

# 워크플로우 마이닝 프레임워크에 대한 고찰

김 학 성\*      김 형 목\*\*      김 광 훈\*\*\*

## ◆ 목 차 ◆

- |              |                         |
|--------------|-------------------------|
| 1. 서 론       | 3. 워크플로우 마이닝 아키텍처 프레임워크 |
| 2. 워크플로우 마이닝 | 4. 결 론                  |

## 1. 서 론

어느 시대 또는 어느 형태의 사회라 할지라도 그 사회가 갖는 사람과 사람들간의 상호 활동과 그들 간의 관계를 지원하기 위한 특징적인 수단과 방법을 제공하고 있다. 오늘날과 같이 사무실에서뿐만 아니라 가정에서도 컴퓨터를 광범위하게 이용하고 있는 정보화 사회에서는 사람들간의 상호 활동 및 관계에 있어서 10년 전이나 20년 전과 별 차이가 없겠지만, 그들을 지원하기 위한 수단이나 방법·면에서는 많은 차이가 있음에 틀림없다. 즉, 컴퓨터 기술과 전자통신 기술의 급진적인 발전과 이러한 기술들간의 수렴은 바로 전자적인 작업환경(Electronic Workspace)이라는 새롭고도 매우 효율적인 상호 작용 지원 수단 및 방법을 잉태하게 되었다. 전자적인 작업환경이란 전(全) 조직체적 통합 시스템으로 정보 처리 활동과 정보 통신 활동의 통합을 통해 조직내의 구성원들간의 상호 활동 및 관계를 정의하고 지원하는 개선된 형태의 조직 활동 수단 및 방법이다.

워크플로우 기술이란 바로 이러한 전자적 작업환경을 구현하기 위한 종합적인 연구 분야로서 컴퓨터 및 통신 분야 뿐 만 아니라 사회학 분야나 언어학 분야, 경영학 분야 등의 다각적인 협력 관계를 통해서만 성공적으로 완성될 수 있는 매우 다중 적인 연구 분야라고 할 수 있다.

최근에는 조직체내에서 처리하고 있는 일련의 사무 업무 처리 과정들을 비즈니스 프로세스(Business Process)를 통해 정의하고 이들의 효율적이면서 효과적인 관리를 위한 자동화된 작업 및 사무환경을 지원하는 사무 운영체제(Business Operating System)라는 용어가 등장하였는데, 워크플로우 기술은 바로 이러한 사무운영체제의 근간이 되는 기술이라고 할 수 있다. 즉, 워크플로우 기술은 조직체내 또는 조직체들간에 처리되고 있는 일련의 사무업무 또는 그룹 작업 및 활동이 어떻게 이루어지는가를 분석하고 컴퓨터 및 통신을 비롯한 첨단 기술들을 이용하여 어떻게 그룹 활동들을 효과적으로 그리고 자동적으로 지원할 수 있는가를 연구하는 광범위한 분야이다.

워크플로우 기술이 전자상거래를 비롯한 고객관리기술(CRM : Customer Relationship Management), 공급망 및 가치사슬망 관리기술(SCM : Supply Chain Management), 데이터 및 응용 프로그램 통합기술(EAI : Enterprise Applications Integration), 전사적 자원관리(ERP : Enterprise Resource Planning) 등과 같은 최첨단 정보기술의 핵심 기반 기술로서 중요하게 인식되고 있는 이유는 현재와 같이 분산되고 복잡한 업무 프로세스를 지원할 수 있고, 고성능, 확장성, 신뢰성, 유연성, 상호 가용성 등과 같은 요구 사항들을 제공할 수 있는 기술이 워크플로우 기술이라고 기대하기 때문이다. 또한 B2C/B2B로 대변되는 전자상거래 및 전자시장(E-Market Place)의 활성화와 함께 전자정부(E-Government) 실현을 위한 기반 구조가 급속하게 확장됨에 따라 워크플로우를 기반으로 한 조직내부 업무 흐름의 자동화를 의미하는 B2E의 구축을 더욱 가속

\* 동남보건대학 컴퓨터응용과 교수  
\*\* 경기대학교 대학원 전자계산학과 박사과정  
\*\*\* 경기대학교 정보과학부 교수

화시키고 있으며, 워크플로우 기술 및 시스템은 결국 기업 또는 조직체내의 모든 업무 처리 절차들을 통합 관리하는 인프라구조(Infrastructure)로 인식되고 있다.

워크플로우는 크게 워크플로우 모델 분야와 워크플로우 관리 시스템 분야로 나눌 수 있고 워크플로우 모델 분야는 일련의 비즈니스 관련 업무들의 흐름(Control Flow)뿐만 아니라 관련 데이터(Data Flow)와 조직내의 참여자(Actor) 또는 참여자(Participant), 역할(Role) 등을 표현하고 정의하는 수단을 제공하고, 워크플로우 관리 시스템 분야는 워크플로우 모델 분야에서 정의된 업무 흐름을 실제 수행하고 그와 관련된 각종 자원들을 관리함으로써 업무 흐름 전반을 관리 및 제어하는 기능을 수행한다. 즉, 워크플로우 관리 시스템의 수행이 정상적으로 종료되었다는 의미는 워크플로우 모델에서 정의된 비즈니스 프로세스의 수행을 마쳤다는 것을 의미하고 있다.

그러나 최근에 조직 내 또는 조직간에 발생되는 비즈니스 프로세스가 복잡해지고, 규모가 커지고, 오랜 수행 시간을 요구하고 있어 비즈니스 프로세스를 디자인하는 작업은 매우 복잡하고, 많은 시간을 요구하는 작업이다. 또한 작성된 비즈니스 프로세스가 워크플로우 관리 시스템에서 정상적으로 수행되리라고 보장 할 수 없다. 즉, 워크플로우 관리 시스템에서 수행되기 위한 완벽한 비즈니스 모델을 정의하는 것이 쉽지 않다는 것을 의미하고 있다.

잘못된 비즈니스 프로세스의 정의로 인하여 워크플로우 관리 시스템의 실행 중 오류 또는 정상적인 종료를 할 수 없다면 이는 조직 또는 기업의 관점에서 볼 때 치명적(Mission Critical)인 문제가 될 수 있다.

이와 같은 이유로 워크플로우 연구 분야에서는 예외상황 처리 및 동적 변경을 가능케 하여 워크플로우 시스템의 유연성을 제공하려는 연구와 비즈니스 프로세스의 개선 및 조직 또는 기업이 보유하고 있는 자원 활용의 극대화를 위한 워크플로우의 자원관리 분야 그리고 수행되는 워크플로우시스템의 성능을 측정하기 위한 워크플로우 시간관리에 대한 연구가 활발하게 진행중이다.

본 논문에서는 워크플로우 관리시스템에서 발생된 실행정보를 이용하여 비즈니스 프로세스의 모델링 정보, 예외처리 또는 동적변경을 위하여 유용한 정보로

이용될 수 있는 워크플로우 마이닝의 개념을 정의하고 최근 연구중인 프로세스 마이닝의 기법에 대하여 기술한다.

마지막으로 워크플로우 관리시스템에서 워크플로우 마이닝을 위하여 어떤 정보를 어떻게 이용할 수 있는지를 설명하는 워크플로우 마이닝 아키텍처 프레임워크를 설명한다.

## 2. 워크플로우 마이닝

### 2.1 워크플로우 마이닝의 개념

지난 10여년 동안 많은 상용 워크플로우 관리시스템 - IBM MQSeries, COSA, Stafware 등등 - 은 다양한 기업 정보기술 분야에 적용되어 사용되어 왔다. 그러나 대부분의 워크플로우 관리 시스템 제품들은 처리해야 할 비즈니스 프로세스의 복잡화, 대규모화, 매우 긴 실행시간(very long execution time) 등과 같은 조건에 만족하기 위한 유연성(flexibility)과 동적 변경(Dynamic Change) 등을 처리하는데 있어서 많은 문제점을 나타내고 있다.

현재 대부분의 워크플로우 관리 시스템은 빌드-타임에 정의된 비즈니스 프로세스를 이용하여 런-타임에서 실행시키는 과정이다. 그러나 자의든 타의든 어떠한 이유로 두 단계의 정보의 모순 또는 불일치로 인하여 워크플로우의 실행이 더 이상 불가능할 때 이런 문제를 처리할 수 있는 유연성을 제공치 못하고 있다. 또한 워크플로우 프로세스에 대한 실행 정보 및 부하 정보를 활용하여 진행중인 처리 프로세스의 속성 -종료예정일, 단위업무 담당자, 종료 단위업무 등과 해당 처리 프로세스에 포함된 단위업무를 변경할 수 있는 동적 변경 기능을 제공해야 한다. 즉, 하나의 프로세스가 워크플로우 시스템에서 수행되었을 경우, 이들의 프로세스 수행 내용이 실행 중에 변화될 필요성이 발생할 수 있다는 것을 의미하고 있다.

비즈니스 프로세스를 정의하기 위한 빌드-타임이 아닌 런-타임에서 정의된 프로세스가 변경되기 위해서는 프로세스의 흐름정의와 관련된 모든 정보들이 프로세스 모델과 처리 프로세스에서 별도로 관리되어야 한다. 이와 같은 동적 변경을 지원하는 워크플로우 관리시스템에는 예외적인 상황이 발생할 경우에 사용

자나 관리자의 요청에 의해서 프로세스의 흐름을 변경하여 업무를 처리할 수 있는 기능을 제공하지 못하고 있다. 유연성과 관련된 문제, 즉 빌드-타임과 런-타임 정보의 불일치로 발생될 수 있는 문제를 해결하기 위하여 프로세스의 실행된 정보를 이용하여 빌드-타임의 프로세스 정의에 이용하는 방법(reverse the process)이나, 워크플로우 시스템의 런-타임 시점에 빌드-타임에서 정의된 프로세스가 변경되기 위해서는 프로세스의 흐름정의 및 실행된 모든 정보를 알기 위한 방법으로 워크플로우 마이닝이라는 개념이 필요하게 되었다.

워크플로우 마이닝 개념은 기업 또는 조직의 프로세스 개선이라는 문제와도 직결될 수 있는 사항이다. 프로세스 개선은 2가지 관점에서 고려할 수 있는데, 첫 번째는 처리가 완료된 프로세스의 통계자료를 분석하고, 처리 지연요인과 상황을 파악하는 통계적 관점에서의 프로세스 개선과 두 번째는 처리 진행중인 프로세스가 워크플로우 시스템에 의해서 자동으로 적절하게 변경되는 실시간 관점에서의 프로세스 개선이 존재한다.

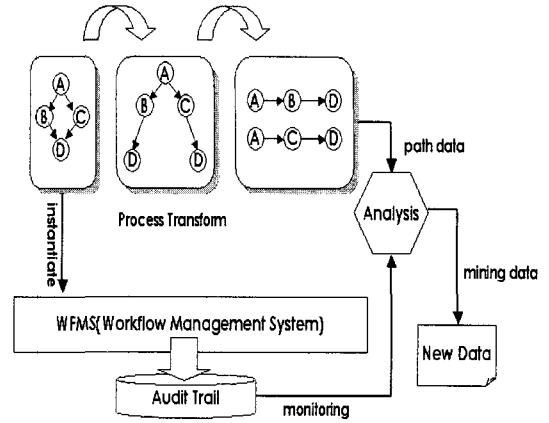
전자의 경우에는 워크플로우 시스템이 제공하는 기능의 측면보다는 워크플로우 시스템의 부가적인 활용 방안에 가까우며, 개선을 위한 처리 내용도 업무 처리 프로세스의 재구성 및 업무 효율 증가에 필요한 리소스 충원 등의 시스템 외적인 요인이 대부분이다.

반면 후자의 경우는 워크플로우 시스템이 프로세스를 처리하는 도중에도 프로세스의 개선을 위해서 시스템이 적절한 조치를 취하는 것으로, 필요에 따라 프로세스 우선순위를 적절히 조절하며, 집중되는 업무를 여러 참여자에게 분배시키는 기능들을 제공하는 시스템의 기능이다. 이와 같은 워크플로우 마이닝의 개념을 정리하여 보면 다음 표 1과 같다.

(표 1) 워크플로우 마이닝 개념

사용자에게 의해 정의된 워크플로우 프로세스를 시스템을 통해 실행된 정보를 바탕으로 현재 실행된 프로세스를 평가하고, 기존의 프로세스를 재구성하거나 새로운 워크플로우 프로세스를 생성할 때 도움을 줄 수 있는 데이터를 추출하는 과정
---

이와 같은 워크플로우 마이닝의 개념을 도식화하면 그림 1에 나타난 것과 같다. 비즈니스 프로세스의 실행



(그림 1) 워크플로우 마이닝 개념도

경로를 마이닝하여 작성된 비즈니스 프로세스의 개선 또는 새로운 비즈니스 프로세스를 생성하기 위한 목적으로 진행되는 개략적인 마이닝 과정이다.

이러한 목적을 달성하기 위한 그림에서 보여지듯이 마이닝 과정은 정의된 비즈니스 프로세스와 실제 실행된 인스턴스의 실행 경로를 비교 분석하는 과정이다. 즉, 빌드-타임에서 정의된 비즈니스 프로세스를 분석하여 모든 가능한 경로를 구한다. 한편 정의된 비즈니스 프로세스가 워크플로우 엔진에서 하나의 인스턴스로 발생되고 실행된 정보를 이용하여 구해진 실제 실행된 경로의 정보를 구한다. 이렇게 얻어진 두 정보를 비교, 분석하여 새로운 실행경로에 대한 유용한 정보를 구할 수 있다.

이와 같이 얻어진 정보를 이용하여 현재 실행된 비즈니스 프로세스를 개선 또는 새로운 비즈니스 프로세스를 작성하는 경우 유용하게 사용될 수 있다

## 2.2 워크플로우 마이닝의 사전연구

오늘날 산업체 및 공공부문에서는 효율적인 업무 처리가 조직체의 경쟁력과 밀접한 관련이 있다. 이러한 이유로 업무 처리의 일부 또는 전체를 자동화하고자 하는 목적에서 워크플로우의 개념이 도입되고 있으며, 이전의 장에서도 설명한 것과 같이 워크플로우의 실행은 빌드-타임에서 정의된 비즈니스 프로세스의 정의에 따라 실행된다.

그러나 현재와 같이 업무 처리의 규모가 거대해져

고, 많은 단계를 거치는 것과 같이 업무 처리가 복잡해짐에 따라 워크플로우를 디자인하는 것은 많은 시간을 요구하는 작업이고, 또한 작업된 워크플로우 디자인이 어떠한 보수도 없이 정의되었다고 보장 할 수 없다. 이러한 이유로 현재 워크플로우 연구 분야에서는 워크플로우의 실행된 정보를 이용하여 워크플로우 디자인에 적용하기 위한 연구 즉, 워크플로우 마이닝 기술이 중요한 연구 주제이다.

본 절에서는 현재 워크플로우 마이닝 개념과 관련되어 연구되고 있는 상황에 대하여 기술하겠다.

### 2.2.1 실행정보를 이용한 프로세스 마이닝 기법

워크플로우 시스템의 실행은 정해진 프로세스의 진행을 의미한다. 즉, 업무 처리를 위한 프로세스에 정의된 액티비티들의 정해진 순서의 실행을 의미한다. 이와 같이 정해진 순서에 의해 실행되는 액티비티 사이에는 종속성(dependency)이 존재하게 되며, 일반적으로 워크플로우 시스템에서는 이러한 액티비티의 종속성 관계를 방향 그래프(Directed Graph)로 표시한다. 또한 액티비티 간의 전이(transition)를 나타내기 위하여 제어 흐름의 결과값(Boolean Value)에 의해서 결정된다. 실행정보를 이용한 마이닝 기법에서는 워크플로우가 실행정보를 이용하여 새로운 프로세스 모델 즉, 액티비티의 종속성을 표현하는 그래프 모델을 제시하기 위한 알고리즘을 제시한다. 즉, 액티비티 사이의 종속성을 나타내어 그 관계를 표현하는 그래프로 나타내는 것이 최종의 목표이며 제시된 알고리즘에 따라 작성된 그래프 모델은 비즈니스 프로세스의 제어 흐름(Control Flow)을 의미하며 다음과 같은 사항을 만족하고 있다.

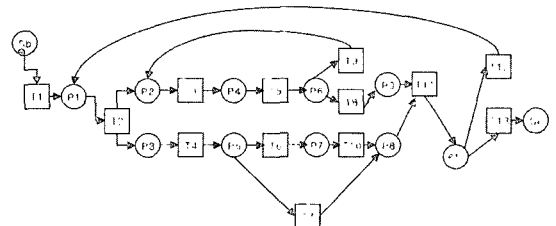
- ▶ 완전성(Completeness)
  - : 실행 결과로 작성된 기록정보 내의 모든 액티비티의 사이의 종속성은 반드시 보존되어야 한다.
- ▶ 중복의 배제(irredundancy)
  - : 작성된 그래프는 항상 논리적인 액티비티의 의존성만을 표시한다.
- ▶ 최소화(Minimality)
  - : 그래프 표현의 간소화를 위하여 최소의 간선(edge)으로 표현되어야 한다.

제안된 알고리즘은 실행정보를 분석하여 각 액티비티 사이의 종속성을 분석하여 표현된 그래프는 워크플로우의 모델링을 위한 방법으로 사용 될 수 있고, 어떠한 워크플로우 엔진에서는 사용할 수 있는 알고리즘을 제시하였다. 그러나, 제안된 알고리즘을 이용하여 워크플로우의 모델 즉, 순차적인 실행을 표현할 수밖에 없는 제약이 있고 또한 개선된 알고리즘을 위하여 지금도 연구가 진행중인 마이닝 기법이다.

### 2.2.2 Event Based Data를 이용한 프로세스 마이닝 기법

상기에서 기술된 연구는 실행정보를 이용하여 각 액티비티의 종속성을 나타내는 방향성 그래프를 도출하여 비즈니스 프로세스 정의에 이용하는 방법이었으나, Event-Based Data를 이용한 프로세스 마이닝 기법은 워크플로우의 실행정보를 이용하여 최종적으로 비즈니스 프로세스를 정확하게 표현하기 위한 페트리넷으로 표현하고 있다. 이렇게 표현함으로써 AND 병합과 분기, OR 병합과 분기, 그리고 반복에 대해서도 정확하게 표현할 수 있다.

다음의 그림 2는 페트리넷을 이용하여 비즈니스 프로세스를 표현한 예를 보여주고 있다.



(그림 2) 페트리넷으로 표현된 비즈니스 프로세스 예제

비즈니스 프로세스의 페트리넷 표현에서 태스크는 트랜지션으로 표현되고 의존성은 아크의 위치로 표현할 수 있다. 예를 들어 상기 그림 7에서 T1, T2, ... T13은 태스크를 나타내고, Sb, P1, P2, ... P10, Se는 의존성을 나타내고 있다. 또한 T2에서 P2와 P3로의 분기는 AND-분기를 표현하고 있고, P8, P9에서 태스크 T11의 병합은 AND-병합을 나타내고 있다. 또한 P5에서 T6, T7의 분기는 OR-분기를 표현하고 있으며, T7과 T10에서 P8로의 병합은 OR-병합을 의미하고 있다.

이와 같이 워크플로우의 실행정보를 마이닝 하여 페트리넷을 작성하기 위하여 다음과 3단계를 거친다.

- [1단계] 각 태스크간의 의존성/빈도 테이블 작성
- [2단계] 의존성/빈도 테이블로부터 의존성/빈도 그래프 작성
- [3단계] 의존성/빈도 그래프로부터 페트리넷 작성

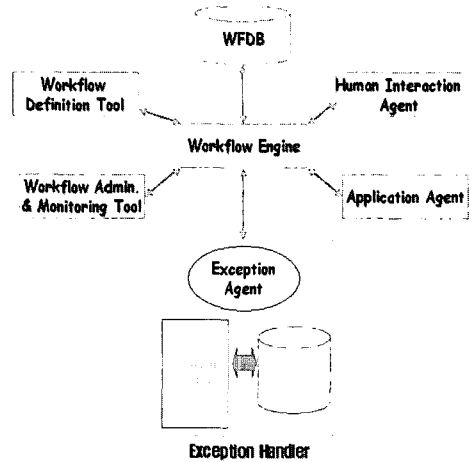
Event-Based Data를 이용한 프로세스 마이닝 기법은 AND 또는 OR 분기와 병합에 대하여 표현할 수 있는 장점이 있지만 현재의 연구결과는 제한된 수의 태스크를 이용하여 실험한 결과이기 때문에 완벽한 비즈니스 프로세스를 표현 할 수 있는지의 여부에 대하여 좀 더 연구가 필요하고, 마이닝 과정을 통하여 얼마나 정확하고 빠르게 비즈니스 프로세스를 표현할 수 있는 것과 같은 질의 향상에 좀 더 많은 연구가 필요하다.

2.2.3 예외처리 마이닝을 이용한 워크플로우 예외처리

워크플로우 시스템의 적용환경이 처리해야 할 비즈니스 프로세스의 대규모화, 복잡성, 매우 긴 수행시간, 사용자 요구의 증가 그리고 관련 데이터의 증가 등과 같이 변화함에 따라 워크플로우 시스템에서 예외처리에 대한 연구는 매우 중요한 분야로 인식되고 있다.

이와 같은 워크플로우 시스템에서 발생하는 예외상황 또는 오류를 처리하기 위하여 많은 방법들이 제공되고, 연구되고 있다. 그러나 대부분의 방법은 빌드-타임에서 예측되는 오류(expected exception)에 대해서만 처리할 수 있는 방법을 제공하고 있다. 즉, 워크플로우 시스템이 실행 중에 발생될 수 있는 예측치 못한 오류 또는 예외 상황(unexpected exception)에 대한 처리는 워크플로우 이외의 부가적인 시스템으로 처리하는 것이 대부분이다.

예외처리 마이닝을 이용한 워크플로우 예외처리 기법에서는 빌드-타임에서 정의될 수 있는 예외상황은 물론 런-타임에서 발생될 수 있는 예외상황, 예를 들어 이미 수행이 완료된 액티비티의 롤백(rollback) 여부를 결정한다든지, 워크플로우 시스템의 동적 변경에 가능여부의 결정하는 것과 같은 작업 그리고, 런-타임에 발생된 예외상황에 대한 해결책을 제시하기 위한



(그림 3) 예외상황 마이닝을 이용한 워크플로우 예외처리 구조

방법으로 이전에 발생된 예외상황과 처리결과와 현재 발생된 예외상황을 비교하여 최적의 해결책을 찾는 다음과 같은 3가지 알고리즘을 제안하고 있다.

- ▶ SEQ-E(SEQuential matching algorithm)
- ▶ SEQ-C(SEQuential matching algorithm on Concept level permutation)
- ▶ BIN-C(BINary search on the sequence of Concept level permutation)

예외상황 마이닝을 이용한 워크플로우 예외처리를 위한 구조는 그림 3과 같이 구성되어 있다. 예외상황 처리기(Exception Handler)의 EHDB(Exception Handler DataBase)는 이전에 발생되었던 예외상황을 기록하기 위한 데이터 베이스이다. 이 때 발생된 오류 또는 예외의 저장을 위한 항목은 다음과 같다.

- ▶ Status : 현재 실행되고 있는 인스턴스의 상태정보를 저장하는 항목으로 예를 들어 running, suspended, terminated, active, complete 등과 같은 내용을 의미한다.
- ▶ Activity : 오류 또는 예외가 발생한 액티비티 정보를 기술한다.
- ▶ Event : 발생된 오류 또는 예외상황을 기술하기 위하여 개발자가 정의한 오류 메시지 등이 저장될 수

있다. 예를 들어 “지정된 수행자(actor)를 찾을수 없습니다.” 등과 같이 개발자가 정의한 오류 또는 예외 상황의 의미를 저장한다.

- ▶ **When** : 오류 또는 예외상황이 발생된 시간을 기록한다.
- ▶ **Who** : 오류 또는 예외상황이 발생된 액티비티의 수행자를 기술한다.

Discovery Module은 상기에서 언급된 3가지의 알고리즘을 이용하여 실행되는 모듈로서 현재 발생된 예외 또는 오류와 가장 유사한 이전의 예외 또는 오류를 찾아내 해결책을 제시한다.

워크플로우 엔진이 실행되는 동안 발생된 예외상황은 예외상황 처리기의 예외상황 에이전트 (Exception Agent)에 의하여 감지된다. 이와 같이 발생된 오류 또는 예외상황과 가장 유사한 과거의 내용 및 해결책을 찾기 위하여 Discovery Module이 수행되고 이 결과로 현재 발생된 오류 또는 예외상황의 해결책을 수행하여 문제를 해결할 수 있다.

### 3. 워크플로우 마이닝 아키텍처 프레임워크

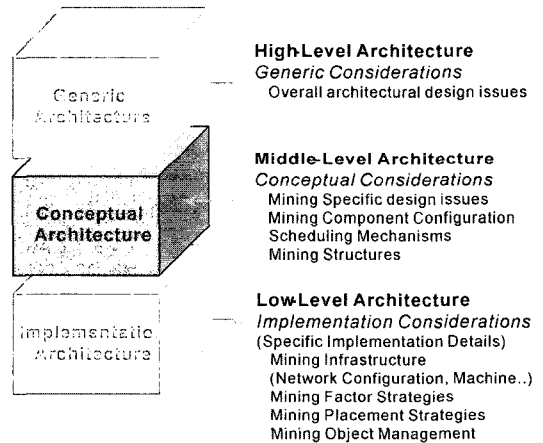
워크플로우 마이닝을 구현하기 위하여 고려해야 할 사항 및 원칙 등을 기술하기 위하여 워크플로우 마이닝 아키텍처 프레임워크를 기술한다. 제안된 워크플로우 마이닝 아키텍처 프레임워크는 그림 4와 같이 3단계 - *Generic Level, Conceptual Level, Implementation Level* - 로 구성되며, 각 단계는 계층적 구조를 갖고 있다. 이와 같은 워크플로우 마이닝 아키텍처 프레임워크에서 제시된 개발 항목 또는 고려사항은 다음과 같은 이유에서 중요한 의미를 갖는다.

- ▶ 개념적 단계에서의 워크플로우 마이닝 시스템에의 공통적인 개발항목 또는 고려사항은 실제 시스템을 구현하기 이전에 워크플로우 마이닝이 적용될 업무, 목적 등을 선택하는데 있어 기준이 될 수 있다.
- ▶ 워크플로우 시스템은 그 자체로서 매우 복잡한 시스템이다. 그러므로 워크플로우 실행정보를 이용하는 워크플로우 마이닝을 구현하고자 하는 경우 잘 정의된 워크플로우 마이닝 구조는 각각의 워크플로

우 시스템에 적용여부를 결정 할 수 있다.

- ▶ 워크플로우 마이닝 시스템에서의 공통적인 개발항목과 고려사항의 정의는 향후 워크플로우 마이닝 시스템의 상호운용성(Inter Operability)을 수행할 수 있다.

워크플로우 마이닝 프레임워크를 위하여 그림 4에서 기술된 항목은 워크플로우 마이닝을 구현하기 위해서는 기본적으로 고려해야 할 필요한 항목들이다. 특히 *Conceptual Level*의 구조는 워크플로우 마이닝을 구현하는데 워크플로우 마이닝의 정책적인 부분과 기술적인 부분을 고려하는 단계로서 이러한 이유로 가장 중요한 부분이고, 고려를 많이 해야 할 부분이다.



(그림 4) 워크플로우 마이닝을 위한 구조적 아키텍처

#### 3.1 Generic 단계 고려사항

*Generic* 단계에서는 워크플로우 마이닝 아키텍처의 전체적인 관점에서 공통적으로 고려되어야 할 사항에 대하여 정의되어 있다. 예를 들어 워크플로우 마이닝의 구현하는 경우 워크플로우 마이닝이 실행될 위치를 결정한다든지, 실행될 경우 사용되는 원시 데이터 즉, 실행정보는 어디에 위치하는지 그리고 언제 실행을 할 것인지에 대한 사항을 정의한다.

*Generic* 단계 아키텍처는 그림 5와 같이 계층적 워크플로우 마이닝 아키텍처 프레임워크의 최상위 단계와 같이 3차원 공간을 이용하여 3×3×3의 큐브 형태

로 되어있는 27개의 박스로 표현할 수 있다. Script (Execution)은 워크플로우 마이닝의 실행모듈의 위치를 의미하고 있다. 또한 Data는 워크플로우 마이닝에서 사용될 데이터로서 각 워크플로우 엔진의 실행정보를 원시데이터로 사용하여 마이닝을 수행한다. 마지막으로 Control은 워크플로우 마이닝의 실행 시점을 의미하는 부분으로 실행구동을 시키는 정보가 어디에 존재하냐에 따라서 구분될 수 있다.

### 3.2 Conceptual 단계 고려사항

그림 5의 계층적 구조에서 두 번째 단계에서 표현된 것과 같이 Conceptual 단계는 Generic 단계의 추상적인 정의를 구체화된 내용을 정의하고 있다. Conceptual 단계에서는 Generic 단계에서 정의된 하나의 인스턴스에 대한 정의라고 할 수 있다.

워크플로우 마이닝 아키텍처 프레임워크에서 가장 핵심적인 부분으로서 워크플로우 엔진에서 발생된 실행정보(Row Data)로부터 어떤 데이터를 이용하여 마이닝 할 것인지, 또는 마이닝의 목적이 무엇인지, 그리고 워크플로우 마이닝 엔진이 워크플로우 엔진과 비교하여 어디에 위치하는지 등과 같은 내용을 정의한다.

이러한 Conceptual 단계 아키텍처는 그림 5에서 표현한 것과 같이 5×3×4의 큐브 형태로 정의되고 각 축에서 고려해야 할 내용은 다음과 같다.

Mining Component 는 워크플로우 마이닝 시스템을 통하여 처리해야 할 대상을 결정하는 부분이다. 즉, 워크플로우 엔진의 실행정보에서 어떤 정보를 이용 할 것인지에 대한 부분으로 예를 들면 아래와 같다.

- ▶ 엔진에서 발생된 instance의 실행시간 및 엔진에서 생성된 instance 수
- ▶ Actor와 Role의 참여 빈도
- ▶ Data의 사용 빈도 및 유형
- ▶ Activity의 실행시간 및 각 Activity 간의 의존성 분석

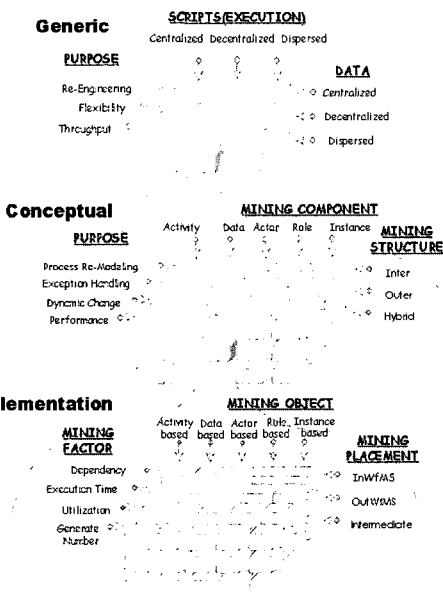
Purpose에서 고려될 사항은 워크플로우 마이닝의 실행목적이 고려될 수 있다. 즉 현재 사용되고 있는 프로세스의 개선을 위한 목적 인지 또는 동적변경을 위한 마이닝 자료의 생성 인지 또는 예외처리를 위한 마이닝 자료 생성 등과 같이 워크플로우 마이닝의 수행 목적에 따라 선택될 수 있다.

Mining Structure는 워크플로우 관리 시스템과 워크플로우 마이닝 시스템과의 관계를 의미한다. 즉, 워크플로우 관리시스템의 내부에 워크플로우 마이닝 시스템이 존재할 수 있고, 또는 워크플로우 관리시스템과 별개의 시스템으로서 워크플로우 마이닝 시스템이 구현될 수 있고 또는 서로 혼합된 형태로서도 구현될 수 있다.

### 3.3 Implementation 단계 고려사항

이전의 두 단계는 워크플로우 마이닝 시스템에 대한 개념적인 부분에 대한 언급되어 있지만 Implementation 단계에서는 실제 워크플로우 마이닝 시스템을 구현함에 있어 고려되어야 할 사항에 대하여 기술하고 있다.

결국 워크플로우 마이닝 아키텍처 프레임워크는 현재 구현되어 있는 워크플로우 마이닝 시스템의 평가 및 특징을 찾아낼 수 있는 기준이 될 뿐만 아니라 새로운 워크플로우 마이닝 시스템을 구현하려 하는 개발자들에게 개발지침을 제공할 수 있다.



(그림 5) 계층 워크플로우 마이닝 아키텍처 프레임워크

## 4. 결 론

최근의 워크플로우 관리 시스템 제품들이 처리해야 할 비즈니스 프로세스의 복잡화(Complexity) 및 대규 모화(Very Large Scale), 매우 긴 실행시간(Very Long Execution Time), 관련 데이터의 증가(Increase Relevant Data) 그리고 고객의 요구가 증가됨에 따라 정의된 프로세스의 동적 변경 또는 예외(Exception)가 발생되었을 경우 처리할 수 있는 유연성(Flexibility)과 같은 기능을 제공하지 못하고 있다. 이와 같은 동적 변경 또는 예외처리를 위한 유연성과 관련된 문제는 비즈니스 프로세스의 정의시점과 실행시점의 정보의 불일치로 발생되고 있다. 이와 같은 문제를 해결하기 위하여 프로세스의 실행된 정보를 이용하여 프로세스 정의에 이용하는 방법이나, 실행 단계에서 정의된 프로세스가 변경되기 위해서는 프로세스의 흐름정의 및 실행된 모든 정보를 알기 위한 방법으로 워크플로우 마이닝이라는 개념이 필요하게 되었다.

이러한 이유로 본 논문에서는 워크플로우 분야에서 최근 주된 관심 분야인 워크플로우 마이닝의 기본 개념과 워크플로우 마이닝과 관련된 사전연구로서 프로세스 마이닝의 연구동향을 설명하였다. 또한 워크플로우 마이닝 아키텍처를 위하여 워크플로우 마이닝 아키텍처 프레임워크에 대하여 제시하였다. 또한 제안된 아키텍처가 워크플로우 관리 시스템에서 활용될 수 있는 방안을 제시하였다. 제안된 워크플로우 마이닝 아키텍처 프레임워크는 Generic-level, Conceptual-level, Implementation-level과 같이 계층적 구조로 이루어져 있으며, 각 level에서 고려되어야 할 부분들을 기술하였다.

향후 각 단계에서 구해진 정보를 이용하여 마이닝을 하기위한 알고리즘에 대해서도 논의가 되어야 할 것이며, 워크플로우 마이닝 시스템의 구현 부분에 대

해서도 고려해야 할 것이다.

## 참 고 문 헌

- [1] Nutt, G., Operating Systems, Addison Wesley, 1997.
- [2] Kim, K. "Architectures for Very Large Scale Workflow Management Systems. PhD Thesis. Computer Science Department, University of Colorado. May 1998.
- [3] A.J.M.M. Weijters, W.M.P van der Aalst, "Process Mining Discovering Workflow Model from Event-Based Data".
- [4] A.J.M.M. Weijters, W.M.P van der Aalst, "Rediscovering Workflow Model from Event-Based Data".
- [5] Frank Leymann and Dieter Roller, "Production Workflow: Concepts and Techniques", 2000 published by Prentice-Hall, Inc.
- [6] R.Agrawal, D.Gunopulos, and F.Leymann. "Mining Process Models from Workflow Logs", in Proc. of the sixth International Conference on Extending Database Technology (EDBT), 1998.
- [7] Clarence A.Ellis and Gary J.Nutt, "Modeling and Enactment of Workflow Systems." In Application and Theory of Petri Nets 1993, Proceedings 14th International Conference, Chicago, Illinois, USA, pages 1-16. Springer-Verlag, 1993.
- [8] John Shafer, Rakesh Agrawal, and Manish Mehta, "SPRINT:A scalable parallel classifier for data mining", the VLDB Conference, Bombay, India, Sep,1996.
- [9] Minos N. Garofalakis, Rajeev Rastogi and Kyuseok Shim, "SPRIT : Sequential Patern Mining with Regular Expression Constraints", the VLDB Conference, Edinburgh, Scotland, UK, 1999.

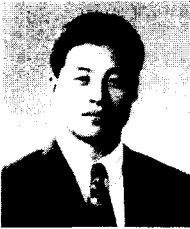


● 저자 소개 ●



**김 학 성**

1993년 경기대학교 전자계산학과(학사)  
1995년 경기대학교 대학원 전자계산학과(석사)  
2003년 경기대학교 대학원 전자계산학과(박사)  
1998년~현재 : 동남보건대학 컴퓨터응용과 교수  
관심분야 : Workflow Mining, CSCW, Workflow Exception Handling



**김 형 목**

1996년 경기대학교 전자계산학과(학사)  
1998년 아주대학교 대학원(석사)  
2000년~현재 : 경기대학교 대학원 전자계산학과 박사과정



**김 광 훈**

1984년 경기대학교 전자계산학과(학사)  
1986년 중앙대학교 대학원 전자계산학과(석사)  
1994년 콜로라도대학교 대학원 컴퓨터과학과(석사)  
1998년 콜로라도대학교 대학원 컴퓨터과학과(박사)  
1998년~현재 : 경기대학교 정보과학부 교수  
관심분야 : Workflow, Groupware, CSCW, Database, 분산처리기술