비즈니스 프로세스 분석을 위한 색채형 워크플로우 모델☆

A Colored Workflow Model for Business Process Analysis

정 우 진* 김 광 훈**
Woo-Jin Jeong Kwang-Hoon Kim

요 약

기업의 업무 활동은 수많은 프로세스로 이루어져 있고 기업들이 업무를 처리하는 모든 활동 속에서 수많은 비즈니스 프로세스가 동시다발적으로 생성되고 처리된다. 전사적 업무환경은 업무의 처리를 간편하게 만든 동시에 더욱 복합적인 업무 형태를 만들어내었고 이에 따라 비즈니스 프로세스의 효율적 관리가 더욱 절실하게 되었다. 이에 기업 및 조직은 프로세스에 대한 개선과 새로운 비즈니스의 발견을 목표로 프로세스를 분석하고 비즈니스와 관련한 지식 도출을 위한 연구 개발에 집중하게 되었다. 프로세스를 분석하고 개선하기 위한 연구들로써 프로세스 마이닝, 프로세스 재발견, BPR(Business Process Reengineering) 등의 개념들이 등장하였고 실제 적용을 위한 연구들이 활발히 진행 중에 있다. 하지만 이러한 연구들은 대부분 프로세스 인스턴스들의 이력 정보에 대한 데이터웨어하우징을 통한 접근법을 따르고 있다. 이러한 접근법은 점점 복잡화되는 최근의 기업 환경과 수시로 변화하는 사용자의 요구를 충족시키기에 부족하다. 분석적인 측면을 고려하여 설계된 프로세스 인스턴스들은 효과적인 그룹화가 가능하고, 분석 도메인 내에서 사용자의 분석 요구가 변화하게 되는 경우에도 분석 비용을 줄일 수 있다. 본 연구에서는 기존 프로세스 모델링 방법론인 ICN을 확장하여 효과적인 프로세스 분석과 비즈니스 관련 지식 도출을 목표로 색채형 워크플로우 모델을 정의한다. 프로세스 인스턴스들에 대한 실행 정보를 통해 수동적인 프로세스 분석이 아닌, 프로세스 정의 시점부터 비즈니스 지식 도출을 목표로 하여 적극적이고 사용자 중심적인 프로세스 분석이 가능하게 한다.

Abstract

Abstract Corporate activities are composed of numerous working processes and during the working flow, various business processes are being created and completed simultaneously. Enterprise Resources Planning (ERP) makes the working process simple, yet creates more complicated work structure and therefore, there is an absolute need of efficient management for business processes. The workflow literature has been looking for efficient and effective ways of rediscovering and mining workflow intelligence and knowledge from their enactment histories and event logs. As part of studies to analyze and improve the process, the concepts of 'Process Mining', 'Process re-discovery', 'BPR (Business Process Reengineering)' have appeared and the studies for practical implementation are proactively being done. However, these studies normally follow the approach throughout data warehousing for log data of process instances. It is very hard for these approaches to reflect user's intention to the rediscovering and mining activities. The process instances designed based on the consideration of analysis can make groupings effectively and when the analysis demand of user changes within the analysis domain can also reduce the cost of analysis. Therefore, the thesis proposes a special type of workflow model, which is called a colored workflow model, that is extended from the ICN (information control net) modeling methodology by reinforcing the concept of colored token. The colored tokens represent the conceptual types of constraints and criteria that can be used to classifying and grouping the workflow intelligence and knowledge extracted from the corresponding workflow models' enactment histories and event logs. Through the runtime information of process instances, it makes possible to analyze proactive and user-oriented process with the goal of deriving business knowledge from the beginning of process definition.

☞ keyword : workflow, BPM, process, model, 워크플로우,BPM, 프로세스, 모델

* 준 회 원 : 경기대학교 일반대학원 컴퓨터학과 석사과정 woojin@kgu.ac.kr
** 종신회원 : 경기대학교 정보과학부 교수 kwang@kgu.ac.kr

[2008/12/08 투고 - 2008/12/23 심사 - 2009/04/06 심사완료]
☆ 이 논문은 2008년도 GRRC (경기도지역협력연구사업) 지원에 의한 연구이었음(This work was carried out with financial support from the 2008 GRRC)

1. 서 론

기업의 활동은 날로 복잡성이 증가되고 있으며, 대내외적인 경영효율의 극대화가 기업 성공의 최대 관건으로 대두되고 있다. 이러한 현실 속에서 기업은 효율적인 경영을 위해 고객, 파트너, 임직원 및 정보시스템과 비즈니스 프로세스 등이 효과적으로 운영/관리되어야 함을 인지하게 되었다. 비즈니스 프로세스에 대한 효과적인 관리는 업무의 중복, 업무 시간의 증가, 데이터 중복으로 인한 기업 내의 손실을 줄일 수 있으며, 특히 급격히 변화하는 시장 상황, 고객의 요구 및 내부 조직의 변경 등으로 인한 비즈니스 프로세스 변화에 신속히 대응할 수 있다.

최근에 더욱 더 많은 기업과 조직들이 프로세스 기반의 정보기술의 중요성과 필요성을 인식하기 시작하면서 워크플로우 및 BPM(Business Process Management) 시스템의 각 산업부문별 적용사례가 급속하게 증가하고 있다. 이러한 워크플로우 기술의 적용사례 증가와 함께 프로세스 기반 업무처리량 및 그에 따른 이력정보들이 늘어나면서 워크플로우 및 업무 프로세스들에 대한 개선과 새로운 유형의 업무 프로세스 또는 비즈니스 프로세스를 재발견하기 위한 워크플로우 기술이 요구되고 있다.

워크플로우 기술은 데이터베이스 시스템 분야는 기본이고 더 나아가 BPM 기술로 발전되고 있다. 최근 워크플로우 개발 환경의 동향은 네트워크의 발전으로 개인 컴퓨팅 환경이 강화되었고 또한 인터넷의 발전으로 인하여 워크플로우 기술 역시 몇 년 사이 많은 변화를 가져오게 되었다. 이러한 발전은 워크플로우 응용 분야가 점점 대규모화 되고 복잡해지는 양상을 띠게 되었다[29].

최근 기업들의 프로세스 관리 요구와 워크플로우 분야의 확대로 인해 비즈니스 프로세스를 효과적으로 분석하고 개선하기 위한 프로세스 마이닝, 프로세스 재발견, BPR(Business Process Reengineering), BPI(Business Process Intelligence)등의 개념이 대두되었다. 이러한 개념들은 비즈니스 프로세스의 개선을 최종 목표로 하고 있지만 점차로 복잡화되는 요즘의 기업환경에 적절하지 못하다. 지금까지의 프로세스 분석, 정보 도출 등의 방법은 생성된 프로세스 인스턴스들 전체에 대한 작업을 전제로 한다. 이러한 접근법은 총체적인 프로세스 분석에 대해서는 효과적일 수 있지만, 점점 복잡화 되고 대규모화되고 있는 기업환경에 적용하기 위해서는 많은 비용을 필요로 한다. 또한 각 비즈니스의 범위 내에서 수시로 변화하는 사용자의 지식 요구에 대응하기 위해서는 기존의 프로세스 분석 방법이 아닌 새로운 접근 방법이 필요하다.

비즈니스 프로세스를 정의하는 시점에서부터 프로세스 분석을 고려한다면 추후 프로세스 분석이 이루어지는 단계에서 효율적으로 프로세스 인스턴스들을 그룹화할 수 있고 적은 비용으로 분석을 수행할 수 있다. 또한 프로세스를 분석하려는 요소들에 대한 선정의를 통하여 더욱 적극적인 비즈니스 분석이 이루어 질 수 있다.

이에 본 연구에서는 생성되는 프로세스 인스턴스들을 효과적으로 그룹화하여 프로세스 분석 작업이 더욱 체계적으로 이루어질 수 있게 하며, 적극적인 비즈니스 지식 도출을 가능하게 하는 모델 방법론을 제안하고자 한다. 프로세스 모델링 방법론인 ICN에 색채 토큰 개념을 적용하여 비즈니스 프로세스를 그룹화하거나 분석하고자 하는 항목들에 대한 내용을 토큰에 포함시켜, 적극적이고 분석 지향적인 프로세스 모델링이 가능할 수 있도록 한다.

2. 관련 연구

2.1 컬러드 페트리 넷 (CPN, Coloured Petri-Net)

페트리넷[14]은 시스템의 모델링과 분석에서 오랫동안 사용되어 온 모델링 방법이다. 이것은 동시성(concurrency)을 나타내기 쉽고 정형적인 의미(formal syntax)가 명확하며 시각화가 용이하다

는 장점을 가지고 있다. 따라서 통신 프로토콜, 제어 시스템, 병렬 프로그램과 모델링에 널리 사용되고 있다. 프로세스의 다양한 상태를 나타내기 위해서는 여러 토큰들을 사용해야 하는데, 기본의 페트리넷에서는 각 상태를 가리키는 토큰들을 서로 구별할 방법이 없었다. 기본 페트리넷을 이용하면서 많은 사용자들은 페트리넷이 너무 기초적(primitive)이어서 표현력이 부족하다는 것을 지적해왔다.

이러한 지적을 보완하기 위해 제안된 것이 고급 페트리 넷(high level Petri-net)이다. 여러 가지 제안된 모델 중에서 토큰의 색이라는 개념을 추가하여 표현력을 향상시킨 것이 컬러드 페트리넷(이하 CPN)이다. 이는 특정 용도를 가정하지 않은 일반 목적 언어라는 점과 ML이라는 기존의 고급 언어를 사용하므로 이해하기 쉽다는 장점이 있다[15].

CPN의 정의를 살펴보면 다음과 같다[16].

CPN = (∑, P, T, A, N, C, G, E, I).
- ∑ 은 색(color, type)들의 집합이다.
- P, T, A 는 각각 place, transition, arc의 집합이다.
- N은 노드 함수로, A에서 P x T ∪ T x P 로의 함수이다.
- C 는 색 함수로 P에서 ∑ 로의 함수이다.
- G 는 조건 함수로, 각 transition에 대해 정의된다.
- E 는 arc expression 으로, 각 arc에 대해 정의된다.
- I 는 초기치 함수로, 각 place에 대해 정의된 다.

여기서 플레이스는 시스템의 상태 혹은 조건을 나타내며 원으로 표시한다. 트랜지션은 시스템의 상태를 변화시키는 동작을 나타내며 선분으로 표시한다. 아크는 방향성을 가진 화살표로 표시하며 흐름을 나타낸다. 각 플레이스에 위치하게 되는 토큰은 시스템의 동적이며 병행적인 동작의 특성을 나타내기 위해 사용된다. 동작이 일어나는데 필요한 조건을 만족하는 경우, 플레이스에 토큰을 위치시킴으로써 표현한다.

기본적인 페트리넷과는 달리 CPN에서는 토큰이 데이터 값을 전달할 수 있고, 이를 통해 토큰들 간의 구분이 가능하다. [17.18].

토큰들이 가진 데이터 값을 토큰 컬러(token colour)라고 부르며 위와 같이 서로 다른 값을 가진 토큰들은 각각 구별되는 색상을 가졌다고 한다. 위의 토큰들은 각 플레이스와 트랜지션에서 요구되는 데이터 형태에 맞게 값이 대응되어, 모델링 된 페트리넷에서 운용되게 된다. 이를 통해 각 플레이스에서의 입출력 값에 대한 추적과 프로세스 수행 성능에 대한 측정이 가능하다.

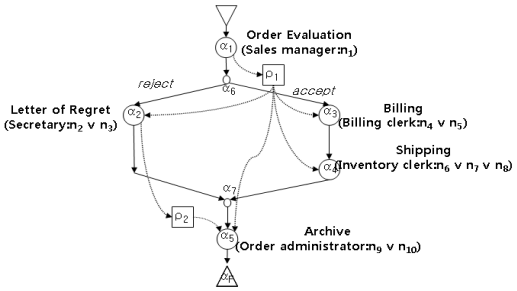
CPN에서 사용되는 색채 토큰은 비즈니스 프로세스를 진행하는 데에 사용되어지는 데이터 패킷이다. 이는 프로세스가 시작될 때 초기 플레이스에 마킹되어 있는 데이터 패킷 집합으로부터 시작하며 비즈니스 분석이 목적이 아닌, 프로세스 운용에 목적을 둔다. 따라서 CPN을 통한 비즈니스 프로세스 분석이 이루어지는 경우에는, 분석하려는 항목에 포함되지 않는 데이터 항목에 대한 처리도 같이 이루어져야 하기 때문에 분석 비용이 높고 비효율적인 분석이 이루어질 수 있다.

2.2 ICN (Information Control Net)

본 논문에서는 워크플로우를 정의하기 위한 기본적인 방법으로 표현 방법이 비교적 간단하고 정형명세가 가능한 ICN 모델링 방법론을 이용하고자 한다. ICN은 사무환경에서의 정보의 흐름을 기술하고 분석하기 위해 제안된 모델이다[19]. ICN은 사무실(Office)의 개념을 일련의 연관된 프로시저(procedure)의 집합으로 정의하며 이러한 프로시저는 선후관계가 존재하는 액티비티들의 집합으로 표현된다. ICN은 그림 형태대로 프로시저, 액티비티, 저장소(repositories), 선후관계를 나타내는 제어 흐름(control flow)과 데이터 흐름(data flow)을 표현한다. 이러한 방법으로 형성된 ICN은 각 단계를 정형화한 방법으로 구성 및 분석할 수

있다.

ICN 제어흐름 그래프는 큰 원으로 표현되는 일련의 액티비티와 작고 빈 원으로 표현되는 OR 노드, 작고 채워진 원으로 표현되는 AND노드, 그리고 이러한 노드들을 연결하는 선(edge)로 구성된다. 화살표(Arc)는 실선(Solid)과 점선(Dashed)으로 표현되는데 이들은 노드들 간의 선후관계 및 자료저장소와의 입출력을 표현한다.



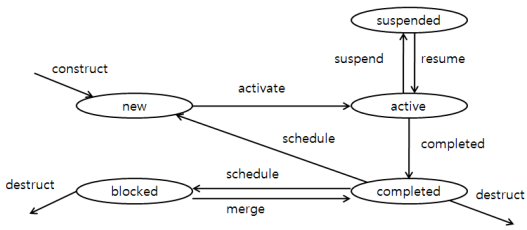
(그림 1) 주문 처리 프로세스

그림 1에서 전체 프로시저의 이름은 주문 처리 프로세스이며, 각 과정을 담당하는 액티비티는 Order Evaluation, Letter of Regret, Billing, Shipping, Archive(α1, α2, α3, α4, α5)이다. 그리고 α6, α7은 OR분기를 나타내는 노드이며, ρ1, ρ2는 저장소를 나타낸다. 또한 직선 형태로 된 실선은 제어의 흐름을 나타내고 곡선형태의 실선은 정보의 흐름을 나타낸다.

ICN에서의 토큰은 프로세스 인스턴스인 워크케이스내의 액티비티 쓰레드를 나타낸다. 워크케이스의 초기 토큰은 시작 액티비티에 위치하며 프로세스가 진행됨에 따라 각 액티비티에서의 위치정보를 가진다. 토큰은 토큰이 위치한 액티비티에 대한 상태(status)를 갖는다. 토큰 상태에 대한 값은 다음과 같다[20].

- new : 새로 생성된 토큰의 초기 상태
- active : 토큰이 위치한 액티비티의 업무가 활성화된 상태
- suspended : 사용자가 업무를 지연시킨 상태
- completed : 사용자가 업무를 끝낸 상태
- blocked : 토큰이 분기 이후의 병합 위치에 있는 상황에서 토큰의 수가 맞지 않는 경우의 상태

위의 그림 2는 토큰의 상태가 변화되는 모습을 나타낸다. 토큰의 상태 변화는 곧 워크케이스 내의 액티비티의 상태변화를 나타낸다.



(그림 2) 토큰 상태 전이 다이어그램

ICN에서의 토큰 개념은 업무의 상태를 나타낸다는 면에서 페트리넷에서의 토큰과 유사한 면을 보이지만, 업무 상태의 다양성을 나타낼 수 있다는 점에서 차이를 보인다. 또한 CPN에서의 토큰이 데이터 값을 가지는 반면 ICN에서는 토큰에 데이터 값을 포함시킬 수 없다는 점도 구별된다. ICN은 프로세스를 정의하거나 상태를 파악하는 경우에는 매우 효과적인 방법이 될 수 있으나, 프로세스에 대한 분석을 수행하려는 경우에는 충분한 정보제공이 어렵다.

2.3 비즈니스 프로세스 인텔리전스

비즈니스 프로세스 인텔리전스(BPI)는 기업 내외부에서 발생하는 수많은 데이터를 적절히 수집, 가공, 분석하여 이를 전략 수립에 응용할 수 있는 기업의 역량을 의미하는 BI(Business Intelligence)의 개념[3,8]과 기업 비즈니스내의 프로세스를 명확히 정의하고 최적화함으로써 업무의 효율성을 높이는 BPM 개념의 결합이다. 즉, 비즈니스 프로세스 인텔리전스란 비즈니스 프로세스 운영에서의 이상 행위의 검출, 미래 상황에 대한 예측, 프로세스의 특정 행태에 대한 원인 분석 등을 통해

비즈니스 프로세스의 개선을 목표로 하는 프로세스 관리 기술이다[1,2,10].

[4]의 연구에서는 이러한 연구 영역을 BPI라고 말하면서 이를 지원하기 위한 솔루션을 개발하였다. [4]에서 개발된 솔루션은 설계된 프로세스 정의를 실행하여 생성된 로그데이터들을 바탕으로 지식(knowledge)을 이끌어내는 것을 목표로 한다. 또한 이를 위하여 이력 정보들에 대한 데이터베이스 스키마를 제시하고 있다. 이 솔루션은 세 가지 주요 컴포넌트로 구성되어 있는데 프로세스 로그로부터 데이터를 추출해내는 PDW loader, 프로세스 마이닝을 위한 PME(Process Mining Engine), 그래픽 디스플레이 리포팅을 위한 Cockpit 으로 구성된다.

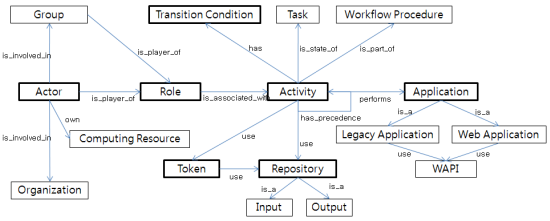
Geppert & Tombros[5] 는 이벤트 기반 워크플로우 실행에서의 로깅과 분석에 관한 접근법을 제시한다. 이 연구에서의 분석방법은 ECA(Event-Condition-Action) 룰을 지원하는 데이터베이스 시스템에 저장된 이벤트 이력에 대한 질의를 통해 이루어진다. 요구된 정보에 대한 분석은 워크플로우 분석 질의 서비스의 설계와 개발을 유도했다.

비즈니스 프로세스 수행에 대한 가시적 분석에 대한 접근은 [12]에서 찾아볼 수 있다. 제안된 가시화 도구인 VisImpact는 'impact factors'라고 불리는 비즈니스 지표(business metric)를 구별해내기 위해 스키마와 프로세스 인스턴스의 분석을 이용한다. 프로세스 비즈니스 지표와 그와 관련된 프로세스 인스턴스는 프로세스 플로우의 관계 등을 보여주는 환형 그래프에 매핑된다.

[13]의 연구에서는 프로세스 수행 관리를 위한 비즈니스 프로세스 인텔리전스 개념과 기술을 소개한다. 이 연구에서는 업무 수행 관리에 대한 비즈니스 프로세스 인텔리전스 개념을 정리하고 액티비티 흐름, 생산 흐름, 자원 흐름, 비용 흐름, 금전 흐름, 이윤 흐름 등의 여섯 가지 프로세스 흐름에 대한 분석 모델을 제시했다. 또한 시간, 질, 서비스, 비용, 속도, 효율성, 중요도 등의 일곱 가지 분류를 이용한 프로세스 수행 평가 방법을 제안했다. 이 연구에서 제시된 애플리케이션을 이용하면 중소규모의 기업 환경에 대한 프로세스 분석과 BPR이 효과적으로 적용될 수 있을 것으로 예측하고, 대규모 기업환경에서의 보완점이 필요하다는 한계를 밝혔다.

이렇듯 비즈니스 프로세스를 지능적으로 분석하고 향상시키기 위한 연구와 노력은 여러 가지가 있었고 지금도 많은 분야에서 연구가 이루어지고 있다. 기존의 BPI에 대한 연구는 대부분 데이터웨어하우징을 통한 프로세스 분석이 주를 이루고 있다. 즉, 존재하는 시스템내의 비즈니스 프로세스를 수행하여 프로세스 인스턴스들이 진행되면서 발생한 여러 이벤트 데이터들에 대한 수집과 분석을 통하여 비즈니스 개선을 위한 지식(knowledge)을 도출해내는 방식이었다. 이는 프로세스의 형태가 복잡화되고 고객의 요구사항이 수시로 변화하는 형태의 비즈니스에 대해서는 상당한 비용이 소모되게 된다. 프로세스를 수행한 결과들로부터 비즈니스 분석을 시작하는 것이 아니라, 비즈니스 프로세스를 정의하는 시점에서부터 프로세스 분석을 고려한다면 추후 프로세스 분석이 이루어지는 단계에서 좀 더 효율적으로 프로세스 인스턴스들을 그룹 화할 수 있고 적은 비용으로 분석을 수행할 수 있다. 또한 프로세스를 분석하려는 요소들에 대한 선정의를 통하여 더욱 적극적인 비즈니스 분석이 이루어 질 수 있다.

이에 본 논문에서는 프로세스를 정의하는 시점에서부터 비즈니스 분석을 고려한 모델링 방법을 제안한다. 프로세스 모델링 방법론인 ICN에 색채 토큰 개념을 적용하여 비즈니스 프로세스를 그룹화하거나 분석하고자 하는 항목들에 대한 내용을 토큰에 포함시켜, 적극적이고 분석 지향적인 프로세스 모델링이 가능할 수 있도록 한다.



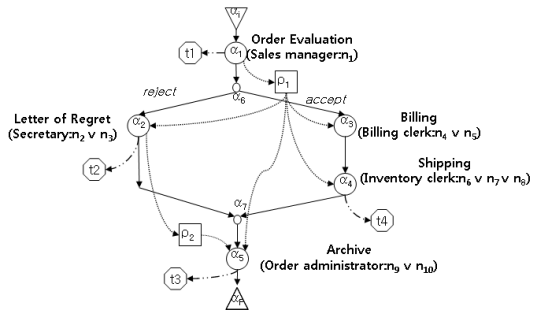
(그림 3) 색채형 워크플로우 메타 모델

3. 색채형 워크플로우

3.1 색채형 워크플로우 메타 모델

워크플로우 메타 모델은 프로세스 정의 내에 포함된 상위 레벨 엔티티들의 관계와 속성들에 대해 기술하며, 상이한 프로세스 모델들에 대한 공통적인 정의와 관련된 프로세스 모델들 간의 프로세스 정의를 그룹화하기 위한 다양한 규약들을 명시한다[21]. 색채형 워크플로우 모델은 기본 워크플로우 모델에 색채 정보를 추가함으로써 정의되며, 이것은 색채형 워크플로우 메타 모델에 따라 정의된다. 색채형 워크플로우 메타 모델은 기본 워크플로우 메타 모델에 색채 정보를 이용할 수 있는 토큰 객체를 추가함으로써 정의된다.

색채형 워크플로우 메타 모델이 포함한 토큰 엔티티는 프로세스 정의의 범위에서 영향을 받는다. 토큰은 정의된 색채형 워크플로우 모델에서 프로세스 진행에 따라 이동하며 액티비티에서 처리되는 데이터들을 이용할 수 있다. 색채형 워크플로우 모델에서 생성되는 토큰 인스턴스는 고유의 데이터 집합을 가질 수 있으며 이러한 데이터 내용은 각 토큰을 구별할 수 있는 색채(Color)개념으로 생각한다. 여기에서의 색채개념은 메타모델의 엔티티에 포함시키지 않는다. 본 논문에서 제시하는 토큰에서의 색채는 모델내의 엔티티로써 구분되어 정의되는 것이 아니라, 프로세스의 타입, 혹은 액티비티의 타입을 구분하는 개념적인 정의로 사용한다. 이와 같은 사항을 기반으로 하여 구성된 색채형 워크플로우 메타 모델은 그림 3과 같다.



(그림 4) 주문처리 관계를 나타내는 색채형 워크플로우

3.2 색채형 워크플로우 모델의 정형적 정의

색채형 워크플로우 모델을 정의하기 위하여 본 논문에서는 ICN을 활용한다. 비즈니스 프로세스를 정의하는데 있어 다른 비즈니스 프로세스 정의 언어들을 배제하고, ICN을 활용한 이유로는 ICN이 정형화 방법으로 비즈니스 프로세스 모델을 정형적으로 정의를 할 수 있으며, 그로 인해 모델이 타당한지를 판단 할 수 있다는 점 때문이다.

색채형 워크플로우 모델을 위한 ICN의 튜플은 $\Gamma = (\delta, \gamma, \varepsilon, \pi, \kappa, \tau, I, O)$과 같이 표현되며, 각각의 정의는 다음과 같다.

- $\delta = \delta i \cup \delta o$

이때 $\delta o : A \rightarrow \mathcal{P}(S \subset \mathcal{P}(A))$는 하나의 액티비티가 수행된 후 수행하는 액티비티들의 집합에 연결하는 관계를 나타내며 $\delta i : A \rightarrow \mathcal{P}(S \subset \mathcal{P}(A))$는 하나의 액티비티가 수행되기 전에 수행하는 액티비티들의 집합에 연결하는 관계를 나타내는 관계이다.

- $\gamma = \gamma i \cup \gamma o$

이때 $\gamma o : A \rightarrow R$은 하나의 액티비티의 이후에 수행되는 액티비티 집합들을 단일 출력 저장소들의 집합과 연결하는 것 중 하나이며, $\gamma i : R \rightarrow \mathcal{P}(A)$은 하나의 액티비티의 이전에 수행되는 액티비티 집합들을 입력 자료저장소들의 집합과 연결하는 관계를 나타내는 것 중 하나이다.

- $\varepsilon = \varepsilon a \cup \varepsilon p$

이때 $\varepsilon p : A \rightarrow P$는 하나의 액티비티를 조합된 역할들의 집합에 액티 비티를 단일 값으로 연결

하는 것이고, εa : P → $\mathscr{P}$ (A)는 조합된 액티비티들의 집합을 역할에 다중 값으로 연결하는 것이다.

- π = πp ∪ πc

이때 πc : P → $\mathscr{P}$ (C)는 조합된 수행자들의 집합을 역할에 다중 값으로 연결하는 것이고, πp : C → $\mathscr{P}$ (P)는 조합된 역할들을 수행자에 다중 값으로 연결하는 것이다.

- κ = κi ∪ κo

이때 κi는 α ∈ A일 때, (δi(α), α) 사이에서, 제어흐름(T) 조건들의 집합이 고, κo는 α ∈ A일 때, (α, δo(α)) 사이에서, 제어흐름(T) 조건들의 집합이다. 이때 집합 T = {default, or(conditions), and(conditions)}이다.

- τ = τt ∪ τa

이때 τt : A → $\mathscr{P}$ (K)는 조합된 액티비티들의 집합에 토큰을 단일 값으로 연결하는 것이고, τa : K → $\mathscr{P}$ (A)는 조합된 토큰들의 집합에 액티비티를 단일 값으로 연결하는 것이다.

- I 는 초기 입력되는 자료 저장소들의 유한 집합이며, ICN의 실행 전에 어떠한 외부의 프로세스에 의해서 로드 되어야 한다.
- O 는 마지막으로 출력되는 자료저장소들의 유한 집합이며 ICN의 실행 후 에 어떠한 외부의 프로세스에 의해서 이용되는 정보를 포함하고 있다고 가정한다.

다음 표 1~7 은 위의 그림 4에서 나타낸 주문처리 관계를 나타내는 색채형 워크플로우 모델을 새롭게 정의한 ICN의 정형 명세 방법에 따라 표현한 것이다.

(표 1) 색채형 ICN 튜플 구성 요소 정의

TheOrderProcedurein ICN
Γ = (δ, γ, ε, π, κ, I, O) over A, R, P, C, T

Activities
A = {α_1, α_2, α_3, α_4, α_5, α_6, α_7, α_I, α_F}

Repositories
R = {ρ_1, ρ_2}

Roles
P = {ω_1, ω_2, ω_3, ω_4, ω_5}

Actors
C = {η_{l1}, η_{l2}, η_{l3}, η_{l4}, η_{l5}, η_{l6}, η_{l7}, η_{l8}, η_{l9}, η_{l10}}

TransitionConditions
T = {d(default), or_1('reject'), or_2('accept')}

Tokens
K = {t1, t2, t3, t4}

InitialInputRepositories
I = {0}

FinalOutputRepositories
O = {0}

(표 2) 액티비티 간 관계

δ = δ_i ∪ δ_o	
$\delta_i(\alpha_I)$ = {0}	$\delta_o(\alpha_I)$ = {α_I}
$\delta_i(\alpha_1)$ = {α_I}	$\delta_o(\alpha_1)$ = {α_6}
$\delta_i(\alpha_2)$ = {α_6}	$\delta_o(\alpha_2)$ = {α_7}
$\delta_i(\alpha_3)$ = {α_6}	$\delta_o(\alpha_3)$ = {α_4}
$\delta_i(\alpha_4)$ = {α_3}	$\delta_o(\alpha_4)$ = {α_7}
$\delta_i(\alpha_5)$ = {α_7}	$\delta_o(\alpha_5)$ = {α_F}
$\delta_i(\alpha_6)$ = {α_1}	$\delta_o(\alpha_6)$ = {α_2, α_3}
$\delta_i(\alpha_7)$ = {α_2}	$\delta_o(\alpha_7)$ = {α_5}
$\delta_i(\alpha_F)$ = {α_5}	$\delta_o(\alpha_F)$ = {0}

(표 3) 저장소와 액티비티 간 관계

γ = γ_i ∪ γ_o	
$\gamma_i(\alpha_I)$ = {0}	$\gamma_o(\alpha_I)$ = {0}
$\gamma_i(\alpha_1)$ = {0}	$\gamma_o(\alpha_1)$ = {ρ_1}
$\gamma_i(\alpha_2)$ = {ρ_1}	$\gamma_o(\alpha_2)$ = {ρ_2}
$\gamma_i(\alpha_3)$ = {ρ_1}	$\gamma_o(\alpha_3)$ = {0}
$\gamma_i(\alpha_4)$ = {ρ_1}	$\gamma_o(\alpha_4)$ = {0}
$\gamma_i(\alpha_5)$ = {ρ_1, ρ_2}	$\gamma_o(\alpha_5)$ = {0}
$\gamma_i(\alpha_6)$ = {0}	$\gamma_o(\alpha_6)$ = {0}
$\gamma_i(\alpha_7)$ = {0}	$\gamma_o(\alpha_7)$ = {0}
$\gamma_i(\alpha_F)$ = {0}	$\gamma_o(\alpha_F)$ = {0}

(표 4) 역할과 액티비티 간 관계

$\varepsilon = \varepsilon a \cup \varepsilon p$	
$\varepsilon_p(\alpha_1) = \{0\}$	
$\varepsilon_p(\alpha_1) = \{\omega_1\}$	
$\varepsilon_p(\alpha_2) = \{\omega_2\}$	$\varepsilon_a(\omega_1) = \{\alpha_1\}$
$\varepsilon_p(\alpha_3) = \{\omega_3\}$	$\varepsilon_a(\omega_2) = \{\alpha_2\}$
$\varepsilon_p(\alpha_4) = \{\omega_4\}$	$\varepsilon_a(\omega_3) = \{\alpha_3\}$
$\varepsilon_p(\alpha_5) = \{\omega_5\}$	$\varepsilon_a(\omega_4) = \{\alpha_4\}$
$\varepsilon_p(\alpha_6) = \{0\}$	$\varepsilon_a(\omega_5) = \{\alpha_5\}$
$\varepsilon_p(\alpha_7) = \{0\}$	
$\varepsilon_p(\alpha_F) = \{0\}$	

(표 5) 수행자와 역할 간 관계

$\pi = \pi p \cup \pi c$	
$\pi_P(\eta_1) = \{\omega_1\}$	
$\pi_P(\eta_2) = \{\omega_2\}$	
$\pi_P(\eta_3) = \{\omega_2\}$	$\pi c(\omega_1) = \{\eta_1\}$
$\pi_P(\eta_4) = \{\omega_3\}$	$\pi c(\omega_2) = \{\eta_2, \eta_3\}$
$\pi_P(\eta_5) = \{\omega_3\}$	$\pi c(\omega_3) = \{\eta_4, \eta_5\}$
$\pi_P(\eta_6) = \{\omega_4\}$	$\pi c(\omega_4) = \{\eta_6, \eta_7, \eta_8\}$
$\pi_P(\eta_7) = \{\omega_4\}$	$\pi c(\omega_5) = \{\eta_9, \eta_{10}\}$
$\pi_P(\eta_8) = \{\omega_4\}$	
$\pi_P(\eta_9) = \{\omega_5\}$	
$\pi_P(\eta_{10}) = \{\omega_5\}$	

(표 6) 트랜지션과 액티비티 간 관계

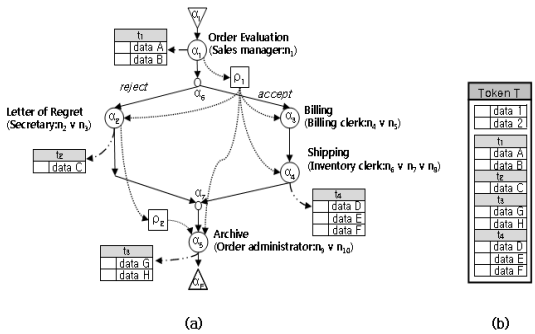
$\kappa = \kappa_i \cup \kappa_o$	
$\kappa_i(\alpha_1) = \{0\}$	$\kappa_o(\alpha_1) = \{d\}$
$\kappa_i(\alpha_1) = \{d\}$	$\kappa_o(\alpha_1) = \{d\}$
$\kappa_i(\alpha_2) = \{or_1\}$	$\kappa_o(\alpha_2) = \{d\}$
$\kappa_i(\alpha_3) = \{or_2\}$	$\kappa_o(\alpha_3) = \{d\}$
$\kappa_i(\alpha_4) = \{d\}$	$\kappa_o(\alpha_4) = \{d\}$
$\kappa_i(\alpha_5) = \{d\}$	$\kappa_o(\alpha_5) = \{d\}$
$\kappa_i(\alpha_6) = \{d\}$	$\kappa_o(\alpha_6) = \{or_1, or_2\}$
$\kappa_i(\alpha_7) = \{d\}$	$\kappa_o(\alpha_7) = \{d\}$
$\kappa_i(\alpha_F) = \{d\}$	$\kappa_o(\alpha_F) = \{0\}$

(표 7) 토큰과 액티비티 간 관계

$\kappa = \kappa_i \cup \kappa_o$	
$\kappa_i(\alpha_1) = \{0\}$	$\kappa_o(\alpha_1) = \{d\}$
$\kappa_i(\alpha_1) = \{d\}$	$\kappa_o(\alpha_1) = \{d\}$
$\kappa_i(\alpha_2) = \{or_1\}$	$\kappa_o(\alpha_2) = \{d\}$
$\kappa_i(\alpha_3) = \{or_2\}$	$\kappa_o(\alpha_3) = \{d\}$
$\kappa_i(\alpha_4) = \{d\}$	$\kappa_o(\alpha_4) = \{d\}$
$\kappa_i(\alpha_5) = \{d\}$	$\kappa_o(\alpha_5) = \{d\}$
$\kappa_i(\alpha_6) = \{d\}$	$\kappa_o(\alpha_6) = \{or_1, or_2\}$
$\kappa_i(\alpha_7) = \{d\}$	$\kappa_o(\alpha_7) = \{d\}$
$\kappa_i(\alpha_F) = \{d\}$	$\kappa_o(\alpha_F) = \{0\}$

3.3 색채 토큰의 정의

토큰은 프로세스 정의 내에서 프로세스 인스턴스, 즉 워크케이스와 관련되어 프로세스 관련 데이터나 사용자 정의 데이터 등을 포함하는 데이터 집합이라고 정의한다. 토큰은 액티비티 토큰과 프로세스 토큰으로 구분지을 수 있으며, 프로세스 토큰은 각 액티비티 토큰들의 집합이다.



(그림 5) 주문처리 색채형 워크플로우 모델의 토큰 구조

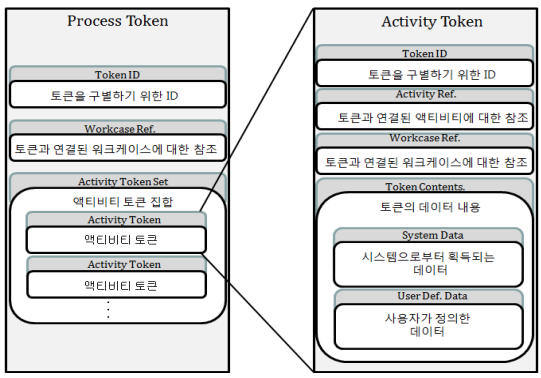
토큰은 비즈니스 프로세스를 모델링하는 시점에 정의된다. 토큰은 프로세스를 모델링하는 시점에서, 비즈니스로부터 도출하려는 지식과 연관된 데이터 항목들을 포함한다. 이 경우, 토큰에 포함되는 내용들은 사용자가 정의하거나, 프로세스내의 각 액티비티에 속해있는 데이터 항목이며 이러한 데이터 항목은 토큰 정의 시에 명시적인 데이터 값을 가지거나 프로세스가 진행되면서 값이 채워진다. 토큰이 가져야 하는 내용들 중 액티비티를 수행하면서 도출될 수 있는 데이터들은 해당 액티비티에 대한 액티비티 토큰의 항목으로 채워진다.

아래 그림 5는 토큰을 적용하여 주문처리 프로세스를 모델링하는 경우에 액티비티 토큰과 프로세스 토큰이 어떻게 구성되어질 수 있는지를 보여주고 있다. 액티비티 α1, α2, α4, α5에 연결되어 있는 각각의 토큰 t1, t2, t3, t4는 (a)에서 보여지는 대로 고유의 데이터구조를 가진다. 이렇게 구성된 액티비티 토큰들은 (b)의 프로세스 토큰 T에 포함

된다. 프로세스 토큰은 고유의 데이터 항목들과 액티비티 토큰들의 집합을 포함함으로써 완성된다.

완성된 프로세스 토큰 모델은 비즈니스 프로세스 인스턴스가 생성되어 진행됨에 따라 내용이 채워진다. 프로세스가 진행되면서 각 액티비티 토큰은 필요로 하는 데이터들을 액티비티로부터 수집하고 이것은 프로세스 토큰의 해당 영역에 기록된다. 즉 어떤 워크케이스 wc1를 처리할 때 액티비티 α1를 수행하는 동안 액티비티 토큰 t1은 정의되어 있는 데이터 항목 data A 와 data B에 대한 처리 내용을 액티비티 α1로부터 받아온다. 이것은 프로세스 토큰 T가 포함하는 액티비티 토큰 t1의 영역인 T.t1.dataA와 T.t1.dataB에 채워지게 된다. 또 다른 워크케이스 wc2가 생성이 되어 진행된다면, 각 액티비티에서 처리되는 데이터값이 달라질 수 있으므로 프로세스 토큰의 구조는 동일하더라도 채워지는 내용이 달라지게 된다. 이 경우, 워크케이스 wc1과 워크케이스 wc2에 대한 프로세스 토큰을 wc1.T , wc2.T 라고 한다면 각 토큰 wc1.T와 wc2.T 는 서로 다른 색상을 가지고 있다고 표현한다.

색채 토큰은 다른 토큰들과 구별될 수 있는 데이터 내용을 가진 토큰이며 이러한 데이터는 각 토큰이 가진 고유한 색채(color)로 간주한다. 토큰이 갖는 색상은 토큰을 구별하기 위한 개념적인 것으로써, 실제의 인스턴스 형태를 갖지 않는다. 색상은 토큰들을 서로 구별해주는 의미적 방법이고 토큰 자체가 지정된 색상을 갖는 것은 아니다. 예를들어, Color::Red 로 표현되는 토큰은 반드시 어떤 항목들을 포함해야 하고 Color::Blue 인 토큰은 반드시 어떤 항목을 포함해야 한다는 의미가 아니라, 단지 각 토큰들은 서로 다른 고유의 내용으로 구성되어 있다는 것을 의미한다. 앞서 언급한 대로 토큰에 포함되는 내용은 프로세스 정의시에 결정되며 그 값은 토큰 정의 시에 할당되거나 프로세스 진행 중에 액티비티로부터 할당 받을 수 있다. 정의된 프로세스 토큰은 동일한 구조라 하더라도 프로세스 인스턴스에 따라 서로 다른 값을 가질 수 있다. 또한 같은 프로세스라 하더라도 프로세스 토큰의 구조는 서로 다를 수 있다. 이러한 경우 토큰은 각각 고유의 색상을 가진다.



(그림 6) 액티비티 토큰과 프로세스 토큰의 구조

액티비티 토큰과 프로세스 토큰의 구조는 다음 그림 6과 같다.

각 토큰은 식별 가능한 고유의 '토큰 ID(Token ID)'를 가지며 이 ID를 이용하여 각각의 토큰에 접근이 가능하다. '워크케이스 참조(Workcase Ref.)'는 이 토큰이 어느 워크케이스에서 사용되고 있는지 알 수 있도록 워크케이스의 ID를 가진다.

액티비티 토큰이 가진 '액티비티 참조(Activity Ref.)' 는 이 토큰이 어느 액티비티에 연결되어 있는지 해당 액티비티에 대한 ID를 가진다. 각 액티비티 토큰의 실질적인 내용은 '토큰 컨텐츠(Token Contents)' 안에 저장된다. 토큰을 구성하는 내용, 즉 워크케이스의 관련 데이터(Relevant Data)같은 '시스템 데이터(System Data)' 또는 '사용자 지정 데이터 (User Def. Data)'의 집합으로 이루어진 구성요소는 '컨텐츠 요소(Contents Element)'라고 정의한다. 각 컨텐츠 요소는 l-value와 r-value 의 쌍으로 이루어지며 색채 토큰 정의시에 해당 값이 모두 할당이 되거나, 프로세스 인스턴스가 진행되어 감에 따라 선정의된 l-value에 해당하는 r-value가 채워진다.

액티비티 토큰 스키마 구조와 프로세스 토큰 스키마 구조는 다음 표8과 표9의 내용과 같다.

(표 8) 액티비티 토큰 스키마 구조

```
<xs:element name="Token_Activity">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Activity_Ref"
type="xs:string"/>
      <xs:element name="Workcase_Ref"
type="xs:string"/>
      <xs:element name="Token_Contents">
    <xs:complexType>
      <xs:sequence>
       <xs:element name="System_Data"
maxOccurs="unbounded"/>
        <xs:element name="User_Def_Data"
minOccurs="0i° maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
      </xs:element>
    </xs:sequence>
   <xs:attribute name="Id" use="required"/>
   <xs:attribute name="Name"/>
  </xs:complexType>
</xs:element>
```

(표 9) 프로세스 토큰 스키마 구조

```
<xs:element name="Token_Process">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Workcase_Ref"/>
      <xs:element name="Token_Contents">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="Token_Activity"
minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
     </xs:element>
    </xs:sequence>
  <xs:attribute name="Id" type="xs:string"
use="required"/>
  <xs:attribute name="Name" type="xs:string"/>
  </xs:complexType>
</xs:element>
```

프로세스 토큰은 '액티비티 토큰 집합(Activity Token Set)'을 가진다. 액티비티 토큰 집합은 프로세스가 가진 액티비티들에 연결되어 있는 액티비티 토큰들을 포함한다. 비즈니스 프로세스를 모델링하면서 정의되는 토큰의 실질적인 구조는 프로세스 토큰의 구조가 된다. 프로세스 토큰의 구조는 해당 프로세스에 대해 어떤 데이터들을 이용하여 어떤 지식을 도출하고 분석해낼 것인가에 따라 변경된다. 이것은 결국 프로세스 토큰을 구성하는 액티비티 토큰 집합의 내용이 고정적이지 않다는 것을 의미하며, 한 프로세스에서 여러 프로세스 토큰 구조가 생성될 수 있다는 것을 의미한다.

프로세스 토큰은 액티비티 토큰에 대한 참조를 가진다. 비즈니스 프로세스를 모델링하는 시점에서 액티비티 토큰들에 대한 정의가 모두 완료된 뒤에 프로세스 토큰에 대한 구조가 완성된다.

4. 색채형 워크플로우 모델의 적용

4절에서는 비즈니스 프로세스를 운용함에 있어서 색채형 워크플로우 모델이 어떤 방향으로 이용될 수 있는지 적용 예시를 언급한다.

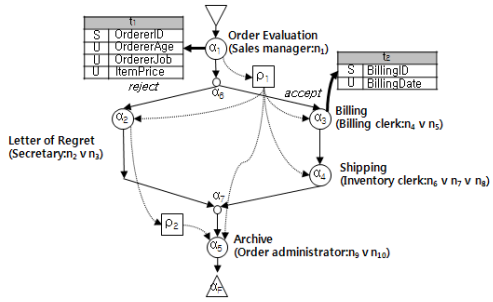
4.1 비즈니스 프로세스 정보 도출

주문처리 관계를 나타내는 색채형 워크플로우에서 사용될 수 있는 모든 데이터 내용이 표 10과 같이 구성된다고 했을 때, 색채형 워크플로우 모델에서 사용되는 색채 토큰의 데이터 항목들은표 10에서 선택적으로 포함되거나 이 외에 사용자가 정의하는 데이터 항목들이 포함된다.

발생된 수천건의 인스턴스로부터 도출해내려는 정보가 특정 기간 동안의 나이별, 직업별 구매력이라고 가정하는 경우에 토큰의 데이터 구성 모습은 다음 그림 7과 같다.

(표 10) 주문처리 관계를 나타내는 색채형 워크플로우 모델의 데이터 항목 예

구분	업무명	데이터 항목			
α1	Order Evaluation	주문자 정보 (Orderer Info)	주문자 ID (OrdererID) 이름 (OrdererName) 나이 (OrdererAge) 성별(Orderer Gender) 직업 (OrdererJob) 주소(Orderer Address)	물품 정보 (Item Info)	물품 ID (ItemID) 물품명 (ItemName) 가격 (ItemPrice) 재고 수량 (ItemStock)
α2	Letter of Regret	주문 오류 정보 (Order Fault Info)	오류코드 (Fault Code) 주문 불가 사유 (Fault Description) 메일 통지 여부 (is_sendTo)		
α3	Billing	결재 정보 (Bill Info)	결재 정보 ID (Bill Info ID) 결재자이름(Name) 주민등록번호(Address) 결재일(Billing Date) 카드번호(Credit Card Number)		
α4	Shipping	운송 정보 (Shipping Info)	운송사 이름(Name) 주소(Address) 연락처 (Cell Phone Number)		
α5	Archive	주문 내역 (Order Info)	주문서 번호 (Order Number) 주문자 ID (Orderer ID) 운송사 정보 (Logistics Info) 송장번호 (Invoice) 물품 위치 (Goods Location) 물품처리상황 (Goods Status)		



(그림 7) 주문처리 프로세스에 적용된 색채 토큰

(표 11) OrderEvaluation 액티비티의 XPDL 정의

```
<Activity Id="Activity1224828273875"
Name="OrderEvaluation">
  <Limit />
  <Implementation>
    <Tool Id="Application_OrderForm"
Type="APPLICATION" />
  </Implementation>
  <Performer>Role_SalesManager</Performer>
  <StartMode>
    <Manual />
  </StartMode>
  <FinishMode>
    <Manual />
  </FinishMode>
  <ExtendedAttributes>
    <ExtendedAttribute Name="XPos" Value="383" />
    <ExtendedAttribute Name="YPos" Value="107" />
  </ExtendedAttributes>
</Activity>
```

(표 12) Billing 액티비티의 XPDL 정의

```
<Activity Id="Activity1224828291093" Name="Billing">
  <Limit />
  <Implementation>
    <Tool Id="Application_Billing" Type="APPLICATION" />
  </Implementation>
  <Performer>role_BillingClerk</Performer>
  <StartMode>
    <Manual />
  </StartMode>
  <FinishMode>
    <Manual />
  </FinishMode>
  <ExtendedAttributes>
    <ExtendedAttribute Name="XPos" Value="612" />
    <ExtendedAttribute Name="YPos" Value="311" />
  </ExtendedAttributes>
</Activity>
```

(표 13) 프로세스 토큰의 XPDL 정의

```
<Token_Process Id="ProcessToken6428042627021628">
  <Workcase_Ref></Workcase_Ref>
  <Token_Contents>
    <Token_Activity Id="ActivityToken46720724302576">

<Activity_Ref>Activity1224828273875</Activity_Ref>
        <Workcase_Ref></Workcase_Ref>
        <Token_Contents>
          <System_Data Name="OrdererID"></System_Data>
              <User_Def_Data
Name="OrdererAge"></User_Def_Data>
              <User_Def_Data
Name="OrdererJob"></User_Def_Data>
          <User_Def_Data Name="ItemPrice"></User_Def_Data>
        </Token_Contents>
    </Token_Activity>
    <Token_Activity Id="ActivityToken67407223025287">
 <Activity_Ref>Activity1224828291093</Activity_Ref>
        <Workcase_Ref></Workcase_Ref>
        <Token_Contents>
          <System_Data Name="BillingID"></System_Data>
          <User_Def_Data
Name="BillingDate"></User_Def_Data>
        </Token_Contents>
    </Token_Activity>
  </Token_Contents>
</Token_Process>
```

그림 7은 주문처리 프로세스 모델에 실제 데이터 항목을 가진 토큰의 적용을 나타내고 있다. OrderEvaluation 액티비티가 가진 색채토큰 t1은 시스템데이터로써 OrdererID를 가지고, 사용자정의데이터로써 OrdererAge, OrdererJob, ItemPrice를 가진다. Billing 액티비티의 토큰 t2는 시스템데이터로써 BillingID와 BillingDate를 가진다. 또한 이러한 액티비티 토큰의 구조를 통해 완성된 프로세스 토큰 T의 구조는 표 13과 같다. 표 13에서의 프로세스 토큰은 표 11과 표 12에 정의된 액티비티에 대한 참조를 가진다. 위와 같이 구성된 토큰은 각 인스턴스마다 동일한 항목에 대해 다른 데이터 값을 가질 수 있으며 그 예는 다음 표 14와 같다.

(표 14) 주문처리 프로세스 토큰의 워크케이스별 데이터 내용

Work case	Orderer ID	Orderer Age	Orderer Job	Item Price	Billing ID	Billing Date
wc1	Kim	45	Driver	12,000	pay_kim	08/09/25
wc2	Jeon	34	Clerk	130,000	pay_jeon	08/10/03
wc3	Kwon	48	House-Keeper	70,000	pay_kwon	08/10/12
wc4	Lee	29	Clerk	300,000	pay_lee	08/10/07
wc5	Park	22	Student	100,000	pay_park	08/10/21
wc6	Cho	42	Sales Man	210,000	pay_cho	08/10/11
wc7	Joo	26	Clerk	34,000	pay_joo	08/11/01
wc8	Han	33	Sales Man	30,000	pay_han	08/10/03
wc9	Ku	49	Driver	325,000	pay_ku	08/10/30
wc10	Seo	21	Student	34,000	pay_seo	08/10/17

각 워크케이스 wc1부터 wc10 까지에 대한 색채 토큰은 서로 다른 색을 가진다고 할 수 있다.

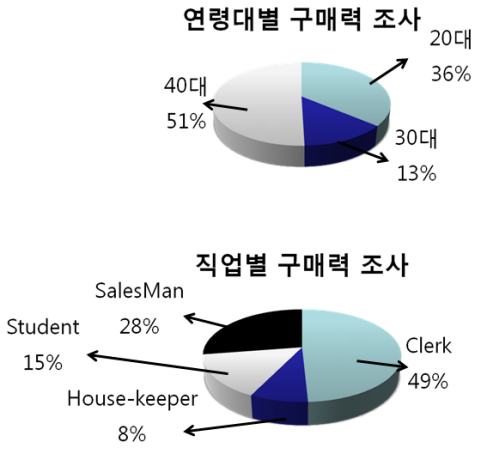
고유하게 구성된 색채토큰들을 이용하여 프로세스 인스턴스들의 지식 도출을 위한 범주를 정하는 것을 컬러링(Coloring)한다고 정의한다. 예를 들어, 10월 동안에 연령대별 구매력에 대한 정보를 도출하려는 경우에는 토큰 집합 내에서 사용자가 정의한 BillingDate 항목 중 10월 구매에 해당하는 wc2, wc3, wc4, wc5, wc6, wc8, wc9, wc10 색채토큰들을 구별해낸다. 다음으로 사용자정의 데이터인 OrdererAge를 각 연령대별 색상으로 구분하여 보여줄 수 있다. 직업별 구매력에 대한 지식 도출도 마찬가지로 행해질 수 있다. 이에 해당하는 데이터는 표 15와 표 16과 같이 나타날 수 있으며 이를 바탕으로 그림 8과 같이 손쉽게 가시적인 프로세스 지식 도출이 가능하다.

(표 15) 특정 기간(10월) 내의 연령대별 워크케이스 구분

	Work case	Orderer ID	Orderer Age	Orderer Job	Item Price	Billing ID	Billing Date
20s	wc4	Lee	29	Clerk	300,000	pay_lee	08/10/07
	wc5	Park	22	Student	100,000	pay_park	08/10/21
	wc10	Seo	21	Student	34,000	pay_seo	08/10/17
30s	wc2	Jeon	34	Clerk	130,000	pay_jeon	08/10/03
	wc8	Han	33	Sales Man	30,000	pay_han	08/10/03
40s	wc3	Kwon	48	House-Keeper	70,000	pay_kwon	08/10/12
	wc6	Cho	42	Sales Man	210,000	pay_cho	08/10/11
	wc9	Ku	49	Driver	325,000	pay_ku	08/10/30

(표 16) 특정 기간(10월) 내의 직업별 워크케이스 구분

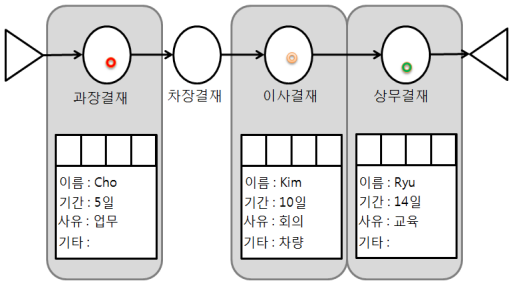
	Work case	Orderer ID	Orderer Age	Orderer Job	Item Price	Billing ID	Billing Date
Clerk	wc2	Jeon	34	Clerk	130,000	pay_jeon	08/10/03
	wc4	Lee	29	Clerk	300,000	pay_lee	08/10/07
House-Keeper	wc3	Kwon	48	House-Keeper	70,000	pay_kwon	08/10/12
Student	wc5	Park	22	Student	100,000	pay_park	08/10/21
	wc10	Seo	21	Student	34,000	pay_seo	08/10/17
Sales Man	wc6	Cho	42	Sales Man	210,000	pay_cho	08/10/11
	wc8	Han	33	Sales Man	30,000	pay_han	08/10/03
Driver	wc9	Ku	49	Driver	325,000	pay_ku	08/10/30



(그림 8) 차트로 도출한 프로세스 관련 지식의 예

4.2 고정적 비즈니스 프로세스 분석

색채토큰은 프로세스의 흐름은 고정적이지만 처리되는 데이터 내용이 모두 다른 형태의 비즈니스에 유용하게 적용된다. 예를 들어 그림 12의 전자결재 프로세스의 경우, 결재가 처리되는 프로세스 흐름은 일정하지만 그 안에서 실질적으로 처리 되는 내용, 즉 프로세스에서 사용되는 애플리케이션에 채워지는 내용들은 모두 다르다. 즉, 출장신청이나 물품구매신청 등과 같이 업무 내용은 다르더라도 프로세스의 흐름이 과장결재, 차장결재, 이사결재, 상무결재 와 같이 고정적인 프로세스인 경우에 색채토큰이 유용하게 사용된다. 이 경우 색채토큰은 각각의 인스턴스에서 사용되는 애플리케이션 자체가 된다.



(그림 12) 색채토큰을 적용한 전자결재 프로세스의 예

이러한 형태의 비즈니스에 색채토큰을 적용함으로써 프로세스 내에서 처리되고 있는 각 인스턴스의 내용 파악이 용이하게 된다. 이러한 형태의 프로세스는 그동안 처리되는 실제 내용을 알기 위해서는 복잡한 분석 절차가 필요했었지만 색채토큰을 이용함으로써 운용되는 데이터에 대한 직접적인 접근이 가능하다.

4.3 프로세스 흐름의 경향성 파악

비즈니스 프로세스를 운용하면서 발생되는 수많은 인스턴스들을 분석하는 경우에, 인스턴스들의 제어 흐름 파악은 중요한 문제이다. 인스턴스들이 어느 방향으로 집중되고 어느방향을 기피하는지를 파악하면 더욱 최적화되고 효율적인 프로세스 재설계가 이루어질 수 있기 때문이다. 흐름이 집중되는 부분에 대해서는 더 상세한 경로들로 나눠주어 프로세스 흐름이 원활하게 하고 플로우가 진행되지 않는 부분에 대해서는 삭제나 통합을 통해 비용낭비를 줄일 수 있다. 이러한 작업을 위한 전단계로 프로세스 제어흐름 파악이 이루어지게 되는데 이러한 경우에도 색채 토큰이 유용하게 사용될 수 있다.

액티비티가 가질 수 있는 액티비티 토큰에 제어흐름에 대한 조건, 즉 분기가 이루어지는 경우에 대한 데이터 내용을 포함시키면 토큰들에 대한 분석을 통하여 프로세스 흐름을 파악할 수 있다. 은행의 대출 프로세스를 간단한 예로 들자면, 대출 신청자의 신용등급에 따라 대출 가능 금액이 달라질 수 있다. 대출 신청자의 신용등급이 A인 경우 대출액은 1억원, 신용등급이 B인 경우 대출액은 5천만원, 신용등급이 C인 경우 대출액은 1천만원, 신용등급이 D인 경우 대출이 불가한 프로세스가 있다고 가정한다. 분기가 일어나는 OR 액티비티에 대한 액티비티 토큰은 시스템 데이터 내용에 분기 조건을 판단하는 데이터 항목, 즉 대출 신청자의 신용등급(credit)을 포함시킨다. OR액티비티 토큰에 데이터가 채워지는 경우 대출 신청자의 신용등급도 같이 수집되고, 이러한 경우

credit==A 또는 credit==C 와 같이 실질적으로 해당 인스턴스 내에서 처리되는 분기 조건 내용이 어떤 것인지를 파악할 수 있게 된다. 이렇게 적용된 색채토큰들을 분석함으로써 실제로 어떤 데이터가 사용되어서 어떤 플로우가 진행되었는지 파악할 수 있게 된다.

이렇듯 색채형 워크플로우 모델을 사용하면 비즈니스 프로세스로부터 도출해내려는 정보가 프로세스를 모델링하는 시점에서부터 정의되어 있기 때문에 수많은 프로세스 인스턴스가 발생한다 하더라도 손쉽게 프로세스들에 대한 그룹화가 가능하며 이를 통해 비즈니스에 대한 지식 도출을 간단하게 처리할 수 있다. 기존의 분석방법은 인스턴스들이 모두 완료된 로그 데이터들을 수집한 뒤, 여러 작업을 거쳐 원하는 지식을 이끌어내는 방식이었다면 색채형 워크플로우 모델은 더욱 간단하고 적극적인 방법으로 로그데이터 뿐만 아니라 런타임중인 데이터들을 통하여 비즈니스 프로세스 지식도출이 가능하다.

5. 결론 및 향후 연구

기업 환경의 복잡화와 다양화, 대규모화는 비즈니스 프로세스에 대한 지능적인 접근을 필요로 하게 되었다. 또한 빈번한 사용자 요구에 적절하게 대응하기 위해서는 비즈니스와 관련된 여러 요구사항들이 복합적으로 고려되어야 한다. 이를 위해 워크플로우 시스템이 갖춰야 하는 요건은 비즈니스 관련 지식을 도출하기 위한 프로세스 분석 능력에 대한 높은 효율성이다. 효과적인 비즈니스 분석과 개선을 위한 프로세스 분석 작업을 위해서는 기업 환경에서 생성되는 수많은 프로세스 인스턴스들에 대한 그룹화가 적절하게 이루어져야 하고 사용자의 빈번한 요청 사항에 적절하게 대응할 수 있어야 한다.

본 논문에서는 이를 위하여 비즈니스 프로세스를 설계하는 시점부터 비즈니스 도메인과 관련한 분석 요소를 고려할 수 있도록 색채형 워크플로우 모델을 제안하였다. 또한 실제 모델링시에 적용될 수 있도록 색채형 워크플로우에서 이용되는 색채 토큰에 대한 스키마 구조를 정의하였다. 색채형 워크플로우 모델은 색채 토큰을 이용함으로써 생성된 인스턴스들의 유형별로 그룹화가 가능하고 이를 이용하여 비즈니스 관련 지식 도출을 손쉽게 할 수 있다. 또한 비즈니스 도메인내에서 요구되는 사용자들의 요구에 효과적으로 대응할 수 있으며 사용자 관점에서의 적극적인 비즈니스 분석이 가능하다.

색채형 워크플로우 모델이 실제 시스템에서 적용되기 위해서는 색채토큰을 적용한 XPDL 문서 생성이 가능한 프로세스 모델러와 이를 운용할 수 있는 워크플로우 엔진의 확장이 필요하다. 색채형 워크플로우 엔진은 각 토큰들을 운용할 수 있는 플랫폼 역할을 수행한다. 또한 이와 아울러 각 프로세스들에 대한 색채토큰 분석이 가능한 프로세스 분석도구와 알고리즘이 마련된다면 비즈니스 프로세스 개선을 위한 지식도출이 가능해질 것이다.

참 고 문 헌

[1] Schiefer J, Jeng J, Kapoor S, Chowdhary P. (2004). Process information factory: a data management approach for enhancing business process intelligence. In: Bichler M, Chung J Y, eds. Proceedings of the IEEE International Conference on E-Commerce Technology. San Diego, CA: IEEE COMPUTER SOC. 162-169.

[2] Bonifati A, Casati F, Dayal U, Shan M C. (2001). Warehousing workflow data: challenges and opportunities. In: Apers PM G, Atzeni P, Ceri S, et al., eds. Proceedings of the 27th VLDB Conference. Rome: Morgan Kaufmann. 649-652.

[3] Schiefer J, Jeng J, Bruckner R M. (2003). Real-time workflow audit data integration into data warehouse systems. In: Proc.of the 11th European Conference

on Information Systems, Naples, Italy. http://is2.lse.ac.uk/asp/aspecis/20030134.pdf

[4] Grigori D, Casati F, Castellanos M, Dayal U, Sayal M, Shan M C. (2004). Business process intelligence. Computers in Industry, 53(3): 321-343.

[5] Geppert A, Tombros D. (1997). Logging and post-mortem analysis of workflow executions based on event histories. In: Geppert A, Berndtsson M, eds. Proc. 3rd Intl. Conf. on Rules in Database Systems, Lecture Notes in Computer Science, Vol. 1312. Heidelberg: Springer. 67-82.

[6] Koksal P, Alpinar S N, Dogac A. (1998). Workflow history management. ACM Sigmod Record, 27 (1): 67-75.

[7] W. Colin, "The IBM Business Intelligence Software Solution," Database Associates, 2000.

[8] B. Azvine, Z. Cui, D.D. Nauck, B. Majeed, "Real Time Business Intelligence for the Adaptive Enterprise" The 8th IEEE International Conference on and Enterprise Computing, E-Commerce, and E-Services, pp29-29, 2006

[9] 이준욱, 백옥현, 류근호, "e - Business에서의 BI지원 데이타마이닝 시스템", 한국정보과학회, 정보과학회논문지 : 컴퓨팅의 실제 제8권 제5호, 10, pp. 489 ~ 500, 2002

[10] 허순영, "비즈니스 인텔리전스(Business Intelligence) 개괄", 한국정보과학회, 정보과학회지 제21권 제10호, pp. 5 ~ 11, 2003

[11] "BPM(Business Process Management) 도입, 운영 실태 조사", 한국소프트웨어산업협회, 2006.09.01~2006.10.15

[12] Hao, M.C, Keim, D.A, Dayal, U., Schneidewind, J.: Business process impact visualization and anomaly detection. Information Visualization 5, 15 – 27 (2006)

[13] W.Tan, W.Shen, L. Xu, B. Zhou, L.Li, "A Business Process Intelligence System for Enterprise Process Performance Management" , IEEE Transactions on System, Man, and Cybernetics-Part C: Applications and Reviews, Volume PP, Forthcoming, 2003 Page(s):1 - 12

[14] J.Peterson, "Petri net Theory and Modeling of Systems", Prentice Hall, 1981

[15] 조승모, 홍형석, 차성덕 "컬러드 페트리 네트를 이용하는 안전성 분석 기법", 한국정보과학회, 한국정보과학회 1995년도 가을 학술발표논문집 제22권 제2호(B), 1995. 10, pp. 1415 ~ 1418 (4pages)

[16] K. Jensen, "Coloured Petri Nets: Basic Concepts, Analysis methods, and Practical Use." Volume 1, Springer Verlag, 1992

[17] K. Jensen, "A brief introduction to Colored Petri Nets", Workshop on the Applicability of Formal Models, 2 June 1998, Aarhus, Denmark, pages 55-58.

[18] K. Jensen, L. M. Kristensen, and L. Wells. Coloured Petri Nets and CPN Tools for modelling and validation of concurrent systems. Int. J. on Software Tools for Technology Transfer (STTT), Sp. Sec. CPN 04/05, 2007.

[19] Clarence A. Ellis and Gary J. Nutt, 'Office Information Systems and Computer Science', Computing Surveys, Vol. 12, No. 1, March 1980.

[20] C.A. Ellis, C. Maltzahn, The Chautauqua workflow system, in: Proceedings of the International Conference on System Science, Maui, HI, 1997.

[21] Workflow Management Coalition Specification Document, "Workflow Coalition Interface 1: Process Definition Interchange Process Model", Document Number: WFMC TC-1016-P, August 5, 1998.

[22] Workflow Management Coalition Specification Document, "The Workflow Reference Model.", Version 1.1, November 1994.

[23] Workflow Management Coalition Specification

Document, "Workflow Management Coalition Audit Data Specification", Version 1.1, Document Number: WFMC-TC-1015, September 1998.

[24] Workflow Management Coalition Specification Document, "Workflow Coalition Interface 1: Process Definition Interchange Process Model", Document Number: WFMC TC-1016-P, August 5, 1998.

[25] Kwang-Hoon Kim , "Workflow Reduction for Reachable-path Rediscovery", IEEE Proceeding of the ICDM Workshop 2003, 2003.11

[26] Workflow Management Coalition Specification Document, "Workflow Client Application (Interface 2) Application Programming Interface (WAPI) Naming Conventions", Version 1.4, Document Number: WFMC-TC-1013, November 1997.

[27] Workflow Management Coalition Specification Document, "Workflow Standard - Interoperability Abstract Specification", Version 1.0, Document Number: WFMC-TC-1012, October 1996.

[28] Kwang-Hoon Kim, Clarence A. Ellis , "Workflow Reduction for Researchable-path Rediscovery in Workflow Mining", Lecture Notes in Computer Science, pp.3-24, 2005.06

[29] G. Alonso and H. Schek, 'Research Issues in Large Workflow Management Systems'. In Proc. of the NSF workshop on workflow and process automation in information systems, Athens, GA, 1996

[30] http://www.wfmc.org

[31] http://www.petrinet.info

◐ 저 자 소 개 ◑

정 우 진 (Woo-Jin Jeong)

2007년 경기대학교 전자계산학과 졸업(이학사)
2007년 ~ 경기대학교 일반대학원 컴퓨터과학과 석사과정
관심분야 : BPM, RTE
E-mail : woojin@kgu.ac.kr

김 광 훈 (Kwang-Hoon Kim)

1984년 : 경기대학교 전자계산학과 졸업(이학사)
1986년 : 중앙대학교 전자계산학과 졸업(이학석사)
1994년 : Univ.of Colorado at Boulder, 컴퓨터과학과 졸업(이학석사)
1998년 : Univ.of Colorado at Boulder, 컴퓨터과학과 졸업(이학박사)
1986년 ~ 1991년 : 한국 전자 통신 연구원 연구원
1998년 ~ 현재 : 경기대학교 정보과학부 교수
관심분야 : 워크플로우, 비피엠, 그룹웨어, CSCW, 분산처리기술, 데이터베이스
E-mail : kwang@kgu.ac.kr