participation relations between activities and performers. If each of the relations has something to do with differentiated values or weights except existence relations, the algorithm has to assign the corresponding values greater than 1.0 to the variable, **weight-value**. Then, it implies that the algorithm is able to discover a valued activity-performer affiliation network model.

As an example, we apply the knowledge discovering algorithm to the product-order workflow procedure [6] in order to verify the algorithm working correctly; the input of the algorithm is the internal property sets of the ICN-based product-order workflow model. Fig. 3 and Table 1 are the graphical and formal representations of the model, respectively, as input of the knowledge discovering algorithm. The model is not a full description of the original ICN-based workflow model, but a partial description only showing the activities' precedence(δ), roles' assignment(ε), performers' assignment(π), and the transition conditions(κ) that are directly related with the social perspective's point of view. Fig. 4 and Table 2 depict the graphical and formal representations of the activity-performer affiliation network model, respectively, discovered by the knowledge discovering algorithm. As you see, the discovered activity-performer affiliation network model represents the involvement knowledge(σ) as well as the participation knowledge(ψ) between the activities and the performers associated with the ICN-based workflow model.

3.4 Knowledge Analytics: Affiliation Matrices

Eventually, it is necessary for activity-performer affiliation knowledge to be analyzed in a mathematical representation. The activity-performer affiliation knowledge is graphically represented by a bipartite graph, and at the same time it can be mathematically represented by an affiliation matrix. Based upon the activity-performer affiliation matrices, it is possible to analyze a variety of knowledge analytics issues [13], such as mean rates analysis [16], density measurements [14], centrality measurements [25], and so on, raised from the social networking literature.

Table 1. Formal representation of the ICN-based workflow model.

$\Gamma = (\delta, \varepsilon, \pi, \kappa, I, O)$ over A, R, P, T /* An ICN-based Workflow Model		
$A = \{\alpha_A, \alpha_B, \alpha_C, \alpha_D, \alpha_E, \alpha_F, \alpha_{start}, \alpha_{end}\}$ /* Activities		
$R = \{\eta_{start}, \eta_{R-X}, \eta_{R-Y}, \eta_{R-Z}, \eta_{end}\}$ /* Roles		
$P = \{\phi_{start}, \phi_{jack}, \phi_{joe}, \phi_{lisa}, \phi_{matthew}, \phi_{shawn}, \phi_{end}\}$ /* Performers		
$T = \{d(\text{default}), tc_1, tc_2\}$ /* Transition Conditions		
$I = \emptyset$ /* Initial Input Repositories		
$O = \emptyset$ /* Final Output Repositories		
$\delta = \delta_i \cup \delta_o$	$\delta_i(\alpha_{start}) = \emptyset;$ $\delta_i(\alpha_A) = \{\{\alpha_{start}\}\};$ $\delta_i(\alpha_B) = \{\{\alpha_A\}\};$ $\delta_i(\alpha_C) = \{\{\alpha_B\}\};$ $\delta_i(\alpha_D) = \{\{\alpha_B\}\};$ $\delta_i(\alpha_E) = \{\{\alpha_C\}\};$ $\delta_i(\alpha_F) = \{\{\alpha_C\}\};$ $\delta_i(\alpha_{end}) = \{\{\alpha_D\}, \{\alpha_E, \alpha_F\}\};$	$\delta_o(\alpha_{start}) = \{\{\alpha_A\}\};$ $\delta_o(\alpha_A) = \{\{\alpha_B\}\};$ $\delta_o(\alpha_B) = \{\{\alpha_C, \{\alpha_D\}\}\};$ $\delta_o(\alpha_C) = \{\{\alpha_E, \alpha_F\}\};$ $\delta_o(\alpha_D) = \{\{\alpha_{end}\}\};$ $\delta_o(\alpha_E) = \{\{\alpha_{end}\}\};$ $\delta_o(\alpha_F) = \{\{\alpha_{end}\}\};$ $\delta_o(\alpha_{end}) = \emptyset;$
$\varepsilon = \varepsilon_a \cup \varepsilon_r$	$\varepsilon_r(\alpha_{start}) = \{\eta_{start}\};$ $\varepsilon_r(\alpha_A) = \{\eta_{R-X}\};$ $\varepsilon_r(\alpha_B) = \{\eta_{R-Y}\};$ $\varepsilon_r(\alpha_C) = \{\eta_{R-Z}\};$ $\varepsilon_r(\alpha_D) = \{\eta_{R-X}\};$ $\varepsilon_r(\alpha_E) = \{\eta_{R-X}\};$ $\varepsilon_r(\alpha_F) = \{\eta_{R-Y}\};$ $\varepsilon_r(\alpha_{end}) = \{\eta_{end}\};$	$\varepsilon_a(\eta_{start}) = \{\alpha_{start}\};$ $\varepsilon_a(\eta_{R-X}) = \{\alpha_A, \alpha_D, \alpha_E\};$ $\varepsilon_a(\eta_{R-Y}) = \{\alpha_B, \alpha_F\};$ $\varepsilon_a(\eta_{R-Z}) = \{\alpha_C\};$ $\varepsilon_a(\eta_{end}) = \{\alpha_{end}\};$
$\pi = \pi_r \cup \pi_p$	$\pi_p(\eta_{start}) = \{\phi_{start}\};$ $\pi_p(\eta_{R-X}) = \{\phi_{jack}, \phi_{joe}, \phi_{lisa}\};$ $\pi_p(\eta_{R-Y}) = \{\phi_{shawn}\};$ $\pi_p(\eta_{R-Z}) = \{\phi_{matthew}\};$ $\pi_p(\eta_{end}) = \{\phi_{end}\};$	$\pi_r(\phi_{start}) = \{\eta_{start}\};$ $\pi_r(\phi_{jack}) = \{\eta_{R-X}\};$ $\pi_r(\phi_{joe}) = \{\eta_{R-X}\};$ $\pi_r(\phi_{lisa}) = \{\eta_{R-X}\};$ $\pi_r(\phi_{shawn}) = \{\eta_{R-Y}\};$ $\pi_r(\phi_{matthew}) = \{\eta_{R-Z}\};$ $\pi_r(\phi_{end}) = \{\eta_{end}\};$
$\kappa = \kappa_i \cup \kappa_o$	$\kappa_i(\alpha_{start}) = \emptyset;$ $\kappa_i(\alpha_A) = \{d\};$ $\kappa_i(\alpha_B) = \{d\};$ $\kappa_i(\alpha_C) = \{tc_1\};$ $\kappa_i(\alpha_D) = \{tc_2\};$ $\kappa_i(\alpha_E) = \{d\};$ $\kappa_i(\alpha_F) = \{d\};$ $\kappa_i(\alpha_{end}) = \{d\};$	$\kappa_o(\alpha_{start}) = \{d\};$ $\kappa_o(\alpha_A) = \{d\};$ $\kappa_o(\alpha_B) = \{\{tc_1\}, \{tc_2\}\};$ $\kappa_o(\alpha_C) = \{d\};$ $\kappa_o(\alpha_D) = \{d\};$ $\kappa_o(\alpha_E) = \{d\};$ $\kappa_o(\alpha_F) = \{d\};$ $\kappa_o(\alpha_{end}) = \emptyset;$

Table 2. Formal representation of the discovered affiliation network model.

$\Lambda = (\sigma, \psi, S)$ over C, A, S /* An Activity-performer Affiliation Network Model			
$C = \{\phi_{jack}, \phi_{joe}, \phi_{lisa}, \phi_{matthew}, \phi_{shawn}\}$ /* Performers			
$A = \{\alpha_A, \alpha_B, \alpha_C, \alpha_D, \alpha_E, \alpha_F\}$ /* Activities			
$S = \emptyset$ /* Performers of External Models			
$\sigma = \sigma_p \cup \sigma_v$ (Involvement Knowledge)	$\sigma_p(\phi_{jack}) = \{\{\alpha_A, \alpha_D, \alpha_E\}\};$ $\sigma_p(\phi_{joe}) = \{\{\alpha_A, \alpha_D, \alpha_E\}\};$ $\sigma_p(\phi_{lisa}) = \{\{\alpha_A, \alpha_D, \alpha_E\}\};$ $\sigma_p(\phi_{matthew}) = \{\{\alpha_C\}\};$ $\sigma_p(\phi_{shawn}) = \{\{\alpha_B, \alpha_F\}\};$	$\sigma_v(\phi_{jack}, \alpha_A) = \{\{I\}\};$ $\sigma_v(\phi_{jack}, \alpha_E) = \{\{I\}\};$ $\sigma_v(\phi_{joe}, \alpha_A) = \{\{I\}\};$ $\sigma_v(\phi_{joe}, \alpha_E) = \{\{I\}\};$ $\sigma_v(\phi_{lisa}, \alpha_D) = \{\{I\}\};$ $\sigma_v(\phi_{lisa}, \alpha_E) = \{\{I\}\};$ $\sigma_v(\phi_{matthew}, \alpha_C) = \{\{I\}\};$ $\sigma_v(\phi_{shawn}, \alpha_B) = \{\{I\}\};$ $\sigma_v(\phi_{shawn}, \alpha_F) = \{\{I\}\};$	$\sigma_v(\phi_{jack}, \alpha_D) = \{\{I\}\};$ $\sigma_v(\phi_{joe}, \alpha_D) = \{\{I\}\};$ $\sigma_v(\phi_{lisa}, \alpha_A) = \{\{I\}\};$ $\sigma_v(\phi_{lisa}, \alpha_E) = \{\{I\}\};$
$\psi = \psi_a \cup \psi_v$ (Participation Knowledge)	$\psi_a(\alpha_A) = \{\{\phi_{jack}, \phi_{joe}, \phi_{lisa}\}\};$ $\psi_a(\alpha_B) = \{\{\phi_{shawn}\}\};$ $\psi_a(\alpha_C) = \{\{\phi_{matthew}\}\};$ $\psi_a(\alpha_D) = \{\{\phi_{jack}, \phi_{joe}, \phi_{lisa}\}\};$ $\psi_a(\alpha_E) = \{\{\phi_{jack}, \phi_{joe}, \phi_{lisa}\}\};$ $\psi_a(\alpha_F) = \{\{\phi_{shawn}\}\};$	$\psi_v(\alpha_A, \phi_{jack}) = \{\{I\}\};$ $\psi_v(\alpha_A, \phi_{joe}) = \{\{I\}\};$ $\psi_v(\alpha_B, \phi_{shawn}) = \{\{I\}\};$ $\psi_v(\alpha_C, \phi_{matthew}) = \{\{I\}\};$ $\psi_v(\alpha_D, \phi_{jack}) = \{\{I\}\};$ $\psi_v(\alpha_D, \phi_{lisa}) = \{\{I\}\};$ $\psi_v(\alpha_E, \phi_{joe}) = \{\{I\}\};$ $\psi_v(\alpha_F, \phi_{shawn}) = \{\{I\}\};$	$\psi_v(\alpha_A, \phi_{lisa}) = \{\{I\}\};$ $\psi_v(\alpha_D, \phi_{joe}) = \{\{I\}\};$ $\psi_v(\alpha_E, \phi_{jack}) = \{\{I\}\};$ $\psi_v(\alpha_E, \phi_{joe}) = \{\{I\}\};$ $\psi_v(\alpha_E, \phi_{lisa}) = \{\{I\}\};$

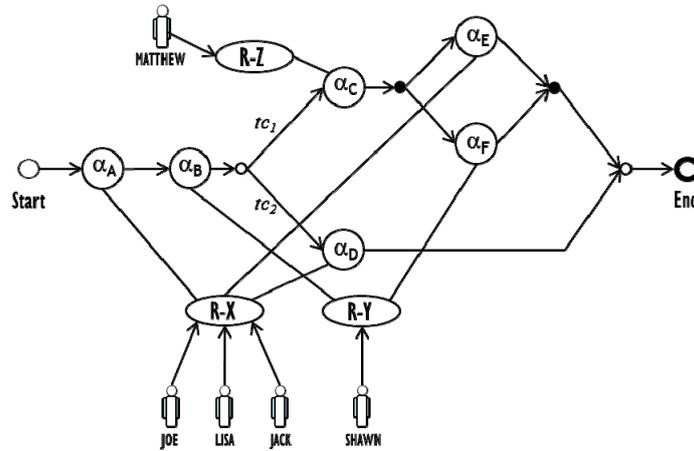


Fig. 3. Graphical representation of the ICN-based workflow model.

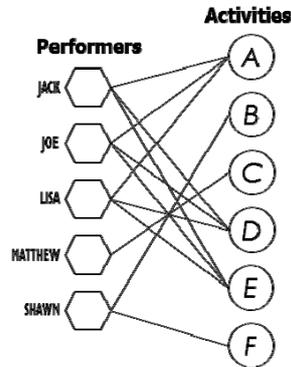


Fig. 4. Graphical representation (bipartite graph) of the discovered affiliation network model.

Defining Affiliation Matrix The affiliation matrix can be realized by either an involvement matrix or a participation matrix. That is, an activity-performer affiliation network model is mathematically transformed into an activity-performer affiliation matrix that records the presence and absence of \mathbf{g} performers at \mathbf{h} workflow activities; thus its dimensions are \mathbf{g} rows and \mathbf{h} columns, respectively. If a certain performer ϕ_i attends a workflow activity α_j , then the entry in the i th and j th cell in the matrix equals to 1; otherwise the entry is 0. Denoting a binary activity-performer affiliation matrix as \mathbf{Z} , its $x_{i,j}$ values meet these conditions:

$$x_{i,j} = \begin{cases} 1 & \text{if performer } \phi_i \text{ is affiliated with workflow activity } \alpha_j \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

- The row total, also called row marginals, (\mathbf{D}), of activity-performer affiliation matrix \mathbf{Z} sum to the number of workflow activities that each performer will attend, which im-

plies the involvement relations between activities and performers in a corresponding workflow model.

$$\bar{D}_r = \left[\sum_{j=1}^h x_{i,j} \right]_{i=1}^g \quad (2)$$

- The column marginals, (D_c), indicate the number of performers who will attend each workflow activity' enactment, which implies the participation relations between performers and activities in a corresponding workflow model.

$$\bar{D}_c = \left[\sum_{i=1}^g x_{i,j} \right]_{j=1}^h \quad (3)$$

Generating Affiliation Matrix This paper conceives an algorithm that automatically transforms activity-performer affiliation knowledge into an affiliation matrix. The following pseudo-coded algorithm is the binary affiliation matrix generation algorithm generating affiliation matrices from the formal representation of an activity-performer affiliation network model. Particularly, it distinguishes the involvement affiliation matrix (Z_p) from the participation affiliation matrix (Z_a), which easily calculate and represent the row marginals (D_r) and the column marginals (D_c), respectively.

Binary Activity-performer Affiliation Matrix Generation Algorithm:

Input An Activity-performer Affiliation Network Model, $A = (\sigma, \psi, S)$;

Output Two Binary Affiliation Matrices, $Z_p[g, h]$ and $Z_a[h, g]$

- g is the number performers in the set of C .
- h is the number activities in the set of A .

Begin Procedure

Initialize

Set Zero To all entries of $Z_p[g, h]$;

Set Zero To all entries of $Z_a[h, g]$;

For ($\forall \phi \in C$) **Do**

Begin

/ The Involvement Relations of $Z_p[g, h]$ */*

Set One To entries of $Z_p[\phi, \text{each member (activity) of } \sigma_p(\phi)]$;

End

For ($\forall \alpha \in A$) **Do**

Begin

/ The Participation Relations of $Z_a[h, g]$ */*

Set One To entries of $Z_a[\alpha, \text{each member (performer) of } \psi_a(\alpha)]$;

End

End Procedure

Analyzing Affiliation Matrix As an example, we apply the algorithm to the activity-performer affiliation knowledge discovered from the product-order workflow procedure [6], in the previous subsection; the input of the algorithm is the formal representation property sets of the activity-performer affiliation network model, and its output is two

binary activity-performer affiliation matrices, \mathbf{Z}_p and \mathbf{Z}_a , which correspond to the involvement affiliation matrix and the participation affiliation matrix, respectively. Table 3 shows the binary activity-performer affiliation matrix (\mathbf{Z}_p) generated from the discovered activity-performer affiliation knowledge in Fig. 4. Based on the affiliation matrix table, we calculate the row marginals (\mathbf{D}_r) as well as the column marginals (\mathbf{D}_c), as the following Eqs. (4) and (5):

$$\bar{D}_r = \left[\sum_{j=1}^6 x_{i,j} \right]_{i=1}^5 = [3, 3, 3, 1, 2], \quad (4)$$

$$\bar{D}_c = \left[\sum_{i=1}^5 x_{i,j} \right]_{j=1}^6 = [3, 1, 1, 3, 3, 1]. \quad (5)$$

Table 3. Binary activity-performer affiliation matrix of Fig. 4.

\mathbf{Z}_p	α_A	α_B	α_C	α_D	α_E	α_F
ϕ_{jack}	1	0	0	1	1	0
ϕ_{joe}	1	0	0	1	1	0
ϕ_{lisa}	1	0	0	1	1	0
$\phi_{matthew}$	0	0	1	0	0	0
ϕ_{shawn}	0	1	0	0	0	1

In interpreting the Eq. (4), each value of \mathbf{D}_r implies the number of activities, in which the corresponding performer is involved; so, we can infer that, for instance, the performer ϕ_{jack} is involved in three workflow activities' enactments. As a result, because the entries of \mathbf{D}_r are the numbers of workflow activities affiliated by each performer, summing them and dividing by the number of performers (\mathbf{g}) yields the mean performer affiliation (involvement) rate, $12/5 = 2.4$ activities. Also, in terms of interpretation of the Eq. (5), each value of \mathbf{D}_c reveals the number of performers, who are participating to the corresponding workflow activity; so, the workflow activity α_D is enacted by a total of three performers. Likewise, because the entries of \mathbf{D}_c are the numbers of performers participating to the corresponding workflow activities' enactments, summing them and dividing by the number of activities (\mathbf{h}) yields the mean activity affiliation (participation) rate, $12/6 = 2.0$ performers.

Supplementing Affiliation Matrix In general, an affiliation networking graph is a bipartite graph, as described in the previous section, in which non-directed lines connect performers aligned on one side of the diagram to the workflow activities aligned on the other side. So, it is possible for an affiliation networking graph to be mathematically represented in a bipartite matrix containing both sets of performers and activities in the rows and columns; assuming an affiliation networking graph has \mathbf{g} performers and \mathbf{h} activities, then its bipartite affiliation matrix has dimensions $(\mathbf{g} + \mathbf{h}) \times (\mathbf{g} + \mathbf{h})$. Consequently, using the involvement affiliation matrix (\mathbf{Z}_p) and the participation affiliation matrix (\mathbf{Z}_a) forms an affiliation bipartite matrix, $\mathbf{X}^{P,A}$, which can be schematically represented as the following Eqs. (6) and (7).

$$\mathbf{X}^{P,A} = \begin{bmatrix} 0 & \mathbf{Z}_p \\ \mathbf{Z}_a & 0 \end{bmatrix} \quad (6)$$

$$\mathbf{X}^P = \mathbf{Z}_p \cdot \mathbf{Z}_a \quad \mathbf{X}^A = \mathbf{Z}_a \cdot \mathbf{Z}_p; \quad (7)$$

As an example, we apply the algorithms to the ICN-based workflow procedure introduced in Fig. 3; its eventual workflow affiliation bipartite matrix and its involvement and participation relationships are shown in Table 4 and Eqs. (8) and (9), respectively. As you see, the diagonal of \mathbf{X}^P (co-involvement) shows that performer 1, 2, and 3 are involved in 3 activities, performer 4 and 5 are involved in 1 activity and 2 activities, respectively. In the same context, the diagonal of \mathbf{X}^A (co-participation) shows that activity 1, 4, and 5 are allotted 3 performers, and activity 2, 3, and 6 are allotted 1 performer. Also, we know that these diagonals are exactly same to the results of Eqs. (4) and (5).

Table 4. Binary workflow affiliation bipartite matrix of Fig. 3.

	ϕ_{jack}	ϕ_{joe}	ϕ_{lisa}	ϕ_{matt}	ϕ_{sha}	α_A	α_B	α_C	α_D	α_E	α_F
ϕ_{jack}	0	0	0	0	0	1	0	0	1	1	0
ϕ_{joe}	0	0	0	0	0	1	0	0	1	1	0
ϕ_{lisa}	0	0	0	0	0	1	0	0	1	1	0
ϕ_{matt}	0	0	0	0	0	0	0	1	0	0	0
ϕ_{sha}	0	0	0	0	0	0	1	0	0	0	1
α_A	1	1	1	0	0	0	0	0	0	0	0
α_B	0	0	0	0	1	0	0	0	0	0	0
α_C	0	0	0	1	0	0	0	0	0	0	0
α_D	1	1	1	0	0	0	0	0	0	0	0
α_E	1	1	1	0	0	0	0	0	0	0	0
α_F	0	0	0	0	1	0	0	0	0	0	0

$$\mathbf{X}^P = \mathbf{Z}_p \cdot \mathbf{Z}_a = \begin{pmatrix} 100110 \\ 100110 \\ 100110 \\ 001000 \\ 010001 \end{pmatrix} \times \begin{pmatrix} 11100 \\ 00001 \\ 00010 \\ 11100 \\ 11100 \\ 00001 \end{pmatrix} = \begin{pmatrix} 33300 \\ 33300 \\ 33300 \\ 00010 \\ 00002 \end{pmatrix} \quad (8)$$

$$\mathbf{X}^A = \mathbf{Z}_a \cdot \mathbf{Z}_p = \begin{pmatrix} 11100 \\ 00001 \\ 00010 \\ 11100 \\ 11100 \\ 00001 \end{pmatrix} \times \begin{pmatrix} 100110 \\ 100110 \\ 100110 \\ 001000 \\ 010001 \end{pmatrix} = \begin{pmatrix} 300330 \\ 010001 \\ 001000 \\ 300330 \\ 300330 \\ 010001 \end{pmatrix} \quad (9)$$

By applying the affiliation bipartite matrix, we are able to measure various social networking knowledge analytical properties on activity-performer affiliation networking knowledge, such as density [16] and centrality [16, 25]. Density measures on an affiliation bipartite matrix give us important basic knowledge, like the average number of activities jointly attended by the pairs of performers, the average number of performers simultaneously assigned to the pairs of activities, and so on. There are, also, four types of centrality [25] possibly measuring on an affiliation bipartite matrix: degree, closeness, betweenness, and eigenvector centrality. The details of these centrality properties won't be described anymore, because of beyond the scope of the paper.

Moreover, the activity-performer affiliation networking knowledge, as exemplified in the previous subsection, shows that five performers are involved in six activities with involvement's weight-value (= 1), and that five performers participate in six activities with participation's weight value (= 1), as well. In terms of interpreting the weight-values, the involvement's weight-values on the performer-activity edges imply designed (or planned) work-involvements, while the participation's weight-values on the activity-performer edges represent "show (or no-show)" implying designed (or planned) work-participations. At this moment, it is necessary to remind the workflow-supported affiliation knowledge rediscovery issue concerning about fulfilled work-affiliation (work-involvement/participation) relations that can be rediscovered from workflow execution logs. The weight-values of these fulfilled work-affiliation relations imply "frequencies or the number of times" counted from the corresponding workflow instances. This is why we need to differentiate the workflow-supported affiliation networking knowledge *discovery* issue from the workflow-supported social networking knowledge *rediscovery* issue.

4. RELATED WORK

Recently, technology-supported social networks and organizational behavioral analytics issues [13, 14, 23, 24] have been raised in the IT literature. Naturally, the workflow literature has just started to transit to and focusing on social and collaborative work analysis on process-oriented organizations. Particularly, our work, workflow-supported affiliation networking knowledge discovery, is directly related with a converged issue of model-log comparison issue and social networking knowledge discovery and analysis issue. With respect to this converged issue, there have been existing two main branches of research approaches: workflow-supported social networking knowledge *discovery* issue and workflow-supported social networking knowledge *rediscovery* issue.

The workflow-supported social networking knowledge rediscovery issue stems from the workflow mining issue that tries to rediscover workflow processes from workflow execution event logs; while on the other, the workflow-supported social networking knowledge discovery issue, that explores social aspects or human behaviors from workflow models, hasn't been attracting attentions in the literature, so yet. A typical research publication concerning the rediscovery issue might be [15], in which the authors suggested a methodology and system to rediscover social networks from the petri-net based workflow enactment event logs. Also, many research groups pointed out the necessity of rediscovering the actor or human behaviors from workflow enactment event logs through those publications, [4-7, 9, 11-13, 19], so far. Particularly, [19] proposed an automatic

rediscovery framework covering almost all perspectives of workflow meta-model including the actors' behaviors; however, it was not directly coping with the social networking knowledge discovery and analysis issues.

A typical one of a few research results on the workflow-supported social networking knowledge discovery issue might be [18]. In this Ph.D. research, the thesis tried to build a fundamental theory of discovering organizational work-sharing networks, which would be a special type of social networks, from a specific workflow procedure. The organizational work-sharing networks discovered from the workflow procedure consist of two kinds of networks; one is role-based organizational work-sharing network, the other is human-based organizational work-sharing network. Also, the thesis suggested a new statistical analysis approach for analyzing organizational work-sharing networks; however, the proposed statistical approach is not directly related with the social networking knowledge analysis methods. The most recently published research results on the discovery issue might be [20] and [21]. Through these two research publications, the authors proposed a conceptual framework and implemented the framework only for discovering workflow-supported social networks from ICN-based workflow models.

Fortunately, this special type of workflow-supported social networking knowledge, the activity-performer affiliation knowledge proposed in this paper, has been firstly addressed in [22]. This paper is the conceptual and contextual extension of [22]. It should so be possible to raise again the discovery and rediscovery issues on the workflow literature. Through this paper, we have showed a possible approach for the workflow-supported affiliation networking knowledge discovery issues.

5. CONCLUSION

The recent trends in working environments require new types of enterprise information systems not only which provide collaborative working facilities but also by which group of people works together simultaneously. A typical one of those enterprise information systems satisfied with the requirement is undoubtedly a large-scale workflow management system with increasingly large and complex workflow applications. The large-scale workflow management system ought to be reflecting the typical organizational perspectives like behavioral, social, informational, collaborative, and historical perspectives, which implies that it is a "human conceptual system" that must be designed, deployed, and understood within their social and organizational contexts; it also starts from the strong belief that relationships and collaborative behaviors among people who are involved in enacting the specific workflow procedures affect the overall performance and being crowned with great successes in the real businesses and the working productivity as well. In this paper, we suggested a possible way of projecting the special affiliation knowledge of the workflow-supported affiliation relations (involvement and participation behaviors) between workflow-based people and workflow-based activities by integrating the social network techniques to the workflow discovering and rediscovering techniques. As a consequence of this suggestion, we newly defined a term, activity-performer affiliation networking knowledge, and proposed an algorithm to discover an activity-performer affiliation network model from an ICN-based workflow model. Additionally, this paper developed another algorithm generating an activity-performer affilia-

tion matrix from the discovered activity-performer affiliation networking knowledge. Conclusively, we successfully verified the proposed algorithms through applying to the product-order workflow model already introduced in our research group's previous work.

However, the proposed algorithms only work for very limited functionality. In other words, it doesn't cover the activity-performer affiliation knowledge analysis and rediscovery issues. So, we would leave those insufficient functionalities to the future works of this paper. Especially, the author's research group, in the near future, would try to extend the basic ideas of the activity-performer affiliation knowledge discovery issue to the rediscovery issue.

REFERENCES

1. C. A. Ellis and G. J. Nutt, "Office information systems and computer science," *ACM Computing Surveys*, Vol. 12, 1980, pp. 27-60.
2. C. A. Ellis and G. J. Nutt, "The modeling and analysis of coordination systems," Technical Report, No. CU-CS-639-93, Department of Computer Science, University of Colorado, 1993.
3. C. A. Ellis, "Formal and informal models of office activity," in *Proceedings of the World Computer Congress*, 1983, pp. 11-22.
4. M. J. Park and K. Kim, "Control-path oriented workflow intelligence analyses," *Journal of Information Science and Engineering*, Vol. 24, 2008, pp. 343-359.
5. K. Kim and C. A. Ellis, " σ -Algorithm: Structured workflow process mining through amalgamating temporal workcases," *Lecture Notes in Artificial Intelligence*, Vol. 4426, 2007, pp. 119-130.
6. C. A. Ellis, *et al.*, "Beyond workflow mining," *Lecture Notes in Computer Science*, Vol. 4102, 2006, pp. 49-64.
7. K. Kim, "A XML-based workflow event logging mechanism for workflow mining," *Lecture Notes in Computer Science*, Vol. 3842, 2006, pp. 132-136.
8. K. Kim, "Actor-oriented workflow model," in *Proceedings of the 2nd International Symposium on Cooperative Database Systems for Advanced Applications*, 1999, pp. 163-177.
9. J. Wainer, K. Kim, and C. A. Ellis, "A workflow mining method through model rewriting," *Lecture Notes in Computer Science*, Vol. 3706, 2005, pp. 184-191.
10. K. H. Kim and H. S. Kim, "A peer-to-peer workflow model for distributing large-scale workflow data onto grid/P2P," *Journal of Digital Information Management*, Vol. 3, 2005, pp. 64-70.
11. K. H. Kim and C. A. Ellis, "Workflow reduction for reachable-path rediscovery in workflow mining," *Foundations and Novel Approaches in Data Mining*, Vol. 9, 2006, pp. 288-309.
12. M. J. Park and K. Kim, "A workflow event logging mechanism and its implications on quality of workflows," *Journal of Information Science and Engineering*, Vol. 26, 2010, pp. 1817-1830.
13. H. Oinas-Kukkonen, *et al.*, "Social networks and information systems: Ongoing and future research streams," *Journal of the Association of Information Management*, Vol. 11, 2010, pp. 61-68.

14. M. Griffiths, "Oxygen: Social intranets, collective intelligence and government practices," *Electronics Journal of e-Government*, Vol. 5, 2007, pp. 177-190.
15. W. M. P. van der Aalst, H. A. Reijers, and M. Song, "Discovering social networks from event logs," *Computer Supported Cooperative Work*, Vol. 14, 2005, pp. 549-593.
16. D. Knoke and S. Yang, "Social network analysis," *Series: Quantitative Applications in the Social Sciences*, 2nd ed., SAGE Publications, 2008.
17. K. Kim and C. A. Ellis, "Section II/Chapter VII – An ICN-based workflow model and its advances," *Handbook of Research on BP Modeling*, IGI Global, ISR, 2009, pp. 142-172,
18. J. Won, "A framework: Organizational network discovery on workflows," Ph.D. Dissertation, Department of Computer Science, Kyonggi University, 2008.
19. A. J. Rembert, "Automatic discovery of workflow models," Ph.D. Thesis Dissertation, Department of Computer Science, University of Colorado at Boulder, 2008.
20. J. Song, M. Kim, H. Kim, and K. Kim, "A framework: Workflow-based social network discovery and analysis," in *Proceedings of the 13th IEEE International Conference on Computational Science and Engineering*, 2010, pp. 421-426.
21. K. Kim, "A workflow-based social network discovery and analysis system," in *Proceedings of the International Symposium on Data-Driven Process Discovery and Analysis*, 2011, pp. 163-176.
22. H. Kim, H. Ahn, and K. Kim, "A workflow affiliation network discovery algorithm," *ICIC Express Letters*, Vol. 6, 2012, pp. 765-770.
23. E. Ferneley and R. Helms, "Editorial of the special issue on social networking," *Journal of Information Technology*, Vol. 25, 2010, pp. 107-108.
24. M. Skerlavaj, V. Dimovski, and K. C. Desouza, "Patterns and structures of intra-organizational learning networks within a knowledge-intensive organization," *Journal of Information Technology*, Vol. 25, 2010, pp. 189-204.
25. K. Faust, "Centrality in affiliation networks," *Journal of Social Networks*, Vol. 19, 1997, pp. 157-191.



Kwanghoon Pio Kim is a full Professor of Computer Science Department and the founder and supervisor of the collaboration technology research laboratory at Kyonggi University, South Korea. Also, he is in charge of the director of the computerization and information institute in Kyonggi University, and was in charge of the director of the contents convergence software research center established at 2007 as a new GRRC project funded by the Gyeonggi Provincial Government, Republic of Korea. He received B.S. degree in computer science from Kyonggi University in 1984. And he received M.S. degree in computer science from Chungang University in 1986. He also received his M.S. and Ph.D. degree from the Computer Science Department of University of Colorado at Boulder, in 1994 and 1998, respectively. He had worked as researcher and developer at Aztek Engineering, American Educational Products Inc., and IBM in USA, as well as at Electronics and Telecommunications Research Institute (ETRI) in South Korea. In present, he is a vice-chair of the BPM Korea Forum. He has been in charge of a country-chair (Korea) and ERC vice-chair of the Workflow Management Coalition. He has also

been on the editorial board of the journal of KSII, and the committee member of the several conferences and workshops. His research interests include groupware, workflow systems, BPM, CSCW, collaboration theory, Grid/P2P distributed systems, process warehousing and mining, workflow-supported social networks and analysis, and process-aware information systems.