

워크플로우 마이닝을 위한 제어 경로 분석 메커니즘

A Control Path Analysis Mechanism for Workflow Mining

민준가*
Jun Ki Min

김광훈**
kwang Hoon Kim

정중수***
Jung Su Chung

요약

본 논문에서는 워크플로우 제어경로 분석 메커니즘을 제안한다. 이는 워크플로우 및 비즈니스 프로세스 기술의 활성화와 더불어 이의 적용사례가 급속히 늘어나면서 워크플로우 및 비즈니스 프로세스의 추적성과 재발견성을 최대화시키고자 최근에 이슈화되고 있는 워크플로우 마이닝 또는 프로세스 재발견 기법으로 활용될 수 있다. 특히, 본 논문에서는 제안하는 메커니즘은 두 가지 주요요소로 구성되는데, 하나는 워크플로우의 제어경로 분석을 통해 제어경로 결정 트리를 생성하는 부분이며, 다른 하나는 워크플로우의 모니터링과 실행 로그 정보로부터 워크플로우의 제어 경로를 기반으로 하는 워크플로우 실행 이력을 마이닝하는 부분이다. 결과적으로, 이 메커니즘을 통해 습득된 워크플로우 제어경로 기반 재발견 지식과 실행 이력 정보는 워크플로우의 제어경로 관련 고급정보를 구축하는데 이용될 수 있을 뿐 만 아니라 최종적으로 해당 워크플로우의 품질을 고급화시키기 위한 리엔지니어링의 주요 기반정보로 활용될 수 있다.

Abstract

This paper proposes a control path analysis mechanism to be used in the workflow mining framework maximizing the workflow traceability and re-discoverability by analyzing the total sequences of the control path perspective of a workflow model and by rediscovering their runtime enactment history from the workflow log information. The mechanism has two components. One is to generate the total sequences of the control paths from a workflow mode by transforming it to a control path decision tree, and the other is to rediscover the runtime enactment history of each control path out of the total sequences from the corresponding workflow's execution logs. Eventually, these rediscovered knowledge and execution history of a workflow model make up a control path oriented intelligence of the workflow model, which ought to be an essential ingredient for maintaining and reengineering the quality of the workflow model. Based upon the workflow intelligence, it is possible for the workflow model to be gradually refined and finally maximize its quality by repeatedly redesigning and reengineering during its whole life long time period.

☞ Keyword : Web Integration, Web Management Tool

1. 서론

최근 워크플로우 기술은 사무 정보 시스템과 데이터베이스 시스템 분야에서 관심의 대상이 되고 있다. 워크플로우는 정상적으로 액티비티의 집합으로 이루어진 비즈니스의 절차를 기술한 것이다. 각각

의 액티비티는 워크플로우 엔진의 제어를 통해 자동적으로 수행될 수 있다. 또한, 워크플로우 관리 시스템은 사전의 분석과 요약, 정보 중심의 작업과 액티비티의 자동화에 대한 도구의 집합이다. 최근 이러한 워크플로우 개발 환경의 동향을 살펴보면[10], 두 가지 현상으로 요약할 수 있다. 하나는 네트워크에서의 개인 컴퓨팅 환경이 강화되고 있다는 점이고, 다른 하나는 워크플로우 응용 분야가 더욱 더 대형화 되고 복잡해 지고 있다는 점이다. 또한 워크플로우 시스템의 기술은 이러한 동향에 맞추어 변화되고 있다. 즉, 일반적으로 워크플로우 시스템은 제어, 기술, 데이터 조작 기술에서 분산 구조를

* 정 회 원 : TTA

jakimin@tta.or.kr(제1저자)

** 종신회원 : 경기대학교 정보과학부 부교수

kwang@kyonggi.ac.kr

*** 정 회 원 : 안동대학교 전자정보산업학부 정교수

jschung@andong.ac.kr

[2005/06/02 투고 - 2005/06/20 1차 심사 - 2005/08/20

2차 심사 - 2005/09/30 심사완료]

가지며 개념적으로 내포된 워크플로우 모델의 구체화에 의해 설계되고, 물리적으로는 객체 지향 방법론에 의해 구현된다. 또한, 전적으로 웹 기반 환경하에 동작을 한다.[5,7-9]

더욱이 최근에 와서 B2C(Business-To-Customer)/B2B(Business-To-Business)로 대변되는 전자상거래 및 전자시장(E-Market Place)의 활성화가 급속하게 확장됨에 따라, 워크플로우를 기반으로 한 기업 내부 업무 흐름의 자동화를 의미하는 B2E(Business-To-Enterprise)의 구축을 더욱 가속화시키고 있으며, 워크플로우 기술 및 시스템은 결국 기업 또는 조직체내의 모든 업무 처리 절차 (business procedure)들을 통합 관리하는 인프라구조(Infrastructure)로 인식되고 있다. 이렇게 변화되는 환경 하에서 워크플로우의 중요한 요소 중에 하나인 워크플로우 모니터링 역시 변화가 요구된다. 기존의 워크플로우 모니터링의 기능은 단순히 워크플로우의 상태 정보나 감시 정보만을 제공하는 것이었다. 그러나 전자상거래에서 모니터링의 역할은 워크플로우를 분석하여 기업이나 고객에게 또 다른 정보를 제공할 수 있어야 한다. 이에 본 논문은 워크플로우의 모니터링 기능을 확장하여 워크플로우를 정의하는 부분에서만 제공하였던 워크플로우에 대한 분석을 기존의 모니터링 정보를 바탕으로 하는 워크플로우 마이닝의 기법에 대하여 기술하고, 이를 위해서는 추가되는 모니터링 정보와, 마이닝의 단계 및 시점을 정의하고자 한다.

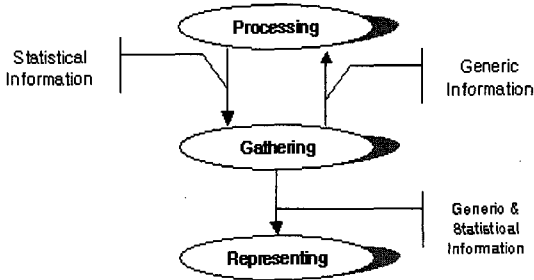
결과적으로, 본 논문의 주요 내용 및 목표는 정의된 워크플로우 및 비즈니스 프로세스를 분석하고, 이로부터 제어경로 결정 트리로 변형하기 위한 알고리즘을 제시하는데 있다. 이의 실현과정을 기술하는 순서로는 2장에서 관련 연구로서 워크플로우 시스템과 마이닝과의 관계를 고급 워크플로우 모니터링 서비스의 정의와 함께 모델링 관점과 시스템의 모니터링 및 감사 정보와의 마이닝과의 관계를 정의한다. 3장에서는 모니터링 정보를 바탕으로 하는 워크플로우 제어경로 분석 메커니즘을 설계하고, 이를 워크플로우 마이닝 기법에 적용할 수 있는 방안을 제시한다.

2. 워크플로우 시스템과 마이닝 메커니즘의 관계

2.1 고급 워크플로우 모니터링 서비스

일반적으로 워크플로우 모니터링[6]이라 함은 워크플로우 시스템에서 일어나는 이벤트를 추적하고, 기록하는 능력을 말한다[2]. 워크플로우 모니터링 서비스는 시스템 수준의 모니터링부터 사용자 수준 혹은 응용 프로그램 수준의 모니터링 서비스까지 매우 다양하다. 전통적인 워크플로우 시스템[8,9]에서는 사용자 수준의 모니터링 서비스를 제공하여 왔지만, 최근에는 워크플로우 시스템의 대형화 추세에 따라 시스템 수준의 모니터링과 통계정보 수준의 모니터링을 중심으로 하는 고급 모니터링 서비스의 필요성이 요구되고 있다. 즉, 시스템 수준의 전형적인 예를 들면 실행 요소들 즉, 분산 엔진등의 수행도나 적재량을 모니터링하는 것이다. 워크플로우 인스턴스와 관련되어 각각의 액티비티의 상태를 보여주는 서비스는 전형적인 사용자 수준의 모니터링이라 한다. 이러한 서비스들은 워크플로우 실행 객체 간의 동작을 감지하는 부분과 실행 시간의 정보들을 접근하는 부분, 어떠한 변화가 일어날 때 마다 알릴 수 있는 부분을 통해 제공될 수 있다. 결과적으로, 고급 워크플로우 모니터링 서비스는 Push형태와 Pull형태로 나눌 수 있다. 일반적으로 시스템 수준의 모니터링 서비스는 Push형태로, 사용자 수준의 모니터링 서비스는 Pull형태로 서비스를 제공한다[6].

그림 1은 모니터링 정보 처리부, 모니터링 정보 수집부, 모니터링 정보 표현부로 이뤄진 모니터링의 기능과 그들간의 상호 정보 교환을 설명한 것이다. 정보 처리부는 수집된 정보중 처리가 요구되는 정보를 처리하거나 분석하는 역할을 담당한다. 예를 들어 한 워크플로우에 대한 통계 정보를 요구할 때에는 처리부에서 이를 처리하여 서비스를 하게 된다. 모니터링 정보 수집부는 워크플로우 실행부에게 정보를 요청하고, 상태정보나 이벤트 정보를 반환 받는 인터페이스를 가진다. 마지막으로, 정



〈그림 1〉 고급 워크플로우 모니터링의 기능

보 표현부는 모니터링 서비스를 받는 곳에서 쓰인다. 수집되거나 처리된 정보를 사용자에게 보다 인식되기 쉬운 방법으로 표현하는 역할을 담당한다. 예를 들어 통계정보의 표현은 단순한 문자 보다는 그래프 형태가 사용자에게는 인식하기가 쉽다. 이를 구체적으로 정의하면 다음과 같다.

· Processing

일반적으로 사용자는 통계정보와 같은 요약된 정보를 요구한다. 그래서, 수행부에 있는 인스턴스는 변화가 일어날 때 마다 그들의 상태나 이벤트를 특정 저장소에 저장을 해야한다.

· Gathering

이 특성은 트랜잭션 워크플로우 모니터링 시스템의 중요하면서 가장 기본이 되는 기능을 제공한다. 수집부는 수행중이거나 수행이 종료된 워크플로우에 대한 특정 정보의 집합을 수집한다. 동시에 수집부는 거대한 모니터링 정보를 처리부에 전송을 한다. 수집된 정보는 시스템, 워크플로우 인스턴스, 액티비티 인스턴스, 사용자등과 같이 분류할 수 있다.

· Representing

사용자는 복잡하고 거대한 정보를 좀더 정리되고 요약된 정보를 더 쉽게 이해할 수 있는 방법을 원한다. 예를 들어 마이크로 소프트의 윈도우 익스플로러와 같은 트리 형태로 인스턴스들을 리스트화하거나 통계정보에 대해서는 막대그래프를 이용하여 표현하는 것을 말한다. 표현부에서 가장 중요하게 다루어야 하는 부분은 워크플로우 인스턴스에 대한 흐름을 보여주는 것이다. 이것은 플로우 다이어그램과 같은 형태로 보여 주어야 하고, 그 안에는 상태 및 진행상황을 나타내야 한다.

2.2 워크플로우 모델링과 마이닝

워크플로우 및 비즈니스 프로세스(이하 워크플로우)를 모델링하는 것은 비효율적인 프로세스, 시간 소비와 비용을 고려하는 것이다. 모델링이 워크플로우의 현재 구조를 이해하기 위해 재 설계하는 부분으로 사용되거나 새로운 워크플로우를 정의하는 부분으로 사용되는 이점을 갖지 못한다. 결과적으로 기업은 워크플로우 기술이 추구하는 이익을 제공할 것이라는 확신 없이는 거대한 투자를 하지 않는다. 모델링 비용의 감소와 미리 결과를 얻을 수 있는 능력이 이상적일 것이다. 워크플로우를 다시 설계하는 것의 원초적인 문제와 적은 확장으로 새로운 워크플로우를 생성하는 것은 이를 수행하기 위해 워크플로우 관리 시스템으로 전송하기 전에 완벽한 모델을 정의하는 것이 필요하다. 워크플로우의 수행에 할당된 사용자들은 대개 그들이 수행해야 하는 작업에 관한 상세한 지식을 가진다. 그들은 작업에 관련된 툴과 툴에 입력되는 예상 데이터, 툴을 통해 나온 결과 등 작업에 관한 이해를 가진다. [1-3]

일반적으로, 워크플로우 모델러[1]는 적당한 사람들과의 면접을 통해 워크플로우의 부분인 각 작업에 대한 충분한 정보를 얻을 수 있다. 해당 워크플로우에 포함된 대부분의 사람들은 그들이 포함된 워크플로우의 전반적인 구조에 관한 지식이 거의 없다. 그들은 전형적으로 최소한 자신의 액티비티들을 인식하고, 따르는 액티비티를 알고 있다. 즉, 사용자들은 단지 워크플로우의 로컬정보만을 가지고 있다. 그래서, 존재하는 워크플로우의 액티비티들 간 순서를 결정하는 것은 도전과 시간을 필요로 하는 작업이다. 워크플로우에서 구성원과의 면접 같은 기술들은 로컬 정보를 수집하는데 사용되어야 한다. 이러한 로컬 정보는 워크플로우의 전반적인 구조를 포함하는 포괄적인 정보로 옮기고, 작업들의 전체적인 순서를 정한다. 워크플로우에서 액티비티의 순서를 발견하는 것은 반복되는 프로세스가 된다. 조합된 정보는 몇 번이고 포함된 사람에 의해 토의되고, 검증될 필요성이 있다. 심지어 올바른 액티비티의 순서가 결정이 되었다 할지라도, 한 액티비티에서 다음 액티비티로의 제어 흐름과 관련된

비즈니스 규칙은 오류가 있을 수 있다. 우리는 종종 비즈니스 전문가도 어떤 도움없이 전이 조건을 명세 한다는 것은 어려움이 있다. 본 논문에서는 이러한 워크플로우를 정확히 정의하기 위해 워크플로우 마이닝[10]의 개념을 도입하였다.

2.3 워크플로우 시스템의 실행 모니터링 및 감사 정보와 마이닝

워크플로우의 경향을 분석하기 위해 워크플로우 시스템의 모니터링 및 감사 정보 로그[2]에 대해 자세히 살펴본다. 워크플로우 시스템은 그림 2과 같이 모니터링 및 감사 정보를 로그한다. 정의된 워크플로우는 워크플로우 관리 시스템으로 인스턴스 하나를 발생한다. 발생된 인스턴스는 각 액티비티를 수행하기 위해 수행자에게 작업을 할당하고 할당된 작업이 종료되면 다시 시스템으로 그 정보를 넘겨준다. 이것을 반복해서 한 인스턴스의 마지막 액티비티가 종료되면 해당 워크플로우 인스턴스는 종료된다. 그런데, 이 도중에 수행자는 시스템은 저장소에 현재 진행되고 있는 인스턴스에 대한 감시 추적 정보를 저장하게 된다. 이러한 정보는 수행자가 수행 도중에 현 워크플로우에 대한 정보를 얻고자 할 때 모니터링을 이용하여 얻을 수 있고, 워크플로우 디자이너나 시스템 관리자는 전체 워크플로우에 대한 추적 정보를 요구할 수도 있다. 워크플로우 디자이너에게 유용하게 사용될 데이터를 제공하기 위해서는 워크플로우 마이닝[10]

을 통해 워크플로우의 경향을 분석하여 이상적인 워크플로우를 정의할 수 있도록 설계를 하여야 한다.

3. 워크플로우 제어 경로 분석 메커니즘

3.1 요구 데이터

워크플로우 마이닝을 위한 데이터는 모니터링 정보 즉 추적 감시 정보만으로 충분치 않다[2,6]. 추적 감시 정보만으로 워크플로우의 경향을 찾기 어렵기 때문에 본 논문에서 워크플로우 관리 시스템에서 끌어올 데이터를 두 가지의 형태로 나누었다. 그것은 내부적인 데이터와 외부적인 데이터[6]이다.

· 내부 데이터

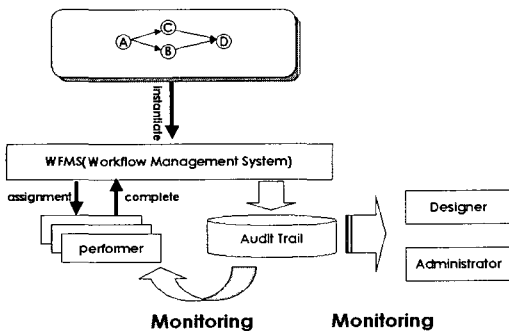
내부 데이터는 워크플로우의 데이터 흐름 정보를 말하는데, 이것은 워크플로우 내부에서 일어나는 관련 데이터(Relevant Data)를 의미한다. 내부 데이터는 워크플로우 마이닝을 위해서 필수적으로 요구되는 데이터는 아니다. 이 데이터는 주로 고객의 정보와 상품의 정보를 나타낸다. 따라서 모니터링은 고객의 나이, 성별등의 정보와 상품정보를 별도로 저장하는 것이 필요하다. 또한, 저장소에 저장시 데이터를 그대로 저장하는 것이 아니라, 나이와 상품, 성별과 상품 등의 연관 관계를 정의하여 저장해야 한다.

· 외부 데이터

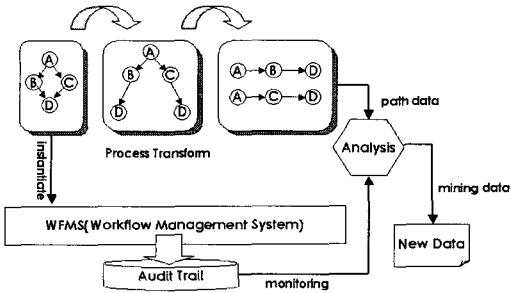
외부 데이터는 워크플로우의 프로세스 흐름 정보를 말하는데, 이것은 워크플로우 외부에서 일어나는 전이 정보(Transition Information)를 의미한다. 마이닝을 위해서는 고객이 어떠한 워크플로우의 경로를 선택하였는지가 중요하고, 이것을 모니터링해야 한다. 이는 워크플로우가 시작하기 전에 모든 경로에 대한 정보를 저장하여, 어떠한 경로를 통해 워크플로우가 진행이 되었는지에 대해 정보를 수집하고 저장한다.

3.2 워크플로우 마이닝의 단계

워크플로우에서 발생하는 데이터를 바탕으로 워크플로우에 관련한 새로운 데이터를 발생하는 것이



〈그림 2〉 워크플로우 시스템의 실행 모니터링 및 감사 정보 로그



〈그림 3〉 워크플로우 마이닝 단계

다. 이를 위해 앞장에서 정의된 데이터를 바탕으로 알고리즘을 통해 새로운 데이터를 생성하고, 워크플로우 로그 정보와 비교하여 워크플로우에 관련된 새로운 데이터를 생성한다.

그림 3은 워크플로우 마이닝의 전체적인 과정[6]을 보여주고 있다. 이를 구체적으로 단계별로 나열하면, 다음과 같다.

- 1단계(워크플로우의 변형): 정의된 워크플로우의 분석을 위한 형태로 변형을 한다.
- 2단계(워크플로우 모니터링): 워크플로우가 진행됨에 따라 감시 추적 정보를 저장소에 저장한다.
- 3단계(워크플로우의 비교 분석): 정의 정보와 수행 정보를 비교하여 통계를 추출한다.
- 4단계(데이터 추출): 고객의 취향, 워크플로우에 대한 경향을 분석하여 새로운 데이터를 추출한다.
- 알고리즘 1 : 처음 시작 액티비티는 트리의 루트 노드로 구성한다. 그리고 다음 액티비티를 검색한다. 다음 액티비티는 워크플로우의 전이 정보(Transition Information)을 이용하여 검색한다.
- 알고리즘 2 : 일반 액티비티나 집합 액티비티는 노드로 구성하지 않고, 다음 액티비티를 검색한다.
- 알고리즘 3 : 분기 액티비티는 분기 수 만큼 수행을 하게 되는데, 먼저 분기 액티비티를 노드로 구성을 하고, 그 노드의 부모 노드를 기존의 노드로 구성한다.
- 알고리즘 4 : 끝 액티비티는 현재의 트리를 반환한다.

〈표 1〉 노드 구조체

Node {		
String	name	//node의 이름
Node	parent	//부모노드
NodeSequence	children	//자식 노드의 집합
} // 구조체		

3.3 워크플로우 모델 변형

워크플로우 모델의 변형은 빌드 타임에서 정의된 워크플로우의 흐름을 마이닝하기 위한 형태로 변형하는 것이다. 마이닝의 목적은 워크플로우의 경향을 찾아내는 것이다. 이는 모든 워크플로우의 경로를 찾아내는 것이 중요하다. 따라서 정의된 워크플로우의 모든 경로로 나열하고자 한다. 그러나 워크플로우 자체를 직접 경로별로 나열하기가 쉽지

〈표 2〉 트리 변형 알고리즘

<pre> ProcessToTree (Input Activity A, Node N) begin if A is start Activity // 알고리즘 1 begin set A to N N is ProcessToTree(Next, N) else if A is normal Activity or join Activity // 알고리즘 2 begin N is ProcessToTree(Next, N) end else if A is split Activity // 알고리즘 3 begin for 1 to the number of split do begin set Nextto N1 parent of N1 is N N1 is ProcessToTree(Next of Next, N1) add N1 to children of N end end else if A is end Activity // 알고리즘 4 begin return N end return N end </pre>

않다. 따라서 워크플로우를 트리 형태로 바꾸고, 트리의 한 가지를 한 경로로 찾는 변형(Transformation)을 하고자 한다.

· 워크플로우 모델 변형 단계

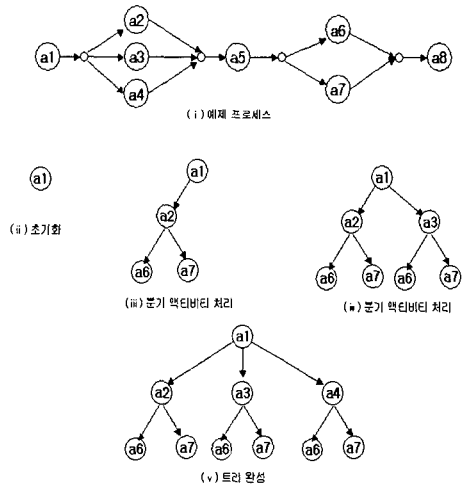
- ① 전이 정보를 이용하여 워크플로우로 나타내며,
- ② 워크플로우를 트리 형태로 변형하고,
- ③ 트리 형태를 루트 노드(root node)에서 최종 노드(leaf node)까지 모든 경로를 나열하며,
- ④ 저장소에 저장한다.

트리는 표 1과 같이 노드 구조로 표현되고 노드는 노드의 이름, 부모 노드, 자식 노드로 구성된다. 빌드 타임에 정의된 워크플로우는 일정한 전이 정보를 바탕으로 저장소에 저장되고, 이를 이용하여 현재 마이닝 할 워크플로우를 나타낸다. 이것을 입력으로 표 1의 트리 변형 알고리즘을 통해 트리로 변형을 시킨다.

워크플로우 모델에 나타나는 액티비티의 종류는 시작 액티비티, 마지막 액티비티, 일반 액티비티, 분기 액티비티, 집합 액티비티이다. 따라서 트리를 구성하기 위하여 이 액티비티의 종류에 따라 법칙을 세워야 한다. 시작 액티비티는 트리의 시작을 의미하므로, 항상 루트를 구성하고, 마지막 액티비티는 최종노드(leaf node)를 구성하게 된다. 일반 액티비티는 경로를 찾기 위해서는 불필요하고, 또한 트리의 복잡도를 줄이기 위해 생략한다. 분기 액티비티는 경로를 결정하는 중요한 액티비티이다. 분기 액티비티의 차수는 분기되는 경우의 수를 의미하며, 분기수 만큼 반복하여 끝 액티비티까지 검색하고, 트리를 구성한다. 트리의 변형이 끝나게 되면, DFS(Depth First Search) 알고리즘을 통해 루트 노드에서 최종 노드까지의 경로를 찾아 낸다. 경우의 수는 최종 노드의 개수이며, 경우의 수 만큼 경로를 나열한다.

그림 4 워크플로우 변형 예제에서는 모두 6개의 경로가 나오고 DFS를 통해 {a1, a2, a6}, {a1, a2, a7}, {a1, a3, a6}, {a1, a3, a7}, {a1, a4, a6}, {a1, a4, a7}를 찾을 수 있다. 이를 예제를 통하여 자세히 알아보도록 한다. 그림 4 에서 (i)는 예제 워크플로우다. 분기 액티비티가 2개이고, 집합 액티

비티가 2개인 워크플로우다. 이것을 위에서 제시한 알고리즘을 통해 트리로 만들려면, 먼저 워크플로우의 시작점을 찾아 입력값으로 한다. 예제 워크플로우에서는 a1이 시작 액티비티 이므로 시작 액티비티를 노드로 구성하고 다음을 검색한다. 다음 액티비티는 분기 액티비티 이므로, 분기 차수 만큼 순환을 하는데, 먼저 a2를 a1의 자식 노드로 구성하고, 다음 액티비티를 검색한다. 다음은 집합 액티비티 이므로 통과하고, 다음은 일반 액티비티이므로 이또한 통과한다. 다음 액티비티는 집합 액티비티이다. 따라서 a2의 자식노드로 a6를 구성하고, 다음 액티비티를 검색한다. 그리고 끝 액티비티이므로, 처음 수행을 마친다. 두번째 분기 액티비티에서 다음 경로를 찾지 못했기 때문에 a7을 a2의 자식노드로 구성한다. 이렇게 하면, 그림 에서 (iii)과 같이 된다. 이것을 반복하면, (iv)(v)그림 같이 변형을 하게 된다.



〈그림 4〉 워크플로우 모델 변형 예제

분석의 단계는 먼저 워크플로우 모니터링의 추적 정보에서 경로를 찾아내는 것이다. 이 경로를 변형된 경로 정보와 비교를 하게 된다. 추적 정보에서 경로 정보를 찾아 내는 시점은 한 워크플로우 인스턴스가 종료되는 시점이다. 만약, 일정한 시점에 그 정보를 찾아 낸다면 방대한 자료이기 때문에 찾는 속도가 떨어질 뿐 아니라, 알고리즘도 복잡할

것이다. 고로 본 논문에서 제안하는 방법은 한 워크플로우가 끝나는 시점으로 정하여 워크플로우가 종료되면 표 3에서 보는 것과 같은 형태로 저장하고, 이 정보를 워크플로우 인스턴스 중심으로 질의를 하여 다음과 같은 과정을 통하여 결과를 얻는다.

- 종료된 워크플로우의 이름으로 모든 액티비티와 시간순서 항목을 질의한다.
- 이 정보를 시간 순서별로 나열한다.
- 같은 액티비티가 연속으로 존재할 경우 하나의 경로로 간주한다.
- 같은 액티비티가 연속적이지 않을 경우, 워크플로우 변형 알고리즘을 통해 트리로 변형한 후 경로를 찾는다.
- 찾은 경로와 먼저 정의된 경로를 비교한다.
- 해당 경로의 데이터베이스에 누적한다.

3.4 워크플로우 실행 모니터링

워크플로우 실행 모니터링은 해당 워크플로우 모델의 실행이 진행 됨에 따라 감시 추적 정보를 저장하는 것을 말한다. 워크플로우의 인스턴스가 진행된다는 의미는 각 액티비티가 수행을 시작한다는 것을 의미한다. 이 액티비티가 수행이 되고 있는 지를 모니터링 하는 것이 추적 정보이고, 워크플로우 마이닝을 위해 이러한 추적 정보를 별도로 저장하는 것이 필요하다. 추적 정보는 별도의 저장소에 저장을 하고 분석을 위해 사용된다.

〈표 3〉 추적정보 저장형태

워크플로우	워크플로우 인스턴스	액티비티	시간 순서	비고
P	P1	A	1	
P	P2	B	25	
P	P3	C	3	

표 3의 저장 형태는 워크플로우, 워크플로우 인스턴스, 액티비티, 순서 등으로 구성된다. 워크플로우는 빌드타임에서 정의된 식별자이고, 워크플로우 인스턴스는 엔진에 실행 중인 또는 수행이 종료된 인스턴스이다. 액티비티는 워크플로우에 해당하는

것으로 인스턴스로 표시하지 않고, 원래의 액티비티로 표시한다. 이것은 워크플로우 인스턴스에서 구분을 하기 때문이다. 마지막으로 시간 순서는 해당 액티비티가 수행된 시각을 의미한다. 이것을 구분하는 기준은 시간 클럭이다. 일정한 클럭을 두어 현재 진행 중일 때 해당 클럭을 기록하게 된다. 예를 들어 P1인스턴스의 A 액티비티가 1클럭에 수행을 하였다면 1로 기록되고, B액티비티가 2에서 5 클럭 까지 수행을 하였다면 2,5로 기록된다. 그러나 C액티비티가 3클럭에 수행을 하였다면 3으로 기록된다. 여기서 추측할 수 있는 정보는 B액티비티와 C액티비티는 동시에 수행되는 병렬의 형태라는 것을 알 수 있다. 이러한 정보는 분석 단계에서 이루어 진다.

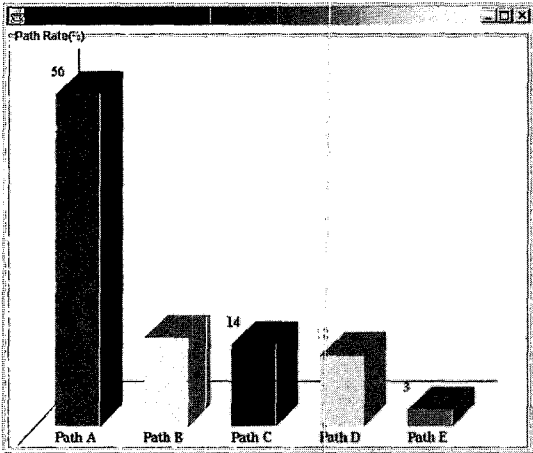
3.5 워크플로우 모델 분석

워크플로우 모델 분석은 워크플로우 모델 변형 및 모니터링 단계를 통해 얻어진 결과를 토대로 두 정보를 비교 분석하는 단계이다. 워크플로우 모델 변형에서 모든 경로에 대한 정보, 워크플로우 실행 모니터링에서 추적 정보를 분석하는 것이다. 이러한 정보를 전자상거래에 적용을 하면, 앞에서 제시한 요구데이터 중 내부 데이터를 경로 정보에 고객의 신상에 관한 정보를 포함 시켜야 한다. 예를 들어 고객의 나이, 성별, 주문 상품 등을 저장 한다면, 나이, 성별, 주문 상품별로 어떠한 경로를 주로 이용을 하였는지에 대한 분석 정보가 추출될 것이다. 이것은 궁극적으로 전자상거래의 CRM(Customer Relationship Management)을 위한 데이터로 충분하다. 본 논문의 범위는 모니터링 시스템에 국한 되어 있으므로 CRM을 위한 데이터 추출이 목표이다.

3.6 데이터 추출

데이터 추출은 분석된 정보를 표현하는 부분으로서 워크플로우 모델 변형, 워크플로우 실행 모니터링, 워크플로우 모델 분석 단계들을 거쳐 워크플로우의 경로에 대한 경향을 파악한다. 그림 5는 경로에 대한 비율을 그래프의 형태로 나타낸 예제로

서, 1000개의 워크플로우 인스턴스를 수행한 결과를 보여주는 것이다. 그림에서 보듯이 A경로를 통해 현저히 많은 양의 작업들이 처리가 되었다. 반면에 E경로는 단지 30개의 인스턴스가 수행이 되었다. 따라서 해당 워크플로우 모델을 모델링하는 사람은 이 정보를 보고 E경로에 대해서 다시 고려를 해 봐야 한다.



〈그림 5〉 데이터 추출의 예

4. 결 론

워크플로우 및 비즈니스 프로세스 기술의 활성화와 더불어 이의 적용사례가 급속히 늘어나면서 최근에는 워크플로우 및 비즈니스 프로세스의 품질 개선과 리엔지니어링을 위해서 워크플로우 마이닝 또는 프로세스 재발견 기법에 대한 연구개발이 중요하게 대두되고 있다. 워크플로우를 정의하는 것은 프로세스를 분석하고, 워크플로우 시스템에 적용할 수 있는 형태로 바꾸는 것이다. 그러나 정확한 정의를 하지 않으면 치명적인 기업의 손해를 일으킬 수 있다. 아무리 잘 정의한 워크플로우라도 실행 중에 일어나는 것은 예측할 수가 없을 뿐 만 아니라 시간의 경과됨에 따라 리엔지니어링의 필요성이 증가하게 된다. 따라서, 본 논문에서 제시한 제어경로 분석 메커니즘은 이러한 워크플로우 재발견 및 리엔지니어링을 수행하기 위한 워크플로우 마이닝 프레임워크에 매우 효과적으로 적용될 수

있을 것으로 기대된다. 결과적으로, 본 논문에서는 워크플로우의 모니터링과 실행 로그 정보로부터 워크플로우의 제어 경로를 기반으로 하는 워크플로우 실행 이력을 마이닝하므로서 해당 워크플로우 및 비즈니스 프로세스의 행위와 품질 경향을 분석하고 개선할 수 있는 워크플로우 마이닝 기법에 대한 연구개발의 첫걸음을 내디디는 시도라는 면에서 매우 의미 있는 기여라고 판단된다.

참 고 문 헌

- [1] Workflow Management Coalition Specification Document, "The Workflow Reference Model," Version 1.1, November 1994
- [2] Workflow Management Coalition Specification Document, "Workflow Management Coalition Audit Data Specification," Version 1.1, Document Number: WFMC-TC-1015, September 1998
- [3] Nortel & University of Newcastle upon Tyne, "Workflow Management Facility Specification," Revised Submission, OMG Document Number: bom/98-03-01, 1998.
- [4] Joint Submitters, "Workflow Management Facility," Revised Submission, OMG Document Number: bom/98-06-07, July 4, 1998.
- [5] Kwang-Hoon Kim and Clarence A. Ellis, "A Framework for Workflow Architectures," University of Colorado/Department of Computer Science, Technical Reports, CU-CS-847-97, Dec. 1997
- [6] H S Hong , B S Lee, K H Kim, S K Paik, "A Web-based Transaction Workflow Monitoring System," WISE 2000 in Hong Kong Proceeding , IEEE PRESS
- [7] Rakesh Agrawal, Dimitrios Gunopulos "Mining Process Models From Workflow Logs," IBM Almaden Research Center, San Joes, CA, USA
- [8] Thomas M. Koulopoulos, "The Workflow

Imperative,” published by Van Nostrand, a division of International Thomson Publishing Inc, 1995

[9] Frank Leymann, Dieter Roller, “Production Workflow Concepts and Techniques,” Prentice Hall PTR 2000

[10] Kwang-Hoon Kim and Clarence A. Ellis, “Workflow Reduction for Reachable-path Rediscovery in Workflow Mining,” Foundations and Novel Approaches in Data Mining, pp. 289-310, Springer-Verlag, 2005

◎ 저자 소개 ◎



민준기(Jun Ki Min)

1993년 2월 한남대학교 대학원 전자계산학과 (석사)
2003년 9월 현재 국립 안동대학교 정보통신공학 박사과정
1993년 2003 우송대, 대덕대, 혜천대, 경기대, 경기대 정보통신대학원 시간강사
1980년 8월 1999년6월 ETRI
1999년 7월 2001년5월 LOCUS
2001년 7월~현재 TTA
관심분야 : 데이터네트워크, 멀티미디어, BPM, Workflow, eBiz, etc.,
email: jakimin@tta.or.kr



김광훈(kwang Hoon Kim)

1986년 2월 중앙대학교 대학원 전자계산학과 (석사)
1994년 University of Colorado at Boulder, Computer Science, MS
1998년 University of Colorado at Boulder, Computer Science, Ph.D
1986년 2월 1991년8월 ETRI
1993년 5월 1994년8월 American Educational Products Inc. Professional DB Consultant
1994년 9월 1995년8월 Colorado Advanced Software Institute. Research Assistant
1995년 9월 1997년2월 Azt다 Engineering Inc, Software Engineer
1998년 3월~현재 경기대학교 정보과학부 부교수
관심분야 : 워크플로우, 그룹웨어, 컴퓨터네트워크, 데이터베이스,
email: kwang@kyonggi.ac.kr



정중수(Jung Su Chung)

1983년 2월 연세대학교 전자공학과 (석사)
1993년 8월 연세대학교 전자공학과 (박사)
1983년 3월 1994년 2월 ETRI
1987년 8월 1989년 8월 벨지움 Alcal/Bell Telephone사 객원연구원
2000년 1월 2001년 1월 미국 UMASS/Lowell 전산학과 객원교수
1994년 3월~현재 국립 안동대학교 공과대학 전자정보산업학부 정교수
관심분야 : 데이터네트워크, 멀티미디어, BPM, etc
email: jschung@andong.ac.kr