Journal of Network and Computer Applications 75 (2016) 355–373

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Workflow-supported social networks: Discovery, analyses, and system

Minjae Park [a], Hyun Ahn [b], Kwanghoon Pio Kim [b,*]

[a] Engineering Innovation Department, BISTel R&D Center, BISTel, INC., BISTel Tower, 128 Baumoe-ro, Seocho-gu, Seoul 06754, South Korea
[b] Collaboration Technology Research Laboratory, Department of Computer Science, Kyonggi University, 154-42 Kwangkyosan-Ro, Youngtong-Gu, Suwon-Si, Kyonggi-do 16227, South Korea

ARTICLE INFO

Article history:
Received 8 March 2015
Received in revised form
23 December 2015
Accepted 15 August 2016
Available online 17 August 2016

Keywords:
Workflow
Business process
Workflow-supported social network
Business analytics and business intelligence
Organizational knowledge
Behavioral structures

ABSTRACT

Technology-supported social networks have been penetrating many aspects of our lives from friendships/blogging sites to working places. The recent BAI (business analytics and business intelligence) systems have also supported the activities of analyzing the social networking issues on employees of companies and their work-sharing relationships, and have used the analyzed issues as a sort of organizational knowledge to deliver quantum improvements in decision-makings and organizational performances. Accordingly this paper focuses on a special type of social networking knowledge called "*workflow-supported social network*" that is formed through the deployments and operations of workflow and BPM (business process management) technologies on a workflow-supported organization. In this paper, we formalize a theoretical framework coping with discovery phase and analysis phase, and conceive a series of formalisms and algorithms for representing, discovering, and analyzing the workflow-supported social network. As a theoretical basis, it uses the conceptual methodology of information control nets that used to formally describe workflow procedures and business processes. The theoretical framework is expansively implemented in the name of a systematic framework that is able to automatically discover a workflow-supported social network from an XPDL-based workflow package, construct SocioMatrices from the discovered workflow-supported social network, analyze the SocioMatrices, and visualize the workload centrality measures of all the actors in the corresponding workflow-supported social network. In order to verify the correctnesses of all the discovery algorithms, analysis equations, and the implemented system, we carried out two operational examples. One is for the theoretical framework, the other is for the systematic framework. We applied the theoretical framework to an information control net model of the typical enterprise Hiring workflow procedure, and presented its analyzed results and visualizations. At the same time, by the implemented system we analyzed an XPDL-based workflow package fulfilled by 17 participants, which comprises two imaginary workflow models: Hiring workflow procedure and Presentation workflow procedure consisting of 17 activities and 7 activities, respectively. The analyzed outputs are presented via a series of the captured screen-snapshots produced from the system. Finally, the paper summarizes the implications of the workflow-supported social networking knowledge and how much it is worth in improving decision-makings and organizational performances of workflow-supported organizations and enterprises.

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 workflow[1] 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 (Kim and Ellis, 2009) have been proposed in the workflow literature, and almost all commonly employ the five essential entity-types (Ellis and Nutt, 1980), such as activity, role, actor, repository, and invoked 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, informational, collaborative, and historical perspectives, onto

* Corresponding author.
E-mail addresses: mjpark@bistel-inc.com (M. Park), hahn@kgu.ac.kr (H. Ahn), kwang@kgu.ac.kr (K.P. Kim).
URLS: http://www.bistel-inc.com (M. Park), http://ctrl.kyonggi.ac.kr (H. Ahn), http://ctrl.kyonggi.ac.kr (K.P. Kim).

[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.

http://dx.doi.org/10.1016/j.jnca.2016.08.014
1084-8045/

workflow models. Therefore, we can conclude that the workflow management systems are “people system (Ellis et al., 2012)” that must be designed, deployed, and understood within their social and organizational contexts.

More recently, individuals as their role of employees and their companies have also started adopting the concept of social networks both intra-organizationally and inter-organizationally, and the collaborative behavior (Griffiths, 2007) of individuals in organizational social networks is also becoming an opportunity for companies to learn about how much their employees are contributing to their products, services, and business operations. So, technology-supported social network (Harri et al., 2010) has received considerable attentions in the fields of information systems and knowledge discovery. Consequently, the recent workflow literature has moved the focus to “*People and their work-collaborating relationships*.” It begins from the strong belief that social relationships and collaborative behaviors among employees who are involved in enacting workflow models affect not only their overall decision-making and organizational performances but also being crowned with great successes in the real businesses and the working productivity.

Accordingly, the workflow literature starts being interested in a new type of social networks (Ferneley and Helms, 2010), dubbed *workflow-supported social networks* (Song et al., 2010). There have been existing two main branches of research approaches in resolving the workflow-supported social network issues. One is the so-called workflow-supported social network *discovery* issues, the other has something to do with the so-called workflow-supported social network *rediscovery* issues (Kim, 2006). The latter is concerned with mining a workflow-supported social network from workflow event logs, which was firstly issued by Aalst et al. (2005); the former is to discover a workflow-supported social network through exploring the human perspective of a specific workflow model, which was firstly issued by Song et al. (2010). More specifically we would differentiate the former from the latter; the former is to explore a *planned*[2] *workflow-supported social network* (social planning aspect) embedded in the underlying workflow model itself, whereas the latter is to explore an *enacted workflow-supported social network* (social enactment aspect) from the execution logs of the model. The paper is directly related with the discovery issues exploring a planned workflow-supported social network embedded in a workflow model. Conclusively, in this paper we propose a theoretical framework as well as a systematic framework for discovering a planned workflow-supported social network. The theoretical framework is a theoretical approach to discover a planned social network from an ICN-based workflow model, and the systematic framework is for implementing a discovery system that is able to systematically discover a planned workflow-supported social network from an XPDL-based workflow model and package. Through these proposed frameworks, we are able to discover, analyze, and visualize a workflow-supported social network and its analysis results, theoretically and systematically.

In terms of making up the paper, the next section gives the technological background and motivation, mainly focusing on the ICN-based workflow model and the human-centered workflow perspectives and knowledge. In the next two consecutive sections, we describe the details of a theoretical framework supporting workflow-supported social network discovery and analysis, and a systematic framework with a series of operational screen-snapshots implementing the discovery and analysis functionalities for workflow-supported social networks. Finally, we give a summary with a brief description of its related works and conclusions including future works.

[2] The planned and the enacted imply the outcomes from the build-time activities and the run-time activities, respectively, under supervision of a corresponding workflow system.

2. Background and motivation

This section introduces the basic concept of ICN-based workflow model (Ellis and Nutt, 1980; Kim and Ellis, 2009) as a technological background, and it illustrates the social perspective of workflow model as a conceptual motivation. In describing the ICN-based workflow model, we start from defining a workflow meta-model (Kim and Ellis, 2009) that is a theoretical basis of the ICN-based workflow model, and next we introduce the graphical notations of the model and their formal representations.

2.1. Workflow meta-model

In describing an ICN-based workflow model, we use the following basic workflow terminology: workflow procedure, activity, job, workcase, role, actor/group, and invoked application including web services. These terms are the primitive entity types to be composed of ICN-based workflow models, and also they have appropriate relationships with each other as shown in Fig. 1. The following 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 with 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 a subworkflow. The elementary activity is an activity that can be realized by a computer program, such as an application program, transaction, script, or web service. The gateway activity implies an activity that is used for controlling execution sequences of elementary/compound activities. The types of gateway activities consist of conjunctive gateways (after activity A, do activities B and C), disjunctive gateways (after activity A, do activity B or C), and loop gateways. Both the disjunctive gateway and the loop gateway need to be set with some specific *transition conditions* in order to select one of the possible transition paths 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 actor, such as organizational staff (role, participant) and a system, which 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. The role assignment (Koschmider et al., 2012) plays a vital task in terms of planning and enacting workflow procedures.
- 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

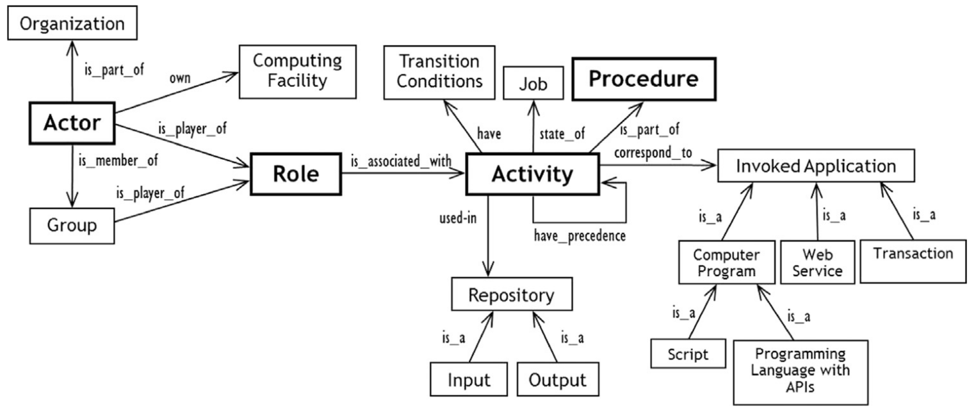

Fig. 1. The workflow meta-model.

with activities and workflow procedures. Note that this actor (or performer) entity type is a vital property in forming a workflow-supported social network that is the essential topic of this paper.

- 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* is a concrete unit of the associated activity, which can be reified by computer programs including *script*, web services, or database transactions. The invoked application associated with an activity can be invoked in a mode of automatic or manual. The automatic mode implies that either a part or all of the invoked application is automatically launched by the workflow enactment component.
- Finally, a *repository* is a set of input and output relevant data of the associated activities. The repository provides a communication channel between the workflow enactment component and the invoked application component, which means that the input and the output repositories play the roles of the input parameters and the output parameters of the associated invoked applications, respectively.

2.2. Background: information control net

An ICN-based workflow model can be defined by capturing the notations of workflow procedures, activities and their control precedences, 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 the ICN-based workflow model. Definition 1 is a formal definition of an 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, actor complexity information, and so forth.

Definition 1. *Information Control Net* (*ICN*) for formally defining workflow model. A basic *ICN* is 8-tuple $\Gamma = (\delta, \rho, \lambda, \varepsilon, \pi, \kappa, \mathbf{I}, \mathbf{O})$ over a set of **A** activities (including a set of group activities), a set **T** of transition conditions, a set **R** of repositories, a set **G** of invoked application programs, a set **P** of roles, and a set **C** of actors (including a set of actor groups), where

- **I** is a finite set of initial input repositories, assumed to be loaded with information by some external process before execution of the ICN.
- **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, δ_o: $\mathbf{A} \longrightarrow \wp(\mathbf{A})$ is a multi-valued function mapping an activity to its sets of (immediate) successors, and δ_i: $\mathbf{A} \longrightarrow \wp(\mathbf{A})$ is a multi-valued function mapping an activity to its sets of (immediate) predecessors.
- $\rho = \rho_i \cup \rho_o$ where ρ_o: $\mathbf{R} \longrightarrow \wp(\mathbf{A})$ is a multi-valued function mapping an activity to its set of output repositories, and ρ_i: $\mathbf{R} \longrightarrow \wp(\mathbf{A})$ is a multi-valued function mapping an activity to its set of input repositories.
- $\lambda = \lambda_a \cup \lambda_g$ where λ_g: $\mathbf{G} \longrightarrow \wp(\mathbf{A})$ is a single-valued function mapping an activity to its invoked application program, and λ_a: $\mathbf{A} \longrightarrow \wp(\mathbf{G})$ is a multi-valued function mapping an invoked application program to its set of associated activities.
- $\varepsilon = \varepsilon_a \cup \varepsilon_p$ where ε_p: $\mathbf{P} \longrightarrow \wp(\mathbf{A})$ is a single-valued function mapping an activity to a role, and ε_a: $\mathbf{A} \longrightarrow \wp(\mathbf{P})$ is a multi-valued function mapping a role to its sets of associated activities.
- $\pi = \pi_p \cup \pi_c$ where π_c: $\mathbf{C} \longrightarrow \wp(\mathbf{P})$ is a multi-valued function mapping a role to its sets of associated actors, and π_p: $\mathbf{P} \longrightarrow \wp(\mathbf{C})$ is a multi-valued function mapping an actor to its sets of associated roles.
- $\kappa = \kappa_i \cup \kappa_o$ where κ_i: $\mathbf{T} \longrightarrow \wp(\mathbf{A})$ is a multi-valued function returning a set of control-transition conditions, **T**, on directed arcs, $(\delta_i(\alpha), \alpha \in \mathbf{A})$ between $\delta_i(\alpha)$ and α; and κ_o: $\mathbf{T} \longrightarrow \wp(\mathbf{A})$ is a multi-valued function returning a set of control-transition conditions, **T**, on directed arcs, $(\mathbf{A}, \delta_o(\alpha))$ between α and $\delta_o(\alpha)$.

Starting and terminating nodes: Additionally, the execution of a workflow model commences by a single χ transition-condition. So, we always assume without loss of generality that there is a single starting node (α_I). At the commencement, it is assumed that all input repositories in the set **I** have been initialized with data by the external system:

$$\exists\ \alpha_I \in \mathbf{A} | \delta_i(\alpha_I) = \{\varnothing\} \wedge \kappa_o(\alpha_I) = \{\{\chi\}\}.$$

The execution is terminated with any one λ output transition-condition. Also we assume without loss of generality that there is a single terminating node (α_F). The set of output repositories O is data holders that may be used after termination by the external system:

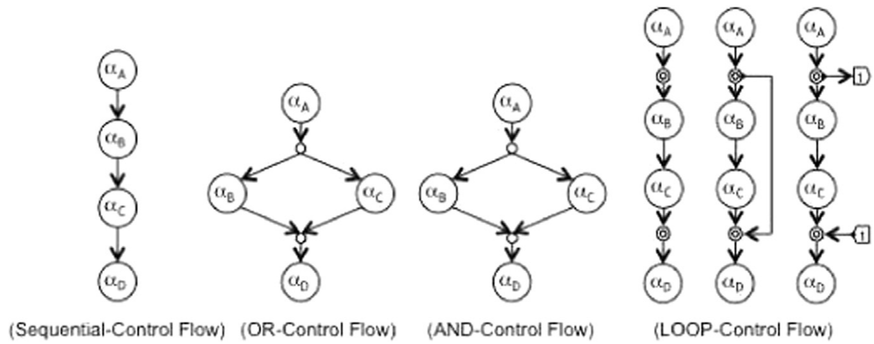

Fig. 2. The information control net primitives.

$\exists\ \alpha_F \in \boldsymbol{A} | \delta_o(\alpha_F) = \{\emptyset\} \wedge \kappa_i(\alpha_F) = \{\{\chi\}\}$.

Control flow: *Temporal ordering of activities*. Given a formal definition, the temporal ordering of activities in a workflow model can be interpreted as follows: For any activity α, in general,

$\delta(\alpha) = \{,,\ldots,$

which means that upon the completion of activity α a transition that simultaneously initiates all of the activities β_{i1} through $\beta_{im(i)}$ occurs, which is called a parallel transition; otherwise only one value out of $i(1 \le i \le n)$ is selected as the result of a decision made within activity α, which is called a decision transition. Note that if $n = 1 \wedge m = 1$, then neither a decision nor a parallel transition is needed after completion of activity α, which means that the transition is a sequential transition. If $m(i) = 1$ for all i, then no parallel processing is initiated by completion of α.

We graphically define these primitive transition types as shown in Fig. 2. An activity having a conjunctive (or parallel) transition is represented by a solid dot (●), and an activity having a disjunctive (or decision) transition is represented by an unfilled dot (○). As defined in the previous section, these special types of activities are called gateway activities, and in order to be syntactically safe, it is very important for these gateway activities to keep the structured properties (proper nesting and matched pair properties). Therefore, each of the gateway activities always keeps a matched pair with split and join types of gateway activity in a workflow procedure, and multiple sets of the gateway activities keep in a properly nested pattern. The following formally describe for the basic transition types modeled by the exclusive-OR and AND gateway activities depicted in Fig. 2.

- Sequential Transition between activities

  incoming $\rightarrow \delta_i(\alpha_B) = \{\{\alpha_A\}\}$; *outgoing* $\rightarrow \delta_o(\alpha_B) = \{\{\alpha_C\}\}$;

- Exclusive OR Transition through *xor-gateway*

  xor-split $\rightarrow \delta_o(\alpha_A) = \{\{\alpha_B\}, \{\alpha_C\}\}$; *xor-join* $\rightarrow \delta_i(\alpha_D) = \{\{\alpha_B\}, \{\alpha_C\}\}$;

- AND Transition through *and*-gateway

  and-split $\rightarrow \delta_o(\alpha_A) = \{\{\alpha_B, \alpha_C\}\}$; *and-join* $\rightarrow \delta_i(\alpha_D) = \{\{\alpha_B, \alpha_C\}\}$;

Loop transition: *Block activity*. We have to take care of an iterative (loop) transition that is the most common and essential construct in modeling the temporal ordering of activities. We do need to graphically define the iterative (loop) transition type as a pair of unfilled dots of gateway activities shown in Fig. 2. At a glance, it can be interpreted as a special type of disjunctive transition type; however, if we replace this transition type with a disjunctive transition type, it is very hard to maintain the structured properties (matched pair and proper nesting) of the workflow model. Therefore, we introduce a concept of block[3] activity in order to keep the structured properties in modeling a workflow procedure and for the sake of the simplification of modeling work, as well. The block activity contains two gateway activities: loop-begin and loop-end. Modeling the temporal ordering of the activities inside of the two gateway activities is done in exactly the same way as the ICN-based workflow modeling approach. We have to specify the loop's exit conditions in the modeling time. Accordingly, the formal definition of a block activity's gateway activities shown in Fig. 2 is as follows:

- *loop-begin* Gateway

  $\rightarrow \delta_i(\alpha_{loop\text{-}begin}) = \{\{\alpha_A\}\}$; $\delta_o(\alpha_{loop\text{-}begin}) = \{\{\alpha_B\}\}$;

- *loop-end* Gateway

  $\rightarrow \delta_i(\alpha_{loop\text{-}end}) = \{\{\alpha_C\}\}$; $\delta_o(\alpha_{loop\text{-}end}) = \{\{\alpha_D\}\}$;

Assigning roles and actors (*performers*): For any activity α, $\varepsilon_p(\alpha) = \{\eta_1, \eta_2, \ldots, \eta_n\}$, where n is the number of roles, $\forall\ \eta \in P$, involved in the activity, means that an activity α is performed by one of the roles. Also, $\varepsilon_a(\eta) = \{\alpha_1, \alpha_2, \ldots, \alpha_m\}$, where m is the number of activities performed by the role, means that a role η is associated with several activities in a workflow procedure. Typically, one or more participants are associated with each activity via roles. As defined in the previous section, a role is a named designator for one or more participants which conveniently acts as the basis for partitioning of work skills, access controls, execution controls, and authority/responsibility, whereas 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 procedures.

2.3. Motivation: social perspectives on the ICN-based workflow model

Almost all the workflow models proposed in other studies commonly employ the five essential entity-types—activity, role, actor, repository and application entity-types—to represent organizational works and their procedural collaborations. These entity-types reflect the typical people-oriented organizational perspectives (Tableau, 2015) like behavioral, social, informational, collaborative, and historical perspectives, in the workflow models and their enactment systems. The workflow management systems accordingly ought to be "people systems (Ellis et al., 2012; Tableau, 2015)" that must be designed, deployed, and understood within their social and organizational contexts.

Role and actor (*Performer*) out of the five essential entity-types

[3] The terminology and concept of block was firstly used in the workflow modeling system of FlowMark Workflow Management System of IBM.

are directly related to the social perspectives on workflow procedures. A role is a named designator for one or more participants which conveniently acts as the basis for partitioning of work skills, access controls, execution controls, and authority/responsibility. An actor is a person that can fulfill roles to execute, be responsible for, or be associated with activities. There exist two types of social perspectives on a workflow procedure: role-based social perspectives and actor-based social perspectives. In the role-based social perspective, for any activity α, $\{\alpha|\varepsilon_p(\alpha) = \{\eta_i\} \wedge \eta_i \in P\}$, where P is the set of roles $(\{\eta_1, \eta_2, \ldots, \eta_n\})$ associated to a workflow model means that the activity α is associated with one of the roles. Also, $\{\eta|\varepsilon_a(\eta) = \{\alpha_1, \alpha_2, \ldots, \alpha_m\} \wedge \eta \in P\}$, where m implies the number of activities associated with a role, means that the role η is associated with one or more activities in a workflow model. In terms of an actor-based social perspective (Kim and Kim, 2005), one or more actors are involved in a role, which means that more than one actor can participate in enacting an activity. The framework developed in this paper can provide a feasible means to quantify and visualize these social perspectives. Particularly, this paper tries to investigate the actor-based social perspective.

As stated above, we are especially interested in the actor-based social perspective that can be discovered from workflow procedures. Moreover, it can be theoretically discovered from the formal definition of the ICN-based workflow model, and it can be systematically discovered from the XPDL[4] workflow models and packages. In other words, the actor-based social perspective becomes a theoretical basis for discovering and analyzing workflow-supported social networks. Fig. 3 shows another graphical view of a workflow model illustrating the actor involvements in a workflow model to emphasize the actor-based social aspect. The figure raises the following questions:

- Who is the most important or prominent actor in a specific workflow procedure?
  1. *Qualitative issue*: Who is the most influential actor in enacting the workflow model?
  2. *Quantitative issue*: How much does the actor contribute in enacting the workflow model?
  3. *Visualizing issue*: How to visualize the actor's contribution in enacting the workflow model?

The qualitative issue and the quantitative issue are reflecting one of the top 10 trends of business intelligence in 2016 (Tableau, 2015), "GOVERNANCE & SELF-SERVICE ANALYTICS BECOME BEST FRIENDS," which implies that organizations have learned that data governance, when done right, can help nurture a culture of analytics and meet the needs of the business, and people are more likely to dig into their data when they have centralized, clean, and fast data sources, and when they know that someone (IT) is looking out for the individual's performance. Also, the visual issue comes from another of the top 10 trends (Tableau, 2015), "VISUAL ANALYTICS BECOMES A COMMON LANGUAGE," which means not only that people are visualizing their data to explore questions, uncover insights, and share stories with both data experts and non-experts alike, but also that visual analytics will serve as the common language, empowering people to reach insights quickly, collaborate meaningfully, and build a community around data.

Conclusively, the answers for the questions should be able to convey a very valuable analytics and uncover meaningful insights (Tableau, 2015) in the workflow-supported organization, which is the primary rationale of the concept of workflow-supported social networks. The framework (Song et al., 2010) theoretically covers discovering a workflow-supported social network that is exerting a shape of the actor-based social perspective to analyzing the discovered workflow-supported social network by mathematically revising some of the well-known equations and algorithms (Faust, 1997; Knoke and Yang, 2008) in the social network analysis literature. Ultimately, the theoretical framework is implemented in this paper, and the implemented system is able to quantitatively answer for the question through the prominence concept and measurement (workload-centrality) at both individual and group levels of analysis.

[4] XPDL stands for XML Process Definition Language, which is a standardized workflow specification language released by WfMC (Workflow Management Coalition).

3. A theoretical framework from discovery to analysis

As pointed out in the previous section, we are especially interested in the actor-based social perspective (Kim, 1999) that can be discovered from the formal definition of the ICN-based workflow model. In other words, we contrive a theoretical framework not only for discovering a workflow-supported social network (Jeon et al., 2012) that is exerting a shape of workload-centralities (Jeon and Kim, 2013) among workflow actors, but for analyzing the discovered workflow-supported social network by borrowing well-known algorithms (Faust, 1997; Knoke and Yang, 2008).

Fig. 4 illustrates an overall view of the theoretical framework and its conceptual phases with a series of associated algorithms. In this section, we formalize the details of the discovery and analysis phases of the framework. In the discovery phase, we describe how to automatically discover a workflow-supported social network from a workflow procedure defined in the ICN-based workflow model. In the analysis phase, we firstly devise two classes of SocioMatrix generation algorithms which automatically generate a *binary directed/non-directed SocioMatrix* (Jeon et al., 2012) and a *valued directed/non-directed SocioMatrix* from the discovered network. Based upon these SocioMatrices, we are able to calculate the prominence measures (centrality and prestige measures) by implementing the corresponding algorithms (Knoke and Yang, 2008).

Note that the most widely used centrality measures are degree, closeness, and betweenness. These measures vary in their applicability to non-directed and directed relations, and differ at the individual actor and the group or complete network levels. This paper uses the degree of centrality algorithm (Knoke and Yang, 2008). Additionally, the workload-centrality analysis equations originating from the degree-centrality analysis equations (Knoke and Yang, 2008) need to be extended to handle a workflow package representing a group of workflow models.

3.1. Discovering workflow-supported social networks

In this subsection, we start with introducing the basic concept and definition of workflow-supported social network model that can be discovered from a workflow procedure represented by the ICN-based workflow model. Next, we expatiate on the discovery algorithm with an example of the Hiring workflow procedure that was introduced by Kim and Ellis (2009) as a typical ICN-based workflow model, and finalize this subsection by explaining the contextual differences between the discovered workflow-supported social network to the traditional social network and its implications on the workflow-supported organizational knowledge and intelligence.

3.1.1. Formal definition of the workflow-supported social network

Basically, the origin of the workflow-supported social network

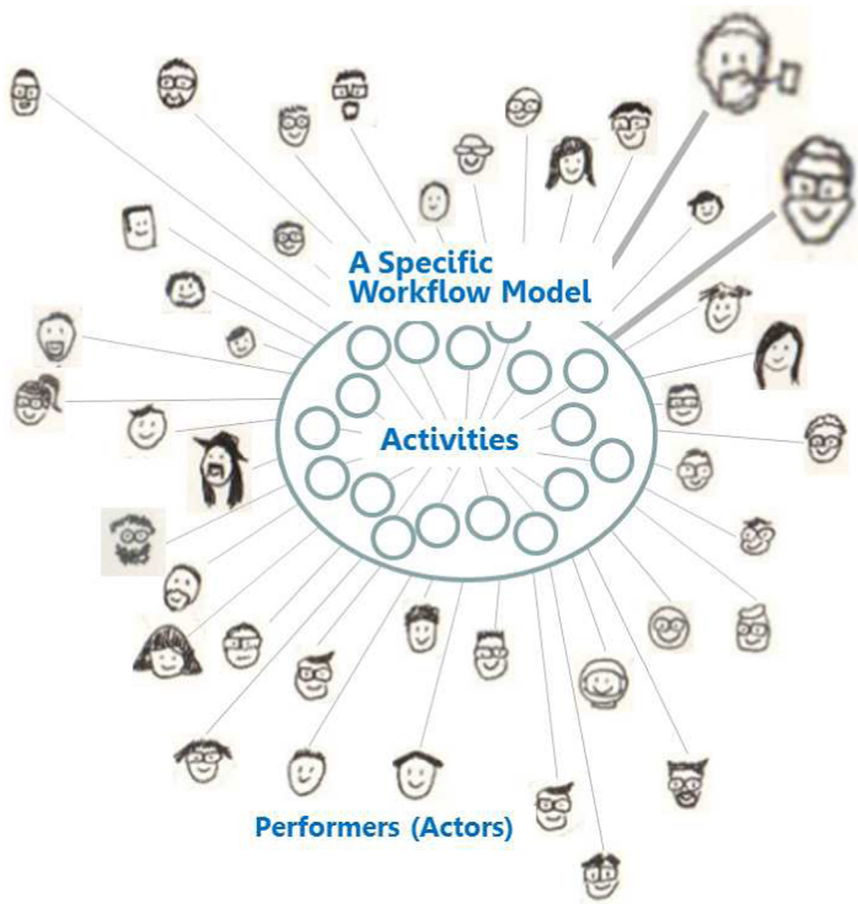

Fig. 3. Actors' involvement on a specific workflow model.

model is the actor-based workflow model (Kim, 1999), and its rationale is based on where it represents the behaviors of acquisition activities among actors in a workflow procedure, which we call workflow-supported social relationships that form this special type of social network. As given in Definition 2 of the workflow-supported social network model (Jeon et al., 2012), the behaviors of the model are revealed through incoming and outgoing directed arcs labeled with activities associated with each of the actors. The directed arcs imply two kinds of behaviors—workflow-supported social relationships and activity acquisition of actors—through which we are able to obtain precedence (candidate-predecessor relationship/candidate-successor relationship) relationships among actors as well as activity acquisition of each actor in a workflow procedure. In terms of defining actors' predecessors and successors, we use the prepositional word, "*candidate*," because a role–actor mapping is a one-to-many relationship, and the actor

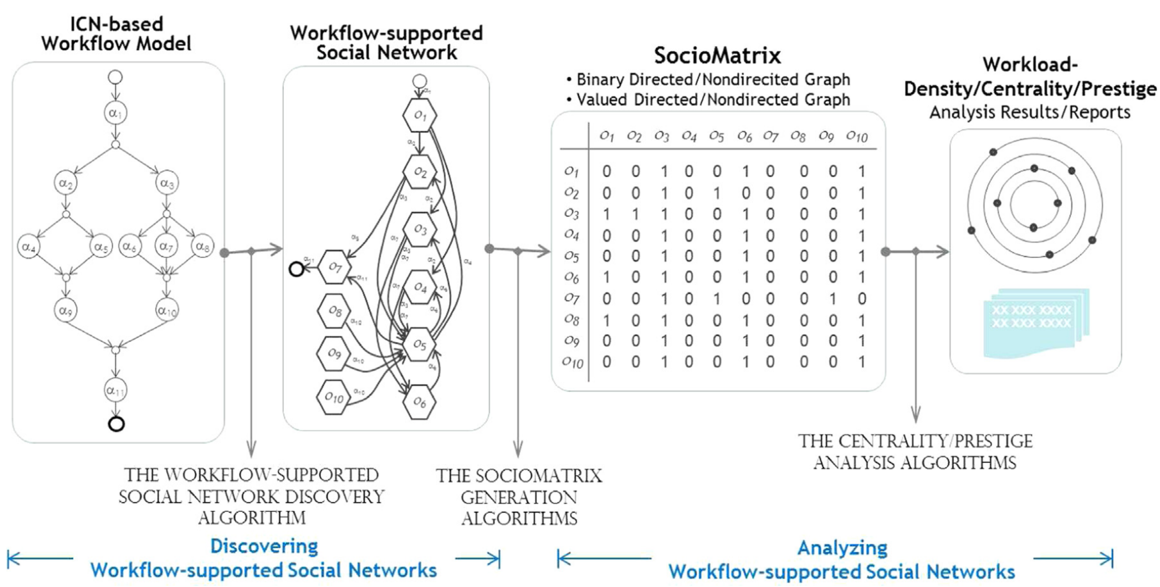

Fig. 4. Discovery and analysis phases of the framework.

selection mechanism will choose one actor out of the assigned actors mapped to the corresponding role during the underlying workflow model's runtime.

The activities on the incoming directed arcs are the previously performed activities by the predecessors of the actor, and the activities on the outgoing directed arcs are the activities acquired by the actor, itself. The activity on the transitive directed arc implies not only the acquisition activities of the actor, but also the previously performed activities by the actor itself. In principle, the workflow-supported social network graph is a directed graph characterized by multiple incoming arcs, multiple outgoing arcs, and cyclic, self-transitive, and multiple activity associations on arcs. However, it can also be transformed to a non-directed graph for the analysis phase of the framework.

Definition 2. *Workflow-supported social network model*. A workflow-supported social network model is formally defined as $\Lambda = (\sigma, \psi, \mathbf{S}, \mathbf{E})$, over a set **C** of actors, and a set **A** of activities, where

- **S** is a finite set of coordinators or coordinator-groups connected from some external workflow-supported social network models.
- **E** is a finite set of coordinators or coordinator-groups connected to some external workflow-supported social network models.
- $\sigma = \sigma_i \cup \sigma_o$/* Social Relationships: successors and predecessors */.
  where σ_o: $\mathbf{C} \longrightarrow \wp(\mathbf{C})$ is a multi-valued function mapping an actor to its sets of (immediate) candidate-successors, and σ_i: $\mathbf{C} \longrightarrow \wp(o \in \mathbf{C})$ is a multi-valued function mapping an actor to its sets of (immediate) candidate-predecessors.
- $\psi = \psi_i \cup \psi_o$/* Acquisition of Activities */.
  where ψ_i: $\mathbf{C} \longrightarrow \wp((\mathbf{A}, \mathbf{C}))$ is a multi-valued function mapping an actor to a set of pairs of predecessors and activities on the directed arcs, $(\sigma_i(o), o)$, $o \in \mathbf{C}$, from $\sigma_i(o)$ to o; and ψ_o: $\mathbf{C} \longrightarrow \wp((\mathbf{A}, \mathbf{C}))$ is a multi-valued function mapping an actor to a set of pairs of successors and activities on the directed arcs, $(o, \sigma_o(o))$, $o \in \mathbf{C}$, from o to $\sigma_o(o)$.

3.1.2. Discovery algorithm for the workflow-supported social network

At this moment, it is important to emphasize that the workflow-supported social network model (Jeon et al., 2012) will not be modeled or designed but automatically discovered from a workflow procedure. So, we devise an automatic discovery methodology for the workflow-supported social network model, which algorithmically explores the internal social properties of an ICN-based workflow procedure. Particularly, in order to discover a workflow-supported social network, it needs the sets of internal properties of the ICN-based workflow procedure as inputs: δ_o (control flow information), ε_p (activity–role mapping information) and π_c (role–actor mapping information). Likewise, we have to remind that we will not differentiate 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 realtime groupware activity type. However, as future work, we need to cope with these social relationships caused from the group-actor binding activities in discovering workflow-supported social networks. The following is an algorithm for automatically discovering a workflow-supported social network from an ICN-based workflow procedure:

Workflow-Supported Social Network Discovery Algorithm:
Input An ICN, $\Gamma = (\delta, \rho, \lambda, \varepsilon, \pi, \kappa, \mathbf{I}, \mathbf{O})$;
Output A Workflow-supported Social Network, $\Lambda = (\sigma, \psi, \mathbf{S}, \mathbf{E})$;
Begin Procedure
For $(\forall\ \alpha \in A)$ **Do**
Begin
$/{*}\sigma = \sigma_i \cup \sigma_o{*}/$
Add *all members of* $\pi_c(\varepsilon_p(\alpha))$
To $\sigma_i\Big($*each member of* $\pi_c\Big(\varepsilon_p\Big(\delta_o\Big(\alpha\Big)\Big)\Big)\Big)$;
Add*all members of* $\pi_c(\varepsilon_p(\delta_o(\alpha)))$
To $\sigma_o\Big($*each member of* $\pi_c\Big(\varepsilon_p\Big(\alpha\Big)\Big)\Big)$;
$/{*}\psi = \psi_i \cup \psi_o{*}/$
Add*all pairs of* (α, o), $\forall\ o \in \pi_c(\varepsilon(\alpha))$
To $\psi_i\Big($*each member of* $\pi_c\Big(\varepsilon_p\Big(\delta_o\Big(\alpha\Big)\Big)\Big)\Big)$;
Add*all pairs of* (α, o), $\forall\ o \in \pi_c(\varepsilon_p(\delta_o(\alpha)))$
To $\psi_o\Big($*each member of* $\pi_c\Big(\varepsilon_p\Big(\alpha\Big)\Big)\Big)$;
End
End Procedure

As an example, we apply the algorithm to the Hiring workflow procedure (Kim and Ellis, 2009), the left-hand side of Fig. 5. The input of the algorithm is the internal property sets of the ICN-based Hiring workflow model, and its output is a workflow-supported social network graphically represented in the right-hand side of Fig. 5. Table 1 gives the formal representation of the workflow-supported social network model discovered from the input model. Unlike the activity–role mapping relationship in which an activity is mapped to just a single role (one-to-one relationship), the workflow-supported social network based on the role–actor mapping relationships has one-to-many relationships in the mappings of activities and actors. Because of the one-to-many relationships between activities and actors, an actor node may have several outgoing directed arcs that have the same activity as their labels. In Fig. 5, for example, the actor node, o_1, has three outgoing directed arcs labeled with the same activity, α_2. However, in the real enactment logs of the Hiring workflow procedure during the runtime, o_1 selects one of the neighbor actors, o_2, o_3, o_4, so as to proceed to the selected actor after performing α_2 during runtime. As a result, according to the actor selection mechanisms, such as random, sequential, heuristic selection mechanisms, and so on, the social relationships on runtime will be differently formed with the social relationships on this discovery result. This is why we need to differentiate the workflow-supported social network discovery work from the workflow-supported social network rediscovery work.

3.1.3. Implication of the workflow-supported social network

The main purpose of the workflow-supported social network is for discovering the work-behavioral relationships among workflow actors, and for eventually analyzing the degrees of workloads and the empowerments for individuals of the workflow actors in enacting a corresponding workflow model. Meanwhile, back at the traditional social network, the purpose is discovering and analyzing the social-behavioral relationships among people. Therefore, there is a cardinal difference between them: the traditional social network must be formed by two or more individuals and its rudimentary analysis, so it starts from a dyadic relationship, whereas the workflow-supported social network ought to be shaped by only a single individual, because it primarily represents the distributive status of individual workloads. As shown in the left-hand side of Fig. 6, we suppose that every activity in the ICN-based workflow model is performed by a single workflow actor, o_1. Then the discovery algorithm will generate a workflow-supported social network with a single node like on right-hand side of the

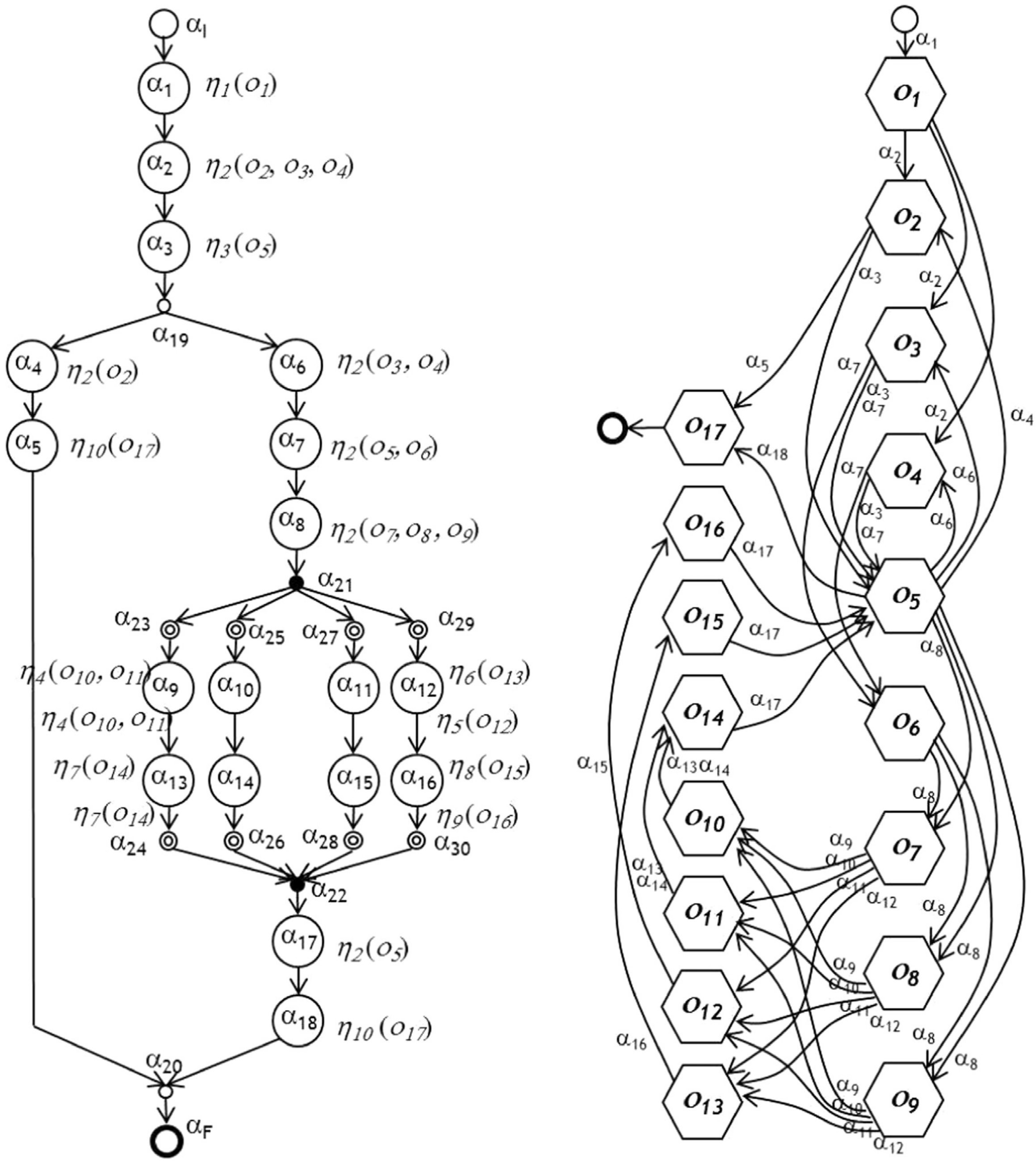

Fig. 5. A workflow-supported social network discovered from the ICN-based Hiring workflow model.

figure, which implies the whole workload for enacting the workflow model has been allotted to the workflow actor, o_1.

This phenomenon gives us very significant intuition in terms of adopting the analysis equations and algorithms (Knoke and Yang, 2008) of the traditional social networks as analysis techniques of the workflow-supported social networks. In other words, we need to adopt and revise the degree centrality analysis equations to properly apply them to analyze the workloads' status of the workflow-supported social networks. In the next subsection, we expand the details of the revisions and extensions of the analysis equations with an example.

3.2. Analyzing workflow-supported social networks

As stated in the beginning part of this section, the framework aims at the prominence concepts and measures—centrality and prestige (Knoke and Yang, 2008) that can be discovered from workflow-supported social networks formally defined in the previous analysis phase. For the sake of the analysis phase, it is necessary for discovered workflow-supported social networks to be represented in SocioMatrices (Jeon and Kim, 2013). By applying a series of social network analysis techniques to the SocioMatrices, we are able to obtain the ultimate answer for the issued question. In this analysis phase, we describe the details of the SocioMatrices generation algorithms and the SocioMatrices analysis equations, and verify them through the workflow-supported social network discovered from the Hiring workflow procedure (Kim and Ellis, 2009).

3.2.1. SocioMatrices from workflow-supported social networks

In general, the social network analysis literature gives suggestions of two classes of SocioMatrices and graphs to analyze cognitive social structures: *binary directed/nondirected SocioMatrix* and *valued directed/nondirected SocioMatrix*. It uses them to

Table 1
Formal representation of the discovered WSSN model.

$\Lambda = (\sigma, \psi, \mathbf{S}, \mathbf{E})$	over **A**, **C**The Workflow-Supported Social Network Model
Elementary activities	$\mathbf{A} = \{\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6, \alpha_7, \alpha_8, \alpha_9, \alpha_{10}, \alpha_{11}, \alpha_{12}, \alpha_{13}, \alpha_{14}, \alpha_{15}, \alpha_{16}, \alpha_{17}, \alpha_{18}, \alpha_I, \alpha_F\}$
Actors	$\mathbf{C} = \{o_I, o_1, o_2, o_3, o_4, o_5, o_6, o_7 o_8, o_9, o_{10}, o_{11}, o_{12}, o_{13}, o_{14}, o_{15}, o_{16}, o_{17}, o_F\}$
$\mathbf{S} = \varnothing$	Initial actors from some external workflow-supported social network models
$\mathbf{E} = \varnothing$	Final actors to some external workflow-supported social network models
$\sigma = \sigma_i \cup \sigma_o$	
σ_i:**Social Predecessors**	σ_o:**Social Successors**
$\sigma_i(o_I) = \varnothing$;	$\sigma_o(o_I) = \{o_1\}$;
$\sigma_i(o_1) = \{o_I\}$;	$\sigma_o(o_1) = \{o_2, o_3, o_4\}$;
$\sigma_i(o_2) = \{o_2, o_5\}$;	$\sigma_o(o_2) = \{o_5, o_{17}\}$;
$\sigma_i(o_3) = \{o_1, o_5\}$;	$\sigma_o(o_3) = \{o_5, o_6\}$;
$\sigma_i(o_4) = \{o_1, o_5\}$;	$\sigma_o(o_4) = \{o_5, o_6\}$;
$\sigma_i(o_5) = \{o_2, o_3, o_4, o_{14}, o_{15}, o_{16}\}$;	$\sigma_o(o_5) = \{o_2, o_3, o_4, o_7, o_8, o_9, o_{17}\}$;
$\sigma_i(o_6) = \{o_3, o_4\}$;	$\sigma_o(o_6) = \{o_7, o_8, o_9\}$;
$\sigma_i(o_7) = \{o_5, o_6\}$;	$\sigma_o(o_7) = \{o_{10}, o_{11}, o_{12}, o_{13}\}$;
$\sigma_i(o_8) = \{o_5, o_6\}$;	$\sigma_o(o_8) = \{o_{10}, o_{11}, o_{12}, o_{13}\}$;
$\sigma_i(o_9) = \{o_5, o_6\}$;	$\sigma_o(o_9) = \{o_{10}, o_{11}, o_{12}, o_{13}\}$;
$\sigma_i(o_{10}) = \{o_7, o_8, o_9\}$;	$\sigma_o(o_{10}) = \{o_{14}\}$;
$\sigma_i(o_{11}) = \{o_7, o_8, o_9\}$;	$\sigma_o(o_{11}) = \{o_{14}\}$;
$\sigma_i(o_{12}) = \{o_7, o_8, o_9\}$;	$\sigma_o(o_{12}) = \{o_{16}\}$;
$\sigma_i(o_{13}) = \{o_7, o_8, o_9\}$;	$\sigma_o(o_{13}) = \{o_{15}\}$;
$\sigma_i(o_{14}) = \{o_{10}, o_{11}\}$;	$\sigma_o(o_{14}) = \{o_5\}$;
$\sigma_i(o_{15}) = \{o_{13}\}$;	$\sigma_o(o_{15}) = \{o_5\}$;
$\sigma_i(o_{16}) = \{o_{12}\}$;	$\sigma_o(o_{16}) = \{o_5\}$;
$\sigma_i(o_{17}) = \{o_5, o_2\}$;	$\sigma_o(o_{17}) = \{o_F\}$;
$\sigma_i(o_F) = \{o_{17}\}$;	$\sigma_o(o_F) = \varnothing$;
$\psi = \psi_i \cup \psi_o$	
ψ_i:**Previous Work**	ψ_o:**Acquisition Work**
$\psi_i(o_I) = \varnothing$;	$\psi_o(o_I) = \{(\alpha_I, o_1)\}$;
$\psi_i(o_1) = \{(\alpha_I, o_I)\}$;	$\psi_o(o_1) = \{(\alpha_1, o_2), (\alpha_1, o_3), (\alpha_1, o_4)\}$;
$\psi_i(o_2) = \{(\alpha_1, o_1), (\alpha_3, o_5)\}$;	$\psi_o(o_2) = \{(\alpha_2, o_5), (\alpha_4, o_{17})\}$;
$\psi_i(o_3) = \{(\alpha_1, o_1, (\alpha_3, o_5)\}$;	$\psi_o(o_3) = \{(\alpha_6, o_5), (\alpha_6, o_6)\}$;
$\psi_i(o_4) = \{(\alpha_1, o_1, (\alpha_3, o_5)\}$;	$\psi_o(o_4) = \{(\alpha_6, o_5), (\alpha_6, o_6)\}$;
$\psi_i(o_5) = \{(\alpha_2, o_2), (\alpha_6, o_3), (\alpha_6, o_4),$ $(\alpha_{13}, o_{14}), (\alpha_{14}, o_{14}), (\alpha_{15}, o_{16}), (\alpha_{16}, o_{15})\}$;	$\psi_o(o_5) = \{(\alpha_3, o_2), (\alpha_3, o_3), (\alpha_3, o_4), (\alpha_7, o_7),$ $(\alpha_7, o_9), (\alpha_{17}, o_{17}), (\alpha_7, o_8)\}$;
$\psi_i(o_6) = \{(\alpha_6, o_3), (\alpha_6, o_4)\}$;	$\psi_o(o_6) = \{(\alpha_7, o_7), (\alpha_7, o_8), (\alpha_7, o_9)\}$;
$\psi_i(o_7) = \{(\alpha_7, o_5), (\alpha_7, o_6)\}$;	$\psi_o(o_7) = \{(\alpha_8, o_{10}), (\alpha_8, o_{11}), (\alpha_8, o_{12}), (\alpha_8, o_{13})\}$;
$\psi_i(o_8) = \{(\alpha_7, o_5), (\alpha_7, o_6)\}$;	$\psi_o(o_8) = \{(\alpha_8, o_{10}), (\alpha_8, o_{11}), (\alpha_8, o_{12}), (\alpha_8, o_{13})\}$;
$\psi_i(o_9) = \{(\alpha_7, o_5), (\alpha_7, o_6)\}$;	$\psi_o(o_9) = \{(\alpha_8, o_{10}), (\alpha_8, o_{11}), (\alpha_8, o_{12}), (\alpha_8, o_{13})\}$;
$\psi_i(o_{10}) = \{(\alpha_8, o_7), (\alpha_8, o_8), (\alpha_8, o_9)\}$;	$\psi_o(o_{10}) = \{(\alpha_9, o_{14}), (\alpha_{10}, o_{14})\}$;
$\psi_i(o_{11}) = \{(\alpha_8, o_7), (\alpha_8, o_8), (\alpha_8, o_9)\}$;	$\psi_o(o_{11}) = \{(\alpha_9, o_{14}), (\alpha_{10}, o_{14})\}$;
$\psi_i(o_{12}) = \{(\alpha_8, o_7), (\alpha_8, o_8), (\alpha_8, o_9)\}$;	$\psi_o(o_{12}) = \{(\alpha_{11}, o_{16})\}$;
$\psi_i(o_{13}) = \{(\alpha_8, o_7), (\alpha_8, o_8), (\alpha_8, o_9)\}$;	$\psi_o(o_{13}) = \{(\alpha_{12}, o_{15})\}$;
$\psi_i(o_{14}) = \{(\alpha_9, o_{10}), (\alpha_{10}, o_{10}), (\alpha_9, o_{11}), (\alpha_{10}, o_{11})\}$;	$\psi_o(o_{14}) = \{(\alpha_{13}, o_5), (\alpha_{14}, o_5)\}$;
$\psi_i(o_{15}) = \{(\alpha_{12}, o_{13})\}$;	$\psi_o(o_{15}) = \{(\alpha_{16}, o_5)\}$;
$\psi_i(o_{16}) = \{(\alpha_{11}, o_{12})\}$;	$\psi_o(o_{16}) = \{(\alpha_{15}, o_5)\}$;
$\psi_i(o_{17}) = \{(\alpha_4, o_2), (\alpha_{17}, o_5)\}$;	$\psi_o(o_{17}) = \{(\alpha_5, o_F), (\alpha_{18}, o_F)\}$;
$\psi_i(o_F) = \{(\alpha_5, o_{17}), (\alpha_{18}, o_{17})\}$;	$\psi_o(o_F) = \varnothing$;

construct a sociogram (Knoke and Yang, 2008), which is a two-dimensional diagram for displaying the relations among actors in a bounded social system. The term "directed" indicates directed relations or ties from the actor at the tail to the actor at the arrowhead (e.g. giving advice), while the term "non-directed" (no arrowheads) implies mutual relations. Likewise, when a directed/non-directed graph is transformed to a SocioMatrix, the term "binary" implies the most basic measurement of the presence or the absence of a tie, a dichotomy indicated by binary values of 1 and 0, respectively. Also SocioMatrices may include non-binary values, reflecting the intensity of relations or ties, such as frequency of contacts, tie strength, or magnitude of associations, and therefore the cell entries in SocioMatrix can vary from 0 to the maximum level of dyadic interaction.

Basically, a workflow-supported social network could be possibly transformed to both of the classes of SocioMatrix simultaneously, even though its original properties pertain to directed and valued graphs. We devise a series of SocioMatrix generation algorithms, which supports those classes of SocioMatrices, each of which automatically generates its corresponding SocioMatrix from a workflow-supported social network. Note that in this paper, we give only two of the algorithms for a binary directed SocioMatrix and binary non-directed SocioMatrix due to the page limitations.

Binary Directed SocioMatrix Generation Algorithm
Input A workflow-supported social network, $\Lambda = (\sigma, \psi, S, E)$;
Output Two symmetric binary SocioMatrices, $\mathbf{Z}_{in}(N, N)$ and $\mathbf{Z}_{out}(N, N)$, where N is the number elements in the set of **C**

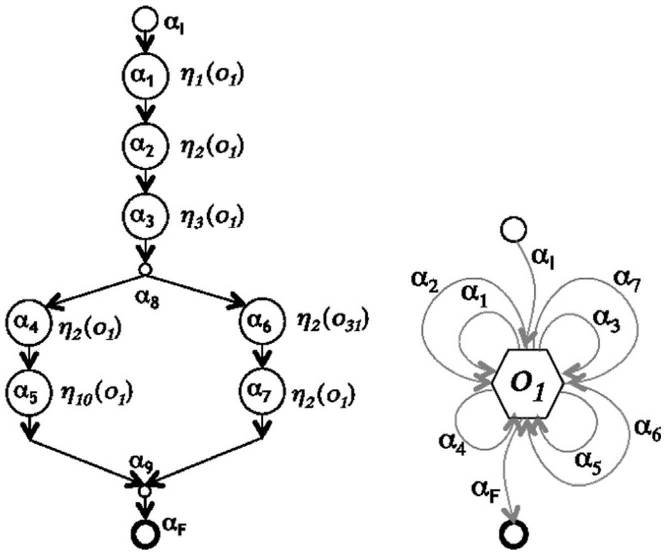

Fig. 6. A workflow-supported social network with a single actor.

actors.
Begin Procedure
Initialize all entries of > $\mathbf{Z}(N, N)$ **To** *Zeroes*;
For $(\forall\ o \in \mathbf{C})$ **Do**
Begin
/* The Incoming Relations of $\mathbf{Z}_{in}(N, N)$*/
Set *One* **To** *entries of* $\mathbf{Z}_{in}(o,$ *each member of* $\sigma_i(o))$;
/* The Outgoing Relations of $\mathbf{Z}_{out}(N, N)$*/
Set *One* **To** *entries of* $\mathbf{Z}_{out}(o,$ *each member of* $\sigma_o(o))$;
End
End Procedure

Binary Non-Directed SocioMatrix Generation Algorithm
Input A workflow-supported social network, $\Lambda = (\sigma, \psi, \mathbf{S}, \mathbf{E})$;
Output A symmetric binary SocioMatrix, $\mathbf{Z}(N, N)$, where N is the number elements in the set of **C** actors.
Begin Procedure
Initialize all entries of $\mathbf{Z}(N, N)$ **To** *Zeroes*;
For $(\forall\ o \in C)$ **Do**
Begin
/* Set the Incoming Relations to $\mathbf{Z}(N, N)$ */
Set *One* **To** *entries of* $\mathbf{Z}(o,$ *each member of* $\sigma_i(o))$;
/* Set the Outgoing Relations to $\mathbf{Z}(N, N)$ */
Set *One* **To** *entries of* $\mathbf{Z}(o,$ *each member of* $\sigma_o(o))$;
End
End Procedure

In order to verify the devised SocioMatrices generation algorithms, we directly apply the above algorithms to the workflow-supported social network on the right-hand side of Fig. 5, discovered from the ICN-based Hiring workflow procedure. Table 2 shows the binary non-directed SocioMatrix successfully generated by the binary non-directed SocioMatrix generation algorithm. Additionally, it is possible to generate a valued non-directed SocioMatrix by adding two of the symmetric binary SocioMatrices, $\mathbf{Z}_{in}(N, N)$ and $\mathbf{Z}_{out}(N, N)$. In this case, the values of the entries might indicate the frequency of activity acquisitions between the paired actors.

3.2.2. *Workload centrality analysis*

Based upon these SocioMatrices, we are able to calculate the prominence measures of centrality and prestige measures by implementing the corresponding algorithms given in Knoke and Yang (2008), Jeon and Kim (2013). Through the centrality and prestige concepts and measures, we can obtain the fully reasonable analysis results enough for answering to the issue question, "Who are the most prominent actors in enacting the corresponding workflow procedure?" The implication of the centrality measures is that a prominent actor has high involvement in many relations, regardless of whether ties are incoming and outgoing. The prestige measures imply that a prominent actor initiates few relations but receives many directed ties. This paper focuses on the centrality measures in analyzing workflow-supported social networks. As mentioned in the previous subsection, there exist several social network analysis techniques and algorithms for the centrality measures. Among them, the most widely used centrality measures are degree, closeness, and betweenness, and these measures vary in their applicability to non-directed and directed relations, and differ at the individual actor and the group or complete network levels. We are interested in how to apply the degree centrality concept (Knoke and Yang, 2008) to analyze workflow-supported social networks.

The workload centrality measures (Jeon and Kim, 2013) can be applied to the individual actor (*actor workload centrality*) as well as the actors group (*group workload centrality*) of complete workflow-supported social networks. Based upon the binary non-directed SocioMatrix of a workflow-supported social network, the actor workload centrality measures the utmost extent to which an actor connects to itself or all other actors in the network. For a binary non-directed workflow-supported social network with g actors, the workload centrality for actor i is the sum of i's adjacent ties to the $g-1$ other actors, and its equation is given in Eq. (1); where $C_d(N_i)$ denotes workload centrality for actor i. That is, the actor-degree centrality analysis equation, $C_d(N_i) = \sum_{j=1}^{g-1} x_{ij} (i \neq j)$ (Knoke and Yang, 2008), must be modified so that the number of adjacent ties with actor i counts the links to itself and the $g-1$ other j actors as follows:

- *Actor-workload centrality analysis equation*:

$$C_d(N_i) = \sum_{j=1}^{g} x_{ij} \tag{1}$$

The actor workload centrality reflects each actor's connectivity to other actors including itself. However, its value depends on g, the sizes of workflow-supported social networks; in other words, the larger the network, the higher the maximum of possible workload centrality value. It means that it needs to be normalized in order to eliminate the effect of network-size variation on calculating the workload centrality, so the normalized actor-degree centrality analysis equation (Knoke and Yang, 2008) as a standardized measure ought to be modified as Eq. (2), and so its value is between 0 and 1, which is independent of the sizes of workflow-supported social networks.

- *Normalized actor-workload centrality analysis equation*:

$$C_n(N_i) = \frac{C_d(N_i)}{g} \tag{2}$$

The group workload centrality measures the extent to which the actors in a workflow-supported social network differ from one another in their individual workload centralities, which is similar to the concept of dispersion, in that the standard deviation that indicates the amount of variation or spread around a central tendency value. In Eqs. (3) and (4) (Knoke and Yang, 2008), $C_A(N^*)$

Table 2
Binary non-directed SocioMatrix of Fig. 5.

Actors	o_1	o_2	o_3	o_4	o_5	o_6	o_7	o_8	o_9	o_{10}	o_{11}	o_{12}	o_{13}	o_{14}	o_{15}	o_{16}	o_{17}
o_1	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
o_2	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1
o_3	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
o_4	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
o_5	0	1	1	1	0	0	1	1	1	0	0	0	0	1	1	1	1
o_6	0	0	1	1	0	0	1	1	1	0	0	0	0	0	0	0	0
o_7	0	0	0	0	1	1	0	0	0	1	1	1	1	0	0	0	0
o_8	0	0	0	0	1	1	0	0	0	1	1	1	1	0	0	0	0
o_9	0	0	0	0	1	1	0	0	0	1	1	1	1	0	0	0	0
o_{10}	0	0	0	0	0	0	1	1	1	0	0	0	0	1	0	0	0
o_{11}	0	0	0	0	0	0	1	1	1	0	0	0	0	1	0	0	0
o_{12}	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	1	0
o_{13}	0	0	0	0	0	0	1	1	1	0	0	0	0	0	1	0	0
o_{14}	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0
o_{15}	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0
o_{16}	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0
o_{17}	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

denotes the largest actor workload centrality observed in a workflow-supported social network, and $C_A(N_i)$ are the workload centralities of the $g-1$ other actors. Conclusively, the index of group workload centrality measurement in a workflow-supported social network may take values between 0.0 and 1.0. The closer that group workload centrality measure is to 1.0, the more uneven or hierarchical is the workload centrality of actors in a workflow-supported social network, while on the other hand, the closer the measure is to 0.0, then the more the workload centrality of the workflow-supported social network is evenly dispersed.

- *Group-workload centrality analysis equations*:

$$C_A = \frac{\sum_{i=1}^{g}\left[C_A(N^*) - C_A(N_i)\right]}{\max \sum_{i=1}^{g}\left[C_A(N^*) - C_A(N_i)\right]} \tag{3}$$

- *Index of group-workload centrality analysis equations*:

$$C_D = \frac{\sum_{i=1}^{g}\left[C_A(N^*) - C_A(N_i)\right]}{g\cdot(g-1)}, \quad (g > 1) \tag{4}$$

As an example, we apply the actor-workload centrality equations and the group workload centrality equations to the SocioMatrix, Table 2, of the workflow-supported social network discovered from the ICN-based Hiring workflow procedure. The final results and their radiational view are shown in Table 3 and Fig. 7, respectively. Based upon these workload centrality measurements, we can guarantee that the framework is surely able to answer to the issued qualitative and quantitative questions as follows:

- *Question*: Who are the most important or prominent actors, and how much do they contribute in enacting the ICN-based Hiring workflow procedure (Kim and Ellis, 2009)?
- *Answer* and *analysis results*: The most prominent actor in enacting the Hiring workflow procedure is the actor, o_5, because his/her workload centrality measure is 10 and its normalized workload is 0.63, which is the largest one out of the actors' values. Also, the group workload centrality measure of the Hiring workflow procedure is 0.495 that comes from the value of $\frac{100}{202}$, and its index is $\frac{100}{240} = 0.417$. The actors' values in the group workload, C_A, row and the index, C_D, row represent the differences (deviations) between the normalized workload centrality value of each actor and the group workload centrality values, respectively. Also it is easily recognized that all of other actors except the actor, o_5, have negative deviations, and the lower values than the values of the group workload centrality and its index.
- *Interpretation*: The closer the group workload centrality and its index values are to 0.0, the more the workload centrality of the workflow-supported social network is evenly dispersed. Based upon this interpretation and the values (0.495&0.417), the example workflow-supported social network of the Hiring workflow procedure ought to be reasonably dispersed and biased to neither of the extreme situations, which implies that the workloads of the actors involved in enacting the workflow procedure seem to be acceptably dispersed and fairly shared. However, the workload of the actor, o_5, is exceptionally higher than others', so we can interpret that the actor is the most important and prominent worker in performing the Hiring workflow procedure. According to the domain, the workflow-supported social network reflects and views the workload sharing or the work-intimacy levels among the actors who are associated with the underlying workflow procedure.

Table 3
Workload centrality measures on the SocioMatrix of Table 2.

Actors		o_1	o_2	o_3	o_4	o_5	o_6	o_7	o_8	
Actor-workload, $C_d(N_i)$		3	3	3	3	**10**	5	6	6	
Normalized workload, $C_n(N_i)$		0.19	0.19	0.19	0.19	**0.63**	0.31	0.38	0.38	
Group-workload, C_A	**0.495**	−0.30	−0.30	−0.30	−0.30	+0.14	−0.18	−0.11	−0.11	
Index of group-workload, C_D	**0.417**	−0.22	−0.22	−0.22	−0.22	+0.22	−0.10	−0.02	−0.02	
Actors		o_9	o_{10}	o_{11}	o_{12}	o_{13}	o_{14}	o_{15}	o_{16}	o_{17}
Actor-workload, $C_d(N_i)$		6	4	4	4	4	3	2	2	2
Normalized workload, $C_n(N_i)$		0.38	0.25	0.25	0.25	0.25	0.19	0.13	0.13	0.13
Group-workload, C_A	**0.495**	−0.11	−0.24	−0.24	−0.24	−0.24	−0.30	−0.36	−0.36	−0.36
Index of group-workload, C_D	**0.417**	−0.02	−0.16	−0.16	−0.16	−0.16	−0.22	−0.28	−0.28	−0.28

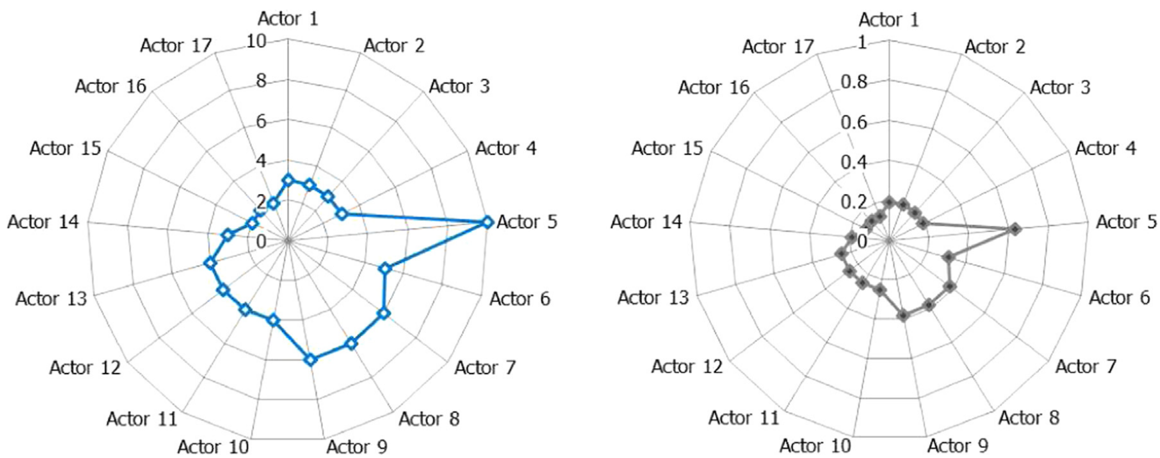

Fig. 7. Radiational views of the actor-workload and normalized actor-workload centralities.

In summary, the framework from the discovery to the analysis of workflow-supported social networks has been described in detail and verified through applying to a real workflow procedure, the Hiring workflow procedure introduced in Kim and Ellis (2009). Particularly, in terms of analyzing workflow-supported social networks, the framework copes with only the workload centrality concepts, so it needs to be refurbished by supplementing some social network analysis techniques and algorithms that can process the closeness and betweenness centrality concepts and the prestige measurements.

3.3. *Extensions of workload centrality analysis equations supporting workflow packages*

In the theoretical framework, the workload-centrality analysis is just applied to a single workflow-supported social network, so the workload centrality equations are developed to measure the centralities of the individual actor (*actor workload centrality*) as well as a group of actors (*group workload centrality*) of a workflow-supported social network. Therefore, in this paper, it is necessary for the workload-centrality analysis equations to be extended so as to handle a group of workflow models, which is a workflow package. The following are the extended workload-centrality analysis equations consisting of actor-workload centrality equation, normalized actor-workload centrality equation, group-workload centrality equation, and indexed group-workload centrality equation. In extending the equations, we assume that a workflow package holds n workflow models, and every workflow model in the same workflow package associates the same number of actors (g actors).

The extended actor-workload centrality analysis equation for a workflow package is based upon a group of the binary non-directed SocioMatrices, which are generated from a group of workflow-supported social networks respectively. For a workflow package with n workflow models and each binary non-directed SocioMatrix with g being actors, the actor-workload centrality measurement for actor i is the sum of i's adjacent ties to the g other actors including itself, and its equations are given in Eq. (5); $C_d(N_{ki})$ denotes the workload centrality of actor i and $\sum_{j=1}^{g} x_{kij}$ in the kth workflow model; $C_d(N_i)$ denotes the workload centrality of actor i in the whole of a workflow package, which comes by summing all of the workflow models, $C_d(N_{1i})$, $C_d(N_{2i})$,..., $C_d(N_{ni})$.

- Actor-workload centrality analysis equation for a group of workflow-supported social networks discovered from the corresponding workflow package:

$$C_d(N_{ki}) = \left[\sum_{j=1}^{g} x_{kij}\right]_{k=1}^{n}; \quad C_d(N_i) = \sum_{k=1}^{n} C_d(N_{ki}) \tag{5}$$

The actor-workload centrality reflects each actor's connectivity to other actors. However, its value depends on g, the sizes of workflow-supported social networks and the sizes of workflow packages; in other words, the larger the network, the higher the maximum of possible workload centrality value. It means that it needs to be normalized in order to eliminate the effect of variation in network size on workload centrality. Therefore, Eq. (6) gives a series of standardized measurements for all workflow-supported social networks of the corresponding workflow package, and the values are between 0 and 1, which is independent of the sizes of workflow-supported social networks as well as the sizes of workflow packages.

- Normalized actor-workload centrality analysis equation for a group of workflow-supported social networks discovered from the corresponding workflow package:

$$C_n(N_{ki}) = \left[\frac{C_d(N_{ki})}{g}\right]_{k=1}^{n}; \quad C_n(N_i) = \sum_{k=1}^{n} \frac{C_d(N_{ki})}{g} \tag{6}$$

The group workload centrality measures the extent to which the actors in a workflow-supported social network differ from one another in their individual workload centralities, which is similar to the concept of dispersion. The standard deviation indicates the amount of variation or spread around a central tendency value. Eqs. (7) and (8) give the group workload centrality measurement and the indexed group workload centrality measurement, respectively, for a workflow-supported social network discovered from a workflow package. In the equations, $C_d(N_k^*)$ denotes the largest actor-workload centrality measurement observed in kth workflow-supported social network, $C_A(N_{ki})$ is the actor-workload centrality of the ith actor, and $C_d(N_k^{max})$ indicates the maximum actor-workload centrality measurement that can be possibly measured in the kth workflow-supported social network.

- Group-workload centrality analysis equations for a group of workflow-supported social networks discovered from the corresponding workflow package:

$$C_A = \sum_{k=1}^{n}\left[\frac{\sum_{i=1}^{g}\left(C_d(N_k^*) - C_d(N_{ki})\right)}{\sum_{i=1}^{g}\left(C_d(N_k^{max}) - C_d(N_{ki})\right)}\right], \quad (g > 1) \tag{7}$$

- Indexed group-workload centrality analysis equations for a group of workflow-supported social networks discovered from the corresponding workflow package:

$$C_D = \sum_{k=1}^{n} \left[\frac{\sum_{i=1}^{g} \left(C_d(N_k^*) - C_d(N_{ki}) \right)}{(g-1)^2} \right], \quad (g > 1) \tag{8}$$

Conclusively, the indexed group workload centrality measurement in a workflow-supported social network discovered from the corresponding workflow package may take values between 0.0 and 1.0. The closer that group workload centrality measure is to 1.0, the more uneven or hierarchical is the workload centrality of actors in a workflow-supported social network. On the other hand, the closer the measure is to 0.0, then the more the group workload centrality of a workflow-supported social network is evenly dispersed.

4. A system and its operational examples

The framework can be implemented in theoretics as well as in systematics. In the previous section, we proved that the proposed framework theoretically corrects and that the framework is practically feasible. Based upon the theoretical evidence, the framework is systematically implemented by using the Java programming language, and a series of snapshots captured from another operational example using the implemented system shows that the system works correctly. This section expands the details of the workload-centrality analysis equations applied to implement the system.

4.1. The systematic framework

As mentioned previously, a systematic framework to discover and analyze workflow-supported social networks has been developed as illustrated in Fig. 8. The systematic framework consists of three architectural components—XPDL workflow package, SocioMatrix, and workload-centrality measurements and their graphical reports—and their related implementing-algorithms. This systematic framework is theoretically supported by the theoretical framework described in the previous section. Nevertheless, unlike the theoretical framework based upon the ICN-based workflow model, it is extended so as to handle an XPDL workflow package holding a group of XPDL workflow models. Furthermore, the system is theoretically backed up by the extended versions of the workload-centrality analysis equations, such as actor-workload centrality analysis equations and group-workload centrality analysis equations, so as to systematically analyze a group of workflow-supported social networks discovered from the corresponding XPDL workflow package.

Fig. 9 illustrates the XPDL's process-level meta-model (WfMC, 2005) made up of three essential components (participant, activity, and transition), which are directly included in XPDL version 1.0, and the extended version of XPDL 2.0 components supporting the BPMN (business process modeling notations) constructs, such as Pool, Lane, Events, Route including Gateway, and others. The systematic framework is currently implemented in version 1.0 of XPDL. However, it does not matter whatever the version of XDPL is used for the input workflow models and packages, because the implemented system touches only the essential components of XPDL-formatted workflow models and packages to discover the workflow-supported social network. The following three skeletons of the basic XPDL tags are corresponding to the three essential components to be used for discovering the workflow-supported social network from an XPDL-based workflow package:

- Activity:

⟨Activities⟩
  ⟨Activity Id=" " Name=" "/⟨
  …
  ⟨Performers ⟩…⟨/Performers ⟩
  …
⟨/Activities⟩

- Participant:

⟨Participants⟩
  ⟨Performer Id=" " Name=" "/⟩…
  …
⟨/Participants⟩

- Transition:

⟨Transitions ⟩
  ⟨Transition Id=""Name=""From=""To=""FlowType=""/⟩
  …
⟨/Transitions⟩

The Systematic Algorithm Generating SocioMatrix from XPDL Workflow Packages
Input A Set of Workflow Models (Package) in XPDL with W, A, T, P;

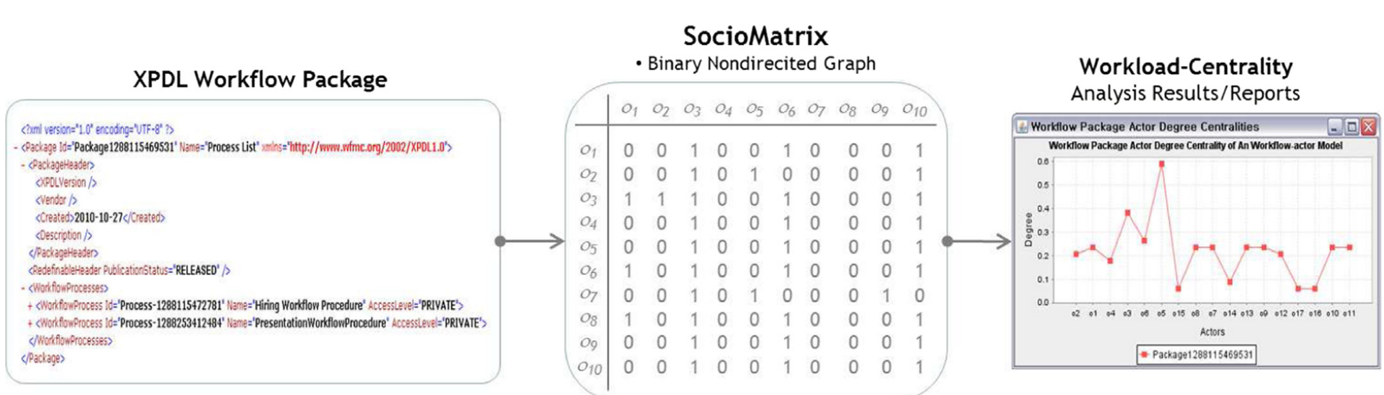

Fig. 8. A systematic framework for W-SN discovery and workload-centrality analysis.

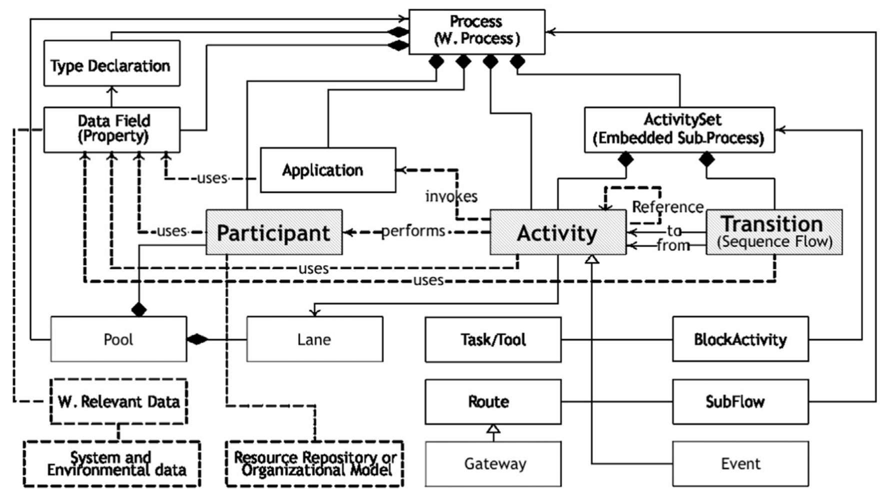

Fig. 9. The XPDL 2.0 process meta-model.

- a set of process-tags: $W = \{\omega_1, \ldots, \omega_n\}$;
- a set of activity-tags: $A = \{\alpha_1, \ldots, \alpha_s\}$;
- a set of transition-tags: $T = \{\tau_1 \ldots, \tau_z\}$;
- a set of performer-tags: $P = \{\phi_1, \ldots, \phi_g\}$;

Output A (Symmetric or Asymmetric) and (Binary or Valued) SocioMatrix, $\mathbf{Z}(N, N)$, where **N** is the total number of performers in the corresponding workflow package-tag.

Begin Procedure

Initialize all entries of $\mathbf{Z}(N, N)$ **To** *Zeroes*;

For $(\forall\, \omega \in W)$ **Do**

For $(\forall\, \tau \in \omega.T)$ **Do**

Switch(the-Type-of-SocioMatrix)

Case 1: /* Bi-Sym-Nondirected SocioMatrix */

For $(\forall\, \phi_s$ *in* α_s *of* τ.From $\wedge \alpha_s =$ '*task-activity*') **Do**

For $(\forall\, \phi_d$ *in* α_d *of* τ.To $\wedge \alpha_d =$ '*task-activity*') **Do**

Set *One* **To** $\mathbf{Z}(\phi_s, \phi_d)$;

Set *One* **To** $\mathbf{Z}(\phi_d, \phi_s)$;

Done;

Done;

Break;

Case 2: /* Bi-Asym-Directed SocioMatrix */

For $(\forall\, \phi_s$ *in* α_s *of* τ.From $\wedge \alpha_s =$ '*task-activity*') **Do**

For $(\forall\, \phi_d$ *in* α_d *of* τ.To $\wedge \alpha_d =$ '*task-activity*') **Do**

Set *One* **To** $\mathbf{Z}(\phi_s, \phi_d)$;

Done;

Done;

Break;

Case 3: /* Val-Sym-Nondirected SocioMatrix */

For $(\forall\, \phi_s$ *in* α_s of τ. From $\wedge \alpha_s =$ '*task-activity*') **Do**

For $(\forall\, \phi_d$ in α_d of τ.To $\wedge \alpha_d =$ '*task-activity*') **Do**

Add *One* **To** $\mathbf{Z}(\phi_s, \phi_d)$;

Add *One* **To** $\mathbf{Z}(\phi_d, \phi_s)$;

Done;

Done;

Break;

Case 4: /* Val-Asym-Directed SocioMatrix */

For $(\forall\, \phi_s$ *in* α_s *of* τ. From $\wedge \alpha_s =$ '*task-activity*') **Do**

For $(\forall\, \phi_d$ *in* α_d *of* τ.To $\wedge \alpha_d =$ '*task-activity*') **Do**

Add *One* **To** $\mathbf{Z}(\phi_s, \phi_d)$;

Done;

Done;

Break;

hctiwS

Done

Done

End Procedure

.

From the essential components' tags (activity, participant, transition) and their relationships' tags (performer, from, to) in an XPDL-based workflow package, the system is able to automatically generate a SocioMatrix that represents the corresponding workflow-supported social network. Basically, the workflow-supported social network is graphically represented by a directed graph with activity-labeled edges, and its mathematical representation can be possibly embodied in a SocioMatrix. Therefore, the system implemented the above systematic algorithm generates a SocioMatrix from an XDPL-based workflow package. The generated SocioMatrix can be characterized by the properties of workloads in the underlying workflow-supported social network. As specified in the systematic algorithm, there are two types of workload properties: weight and direction. The weight property characterizes SocioMatrix as either binary or valued, and the direction property classifies it as either symmetric (non-directed) or asymmetric (directed). Ultimately, SocioMatrix can be classified into binary-symmetric, binary-asymmetric, valued-symmetric, or valued-asymmetric, and these SocioMatrices can be used for analyzing workload-centrality measurements as well as workload–prestige measurements of the corresponding workflow-supported social network. The workload centrality and prestige measurements seek to quantify workflow-performers' prominences within a complete workflow-supported social network by summarizing the structural relations, particularly the structures of choices made, choices received, and indirect ties, among all nodes. Conclusively, the analysis functionality of the systematic framework can be summarized as follows:

- *Workload centrality analysis using binary (or valued) symmetric SocioMatrix*: Measuring how much a prominent workflow-performer has high involvement, which is interpreted as workload, in many relations, regardless of whether sending or receiving ties in a workflow-supported social network.

- *Workload prestige analysis using binary (or valued) asymmetric SocioMatrix*: Measuring how a prominent workflow-performer initiates few relations but receives many direct ties, which is interpreted as work-report, in a workflow-supported social network. For an instance, the upper echelons of the workplace rarely or never report their activities to their subordinates, whereas the lower-echelon employees routinely report to their managerial supervisors about their work activities.

The current system is implemented only for analyzing the workload-centrality measurements, so it is able to handle the binary symmetric SocioMatrix and the valued symmetric SocioMatrix. Finally, the system's development environments are listed as follows. Particularly, we suppose that the XPDL workflow package's release version is XPDL 1.0. So, it is necessary to extend it to support later versions, which reflect the BPMN[5] graphical constructs.

- *Operating system*: Windows XP Pro, '02 Service Pack 3.
- *Programming language*: Java Development Toolkit v.6.0.
- *XPDL version*: XPDL 1.0 & 2.0.
- *SocioMatrix*: Binary symmetric (nondirected).
- *Development tool*: Eclipse Ganymede Release 3.4.
- *Libraries*: SWT/JFace, JFreeChart, etc.

4.2. An operational example

As an operational example, an XPDL workflow package shown in Fig. 10 has been applied to the systematic framework. The specification of the exampled XPDL-based workflow package is summarized in Table 4. That is, the package named "Process List" holds two XPDL workflow models: one is named "Hiring workflow procedure," and the other is "Presentation workflow procedure." The former workflow model consists of seventeen task-activities (DatabaseUpdate task-activity is performed twice) that are enacted by seventeen actors (o_1–o_{17}), and the latter workflow model has seven task-activities to be enacted by the same actors.

In XPDL version 2.0, there are five types of activities: task, subflow, block-activity, route including gateway, and event. Out of them, only the task-activity assigns a group of actors, as shown in the middle part of Fig. 10, where the task-activity, Id="1288115512796" Name="NewApplicationInfo", is allotted into three actors, o_2, o_3, o_4. The following are to project the activity tags of the middle part from the exampled XPDL-based workflow package. By taking into account these workflow-performer tags inside the task-activity tags and From/To information of the transition tags belonging to each ⟨WorkflowProcess⟩ tag, the system is able to generate a binary symmetric SocioMatrix corresponding to each workflow procedure.

```
⟨Activities⟩
  …
  ⟨Activity Id= "1288115512796"
  Name="NewApplicationInfo"⟩
    …
    ⟨Performers⟩
      ⟨Performer⟩ o2 ⟨/Performer⟩
      ⟨Performer⟩ o3 ⟨/Performer⟩
      ⟨Performer⟩ o4 ⟨/Performer⟩
    ⟨/Performers⟩
    …
  ⟨/Activity⟩
  …
⟨/Activities⟩
```

Through the screen-snapshots of Fig. 11, we can recognize that the system successfully discovers two workflow-supported social networks from the exampled XPDL workflow package, and it also analyzes workload-centralities based upon the discovered one. The left-hand side screen-snapshot displays one of the binary symmetric (non-directed) SocioMatrices, whereas the right-hand side screen-snapshot prints out the workload-centrality measurements, such as actor-workload centrality measurements, normalized actor-workload centrality measurements, group-workload centrality measurements, and indexed group-workload centrality measurements, from analyzing the SocioMatrices by applying the extended workload analysis equations.

Additionally, it is worthy to graphically visualize the analyzed workload centrality measurements at a glance. So, the system provides the visualization functionality of the workload centrality measurements, as shown in Fig. 12. The left-hand side of the upper screen-snapshots in the figure graphically displays the analyzed results of the workload[6] centralities of each workflow actor assigned into the two workflow models, "Hiring workflow procedure (Process-1288115472781)" and "Presentation workflow procedure (Process-1288253412484)". At the same time, the right-hand side of the upper screen-snapshots shows the indexed workload centralities normalized between 0 and 1. Finally, the lower screen-snapshot of the figure gives a graphical view of the integrated workload centralities of each workflow actor in enacting the workflow package, "Process List." We can easily recognize through the graphical views that the workflow actor, o_5, ought to be the most influential worker in enacting the workflow package.

5. Related works

In recent, the workflow literature just starts being focused on social and collaborative work analysis on workflow-supported organizations. Particularly, our workflow-supported social network discovery and analysis are directly related to a converged issue of model-log comparison, and social networks generation and analysis. With respect to this converged issue, there have been two main branches of research approaches: social network *discovery* issue and social network *rediscovery* issue. The social network rediscovery issue stems from the workflow mining issue that tries to rediscover workflow processes from workflow execution event logs, while on the other, the social network discovery issue that explores social aspects or human behaviors from workflow models has not been attracting attentions in the literature yet. Aalst et al. (2005) 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 (Ellis et al., 2006; Kim and Ellis, 2006, Kim and Ellis, 2007; Park and Kim, 2008, 2010; Rembert, 2008; Skerlavaj et al., 2010; Wainer et al., 2005), so far. Especially, Rembert (2008) proposed an automatic rediscovery framework covering almost all perspectives of a

[5] BPMN stands for business process modeling notations, and it is released by OMG's BMI (business modeling and integration) Domain Task Force.

[6] In the window's title and the graph's title, we used the label, "Actor Degree Centralities" when the system was implemented at first. The actor degree centrality implies the workload centrality as described in the beginning of the paper.

```
<?xml version="1.0" encoding="UTF-8" ?>
- <Package Id="Package1288115469531" Name="Process List" xmlns="http://www.wfmc.org/2002/XPDL1.0">
  - <PackageHeader>
      <XPDLVersion />
      <Vendor />
      <Created>2010-10-27</Created>
      <Description />
    </PackageHeader>
    <RedefinableHeader PublicationStatus="RELEASED" />
  - <WorkflowProcesses>
    - <WorkflowProcess Id="Process-1288115472781" Name="Hiring Workflow Procedure" AccessLevel="PRIVATE">
      + <ProcessHeader DurationUnit="D">
      + <RedefinableHeader PublicationStatus="RELEASED">
      + <Participants>
      - <Activities>
        + <Activity Id="Activity1288115505250" Name="start1">
        + <Activity Id="Activity1288115510125" Name="ApplyActivity">
        - <Activity Id="Activity1288115512796" Name="NewApplicationInfoAcrivity">
            <Limit />
          + <Implementation>
          - <Performers>
              <Performer>o2</Performer>
              <Performer>o3</Performer>
              <Performer>o4</Performer>
            </Performers>
          + <StartMode>
          + <FinishMode>
          + <ExtendedAttributes>
          </Activity>
        + <Activity Id="Activity1288115515140" Name="DecisionActivity">
        + <Activity Id="Activity1288115517828" Name="or_split5">
        + <Activity Id="Activity1288115520812" Name="RejectingActivity">
        + <Activity Id="Activity1288115522453" Name="DatabaseUpdateActivity">
        + <Activity Id="Activity1288115524953" Name="RequestCompersationActivity">
        + <Activity Id="Activity1288115526593" Name="OfferLetterActivity">
        + <Activity Id="Activity1288115528328" Name="HiringActivity">
        + <Activity Id="Activity1288115555625" Name="and_split11">
        + <Activity Id="Activity1288115566078" Name="EmploymentCheckupActivity">
        + <Activity Id="Activity1288115566421" Name="EmploymentReviewActivity">
        + <Activity Id="Activity1288115566968" Name="SecurityCheckupActivity">
        + <Activity Id="Activity1288115567312" Name="SecurityReviewActivity">
        + <Activity Id="Activity1288115568125" Name="EducationCheckupActivity">
        + <Activity Id="Activity1288115568453" Name="EducationReviewActivity">
        + <Activity Id="Activity1288115569062" Name="MedicalScreenActivity">
        + <Activity Id="Activity1288115569328" Name="MedicalReviewActivity">
        + <Activity Id="Activity1288115591734" Name="and_join21">
        + <Activity Id="Activity1288115600156" Name="ReviewResultsActivity">
        + <Activity Id="Activity1288115600484" Name="DatabaseUpdateActivity">
        + <Activity Id="Activity1288115607515" Name="or_join24">
        + <Activity Id="Activity1288115614046" Name="end25">
        + <Activity Id="Activity1290583702375" Name="loop25">
        + <Activity Id="Activity1290583706812" Name="loop26">
        + <Activity Id="Activity1290583707140" Name="loop27">
        + <Activity Id="Activity1290583707468" Name="loop28">
        + <Activity Id="Activity1290583708375" Name="loop29">
        + <Activity Id="Activity1290583708718" Name="loop30">
        + <Activity Id="Activity1290583709031" Name="loop31">
        + <Activity Id="Activity1290583709375" Name="loop32">
        </Activities>
      + <Transitions>
      + <ExtendedAttributes>
      </WorkflowProcess>
    + <WorkflowProcess Id="Process-1288253412484" Name="PresentationWorkflowProcedure" AccessLevel="PRIVATE">
    </WorkflowProcesses>
</Package>
```


Fig. 10. An operation example: XPDL workflow package.

workflow meta-model including the actors' behaviors; however, it did not directly cope with the social network discovery and analysis issues.

Won (2008) 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: a role-based organizational work-sharing network and a human-based organizational work-sharing network. Also, he suggested a new statistical analysis approach for analyzing organizational work-sharing networks; however, the proposed statistical approach is not directly related to the social network analysis methods developed in this paper. In terms of quantifying the degree of work-

Table 4
Specifications of the XPDL-based pseudo-workflow package.

Package: process list	Workflow activity	Actor
Hiring-workflow-procedure	Apply, NewApplicationInfo, Decision, Rejecting, DatabaseUpdate, RequestCompensation, OfferLetter, Hiring, EmploymentCheckup, EmploymentReview, SecurityCheckup, SecurityReview, EducationCheckup, EducationReview, MedicalScreen, MedicalReview, ReviewResults	o_1–o_{17}
Presentation-workflow-procedure	PreInterview, InterviewAnalysis, InterviewProcess, InterviewSkillPractice, Presentation, OrganizationDiscussion, QnA	o_1–o_{17}

sharing for workflow actors, he used not a social network analysis equation, but a statistical co-relation analysis tool.

Representative pioneering work on the workflow-supported social network discovery issue has been done (Ahn et al., 2012; Jeon et al., 2012; Jeon and Kim, 2013; Kim et al., 2011; Kim, 2011). These works proposed a theoretical framework to discover and analyze workflow-supported social networks, and they implemented the theoretical framework. This paper tries to extend the research work of Song et al. (2010), who firstly proposed a theoretical framework from discovering a workflow-supported social network from a single workflow model to analyze the degree centrality of the discovered network. So, it might be better to say that the conference paper simply introduces the fundamental concept and idea of workflow-supported social networks. Therefore, in this paper, not only we have extensively rebuilt the theoretical framework, but we have also realized the theoretical framework by implementing it as a systematic framework. We have supplemented the workload-centrality analysis equations so as to properly handle the workflow-supported social networks discovered from a workflow package rather than a single workflow model, and we have applied the supplemented equations to the systematic framework in implementing the system. Ultimately, the extended workload-centrality analysis equations can be applied to a group of workflow packages deployed over the entire workflow-driven organization.

In summary, there are two research branches concerning about the workflow-supported social networking knowledge discovery: the discovery issue and the rediscovery issue. The rediscovery issue (Aalst et al., 2005) explores the knowledge (enacted networks) from the workflow enactment logs, whereas the discovery issue of this paper's approach explores the knowledge (planned networks) from the workflow models and packages. As a consequence, the result of exploration from the rediscovery approach is a subset of the discovery approach, because the workflow enactment logs are produced based upon their corresponding workflow models. In terms of analyzing the discovered or rediscovered knowledge, both of the approaches can possibly share their analysis equations and algorithms. However, this paper's discovery approach is able to explore, analyze, and visualize the knowledge much more efficiently, effectively, and plentifully than the rediscovery approach, because the workflow models and packages have much more various and plentiful information than their enactment logs, and the discovery approach is relatively much simpler, easier, and more scalable than the rediscovery approach in terms of the implementation and operation of the corresponding systems. These approaches' results can possibly be used for the model-log

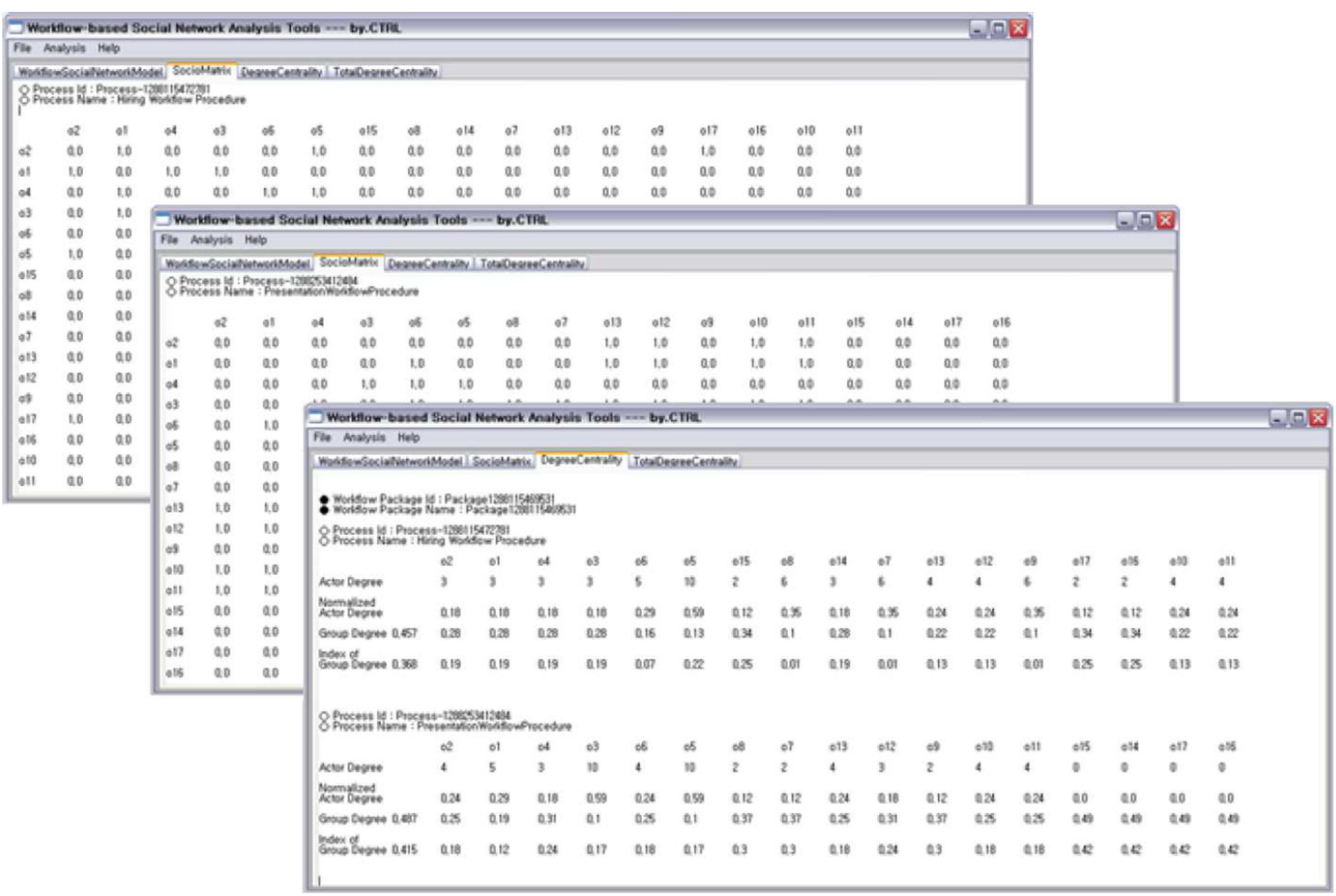

Fig. 11. SocioMatrices of the discovered workflow-supported social networks and their analyzed workload centrality measures.

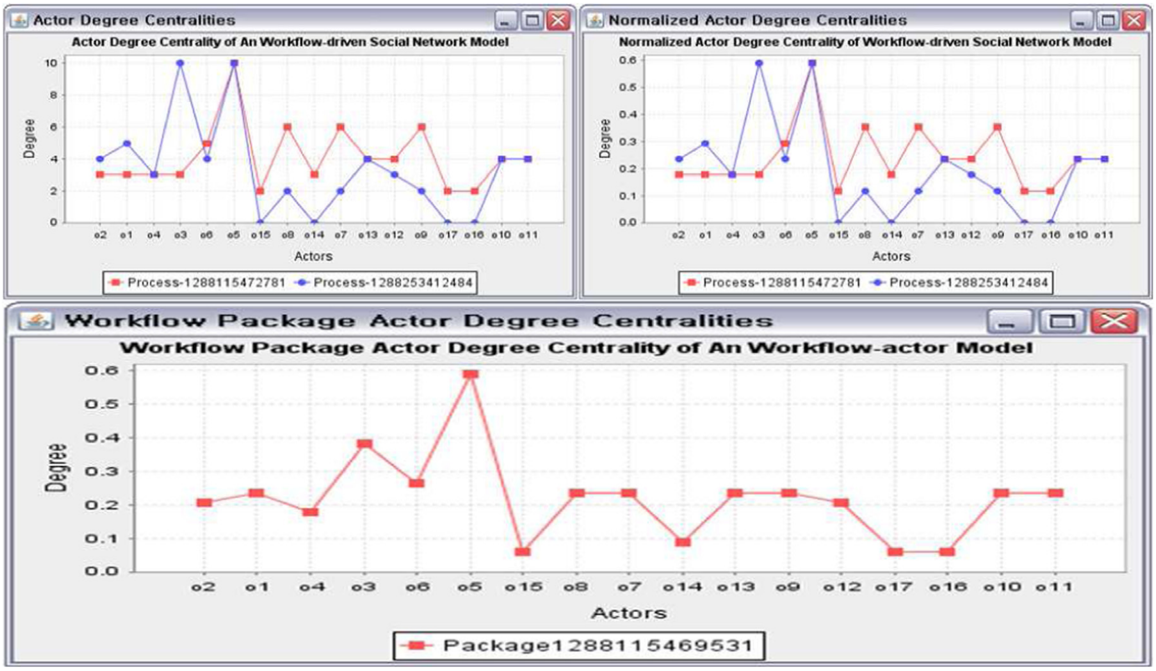

Fig. 12. Graphical views of the analyzed workload centrality measurements.

comparison in terms of every aspect of the knowledge like exploration, analysis, and visualization. The model-log comparison ought to be very important in organizational knowledge and intelligence, because it is directly related to the planning and enacting strategies with performance evaluations in managing organizational resources.

6. Conclusion

The recent trends in working environments require new types of BAI and enterprise information systems, which not only provide collaborative working facilities, but also allow groups of people to work together simultaneously for drastic improvement in decision-making and organizational performance. A typical one of these enterprise information systems that satisfies the requirements is undoubtedly a large-scale workflow management system with increasingly large and complex workflow applications. The large-scale workflow management system ought to reflect the typical organizational perspectives like behavioral, social, informational, collaborative, and historical perspectives, which implies that it is a "people 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 the people who are involved in enacting the specific workflow procedures affect the overall decision-making performance and achieving great successes in real businesses, and working productivity as well.

In this paper, we suggested a possible way of viewing the relationships and collaborative behaviors among workflow-supported people by converging the social network analysis techniques and the workflow discovering and rediscovering techniques. As a consequence of this suggestion, we newly defined workflow-supported social networks, proposed a theoretical framework, and implemented a systematic framework. Through the theoretical framework, we are able to not only discover a workflow-supported social network from an ICN-based workflow procedure, but also analyze the discovered workflow-supported social network by the well-known social network analysis technique, the workload centrality concept, and a measurement technique. For the sake of the frameworks, we devised a series of algorithms from discovering to analyzing, and we successfully verified the proposed frameworks not only through applying the algorithms to the enterprise Hiring workflow procedure already introduced in the previous work, but also through implementing the systematic framework with an operational example of an XPDL-based workflow package comprising two imaginary workflow procedures.

However, the proposed frameworks provide very limited functionality. In other words, they need to cope with the workflow-supported social network rediscovering functionality, and other advanced analysis techniques such as betweenness, closeness, eigenvalue centralities, and prestige measurements issues. We leave those advanced functionalities and techniques for the future works. In the near future, we will try to extend the basic ideas of the human-centered workflow knowledge so as to solve the raised issues on the collaborative management and analysis works of very large scale workflow-supported organizations.

Acknowledgment

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (Grant No. 2012006971).

References

van der Aalst, Wil M.P., Reijers, Hajo A., Song Minseok, 2005. Discovering social networks from event logs. Comput. Support. Cooperative Work 14(6), 549–593.
Ahn, Hyun, et al., 2012. A workflow-supported activity-performer affiliation network model. In: Proceedings of the International Conference on Computers, Networks, Systems, and Industrial Applications, Jeju Island, South Korea, pp. 307–312.
Ellis, Clarence A., Nutt, Gary J., 1980. Office information systems and computer science. ACM Comput. Surv. 12 (1), 27–60.
Ellis, Clarence A., et al., 2006. Beyond Workflow Mining. Lecture Notes in Computer Science, vol. 4102, pp. 49–64.
Ellis, Clarence, Rembert, Aubrey, Wainer, Jacques, Kim, Kwanghoon, 2012.

Investigations on stochastic information control nets. Inf. Sci. 194 (120), 120–137.
Faust, Katherine, 1997. Centrality in affiliation networks. J. Soc. Netw. 19, 157–191.
Ferneley, E., Helms, R., 2010. Editorial of the special issue on social networking. J. Inf. Technol. 25 (2), 107–108.
Griffiths, Mary, 2007. Oxygen: social intranets, collective intelligence and government practices. Electron. J. e-Gov. 5 (2), 177–190.
Harri, Oinas-Kukkonen, et al., 2010. Social networks and information systems: ongoing and future research streams. J. Assoc. Inf. Syst. 11 (2), 61–68.
Jeon, Myounghoon, et al., 2012. A workflow-supported social network model. In: Proceedings of the International Conference on Computers, Networks, Systems, and Industrial Applications, Jeju Island, South Korea, pp. 457–461.
Jeon, Myounghoon, Kim, Kwanghoon Pio, 2013. Workload-centrality analysis and visualization of workflow-supported social networks. ICIC Express Lett. 7 (3(B)), 1049–1054.
Kim, Kwanghoon, Ellis, Clarence A., 2007. σ-Algorithm Structured Workflow Process Mining Through Amalgamating Temporal Workcases. Lecture Notes in Artificial Intelligence, vol. 4426, pp. 119–130.
Kim, Haksung, Ahn, Hyun, Kim, Kwanghoon, 2011. A workflow affiliation network discovery algorithm. ICIC Express Lett. 6 (3), 765–770.
Kim, Kwang-Hoon, Ellis, Clarence A., 2009. Section 2/Chapter VII. An ICN-based workflow model and its advances. In: Handbook of Research on BP Modeling, IGI Global, ISR, pp.142–172.
Kim, Kwanghoon, 1999. Actor-oriented workflow model. In: Proceedings of the 2nd international symposium on Cooperative Database Systems for Advanced Applications, Wollongong, Australia, pp. 150–164.
Kim, Kwanghoon, 2006. A XML-based Workflow Event Logging Mechanism for Workflow Mining. Lecture Notes in Computer Science,vol. 3842, pp. 132–136.
Kim, Kwang-Hoon, Kim, Hak-Sung, 2005. A peer-to-peer workflow model for distributing large-scale workflow data onto grid/P2P. J. Digit. Inf. Manag. 3 (2), 64–70.
Kim, Kwang-Hoon, Ellis, Clarence A., 2006. Workflow reduction for reachable-path rediscovery in workflow mining. In: Foundations and Novel Approaches in Data Mining , vol. 9, Springer, pp. 288–309.
Kim, Kwanghoon, 2011. A workflow-based social network discovery and analysis system. In: Proceedings of the International Symposium on Data-driven Process Discovery and Analysis, Campione d'Italia, Italy, pp. 163–176.
Knoke, David, Yang, Song, 2008. Social Network Analysis. Quantitative Applications in the Social Sciences, 2nd Edition. SAGE Publications.
Koschmider, Agnes, Liu Yingbo, Schuster,Thomas, 2012. Role Assignment in Business Process Models, Lecture Notes in Business Information Processing, vol. 99, 37–49.
Park, Min-Jae, Kim, Kwanghoon, 2008. Control-path oriented workflow intelligence analyses. J. Inf. Sci. Eng. 24 (2), 343–359.
Park, Min-Jae, Kim, Kwanghoon, 2010. A workflow event logging mechanism and its implications on quality of workflows. J. Inf. Sci. Eng. 26 (5), 1817–1830.
Rembert, Aubrey J., 2008. Automatic Discovery of Workflow Models. (Ph.D. Thesis). Department of Computer Science, University of Colorado at Boulder.
Skerlavaj, Miha, Dimovski, Vlado, Desouza, Kevin C., 2010. Patterns and structures of intra-organizational learning networks within a knowledge-intensive organization. J. Inf. Technol. 25 (2), 189–204.
Song, Jihye, Kim, Minjoon, Kim, Haksung, Kim, Kwanghoon, 2010. A framework: workflow-based social network discovery and analysis. In: Proceedings of the International Workshop on Workflow Management in Service and Cloud Computing, Hongkong, China, pp. 421–426.
Tableau, Inc., 2015. TOP 10 Trends in 2016: business intelligence. Whitepaper of Tableau.
Wainer, Jacques, Kim, Kwanghoon, Ellis, ClarenceA., 2005. A Workflow Mining Method Through Model Rewriting, Lecture Notes in Computer Science, vol. 3706, 184–191.
WfMC, 2005. Process Definition Interface – XML Process Definition Language. Workflow Management Coalition Workflow Standard, Document Number WFMC-TC-1025.
Won, Jaekang, 2008. A Framework: Organizational Network Discovery on Workflows (Ph.D. Dissertation). Department of Computer Science, Kyonggi University.