

# 워크플로우 소셜네트워크 인텔리전스 발견 알고리즘<sup>☆</sup>

## A Workflow-based Social Network Intelligence Discovery Algorithm

김 광 훈\*

Kwanghoon Kim

### 요 약

본 논문에서는 워크플로우 소셜네트워크 인텔리전스의 발견 알고리즘을 제안한다. 즉, 워크플로우 인텔리전스 (또는 비즈니스 프로세스 인텔리전스) 기술은 워크플로우 모델들과 그의 실행이력으로부터 일련의 인텔리전스를 발견, 분석, 모니터링 및 제어, 그리고 예측하는 세부기법들로 구성되는데, 본 논문에서는 워크플로우 모델과 그의 실행을 기반으로 형성되는 수행자들간의 협업 인텔리전스를 "워크플로우 소셜네트워크 인텔리전스"라고 정의하고, 그의 발견기법인 정보제어넷(ICN, Information Control Net)기반 워크플로우 소셜네트워크 인텔리전스 발견 알고리즘을 제안한다. 특히, 제안한 알고리즘의 적용 사례를 통해 특정 워크플로우 모델로부터 해당 워크플로우 소셜네트워크 인텔리전스를 성공적으로 생성할 수 있음을 증명하므로써 본 논문에서 제안한 알고리즘의 정확성 및 적합성을 검증한다.

### ABSTRACT

This paper theoretically derives an algorithm to discover a new type of social networks from workflow models, which is termed workflow-based social network intelligence. In general, workflow intelligence (or business process intelligence) technology consists of four types of techniques that discover, analyze, monitor and control, and predict from workflow models and their execution histories. So, this paper proposes an algorithm, which is termed ICN-based workflow-based social network intelligence discovery algorithm, to be classified into the type of discovery techniques, which are able to discover workflow-based social network intelligence that are formed among workflow performers through a series of workflow models and their executions. In order particularly to prove the correctness and feasibility of the proposed algorithm, this paper tries to apply the algorithm to a specific workflow model and to show that it is able to generate its corresponding workflow-based social network intelligence.

□ keyword : 워크플로우, 비즈니스 프로세스, 워크플로우 인텔리전스, 소셜네트워크 발견  
Workflow, Business Process, Workflow Intelligence, Social Network, Knowledge Discovery

## 1. 서 론

최근에 더욱 더 많은 기업과 조직들이 프로세스 기반의 정보기술의 중요성과 필요성을 인식하기 시작하면서 워크플로우 및 비피엔 시스템(이하 워크플로우 시스템)의 각 산업부문별 적용사례가 급속하게 증가하고 있다. 이러한 워크플로우 기술의 적용사례 증가와 함께 프로세스 기반 업무처리량 및 그에 따른 이력정보들이 늘어나면서 멀지 않은 미래에 기존의 워크플로우 및 업무 프로세스들에 대한 개선과 분석[6], 모니터링 및 제어[9], 예측을 가능하게 할 뿐만 아니라 새로운 유형의 업무 프로세스 관련 인텔리전스를 발견[16,18]하거나 마이닝(재발

견)[2]할 수 있는 워크플로우 인텔리전스 기술[8]에 대한 요구가 급속히 증가할 것으로 예상되고 있다. 특히, 연구개발 측면에서의 도전분야는 워크플로우 인텔리전스의 발견 및 재발견 기술이다. 워크플로우 모델과 일련의 워크플로우 모델들의 그룹으로 정의되는 워크플로우 패키지로부터 워크플로우 인텔리전스를 발견해내는 기법을 워크플로우 인텔리전스 발견기법[16]이라고 하며, 워크플로우 모델들의 실행이력인 로그정보로부터 워크플로우 인텔리전스를 발견해내는 기법을 워크플로우 인텔리전스 재발견기법[10]으로 정의되는데, 최근에 이러한 워크플로우 인텔리전스 발견 및 재발견 기법들에 대한 연구가 워크플로우 기술분야의 주요 연구이슈로 등장하고 있다. 결과적으로, 대량의 워크플로우 및 비즈니스 프로세스 모델과 그들의 실행 이력을 기반으로 한 워크플로우 인텔리전스의 발견 및 재발견 알고리즘들과 이를 위한 프레임워크의 확보 유무는 워크플로우 관리 시스템과 비즈니스 프로세스 관리 시스템의 가치를 평가하는데 중

\* 종신회원 : 경기대학교 일반대학원 컴퓨터학과  
kwang@kgu.ac.kr

[2011/11/18 투고 - 2011/11/26 심사 - 2012/03/07 심사완료]

☆ 본 연구는 중소기업청에서 지원하는 2011년도 산학연공동기술개발사업(No. 2011-0279)의 연구수행으로 인한 결과물임.

요한 요소로 인식되고 있다.

본 논문에서는 워크플로우 인텔리전스 발견기법의 하나로 프로세스기반 엔터프라이즈에서 워크플로우 모델들의 실행에 기반하여 형성될 수 있는 업무수행자들간의 소셜네트워크 (또는 협업 네트워크) 인텔리전스를 발견하기 위한 워크플로우 소셜네트워크 발견 알고리즘을 제안한다. 본 저자의 연구그룹에서는 정보제어넷기반 워크플로우 인텔리전스 발견 및 재발견 프레임워크와 그와 연관된 일련의 알고리즘들과 인텔리전스 시스템 구현에 대한 연구를 수행해오고 있는데, 본 논문의 연구범위는 그 중 정보제어넷기반 워크플로우 인텔리전스 발견 프레임워크에 해당한다. 또한, 이 발견 프레임워크는 발견단계와 분석단계로 구성되는데, 발견단계에서는 정보제어넷기반 워크플로우 모델[12]로부터 소셜네트워크 인텔리전스 모델을 발견하는 알고리즘이 핵심이며, 분석단계에서는 발견단계에서 발견된 소셜네트워크 인텔리전스를 분석하기 위해서 요구되는 소셜행렬(SocioMatrix) 생성 알고리즘과 업무수행자들간의 소셜구심도(Degree Centrality) [11], 업무지시도(Prestige)[11] 등을 분석하기 위한 일련의 소셜네트워크 분석알고리즘들이 핵심이다. 따라서, 본 논문의 핵심범위는 워크플로우 소셜네트워크 인텔리전스 발견 프레임워크의 발견단계를 실현하는 발견알고리즘을 제안하는 것이다.

결과적으로, 본 논문에서는 정보제어넷기반 워크플로우 모델로부터 소셜네트워크 인텔리전스를 발견하는 알고리즘을 제안하고, 적용사례를 통해 제안된 알고리즘의 구현가능성을 제시하고자 한다. 특히, 본 논문에서 제안하는 발견알고리즘을 실존 엔터프라이즈의 대표적인 워크플로우 모델인 주문처리 프로세스 모델을 모방한 가상의 정보제어넷기반 워크플로우 모델을 정의하고, 이를 제안된 발견알고리즘에 적용함으로써 궁극적으로 워크플로우 소셜네트워크 발견 프레임워크의 적합성 및 정확성을 검증하고자 한다.

본 논문의 구성으로서 다음 장에서는 워크플로우 메타모델과 이를 구성하는 객체타입들 중에서 소셜 인텔리전스와 연관된 구성요소들로 정의될 수 있는 워크플로우 소셜 인텔리전스 개념을 정의하고, 그 다음인 제 3 장에서는 본 논문의 핵심인 워크플로우 소셜네트워크 발견 알고리즘을 제시하고, 그의 적용사례를 기술한다. 그리고 마지막으로 제 4 장에서 본 논문의 핵심주제에 대한 관련연구 조사내용과 본 논문의 범위와 핵심 기여 내용을 기술한다.

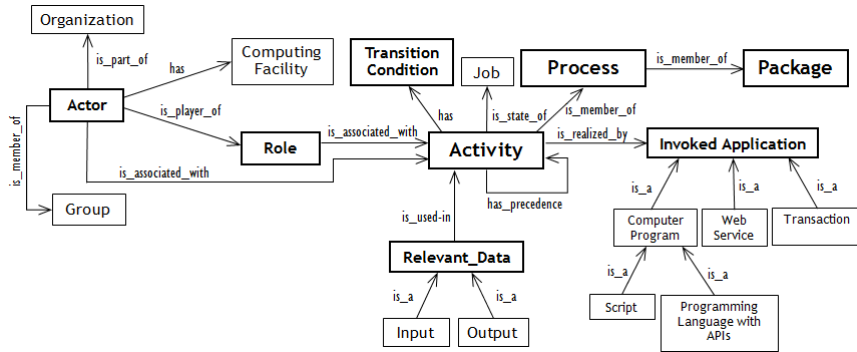
## 2. 워크플로우 소셜 인텔리전스

워크플로우 소셜 인텔리전스[9,10,16]의 기본적인 개념은 워크플로우 모델 및 시스템이 "사람중심의 소프트웨어 시스템"이라는 개념적 정의로부터 시작된다. 즉, 워크플로우 모델은 업무수행자들간의 프로세스기반 협업관계 모델이라고 정의할 수 있으며, 본 논문에서는 특정 워크플로우 모델과 그의 실행을 수행함에 따라 형성될 수 있는 업무수행자들간의 상호협력관계 네트워크를 워크플로우 소셜 네트워크라고 정의한다. 결과적으로, 이러한 워크플로우 소셜 네트워크는 워크플로우 메타모델을 기반으로 정의된 워크플로우 모델의 속성정보나 모델의 실행이력으로부터 발견 또는 재발견될 수 있으며, 이로부터 업무수행자들간의 상호업무협력상황, 업무집중도 및 기여도, 상호업무연관도, 업무소속성 등으로 대표되는 워크플로우 소셜 인텔리전스를 분석해 낼 수가 있다. 따라서, 본 장에서는 워크플로우 소셜 인텔리전스의 이론적 배경이 되는 워크플로우 메타모델과 워크플로우 소셜 네트워크의 기본개념을 정의한다.

### 2.1 워크플로우 메타모델

워크플로우 메타-모델[12]은 워크플로우 모델을 기술하는 모델로서 워크플로우 모델을 정의하는데 필수적으로 요구되는 객체유형들의 집합과 그들간의 관계를 정의한 모델이다. (그림 1)은 워크플로우 메타-모델을 그래픽 표기법으로 표현한 것이다. 워크플로우 메타-모델을 구성하는 객체유형들은 워크플로우 프로세스, 액티버티, 역할, 수행자, 연관데이터, 천이조건, 호출프로그램 등이 있으며, 특히 액티버티 수행으로 형성되는 수행자들간의 업무협업네트워크를 본 논문에서는 워크플로우 소셜네트워크라고 정의한다.

- 프로세스(Process) : 워크플로우 또는 비즈니스 프로세스의 또 다른 명칭 또는 단축어이다. 프로세스는 액티버티라고 정의되는 단위업무들의 집합과 이들간의 실행순서로 정의된다. 특히, 프로세스들의 집합을 패키지라고 정의한다. 최종적으로, 정의된 프로세스는 단위업무들의 실행순서로 정의되는 제어흐름과 각 단위업무의 입력력 데이터들로 정의되는 데이터흐름을 내포하고 있다. 내포된 제어흐름은 프로세스를 구성하는 단위업무(액티버티)들에 대한 4가지의 기본적인 제어흐름 유형, 즉 순차적(Sequential),



(그림 1) 워크플로우 메타-모델

선택적(Disjunctive), 병렬적(Conjunctive), 반복적(Repetitive, Looping) 제어흐름 유형들의 조합으로 표현된다.

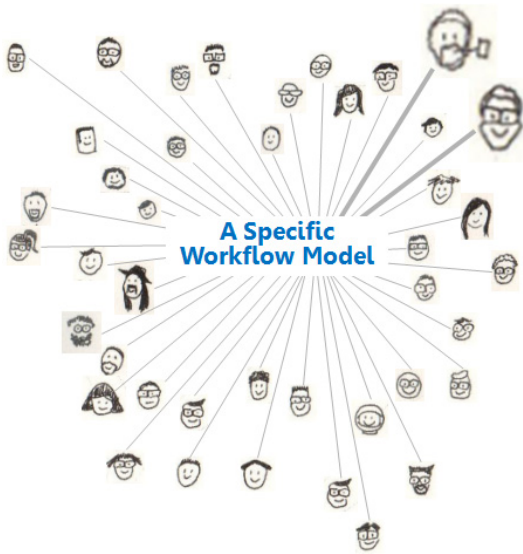
- **액티비티(Activity)** : 프로세스를 구성하는 기본 단위 업무에 대한 개념적 용어이다. 액티비티의 종류는 작업액티비티(Work Activity), 블록액티비티(Block Activity), 서브프로세스액티비티(Subprocess Activity), 게이트웨이액티비티(Gateway Activity), 이벤트액티비티(Event Activity) 등으로 구분된다. 이들 액티비티들간에는 실행시간의 선후관계(have-precedence)가 존재하며, 이러한 선후관계는 앞서 소개한 기본적인 제어흐름 유형을 통해 정의되며, 이는 곧 프로세스의 정의를 의미한다.
- **역할(Role)과 수행자(Actor 또는 Performer)** : 역할과 수행자 객체유형은 워크플로우 모델을 위한 조직정보를 정의하는 핵심요소로서, 본 논문의 연구주제인 워크플로우 소셜 인텔리전스 형성의 대상이 되는 객체유형이다. 프로세스를 구성하는 작업액티비티들에는 그의 실행을 담당하는 수행자를 할당해야 한다. 그러나 각각의 일반액티비티에 수행자를 직접 할당할 수도 있지만, 조직내에서 가장 변화가 많은 구성요소가 바로 인적요소이기 때문에 이로 인해 워크플로우 모델을 너무 자주 변경해야한다는 단점을 갖는다. 따라서 워크플로우 모델의 변경을 최소화하고 다양한 유형의 수행자 결정조건을 효율적으로 활용하고, 수행자 결정의 유연성들을 제공하기 위해서 일반적으로 역할 객체유형을 활용한다. 특히, 역할 객체유형은 조직의 물리적 구조와 논리적 구조를 반영할 수가 있는데, 조직의 물리적 구조를 역할로 정의할 경우는 조직의 부서 구조를 적용

할 수 있으며, 조직의 논리적 구조를 역할로 정의할 경우는 조직내의 기능적 역할을 역할객체로 적용할 수 있다. 물론 워크플로우 모델링 도구의 구현방법에 따라 두 가지를 동시에 역할객체로 활용할 수도 있다. 또한, 하나의 역할객체에 여러 명의 수행자를 할당할 수 있으며, 역시 한 명의 수행자객체는 여러 역할객체에 동시에 할당될 수 있다. 결과적으로, 어느 특정 프로세스 인스턴스의 일반액티비티를 실행할 때는 이를 담당할 한명의 수행자를 해당 역할객체에 배정된 수행자들 중의 한 명을 결정하게 되는데, 이를 수행자의 실시간결정 또는 수행자의 런타임바인딩이라고 한다.

결과적으로, 워크플로우 메타-모델은 워크플로우 모델 뿐만 아니라 워크플로우 관리 시스템의 핵심 구성요소를 정의하는 기본모델이며, 특히 워크플로우 모델상의 역할 배정과 업무수행자 할당은 워크플로우 소셜 네트워크 개념의 등장과 필요성에 대한 이론적 근거가 된다.

## 2.2 워크플로우 소셜 네트워크

앞서 정의한 워크플로우 메타-모델을 기반으로 정의된 워크플로우 모델의 실행시점에는 그를 구성하는 액티비티들과 그의 실제적인 실행을 위해 할당된 역할과 그에 속한 업무수행자들 중의 한 명을 선정함으로써 해당 액티비티와 연관되는 업무(프로그램 실행)를 수행하게 된다. 그렇다면, 워크플로우기반 엔터프라이즈의 관리자 또는 경영자 관점에서 생각할 수 있는 가장 의미심장한 근본적인 궁금증은 무엇일까? 즉, 어느 특정 워크플로우 모델 또는 엔터프라이즈 전체의 워크플로우 모델들과 그의 실행이력로그들로부터 발견할 수 있는 워크플로우 인



(그림 2) 워크플로우 모델의 업무수행자 할당과 잠재적 소셜 네트워크 인텔리전스

텔리전스와 지식은 무엇일까? (그림 2)는 특정 워크플로우 모델의 업무수행을 담당할 업무수행자(인적자원)들의 할당상황을 도식화한 것이며, 이를 통해 본 논문에서 고려하고 있는 핵심개념인 워크플로우 소셜 네트워크 모델의 가치와 필요성을 예감할 수 있을 것으로 기대해 본다. 다음의 질문에 대한 답이 워크플로우 모델과 그의 실행이력으로부터 발견/재발견 할 수 있는 워크플로우 소셜 인텔리전스[16]라고 할 수 있다.

- 해당 워크플로우 모델을 수행하는데 있어서 가장 중요한 역할 또는 영향력이 높은 사람은 누구인가?
- 이 모델에 대한 각 수행자들의 업무적 영향력을 수치화 할 수 있는가?
- 이 모델에 연관된 수행자들간의 상호 협력 관계는 어느 정도이고 그들을 수치화 할 수 있는가?
- 이 모델의 정의시점에 할당된 인적자원으로부터 발견된 소셜네트워크 지식과 실행시점의 실행이력 또는 로그로부터 재발견된 소셜네트워크 지식간의 비교는 가능한가? 즉, 계획된 인적자원 중심의 소셜네트워크와 실제로 실행된 인적자원 중심의 소셜네트워크의 비교를 통한 워크플로우 모델의 충실도(Fidelity) 정도를 수치화 할 수 있는가?

이 모델의 실행이력으로부터 각 액티비티에 대한 인적자원 할당과 그의 수행실적 정도를 수치화할 수 있는가?

이러한 개념을 엔터프라이즈 전체수준으로 확장한다면, 더욱 복잡하고 난해한 그리고 자동화하지 않으면 효율적으로 답할 수 없는 매우 고품질의 지식 또는 인텔리전스로 발전할 수 있음에 틀림없으며, 이에 대한 효율적/효과적 발견 기법들과 그들을 통해 획득된 워크플로우 기반 소셜네트워크 및 소속성네트워크의 분석기법들에 대한 연구가 요구되고 있다. 따라서, 본 논문의 핵심 내용은 수행자 중심으로 형성되는 이러한 고품질의 워크플로우 기반 소셜네트워크 인텔리전스를 발견하는 알고리즘을 고안하는데 있다.

### 3. 워크플로우 소셜네트워크 발견 알고리즘

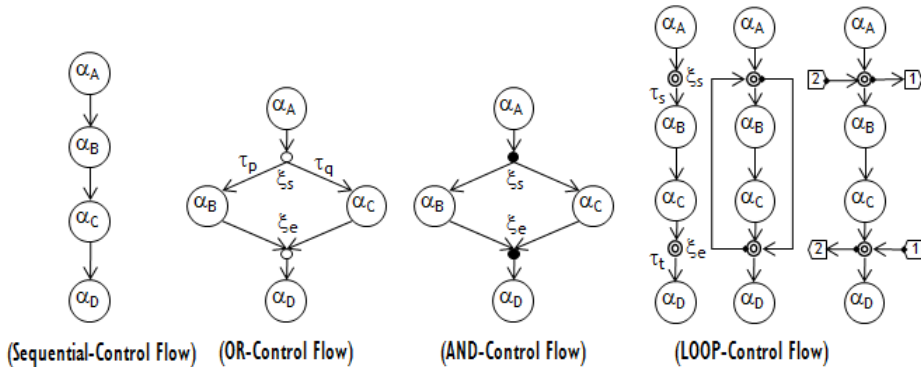
본 논문에서는 워크플로우 메타모델을 기반으로 하는 워크플로우 모델링 방법론으로 정보제어넷을 이용하고자 한다. 여기서는, 정보제어넷 기반의 워크플로우 모델로부터 업무수행자 중심의 소셜 네트워크 모델을 발견하는 알고리즘을 고안하고, 그의 적용사례로서 어느 한 엔터프라이즈의 고용 워크플로우 모델로부터 워크플로우 소셜네트워크 모델을 발견한 결과에 대해서 기술한다.

#### 3.1 정보제어넷기반 워크플로우 모델

정보제어넷[12]에서는 워크플로우 모델을 정형적 표기와 그래픽적 표기로 정의하고 있다. 이 절에서는 정보제어넷기반의 소셜 네트워크를 발견하는데 필수적인 구성요소인 액티비티들의 프로세스(제어흐름)과, 각 액티비티의 역할할당, 그리고 각 역할의 업무수행자배정을 표현하기 위한 정형적 표기법과 그에 해당하는 그래픽적 표기법을 정의한다.

##### 3.1.1 워크플로우 프로세스(제어흐름) 정의

워크플로우 모델의 핵심은 프로세스 즉 액티비티들간의 제어흐름을 정의하는 것이다. 워크플로우 프로세스의 기본구조[12]는 시작이벤트액티비티와 종료이벤트액티비티 그리고 이 두 이벤트액티비티들 사이에 시간적 선후행관계를 이루는 액티비티(단위업무)들의 집합과 각 액티비티가 만족해야 할 전이조건들의 집합으로 정의된다. (그림 3)은 워크플로우 프로세스를 이루는 기본적인



(그림 3) 워크플로우 프로세스(제어흐름) 기본 유형

제어흐름의 유형들과 각 액티비티의 선행행천이조건들의 집합을 그래픽적 표기법으로 나타낸 것이다. 여기서, 워크플로우 프로세스의 그래픽적 표기법을 위한 기본적인 규칙은 다음과 같다.

- 액티비티들간의 실행시간적 선행행관계는 반드시 하나의 화살표로서 표현되어야 한다.
- 각 액티비티의 선행행천이조건들은 해당 화살표상에 위치되어야 한다. 이 때, 화살표상에 위치하는 선행행천이조건들의 예초값(Default)들은 표기를 생략할 수 있다.
- 작업액티비티들간에는 하나 이상의 선행행관계(화살표)를 정의할 수 없다.
- 게이트웨이액티비티인 OR-split/AND-split 액티비티는 하나의 선행액티비티와 두 개 이상의 후행액티비티를 갖으며, OR-join/AND-join 액티비티는 두 개 이상의 선행액티비티와 하나의 후행액티비티를 갖는다.
- 게이트웨이액티비티인 LOOP-split/LOOP-join은 각각 두 개씩의 선행행액티비티들을 갖는다.
- 구조적 워크플로우 프로세스, 즉 게이트웨이액티비티는 반드시 split/join 유형의 쌍을 이루어야 할 뿐만 아니라 다중의 게이트웨이액티비티들은 네스팅 구조를 유지해야 한다.

워크플로우 프로세스의 기본구조[12]에 대한 정형적 정의는 [정의 1]에 나타내었다. 워크플로우 프로세스는 액티비티 집합과 천이조건 집합을 대상으로 하는 두 개의 함수, 즉 해당 프로세스를 구성하는 액티비티들의 선

후행액티비티 집합들을 정의하는 함수와 각 액티비티들의 선행행천이조건 집합들을 정의하는 함수로 정의된다.

[정의 1] 워크플로우 프로세스,  $\Gamma_C = [ \text{function}(\delta, \kappa), \text{set}(A, T) ]$

(집합) 워크플로우 프로세스의 구성객체집합

- 액티비티들의 집합  $A = \{ \alpha_1, \dots, \alpha_k, \xi_1, \dots, \xi_p, \varsigma_1, \dots, \varsigma_q, \varphi_1, \dots, \varphi_s \};$ 
  - $\alpha_1, \dots, \alpha_k$  : 작업액티비티
  - $\xi_1, \dots, \xi_p$  : 게이트웨이액티비티 (명시적 또는 암시적 정의)
  - $\varsigma_1, \dots, \varsigma_q$  : 서브프로세스액티비티
  - $\varphi_1, \dots, \varphi_s$  : 이벤트액티비티 (시작액티비티와 종료액티비티)
- 천이조건들의 집합  $T = \{ \tau_1, \dots, \tau_m \};$

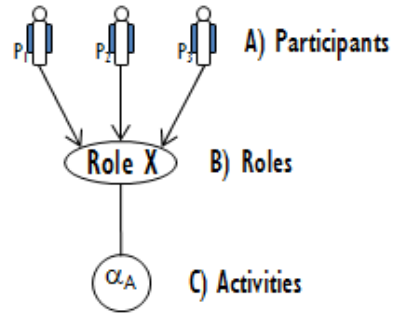
(함수) 선행행액티비티 함수,  $f: \delta = \delta_i \cup \delta_o$ ; 선행행천이조건 함수,  $f: \kappa = \kappa_i \cup \kappa_o$ .

- $f: \delta_i A \rightarrow \mathcal{G}(A)$ 로 정의되는 다중값 매핑함수\*로서 어느 특정 액티비티,  $\alpha$ 와 그의 선행액티비티 집

\*  $A \rightarrow \mathcal{G}(A)$ , 다중값 매핑함수(Multivalued Mapping Function)는 집합 A의 모든 원소들은 그의 멱집합(Powerset)  $\mathcal{G}(A)$ 의 하나 이상의 원소들과 매핑된다는 의미로서, 프로그램의 함수개념을 적용한다면 집합 A의 각 원소를 입력 했을 때 그의 멱집합  $\mathcal{G}(A)$ 의 하나 이상의 원소들을 출력하는 함수를 정의한 것이다. 다중값 매핑함수는 수학적 함수의 개념으로는 잘못 명기된 이름이라고 할 수 있는데, 그 이유는 수학적 함수의 개념은 X영역의 값에는 Y영역의 오직 하나의 값과 매핑될 수 있음을 의미하기 때문이다. 참고적으로, 다중값 매핑함수는 multi-function, set-valued function, set-valued map, multi-valued map 등으로 불리우기도 한다.

합과의 매칭을 정의한 함수이다.

- $f: \delta_o A \rightarrow \mathcal{C}(A)$ 로 정의되는 다중값 매칭함수로서 어느 특정 액티버티,  $\alpha$ 와 그의 후행액티버티 집합과의 매칭을 정의한 함수이다.
- $f: \kappa_i A \rightarrow \mathcal{C}(T)$ 로 정의되는 다중값 매칭함수로서 어느 특정 액티버티,  $\alpha$ 와 그의 선행천이조건 집합과의 매칭을 정의한 함수이다.
- $f: \kappa_o A \rightarrow \mathcal{C}(T)$ 로 정의되는 다중값 매칭함수로서 어느 특정 액티버티,  $\alpha$ 와 그의 후행천이조건 집합과의 매칭을 정의한 함수이다. □



(그림 4) 워크플로우 역할/수행자 정의

### 3.1.2 워크플로우 역할 및 수행자 정의

정보제어넷 워크플로우 모델[12]에 있어서 액티버티들간의 제어흐름과 데이터흐름이 프로세스 측면의 핵심 구성요소라고 한다면, 실제적인 워크플로우 액티버티들의 실행을 담당하는 조직측면의 핵심 구성요소가 바로 역할과 수행자 객체를 정의하는 일이다. 워크플로우 모델에 있어서 역할 객체와 수행자 객체들이 갖는 기본적인 의미는 앞서 설명을 하였듯이, 조직을 이루는 기본요소로서 역할 객체는 논리적 조직개념의 역할과 물리적 조직개념의 부서를 모두 포함한다고 가정하며, 수행자 객체는 하나 이상의 역할들에 소속될 수 있다. 그리고 액티버티 객체의 유형들(작업, 게이트웨이, 서브프로세스, 이벤트) 중에서 오직 작업액티버티들과의 역할/수행자 객체의 연결관계만을 정의할 수 있으며, 나머지 유형의 액티버티들의 실행은 워크플로우 실행엔진에 의해 수행된다고 가정한다.

이들 역할 및 수행자 정의를 위한 그래픽적 표기법과 정형적 표기법은 각각 (그림 4)와 [정의 2]와 같이 정의한다. (그림 4)의 그래픽적 표기법에서 나타난 바와 같이 역할 객체는 타원형으로 수행자(Participant 또는 Actor) 객체는 사람모양으로 표현되며, 역할 객체와 액티버티 객체와의 관계는 방향이 없는 선으로, 수행자 객체들과 역할 객체들의 소속관계는 직선의 화살표로 표현한다. 특히, 정보제어넷 워크플로우 역할 및 수행자 정의의 그래픽적 표기법을 위한 기본적인 규칙은 다음과 같다.

- 역할 객체는 작업액티버티 객체들에게만 할당되고, 수행자 객체는 역할 객체에게만 배정된다.
- 작업액티버티들과 역할들간의 연결관계는 직선으로 정의한다.
- 역할과 수행자들간의 연결관계는 직선의 화살표로

정의한다.

- 액티버티들과 수행자들간의 직접적인 연결은 정의될 수 없다.
- 각 액티버티는 반드시 하나의 역할과의 연결관계를 정의될 수 있다.
- 하나의 역할은 하나 이상의 액티버티들과의 연결관계를 정의할 수 있다.
- 하나의 역할은 하나 이상의 수행자들과 연결관계를 정의할 수 있다.
- 하나의 수행자는 하나 이상의 역할들과 연결관계를 정의할 수 있지만, 그래픽적 연결관계는 표시하지 않고 해당 수행자 객체를 중복해서 정의한다.

[정의 2] 워크플로우 역할 및 수행자 정의,  $\Gamma R = [ \text{function}(\epsilon, \pi), \text{set}(A, R, P) ]$

(집합) 워크플로우 모델의 구성객체집합

- 액티버티들의 집합,  $A = \{ \alpha_1, \dots, \alpha_n \}$ ;  
-  $\alpha_1, \dots, \alpha_n$  : 작업액티버티
- 역할들의 집합,  $R = \{ \nu_1, \dots, \nu_m \}$ ;
- 수행자들의 집합,  $P = \{ \phi, \dots, \phi_k \}$ ;

•  $\mathcal{C}(\ )$  : Power Set

(함수) 역할할당 함수,  $f: \epsilon = \epsilon_a \cup \epsilon_p$ ; 수행자배정 함수,  $f: \pi = \pi_p \cup \pi_c$ ;

- $f: \epsilon_p A \rightarrow R$ 로 정의되는 단일값 매칭함수로서 어느 특정 액티버티,  $\alpha$ 와 그의 실행을 책임맡은 하나의 역할과의 매칭을 정의한 함수이다.
- $f: \epsilon_a R \rightarrow \mathcal{C}(A)$ 로 정의되는 단일값 매칭함수로서 어느 특정 역할,  $\nu$ 와 그 역할이 실행책임을 맡고 있는 액티버티 집합과의 매칭을 정의한 함수이다.
- $f: \pi_c R \rightarrow \mathcal{C}(P)$ 로 정의되는 단일값 매칭함수로서 어느 특정 수행자,  $\phi$ 와 그 수행자 소속한 역할 집합

과의 매칭을 정의한 함수이다.

- $f: \pi_p P \rightarrow \mathcal{C}(R)$ 로 정의되는 단일값 매칭함수로서 어느 특정 역할,  $v$ 와 그 역할에 할당된 수행자 집합과의 매칭을 정의한 함수이다. □

**(1) 역할할당 함수:** 워크플로우 모델의 정형적 정의에서 역할할당 함수의 의미는 해당 액티비티와 그의 실행을 책임지는 역할과의 매핑관계를 집합의 개념으로 정의한 것이다. 즉, 이 함수는 어느 특정 액티비티( $\alpha$ )의 실행을 책임지는 조직의 논리적 구성요소(역할) 또는 물리적 구성요소(부서)들의 집합을 리턴 값으로 제공하는 함수와 어느 특정 역할 또는 부서( $v$ )에 할당된 액티비티들의 집합을 리턴 값으로 제공하는 함수로 구성되며, (그림 4)의 부분적 워크플로우 모델의 액티비티에 대한 함수의 적용결과는 다음과 같다.

- ① 역할할당 함수의 액티비티에 할당된 역할과의 매핑 수행결과

$$f: \varepsilon_p(\alpha_A) = \{ v_x \};$$

- ② 역할할당 함수의 역할에 할당된 액티비티와의 매핑 수행결과

$$f: \varepsilon_a(v_x) = \{ \alpha_A \};$$

**(2) 수행자배정 함수:** 워크플로우 모델의 정형적 정의에서 수행자배정 함수의 의미는 해당 역할과 그에 배정된 수행자들과의 매핑관계를 집합의 개념으로 정의한 것이다. 즉, 이 함수는 어느 특정 역할( $v$ )에 배정된 수행자들의 집합을 리턴 값으로 제공하는 함수와 어느 특정 수행자( $o$ )가 속한 역할 또는 부서의 집합을 리턴 값으로 제공하는 함수로 구성되며, (그림 4)의 부분적 워크플로우 모델에 대한 함수의 적용결과는 다음과 같다.

- ① 수행자배정 함수의 역할에 배정된 수행자들과의 매핑 수행결과

$$f: \pi_c(v_x) = \{ \phi_1, \phi_2, \phi_3 \};$$

- ② 수행자배정 함수의 수행자에 배정된 역할들과의 매핑 수행결과

$$f: \pi_p(\phi_1) = \{ v_x \}; \quad f: \pi_p(\phi_2) = \{ v_x \};$$

$$f: \pi_p(\phi_3) = \{ v_x \};$$

### 3.2 워크플로우 소셜네트워크 모델

앞서 정의했듯이, 정보제어넷기반의 워크플로우 모델 [12]은 일련의 액티비티들과 그들간의 시간적 우선순위를 통해 정의되며, 각 액티비티의 실행은 할당된 역할

(Role)에 속한 업무수행자들에 의해 수행된다. 따라서, 워크플로우 모델상에 정의된 역할할당 및 수행자배정 정보를 바탕으로 업무수행자들간에 형성된 상호관계를 발견한 결과를 정보제어넷기반 워크플로우 소셜 네트워크 모델이라고 정의한다. 본 절에서는 정보제어넷기반의 워크플로우 소셜 네트워크 모델의 정형적 표기법 및 그래픽적 표기법을 기술한다. 궁극적으로, 본 논문에서 제안한 발견 알고리즘의 수행결과로 발견된 워크플로우 소셜네트워크 모델은 기존의 소셜네트워크 분석기법들[11], 중심도(Centrality)[16,17], 명성도(Prestige), 과당성(Clique), 구조동질성(Structural Equivalence) 등을 적용함으로써 앞서 제시한 워크플로우 인텔리전스 관련 질문들에 대한 해답을 구할 수가 있다.

[정의 3]과 (그림 5)는 각각 워크플로우 소셜네트워크 모델의 정형적 표기법과 그래픽적 표기법을 나타낸 것이다. 워크플로우 소셜네트워크 모델의 정형적 표기법은 워크플로우 모델의 구성객체인 액티비티 ( $A = \{ \alpha_1, \dots, \alpha_n \}$ )와 수행자 ( $P = \{ \phi_1, \dots, \phi_k \}$ )를 구성요소로 하는 수행자들간의 소셜관계 함수 ( $\sigma = \sigma_i \cup \sigma_o$ )와 각 수행자의 취득액티비티 함수 ( $\zeta = \zeta_i \cup \zeta_o$ )를 통해 정의될 수 있다. 특히, 워크플로우 소셜네트워크 모델의 정형적 표현은 집합이론(Set Theory)의 특수한 확장형인 가방이론(Bag Theory)\*[12]를 기반으로 한다. 여기서, 가방이론이란 어느 한 집합에서 원소(Member)의 중복성을 허용한 이론으로서 페트리넷이론(Petri Net)[18] 등에서 사용하는 집합이론의 확장형이다.

[정의 3] 워크플로우 소셜 네트워크 모델의 정의,  $\Lambda =$

[ function( $\sigma, \zeta$ ), set(A, P) ]

(집합) 워크플로우 모델의 구성객체집합

- 액티비티들의 집합,  $A = \{ \alpha_1, \dots, \alpha_n \};$

-  $\alpha_1, \dots, \alpha_n$  : 작업액티비티

- 수행자들의 집합,  $P = \{ \phi_1, \dots, \phi_k \};$

- 수행자들간의 소셜링크 집합,  $P \times P = \{ (\phi_i, \phi_j), 1 \leq i, j \leq k \};$

- $\mathcal{C}(\cdot)$ : Power Set

(함수) 소셜관계 함수,  $f: \sigma = \sigma_i \cup \sigma_o$ ; 연관액티비티

함수,  $f: \zeta = \zeta_a \cup \zeta_o$ ;

\* 가방이론(Bag Theory)의 기본적인 배경은 가방(Bag)속에는 여러 개의 동일한 물건을 넣을 수 있다는 생각에서 정의된 집합이론의 확장형이다. 이 논문의 워크플로우 소셜네트워크 모델에서는 원소의 중복성 허용과 함께 원소들의 배열순서도 의미를 갖는다고 가정한다.



$f: \sigma_i P \rightarrow \mathcal{C}(P)$ 로 정의되는 단일값 매칭함수로서 어느 특정 업무수행자,  $\phi$ 와 그의 전입수행자 (자신을 포함함)와의 매칭을 정의한 함수이다.

$f: \sigma_o P \rightarrow \mathcal{C}(P)$ 로 정의되는 단일값 매칭함수로서 어느 특정 업무수행자,  $\phi$ 와 그의 후입수행자 (자신을 포함함)와의 매칭을 정의한 함수이다.

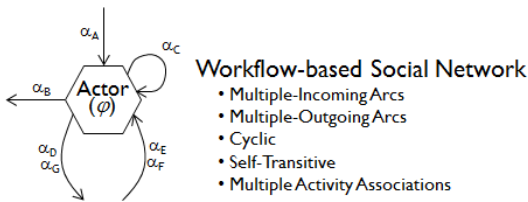
$f: \zeta_a P \rightarrow \mathcal{C}(A)$ 로 정의되는 단일값 매칭함수로서 어느 특정 업무수행자,  $\phi$ 와 그의 전입수행자와 연관된 액티비티 집합과의 매칭을 정의한 함수이다.

즉, 업무수행자들간의 소셜링크인 ( $\sigma_i(\phi)$ ,  $\phi$ )의 형성과 연관된 할당액티비티를 의미한다.

$f: \zeta_o P \rightarrow \mathcal{C}(A)$ 로 정의되는 단일값 매칭함수로서 어느 특정 업무수행자,  $\phi$ 와 그의 후입수행자와 연관된 액티비티 집합과의 매칭을 정의한 함수이다.

즉, 업무수행자들간의 소셜링크인 ( $\phi$ ,  $\sigma_o(\phi)$ )의 형성과 연관된 취득액티비티를 의미한다.

□



(그림 5) 워크플로우 소셜네트워크 모델의 그래픽적 표기법

(그림 5)는 워크플로우 소셜네트워크 모델의 그래픽적 표기법과 그의 특성을 나타낸 것이다. 즉, 수행자(Actor or Performer)는 정육각형으로 표기하고, 수행자들간의 소셜관계는 직선 또는 곡선의 화살표로 표기한다. 그리고 두 수행자들간의 소셜관계를 형성하는 원인이 되는 연관액티비티는 소셜관계선상에 위치시킴으로서 표기될 수 있다. 다음은 워크플로우 소셜네트워크 모델의 그래픽적 표기법을 위한 기본적인 특성 및 규칙을 나타낸 것이다.

각 수행자 객체는 다중의 취득액티비티들에 의한 방향성 있는 소셜관계를 형성할 수 있다.

각 수행자 객체는 자기 자신과의 소셜관계를 형성할 수 있다.

워크플로우 소셜네트워크 모델은 사이클릭 그래프를 형성할 수 있다.

### 3.3 워크플로우 소셜네트워크 발견 알고리즘

정보제어넷기반 워크플로우 모델[12]로부터 워크플로우 소셜네트워크 모델을 발견하는 알고리즘은 다음의 (그림 6)과 같다. 이 알고리즘의 입력은 워크플로우 프로세스 (제어흐름) 모델 ( $\Gamma_C$ )과 워크플로우 역할 및 수행자 할당 모델 ( $\Gamma_R$ )이며, 최종 수행결과인 출력은 발견된 워크플로우 소셜네트워크 모델 ( $\Lambda$ )이다. 특히, 이 발견 알고리즘에서는 게이트웨이액티비티를 암시적으로 정의하는 정보제어넷기반 워크플로우 모델을 입력으로 하며, 따라서 워크플로우 소셜네트워크 모델을 구성하는 액티비티들의 유형은 작업액티비티이다. 결과적으로, 이 알고리즘의 시간복잡성은  $O(N \times E)$ 이며, 여기서  $N$ 은 입력 워크플로우 모델을 구성하는 액티비티의 수,  $E$ 는 입력 워크플로우 모델을 구성하는 액티비티들간의 제어흐름관계 수를 의미한다.

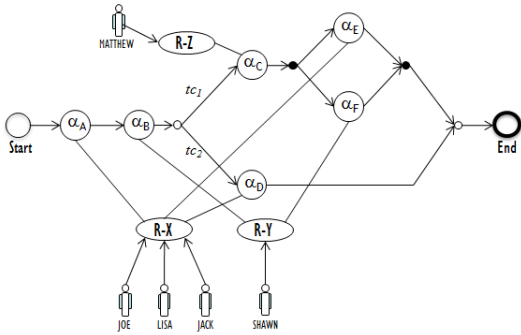
```

[워크플로우 소셜네트워크 발견 알고리즘]
입력:  $\Gamma_C = [ \text{function}(\delta, \kappa), \text{set}(A, T) ]$  (워크플로우 프로세스 모델)
 $\Gamma_R = [ \text{function}(\epsilon, \pi), \text{set}(A, R, P) ]$  (워크플로우 역할 및 수행자 할당 모델)
출력:  $\Lambda = [ \text{function}(\alpha, \zeta), \text{set}(A, P) ]$  (워크플로우 소셜네트워크 모델)
BEGIN PROCEDURE
/* 소셜관계 함수에 의한 소셜링크 발견 */
FOR (  $\forall \alpha \in A$  ) DO /* 여기서,  $\alpha$ 는 이벤트 및 작업 액티비티이다. */
FOR (  $\forall \alpha_o \in \delta_o(\alpha)$  ) DO
ADD all members of  $\pi(\epsilon_o(\alpha))$  TO  $\alpha_o$ (each member of  $\pi(\epsilon_o(\alpha))$ );
DONE
FOR (  $\forall \alpha \in \delta(\alpha)$  ) DO
ADD all members of  $\pi(\epsilon(\alpha))$  TO  $\alpha$ (each member of  $\pi(\epsilon(\alpha))$ );
DONE
/* 연관액티비티 함수에 의한 각 수행자의 할당 및 취득 연관액티비티 발견 */
FOR (  $\forall \alpha \in \delta(\alpha)$  ) DO /* 할당-연관액티비티 발견 */
ADD  $\alpha$  TO  $\zeta_o$ (each member of  $\pi(\epsilon_o(\alpha))$ );
DONE
FOR (  $\forall \alpha_o \in \delta_o(\alpha)$  ) DO /* 취득-연관액티비티 발견 */
ADD  $\alpha$  TO  $\zeta$ (each member of  $\pi(\epsilon(\alpha_o))$ );
DONE
END PROCEDURE
    
```

(그림 6) 워크플로우 소셜네트워크 발견 알고리즘

이 발견 알고리즘의 실행결과인 워크플로우 소셜네트워크 모델은 해당 워크플로우 모델의 제어흐름을 기반으로 형성된 수행자들간의 업무할당 소셜관계를 의미하며, 궁극적으로 중심도(Centrality), 명성도(Prestige), 파당성(Clique), 구조동질성(Structural Equivalence) 등의 소셜네트워크 분석기법[11]을 적용하기 위해서, 이 알고리즘에 의해 발견된 기본 워크플로우 소셜네트워크 모델은 할당-연관액티비티 및 취득-연관액티비티 중심의 워크플로우 소셜네트워크 모델이며, 이로부터 다양한 유형의 소셜네트워크 모델로의 변형이 가능하다. 이 논문에서는 이러한 유형의 워크플로우 소셜네트워크 모델을 생성하는 알고리즘들의 자세한 내용은 기술하지 않는다.





(그림 7) 정보제어넷기반 워크플로우 모델 예의 그래픽적 표현

- 할당-연관액티비티 중심의 워크플로우 소셜네트워크 모델
- 취득-연관액티비티 중심의 워크플로우 소셜네트워크 모델
- 방향성/무방향성 워크플로우 소셜네트워크 모델
- 이진/다중-연관가중치 워크플로우 소셜네트워크 모델

### 3.4 발견 알고리즘 적용과 워크플로우 소셜네트워크 모델의 분석

이 절에서는 앞서 개발한 워크플로우 소셜네트워크 발견 알고리즘을 (그림 7)에 나타난 정보제어넷기반 워크플로우 모델[2]에 적용함으로써 이 알고리즘 동작의 정확성을 검증하고자 한다. 또한, 이 발견 알고리즘의 실행결과인 기본 워크플로우 소셜네트워크 모델은 할당-연관액티비티 중심의 워크플로우 소셜네트워크 모델이고, 이로부터 변형된 연관가중치 워크플로우 소셜네트워크 모델, 그리고 이를 분석하기 위한 소셜행렬(SocioMatrix) [11]의 생성 등 발견부터 분석까지의 일련의 과정을 예를 들어 기술한다. 또한, 발견된 워크플로우 소셜네트워크 모델에 대한 소셜구심도(Degree-Centrality) 분석기법[11] 적용과 그의 분석결과를 예를 들어 기술함으로써 다양한 분석기법들의 적용가능성을 제시한다.

#### 3.4.1 정보제어넷기반 워크플로우 모델 예

워크플로우 소셜네트워크 발견 알고리즘에 적용할 정보제어넷기반 워크플로우 모델의 한 예는 (그림 7)과 같으며, 이 그래픽적 표현에 대한 정형적 표현은 다음과 같다. 즉, 워크플로우 메타-모델을 구성하는 개체들 중에서

워크플로우 소셜네트워크 발견에 직접적인 영향을 미치는 제어흐름 측면과 역할할당 측면 그리고 수행자배정 측면이며, (그림 7)의 워크플로우 모델에 대한 이러한 측면들의 정형적 표현은 다음과 같다.

① (그림 7)의 정보제어넷기반 워크플로우 프로세스 (제어흐름) 모델의 정형적 표현

- $\delta_i(\alpha_{Start}) = \emptyset$ ;  $\delta_o(\alpha_{Start}) = \{ \{ \alpha_A \} \}$ ;  $\delta_i(\alpha_A) = \{ \{ \alpha_{Start} \} \}$ ;  $\delta_o(\alpha_A) = \{ \{ \alpha_B \} \}$ ;
- $\delta_i(\alpha_B) = \{ \{ \alpha_A \} \}$ ;  $\delta_o(\alpha_B) = \{ \{ \alpha_C \}, \{ \alpha_D \} \}$ ;
- $\delta_i(\alpha_C) = \{ \{ \alpha_B \} \}$ ;  $\delta_o(\alpha_C) = \{ \{ \alpha_E, \alpha_F \} \}$ ;
- $\delta_i(\alpha_D) = \{ \{ \alpha_B \} \}$ ;  $\delta_o(\alpha_D) = \{ \{ \alpha_{End} \} \}$ ;  $\delta_i(\alpha_E) = \{ \{ \alpha_C \} \}$ ;  $\delta_o(\alpha_E) = \{ \{ \alpha_{End} \} \}$ ;
- $\delta_i(\alpha_F) = \{ \{ \alpha_C \} \}$ ;  $\delta_o(\alpha_F) = \{ \{ \alpha_{End} \} \}$ ;  $\delta_i(\alpha_{End}) = \{ \{ \alpha_D, \alpha_E, \alpha_F \} \}$ ;  $\delta_o(\alpha_{End}) = \emptyset$ ;

② (그림 7)의 정보제어넷기반 워크플로우 역할할당 모델의 정형적 표현

- $\varepsilon_p(\alpha_{Start}) = \emptyset$ ;  $\varepsilon_p(\alpha_A) = \{ \nu_{R_X} \}$ ;  $\varepsilon_p(\alpha_B) = \{ \nu_{R_Y} \}$ ;  $\varepsilon_p(\alpha_C) = \{ \nu_{R_Z} \}$ ;
- $\varepsilon_p(\alpha_D) = \{ \nu_{R_X} \}$ ;  $\varepsilon_p(\alpha_E) = \{ \nu_{R_X} \}$ ;  $\varepsilon_p(\alpha_F) = \{ \nu_{R_Y} \}$ ;  $\varepsilon_p(\alpha_{End}) = \emptyset$ ;
- $\varepsilon_a(\nu_{R_X}) = \{ \{ \alpha_A, \alpha_D, \alpha_E \} \}$ ;  $\varepsilon_a(\nu_{R_Y}) = \{ \{ \alpha_B, \alpha_F \} \}$ ;  $\varepsilon_a(\nu_{R_Z}) = \{ \{ \alpha_C \} \}$ ;

③ (그림 7)의 정보제어넷기반 워크플로우 수행자배정 모델의 정형적 표현

- $\pi_c(\nu_{R_X}) = \{ \{ \phi_{joe}, \phi_{lisa}, \phi_{jack} \} \}$ ;  $\pi_c(\nu_{R_Y}) = \{ \{ \phi_{shawn} \} \}$ ;  $\pi_c(\nu_{R_Z}) = \{ \{ \phi_{matthew} \} \}$ ;
- $\pi_p(\phi_{joe}) = \{ \nu_{R_X} \}$ ;  $\pi_p(\phi_{lisa}) = \{ \nu_{R_X} \}$ ;  $\pi_p(\phi_{jack}) = \{ \nu_{R_X} \}$ ;
- $\pi_p(\phi_{shawn}) = \{ \nu_{R_Y} \}$ ;  $\pi_p(\phi_{matthew}) = \{ \nu_{R_Z} \}$ ;

#### 3.4.2 발견된 워크플로우 소셜네트워크 모델

(그림 7)의 정보제어넷기반 워크플로우 모델에 대한 정형적 표현들을 (그림 6)의 워크플로우 소셜네트워크 발견 알고리즘에 적용한 결과는 다음과 같다.

① 소셜관계,  $\sigma = \sigma_i \cup \sigma_o$ .

- $\sigma_i(\phi_{Initiator}) = \emptyset$ ;
- $\sigma_o(\phi_{Initiator}) = \{ \{ \phi_{joe}, \phi_{lisa}, \phi_{jack} \} \}$ ;
- $\sigma_i(\phi_{joe}) = \{ \{ \phi_{Initiator}, \phi_{shawn}, \phi_{matthew} \} \}$ ;

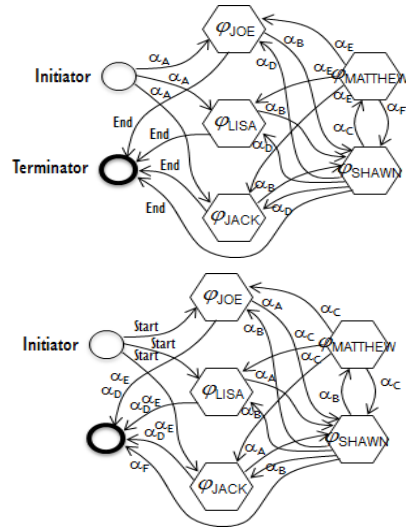
- $\sigma_o(\phi_{joe}) = \{ \phi_{shawn}, \{ \phi_{terminator} \} \};$
- $\sigma_i(\phi_{lisa}) = \{ \{ \phi_{initiator}, \phi_{shawn}, \phi_{matthew} \} \};$   
 $\sigma_o(\phi_{lisa}) = \{ \{ \phi_{shawn}, \{ \phi_{terminator} \} \} \};$
- $\sigma_i(\phi_{jack}) = \{ \{ \phi_{initiator}, \phi_{shawn}, \phi_{matthew} \} \};$   
 $\phi_o(\phi_{jack}) = \{ \{ \phi_{shawn}, \{ \phi_{terminator} \} \} \};$
- $\sigma_i(\phi_{shawn}) = \{ \{ \phi_{joe}, \phi_{lisa}, \phi_{jack}, \phi_{matthew}, \phi_{terminator} \} \};$   
 $\sigma_o(\phi_{shawn}) = \{ \{ \phi_{joe}, \phi_{lisa}, \phi_{jack}, \phi_{matthew}, \phi_{terminator} \} \};$
- $\sigma_i(\phi_{matthew}) = \{ \{ \phi_{shawn} \} \};$   
 $\sigma_o(\phi_{matthew}) = \{ \{ \phi_{joe}, \phi_{lisa}, \phi_{jack}, \phi_{shawn} \} \};$
- $\sigma_i(\phi_{terminator}) = \{ \{ \phi_{joe}, \phi_{lisa}, \phi_{jack}, \phi_{shawn} \} \};$  ;  $\sigma_o(\phi_{terminator}) = \emptyset$  ;

② 연관액티비티,  $\zeta = \zeta_a \cup \zeta_c$

- $\zeta_a(\phi_{initiator}) = \{ \{ \alpha_A, \{ \alpha_A, \{ \alpha_A \} \} \} \};$   
 $\zeta_c(\phi_{initiator}) = \{ \{ \alpha_{start}, \{ \alpha_{start}, \{ \alpha_{start} \} \} \} \};$
- $\zeta_a(\phi_{joe}) = \{ \{ \alpha_B, \{ \alpha_{end} \} \} \};$   
 $\zeta_c(\phi_{joe}) = \{ \{ \alpha_A, \{ \alpha_D, \alpha_E \} \} \};$
- $\zeta_a(\phi_{lisa}) = \{ \{ \alpha_B, \{ \alpha_{end} \} \} \};$   
 $\zeta_c(\phi_{lisa}) = \{ \{ \alpha_A, \{ \alpha_D, \alpha_E \} \} \};$
- $\zeta_a(\phi_{jack}) = \{ \{ \alpha_B, \{ \alpha_{end} \} \} \};$   
 $\zeta_c(\phi_{jack}) = \{ \{ \alpha_A, \{ \alpha_D, \alpha_E \} \} \};$
- $\zeta_a(\phi_{shawn}) = \{ \{ \alpha_D, \{ \alpha_D, \{ \alpha_D, \{ \alpha_C, \{ \alpha_{end} \} \} \} \} \} \};$   
 $\zeta_c(\phi_{shawn}) = \{ \{ \alpha_B, \{ \alpha_B, \{ \alpha_B, \{ \alpha_B, \{ \alpha_F \} \} \} \} \} \};$
- $\zeta_a(\phi_{matthew}) = \{ \{ \alpha_E, \{ \alpha_E, \{ \alpha_E, \{ \alpha_F \} \} \} \} \};$   
 $\zeta_c(\phi_{matthew}) = \{ \{ \alpha_C, \{ \alpha_C, \{ \alpha_C, \{ \alpha_C \} \} \} \} \};$
- $\zeta_a(\phi_{terminator}) = \emptyset$  ;  
 $\zeta_c(\phi_{terminator}) = \emptyset$  ;

여기서, 앞서 기술하였듯이, 워크플로우 소셜네트워크 모델의 정형적 표현에서는 집합이론의 확장형인 가방이론(Bag Theory), 즉 원소의 중복성을 허용할 뿐 만 아니라 원소의 배열순서도 허용한다는 점을 강조하고자 한다. 예를 들어, 아래의 발견결과를 기반으로 가방  $\sigma_o(\phi_{initiator})$ 의 첫 번째 원소  $\{ \phi_{joe} \}$ 와 형성된 소셜관계  $(\phi_{initiator}, \phi_{joe})$ 는 할당-연관액티비티 가방  $\zeta_a(\phi_{initiator})$ 의 첫 번째 원소인 액티비티  $\{ \alpha_A \}$  또는 취득-연관액티비티 가방  $\zeta_c(\phi_{initiator})$ 의 첫 번째 원소인 액티비티  $\{ \alpha_{start} \}$ 에 의해 형성되었음을 유추할 수 있다.

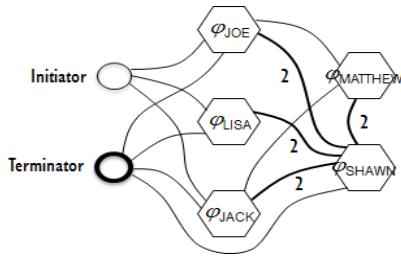
위와 같은 워크플로우 소셜네트워크 모델의 소셜관계 및 연관액티비티에 대한 정형적 표현을 그래픽적으로 표현한 것이 (그림 8)이다. 왼쪽은 각 수행자의 할당-연관액



(그림 8) 발견된 할당-연관액티비티 및 취득-연관액티비티 중심의 워크플로우 소셜네트워크 모델

티비티에 대한 그래픽적 표현이고, 오른쪽은 각 수행자의 취득-연관액티비티에 대한 그래픽적 표현이다. 여기서, (그림 7)의 워크플로우 모델에서 업무시작 액티비티 ( $\alpha_{start}$ )와 업무종료 액티비티( $\alpha_{end}$ )는 각각 업무개시수행자( $\phi_{initiator}$ )와 업무종료수행자( $\phi_{terminator}$ )에 의해 수행된다고 가정한다.

기본적으로 이 논문에서 제시한 알고리즘에 의해 발견된 워크플로우 소셜네트워크 모델의 소셜관계는 방향성을 갖으며, 그 방향성의 의미는 두 수행자들간의 소셜관계를 형성하는데 연관되는 액티비티의 할당 또는 취득관계를 정의하는 것이다. 또한, 발견된 기본적인 워크플로우 소셜네트워크 모델로부터 해당 워크플로우 모델의 실행을 위해 배정된 수행자 그룹내에서 수행자들간의 상호소셜강도(상호협업수준 또는 상호연결강도)를 측정하기 위한 이진 또는 다중의 연관가중치[11]를 갖는 워크플로우 소셜네트워크 모델을 생성할 수 있다. (그림 9)는 (그림 8)의 발견된 할당/취득 연관액티비티 중심의 워크플로우 소셜네트워크 모델로부터 생성된 다중-연관가중치 워크플로우 소셜네트워크 모델을 나타낸 것이다. 즉, (그림 8)의 방향성 워크플로우 소셜네트워크 모델을 무방향성 워크플로우 소셜네트워크 모델로 변형하고, 소셜관계링크들의 중복성을 허용하면 다중(Multiple)-연관가중치가 되고 중복성을 허용하지 않으면 이진(Binary)-연관가중치가 된다.



(그림 9) 발견된 무방향성 다중-연관가중치 워크플로우 소셜네트워크 모델

### 3.4.3 발견된 워크플로우 소셜네트워크의 분석

(그림 9)는 소셜관계링크의 중복성을 허용하여 그 개수를 연관가중치로 한 다중-연관가중치의 워크플로우 소셜네트워크 모델을 나타낸 것이다. 특히, 워크플로우 소셜네트워크 모델에 대해 다양한 유형의 소셜네트워크 분석기법을 적용하기 위해서는 해당 소셜네트워크 모델을 행렬형태로 표현해야 하는데, 이 행렬을 소셜행렬(SocioMatrix)이라고 한다. 결과적으로, (표 1)은 (그림 9)의 다중-연관가중치 워크플로우 소셜네트워크 모델을 소셜행렬(SocioMatrix)로 표현한 것이다. 이러한 다중-연관가중치 워크플로우 소셜네트워크 모델의 소셜행렬(SocioMatrix)에 대해서 중심도(Centrality), 명성도(Prestige), 과당성(Clique), 구조동질성(Structural Equivalence) 등의 소셜네트워크 분석기법[11]을 적용함으로써 궁극적으로 워크플로우 모델상에 내장되어 있는 다양한 소셜네트워크 관련 워크플로우 인텔리전스를 발견하고 분석해 낼 수 있는 것이다. 또한, 이 논문에서는 하나의 워크플로우 모델에 대한 워크플로우 소셜네트워크 모델을 발견하고 분석할 수 있는 접근방법을 기술하고 있지만, 다중의 워크플로우 모델들을 포함하는 워크플로우 패키지[17] 뿐 아니라 다중의 워크플로우 패키지들을 실행하는 수행자 그룹에 형성되는 워크플로우 소셜네트워크 인텔리전스를 발견할 수 있도록 해당 알고리즘을 확장할 필요가 있다.

(표 1) 다중-연관가중치 워크플로우 소셜네트워크 모델의 소셜행렬(SocioMatrix)

|                        | $\varphi_{initiator}$ | $\varphi_{joe}$ | $\varphi_{lisa}$ | $\varphi_{jack}$ | $\varphi_{sha.}$ | $\varphi_{matt.}$ | $\varphi_{terminator}$ |
|------------------------|-----------------------|-----------------|------------------|------------------|------------------|-------------------|------------------------|
| $\varphi_{initiator}$  | 0                     | 1               | 1                | 1                | 0                | 0                 | 0                      |
| $\varphi_{joe}$        | 1                     | 0               | 0                | 0                | 2                | 1                 | 1                      |
| $\varphi_{lisa}$       | 1                     | 0               | 0                | 0                | 2                | 0                 | 1                      |
| $\varphi_{jack}$       | 1                     | 0               | 0                | 0                | 2                | 1                 | 1                      |
| $\varphi_{sha.}$       | 0                     | 2               | 2                | 2                | 0                | 2                 | 1                      |
| $\varphi_{matt.}$      | 0                     | 1               | 0                | 1                | 2                | 0                 | 0                      |
| $\varphi_{terminator}$ | 0                     | 1               | 1                | 1                | 1                | 0                 | 0                      |

$$\textcircled{1} C_d(N_i) = \sum_{j=1}^g x_{ij} \quad \textcircled{2} C_n(N_i) = \frac{C_d(N_i)}{g \times \text{maxweight}}$$

(수행자의 소셜구심도(Degree-Centrality) 분석 및 정규화 수식)

| Actors   | $\varphi_{joe}$ | $\varphi_{lisa}$ | $\varphi_{jack}$ | $\varphi_{sha.}$ | $\varphi_{matt.}$ |
|--|-----------------|------------------|------------------|------------------|-------------------|
| Actor Degree, $C_d(N_i)$                           | 5               | 4                | 5                | 9                | 4                 |
| Normalized Degree, $C_n(N_i)$ $\text{maxweight}=2$ | .36             | .29              | .36              | .64              | .29               |

(그림 10) 수행자의 소셜구심도(Degree-Centrality) 분석결과

(표 1)의 소셜행렬로부터 중심도(Centrality), 명성도(Prestige), 과당성(Clique), 구조동질성(Structural Equivalence) 등의 다양한 워크플로우 소셜네트워크 분석결과를 획득할 수 있다. 이러한 워크플로우 소셜네트워크 분석기법들은 본 논문의 범위에 해당하지 않지만, 워크플로우 소셜네트워크 모델과 그의 소셜행렬에 대한 활용가능성을 증명하기 위하여 간단한 분석기법인 각 수행자별 소셜구심도 분석 수식 및 결과를 기술하고자 한다. 결과적으로, (그림 10)은 대표적인 워크플로우 소셜네트워크 분석기법의 하나인 소셜구심도(Degree-Centrality)의 분석 수식과 그의 적용결과를 나타낸 것이다. 첫 번째 수식(①)[11,16]은 워크플로우 소셜네트워크 개별 수행자의 소셜구심도를 분석하는 수식이며, 두 번째 수식(②)[11,16]은 각 수행자의 소셜구심도 값을 0과 1사이로 정규화하기 위한 수식이다. 수식에서  $g$ 는 소셜행렬을 구성하는 수행자의 수이며,  $\text{maxweight}$ 는 해당 워크플로우 소셜네트워크상에서 형성 가능한 최대 연관가중치를 의미한다. 따라서, (표 1)의 소셜행렬의 경우,  $g=7$ 이며, 최대 연관가중치  $\text{maxweight}=2$ 라고 가정한다. 각 수행자의 소셜구심도 분석결과에서 나타내었듯이, 소셜구심도 정도가 가장 높은 수행자인  $\varphi_{lisa}$ 의 소셜구심도는 9이며, 이를  $\text{maxweight}=2$ 의 경우를 적용하여 0과 1사이의 값으로 정규화하면 0.64이다. 이와 같은 워크플로우 소셜네트워크의 소셜구심도 분석결과는 (그림 7)의 워크플로우 모델을 실행하는데 있어서 가장 업무의 중심도가 가장 높은 수행자는 Lisa라고 해석될 수 있다.

지금까지 본 논문에서는 워크플로우 소셜네트워크 인텔리전스 발견 알고리즘을 제안하고, 그 발견된 기본적인 워크플로우 소셜네트워크 모델과 그로부터 변형가능한 다양한 유형의 확장된 워크플로우 소셜네트워크 모델 그리고 그에 대한 소셜행렬 및 분석기법을 예를 들어 설명하였다. 특히, 본 논문에서 제안한 알고리즘은 하나의 정보제어넷기반의 워크플로우 모델로부터 업무수행자들 간의 소셜네트워크 모델을 발견하는 것을 목표로 하고

있지만, 다중의 워크플로우 모델들, 즉 워크플로우 패키지 뿐 만 아니라 다중의 워크플로우 패키지들로부터 소셜네트워크 모델을 발견할 수 있도록 그의 기능적 확장이 요구된다. 또한, 실세계에 이 알고리즘을 적용하기 위해서는 정보제어넷기반 워크플로우 모델을 XML기반의 국제표준정의언어인 XPDL[12,17]로 표현하고, 이로부터 워크플로우 소셜네트워크 모델을 발견할 수 있도록 수정되어야 할 것이다.

#### 4. 관련연구

최근의 워크플로우 및 비즈니스 프로세스 기술분야에서는 프로세스기반 조직의 복잡성과 혁신성 증대에 따른 프로세스 리엔지니어링을 위한 워크플로우 발견 또는 재발견 이슈[1,6,18]와 업무수행자들간의 소셜이나 협업 인텔리전스 마이닝 및 분석 이슈[8,9,15,16]에 대한 연구가 크게 관심을 받고 있다. 특히, 본 논문의 연구내용과 직접적인 관련이 있는 기술분야인 워크플로우 인텔리전스의 발견 및 재발견 이슈에 대한 연구현황과 본 연구내용과의 차별성에 대한 지속적인 관찰이 필요하다.

워크플로우 소셜 인텔리전스(비즈니스프로세스 인텔리전스)[6,8,15]의 핵심 연구내용 및 범위는 크게 분석, 예측, 모니터링 그리고 제어 부분으로 나뉘질 수 있는데, 본 논문의 워크플로우 소셜네트워크 인텔리전스 발견 알고리즘은 분석과 예측 부분에 해당된다고 할 수 있다. 즉, 워크플로우기반 소셜 네트워크 발견 기법 및 알고리즘들[16,17]은 워크플로우 모델 또는 그의 집합인 워크플로우 패키지들에 대한 분석을 통해 소셜네트워크를 발견할 수 있으므로 워크플로우 인텔리전스 분석 부분에 해당하며, 워크플로우기반 소셜네트워크 재발견 기법 및 알고리즘들[10,18]은 워크플로우의 실행이력 또는 로그[3,7,8]로부터 소셜네트워크를 재발견하고, 이의 결과를 워크플로우 모델로부터 발견된 소셜네트워크 지식과 비교함으로써 해당 워크플로우 모델의 충실도(Fidelity) 분석과 개선을 위한 예측을 수행할 수 있기 때문이다. 기존의 워크플로우 인텔리전스 관련 시스템들[8,9,17]은 대부분 워크플로우 인텔리전스[7]의 모니터링 및 제어 부분에 해당한다고 판단되며, 특히 기존의 워크플로우 지식 발견 및 재발견 기법들[1,2,6,10,13,14]은 프로세스기반 엔터프라이즈 시스템의 실행이력으로부터 워크플로우 프로세스(제어흐름 관점) 지식을 발견 및 재발견하는데 집중되고 있다.

따라서, 본 논문의 주요 연구내용인 워크플로우 수행자들간의 소셜네트워크를 발견하는 알고리즘은 워크플로우 인텔리전스 발견 및 재발견 분야의 새로운 영역을 개척했다는 점에서 의미가 있다고 판단된다. 또한, 워크플로우기반 소셜네트워크 관련 기존의 연구결과는 그 수적인 측면에서도 매우 적을 뿐 만 아니라 주로 페트리넷(Petri-net)[18]을 이론적 기반으로 하고 있다는 점이다. 본 논문에서 제안한 발견 알고리즘의 이론적 배경은 정보제어넷(Information Control Net)[12]을 기반으로 하는 최초의 시도라는 점에서 의미가 있다.

#### 5. 결 론

지금까지 본 논문에서는 워크플로우 소셜네트워크 인텔리전스의 기본 개념을 정의하였고, 정보제어넷기반의 워크플로우 모델로부터 이 새로운 개념의 인텔리전스를 발견하기 위한 알고리즘을 제안하였다. 또한, 제안된 발견 알고리즘의 기능적 정확성을 검증하기 위한 적용 예로서 프로세스기반 엔터프라이즈의 대표적인 워크플로우 모델들 중의 하나인 주문처리 프로세스를 모델링한 정보제어넷기반 워크플로우 모델을 이 알고리즘에 적용하고, 그 결과인 워크플로우 소셜네트워크 모델과 그의 확장 모델에 대한 소셜구심도 분석기법을 적용하였다. 결과적으로, 현재 소셜네트워크 개념을 기반으로 하는 대부분의 시스템 또는 서비스들은 개인 중심, 즉 사적인 일상 중심의 소셜 네트워크 지식에 초점을 두고 있는데 반하여, 본 논문의 초점은 엔터프라이즈 중심, 즉 직장 업무 중심의 소셜 네트워크 지식에 초점을 두고 있다. 따라서, 본 논문의 주요 기여는 기존의 소셜네트워크 개념을 “개인 중심의 소셜네트워크에서 엔터프라이즈 중심의 소셜네트워크로”라는 근본적인 개념 확장의 계기가 된다는 점에서 의미가 크다고 할 수 있다.

추후 연구의 핵심내용으로써, 본 논문의 연구결과는 단 하나의 워크플로우 모델로부터 업무수행자들간의 업무관련 협업 소셜네트워크를 발견하고, 그 결과를 소셜네트워크 분석기법을 통해 분석하는 일련의 과정을 기술하고 있지만, 궁극적으로 엔터프라이즈 전체의 업무프로세스들을 대상으로 워크플로우기반 소셜네트워크 지식을 발견 및 재발견하는 기법 및 알고리즘들을 연구하고 이를 기반으로 하는 소셜네트워크 인텔리전스 시스템 구현할 수 있도록 본 논문에서 제안한 발견 알고리즘의 기능적/성능적 범위를 확장할 필요가 있다.

## 참 고 문 헌

- [1] Kwanghoon Kim, Clarence A. Ellis, "sigma-Algorithm: Structured Workflow Process Mining through Amalgamating Temporal Workcases," Lecture Notes in Artificial Intelligence, Vol. 4426, pp.119-130, 2007
- [2] Clarence A. Ellis, et al., "Beyond Workflow Mining," Lecture Notes in Computer Science, Vol. 4102, pp.49-64, 2006
- [3] Kwanghoon Kim, "An XML-Based Workflow Event Logging Mechanism for Workflow Mining," Lecture Notes in Computer Science, Vol. 3842, pp.132-136, 2006
- [4] Kwanghoon Kim, "Actor-oriented Workflow Model," Proceedings of the 2nd international symposium on Cooperative Database Systems for Advanced Applications, WOLLONGONG, AUSTRALIA, March 27-28, 1999
- [5] Jacques Wainer, Kwanghoon Kim, Clarence A. Ellis, "A Workflow Mining Method Through Model Rewriting," Lecture Notes in Computer Science, Vol. 3706, pp.184-191, 2005
- [6] Kwang-Hoon Kim, Clarence A. Ellis, "Workflow Reduction for Reachable-path Rediscovery in Workflow Mining," Foundations and Novel Approaches in Data Mining, Vol.9, pp.288-309, Springer, 2006
- [7] Min-Jae Park, Kwanghoon Kim, "A Workflow Event Logging Mechanism and Its Implications on Quality of Workflows," JOURNAL OF INFORMATION SCIENCE AND ENGINEERING, Vol.26, No.5, pp.1817-1830, 2010
- [8] Harri Oinas-Kukkonen, et al., "Social Networks and Information Systems: Ongoing and Future Research Streams," JOURNAL OF THE ASSOCIATION OF INFORMATION SYSTEMS, Vol.11, Issue2, pp. 61-68, 2010
- [9] Mary Griffiths, "Oxygen: Social Intranets, Collective Intelligence and Government Practices," ELECTRONICS JOURNAL OF E-GOVERNMENT, Vol.5, Issue2, pp.177-190, 2007
- [10] Wil M. P. van der Aalst, Hajo A. Reijers, Minseok Song, "Discovering Social Networks from Event Logs," COMPUTER SUPPORTED COOPERATIVE WORK, Vol.14, No.6, pp.549-593, 2005
- [11] David Knoke, Song Yang, SOCIAL NETWORK ANALYSIS-2nd Edition, Series: Quantitative Applications in the Social Sciences, SAGE Publications, 2008
- [12] Kwanghoon Kim, Clarence A. Ellis, "Section II / Chapter VII. An ICN-based Workflow Model and Its Advances," Handbook of Research on BP Modeling, pp.142-172, IGI Global, ISR, 2009
- [13] Jaekang Won, "A Framework: Organizational Network Discovery on Workflows," Ph.D. Dissertation, Department of Computer Science, KYONGGI UNIVERSITY, 2008
- [14] Aubrey J. Rembert, "Automatic Discovery of Workflow Models," Ph.D. Thesis Dissertation, Department of Computer Science, UNIVERSITY OF COLORADO AT BOULDER, 2008
- [15] Wil M.P. van der Aalst, "Challenges in Business Process Analysis," LECTURE NOTES ON BUSINESS INFORMATION PROCESSING, Vol. 12, pp.27-42, 2008
- [16] Jihye Song, et al., "A Framework: Workflow-based Social Network Discovery and Analysis," Proceedings of the 2nd International Workshop on Workflow Management in Service and Cloud Computing, Dec. 10~13, 2010, Hongkong, China
- [17] Kwanghoon Kim, "A Workflow-based Social Network Discovery and Analysis System," Proceedings of the 1st International Symposium on Data-driven Process Discovery and Analysis, pp.163-176, Jun 29-Jul 1, 2011, Campione d'Italia, Italy
- [18] A. Rozinat, et al., "Discovering Simulation Models," INFORMATION SYSTEMS, Vol.34, pp.305-327, 2009

● 저 자 소 개 ●



**김 광 훈 (Kwanghoon Kim)**

1984.2 경기대학교 이과대학 전자계산학과 학사

1986.2 중앙대학교 일반대학원 전자계산학과 석사

1994.5 University of Colorado at Boulder Department of Computer Science, MS

1998.5 University of Colorado at Boulder Department of Computer Science, Ph.D.

1986.2~1991.8 한국전자통신연구원 연구원

2005.3~2010.2 Univ. of Colorado at Boulder Department of Computer Science, 방문교수

2007.7~2010.6 콘텐츠융합소프트웨어연구센터장

1998.3~현재 경기대학교 컴퓨터과학과 정교수

2002.3~현재 비피엠코리아포럼 부회장

2003.1~현재 WfMC ERC Vice-chair

2003.1~현재 TTA 정보통신국제표준전문가

2000.1~현재 한국인터넷정보학회 이사, 부회장

2012.2~현재 경기대학교 전산정보원 원장

관심분야: 워크플로우/비피엠 기술, RFID/USN 미들웨어 기술, Collaboration Technology

E-mail: kwang@kgu.ac.kr