

워크플로우 제어 의존성 분석 도구의 구현

박 옹^o 오종태 김광훈 백수기
경기대학교 일반대학원 전자계산학과 그룹웨어연구실
e-mail : {wpark,jtoh,kwang,skpaik}@kuic.kyonggi.ac.kr

A Control Dependency Analyzer for Workflow

Woong park^o Jong-Tae Oh Kwang-Hoon Kim Su-Ki Paik
Dept. of Computer Science Kyonggi University

요 약

본 논문은 워크플로우 모델링 도구인 ICN(Information Control Net)을 기반으로 하는 워크플로우 제어 의존성 분석을 위한 도구를 구현하였다. 즉, ICN 모델로 정의된 워크플로우의 각 단위업무(Activity)들 간에 존재하는 제어 의존 관계를 표현하기 위한 제어 의존 넷(Control Dependency Net)을 정형적인 방법으로 정의, 표현하였다. 본 논문에서 정의된 제어 의존 넷은 워크플로우 빌드타임(Build-time) 측면과 런타임(Run-time)측면에서 여러 가지 기능을 제공한다. 전자의 측면에서는 워크플로우 모델링 시멘틱 에러 테스트 기능을 효과적으로 적용될 수 있으며, 후자의 측면에서는 제어 흐름의 동적 변경(dynamic Change)기능의 완결성 및 프로세스 안의 단위업무(Activity) 간의 경로 및 경로 대한 정보를 제공한다.

1. 서론

최근 정보통신 기술의 발달은 조직체에서의 사무업무처리 프로시저(Business Procedure)에 깊은 영향을 미치고 있다. 최근 몇 년 동안에 많은 사무 업무들이 사라지거나, 변형되거나, 완전히 새로운 형태의 업무가 생겨나고 있다. 이러한 사무업무 형태 및 환경 변화에 대처하고, 보다 효율적인 조직의 운영을 위해 워크플로우와 사무업무 처리과정 리엔지니어링(BRP : Business Process Reengineering)의 개념이 나타났다. 워크플로우 기술은 크게 일련의 단위 업무들로 구성되는 워크플로우를 정의하고 분석하는 워크플로우 모델 분야와 워크플로우의 각 업무들의 실행과 그들 간의 흐름을 제어하는 워크플로우 관리 시스템 분야로 나뉘어진다. 특히, 워크플로우 관리시스템은 어느 조직체내에서 또는 조직체 간에 구성되어 있는 사무 업무 프로시저 또는 사무 업무 프로세스(Business Process)를 기반으로 하는 업무의 흐름을 제어하고 자동화시키기 위한 기술이다. 워크플로우의 사무 업무 프로세스는 일련의 물리적 또는 논리적 단위의 단위 업무(Activity)들과 이들의 수행을 담당하는 역할(Role) 및 참여자(Actor), 그리고 단위 업무들 간에 전달되는 문

서 또는 정보들로 구성된다. 보통의 사무 업무 프로세스는 많은 단위 업무들로 구성되어 있다. 사무 업무 프로세스의 실행을 위해 각각의 단위 업무(Activity)는 응용프로그램 및 사용자 그리고 규칙들로 연결된다. 이러한 사무 업무 프로세스 안에서 단위 업무(Activity)간에 정의되는 실행 순서의 흐름을 제어 흐름(Control flow)이라 한다. 그리고, 각각의 단위 업무(Activity)들은 OR 또는 AND 노드를 통해 선택적 또는 병렬적 실행 제어 순서 관계를 갖는다. 본 논문에서는 이와 같은 단위 업무(Activity) 간에 실행 순서에 대한 제어 흐름을 분석하여 각 단위 업무(Activity)간에 존재하는 실행 제어 의존성 관계를 분석하는 도구를 구현하였다. 본 논문에서 구현한 제어 의존성 분석 도구는 워크플로우 관리 시스템의 주요 실행 모듈인 빌드타임(Build-time)과 런타임(Run-time)에서 여러 가지의 기능을 제공한다. 즉 빌드 타임 측면에서는 워크플로우 응용분야의 대형화와 복잡성의 증가에 따라 그 중요성이 더욱 증가하고 있는 워크플로우 모델의 테스트 기능에 대한 확장을 도모할 수 있다. 기존의 빌드타임 모듈에서는 정의된 워크플로우에 대한 구문(Syntax) 에러 테스트 기능과 의미적(Semantic) 에러

테스트 기능을 제공한다. 런타임 측면에서는 워크플로우의 적용에 가장 저해가 되는 요소인 동적 변경 지원 기능의 완결성 부족 문제에 대한 해결책 및 사무업무 프로세스 안에 단위 업무(Activity)간에 경로 및 경로의 수의 정보를 제공한다. 단위 업무 간에 경로 및 경로의 수는 BPR(Business Process Reengineering)에 대한 중요한 정보가 된다. 다음 장에서는 워크플로우 모델링 도구 중의 하나인 ICN 모델과 제어 의존성 분석을 위한 분석 메커니즘을 기술한다.

$\Gamma = (\delta, \gamma, I, O)$

δ : precedence constraint among activities
 γ : repository input/output requirement of activity
 I : initial input repository
 O : final output repository

<그림 1>에 대한 정형적인 ICN 표현은 아래와 같다.

2. ICN(Information Control Net) 모델

2.1 ICN 모델의 소개

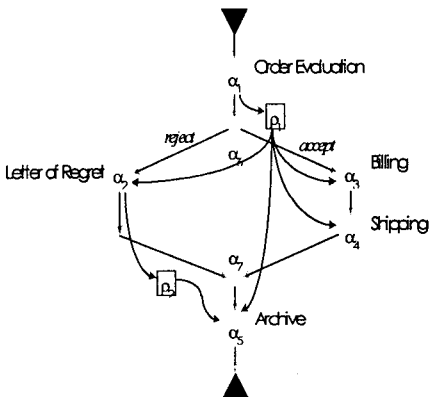
워크플로우 모델이란 조직체나 작업그룹에서의 업무환경과 업무 프로세스를 적절히 표현한 것이다. 워크플로우 모델은 조직의 모습을 할당업무(tasks), 업무자(Actor), 역할, 단위 업무(Activity)와 자료저장소의 모습으로 표현된다. ICN(Information Control Net)은 사무실(Office)의 개념을 일련의 관련된 프로세스(Process)의 집합으로 정의하며 이러한 프로세스는 전후관계가 존재하는 단위 업무들의 집합으로 표현된다. ICN은 그림 형태로 프로세스, 단위업무, 저장소(Repository) 전후관계를 나타내는 제어흐름(Control flow)과 데이터흐름(Data flow)을 표현한다. ICN 제어흐름 그래프는 큰 원으로 표현되는 일련의 액티비티와 작고 빈 원으로 표현되는 OR 노드, 작고 채워진 원으로 표현되는 AND 노드, 그리고 이러한 노드들을 연결하는 선(edge)로 구성된다. 화살표(Arc)는 실선(Solid)과 점선(Dashed)으로 표현되는데 이들은 노드들 간의 선후관계 및 자료저장소와의 입/출력을 표현한다

$\Gamma = (\delta, \gamma, I, O)$

$A = \{\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6, \alpha_7\}$
 $R = \{\rho_1, \rho_2\}$
 $I = \{\}$
 $O = \{\}$

$\delta_i(\alpha_1) = (\lambda), \delta_o(\alpha_1) = (\alpha_6); \gamma_i(\alpha_1) = (\lambda), \gamma_o(\alpha_1) = (\rho_1)$
 $\delta_i(\alpha_2) = (\alpha_6), \delta_o(\alpha_2) = (\alpha_7); \gamma_i(\alpha_2) = (\rho_1), \gamma_o(\alpha_2) = (\rho_2)$
 $\delta_i(\alpha_3) = (\alpha_6), \delta_o(\alpha_3) = (\alpha_4); \gamma_i(\alpha_3) = (\rho_1), \gamma_o(\alpha_3) = (\lambda)$
 $\delta_i(\alpha_4) = (\alpha_3), \delta_o(\alpha_4) = (\alpha_7); \gamma_i(\alpha_4) = (\rho_1), \gamma_o(\alpha_4) = (\lambda)$
 $\delta_i(\alpha_5) = (\alpha_7), \delta_o(\alpha_5) = (\lambda); \gamma_i(\alpha_5) = (\rho_1, \rho_2), \gamma_o(\alpha_5) = (\lambda)$
 $\delta_i(\alpha_6) = (\alpha_1), \delta_o(\alpha_6) = ((\alpha_2), (\alpha_3)); \gamma_i(\alpha_6) = (\lambda), \gamma_o(\alpha_6) = (\lambda)$
 $\delta_i(\alpha_7) = (\emptyset, \alpha_4), \delta_o(\alpha_7) = ((\alpha_5)); \gamma_i(\alpha_7) = (\rho_2), \gamma_o(\alpha_7) = (\lambda)$

2.2 정형적인 ICN 모델



<그림 1>주문처리 관계를 나타내는 워크플로우

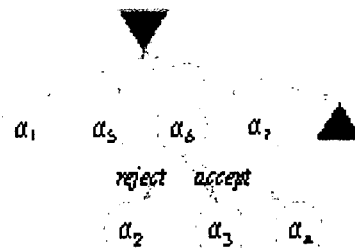
기본 ICN은 4개의 구성요소인 프로세스, 단위업무(Activity), 선후관계(Precedence)와 자료저장소로 구성되면 A를 일련의 단위 업무들의 집합이라고 하고, R을 일련의 자료 저장소의 집합이라고 할 때 수식적인 표현과 정의는 다음과 같다.

3. 제어 의존성 분석 메커니즘

3.1 CDN(Control Dependency Net)의 소개

ICN 모델을 바탕으로 제어 흐름 의존성을 분석하기 위한 CDN(Control Dependent Nets)을 정의한다. 또한 ICN으로부터 CDN을 생성하는 알고리즘도 고안하였다. 워크플로우의 제어 의존성 분석은 각 액티비티들간의 존재하는 실행 의존 관계를 정의함으로써 구현될 수 있다. 즉, ICN 모델상의 OR 노드와 AND 노드 그리고 Loop 부분이 액티비티들간의 실행 의존 관계를 형성하는데 중요한 역할을 한다.

3.2 정형적인 CDN 모델



<그림 2> CDN 모델

CDN은 프로세스의 각 단계에서의 OR 노드 또는 AND 노드를 설계하기 위해 사용된다. CDN을 통해서 각각의 액티비티간의 제어 이행 조건(control-transition conditions)을 효율적으로 발생시킬 수 있다. 제어 이행 조건은 액티비티 간의 제어 흐름 및 실행 중에

제어 흐름 동적 변경을 하기 위해 사용된다. CDN 은 $\Omega = (\varphi, \kappa^c, S, E)$ 정의할 수 있고, A 는 액티비트 집합이고, T 는 control-transition conditions 의 집합이다.

$$\Omega = (\varphi, \kappa^c, S, E)$$

φ = Control Deendency among activities
 κ = Control transition conditions
 S = initial control transition conditions
 E = final control transition conditions

<그림 1>을 기초로 정형화된 CDN 모델을 표현한다

$$\Omega = (\varphi, \kappa, S, E)$$

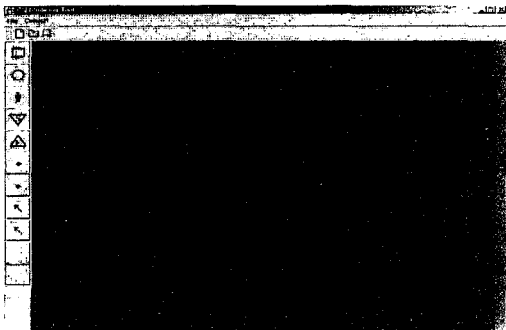
$A = \{\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6, \alpha_7, \alpha_F\}$
 $T = \{\text{default}, \text{or}(\text{hire}=\text{"reject"}), \text{or}(\text{hire}=\text{"accept"}), \text{And}(\text{default})\}$
 $S = \{\emptyset\} \quad E = \{\emptyset\}$

$\varphi_0(\alpha_1) = \{\emptyset\}$ $\varphi_0(\alpha_2) = \{\alpha_1, \alpha_5, \alpha_6, \alpha_7, \alpha_F\}$ $\varphi_1(\alpha_1) = \{\alpha_1\} \quad \varphi_0(\alpha_1) = \{\emptyset\}$ $\varphi_1(\alpha_2) = \{\alpha_6\} \quad \varphi_0(\alpha_2) = \{\emptyset\}$ $\varphi_1(\alpha_3) = \{\alpha_6\} \quad \varphi_0(\alpha_3) = \{\emptyset\}$ $\varphi_1(\alpha_4) = \{\alpha_6\} \quad \varphi_0(\alpha_4) = \{\emptyset\}$ $\varphi_1(\alpha_5) = \{\alpha_1\} \quad \varphi_0(\alpha_5) = \{\emptyset\}$ $\varphi_1(\alpha_6) = \{\emptyset\} \quad \varphi_0(\alpha_6) = \{\alpha_2, \alpha_3, \alpha_4\}$ $\varphi_1(\alpha_7) = \{\alpha_1\} \quad \varphi_0(\alpha_7) = \{\emptyset\}$ $\varphi_1(\alpha_F) = \{\alpha_1\} \quad \varphi_0(\alpha_F) = \{\emptyset\}$	$\kappa_1(\alpha_1) = \{\emptyset\} \quad \kappa_0(\alpha_1) = \{d\}$ $\kappa_1(\alpha_1) = \{d\} \quad \kappa_0(\alpha_1) = \{\emptyset\}$ $\kappa_1(\alpha_2) = \{\text{or}(\text{hire}=\text{"reject"})\}$ $\kappa_0(\alpha_2) = \{\emptyset\}$ $\kappa_1(\alpha_3) = \{\text{or}(\text{hire}=\text{"accept"})\}$ $\kappa_0(\alpha_3) = \{\emptyset\}$ $\kappa_1(\alpha_4) = \{d\} \quad \kappa_0(\alpha_4) = \{\emptyset\}$ $\kappa_1(\alpha_5) = \{d\} \quad \kappa_0(\alpha_5) = \{\emptyset\}$ $\kappa_1(\alpha_6) = \{d\}$ $\kappa_0(\alpha_6) = \{\text{or}(\text{hire}=\text{"reject"}), \text{or}(\text{hire}=\text{"accept"})\}$ $\kappa_1(\alpha_7) = \{d\} \quad \kappa_0(\alpha_7) = \{\emptyset\}$ $\kappa_1(\alpha_F) = \{d\} \quad \kappa_0(\alpha_F) = \{\emptyset\}$
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

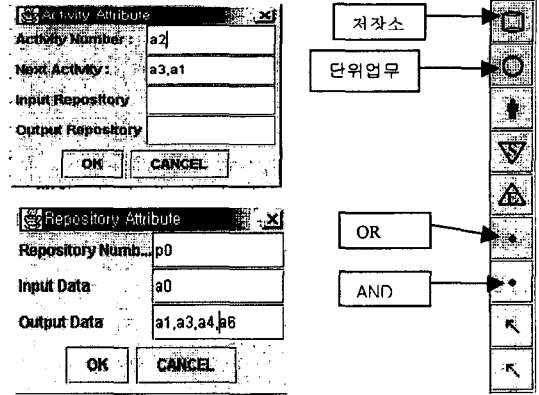
4. 구현

본 논문에서 구현한 제어 의존성 분석 도구는 <그림 3>에서 전체적인 모습을 나타내고 있으면 <그림 4>에서는 특징적인 모습을 표현하고 있다. 개발 환경은 다음과 같다.

- H/W : Intel Pentium MMX-233
- Ram : 128 MB
- S/W : Java 1.2.2
- OS 환경 : Microsoft Window2000

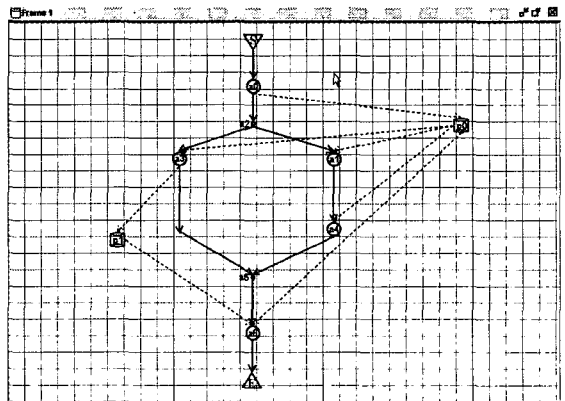


<그림 3> 전체 화면



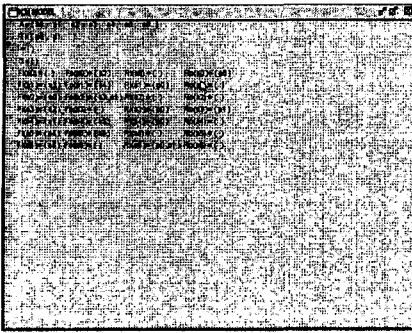
<그림 4> 다이얼 박스와 툴바

<그림 3>은 도구의 전체 화면을 표현한 그림이면, 위에 <그림 4>의 툴바는 ICN 모델에 필요한 아이템들로서 자료저장소(Repository), 단위 업무(Activity), 사용자(Actor)등이 표현되어 있다. 위에 첫번째 다이얼 박스는 단위 업무(Activity)의 속성을 나타내고 있다. 첫번째 필드는 현재 단위 업무의 이름, 두번째 필드는 다음 단위 업무의 이름, 세번째 필드는 입력자료가 저장된 자료 저장소 이름, 마지막 필드는 출력자료가 저장되는 자료저장소의 이름이다. 두번째 다이얼 박스는 자료 저장소의 속성을 나타내고 있다. 첫번째 필드는 자료 저장소의 이름, 두번째 필드는 입력 자료를 주는 단위 업무들의 이름, 세번째 필드는 출력 자료를 주는 단위 업무들의 이름을 나타낸다.



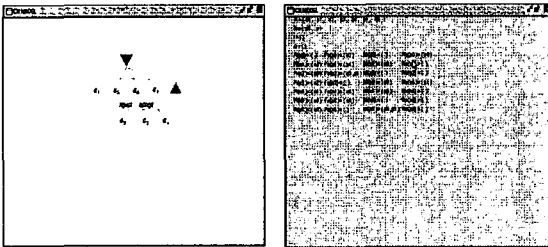
<그림 5> 그래픽적 ICN 모델

<그림 5>는 <그림 1>의 주문처리 관계를 나타내는 워크플로우를 표현한 것이다. <그림 4>의 다이얼 박스는 <그림 5>의 단위 업무와 자료 저장소의 속성을 나타내고 있는 다이얼 박스이다.



<그림 6>정형적인 ICN 모델

위에 <그림 6>은 <그림 5>에 나타내고 있는 그래픽적인 ICN 모델을 정형적인 모델로 나타낸 그림이다. 이 데이터를 가지고 CDN 모델로 나타낼 수 있다.



<그림 7>정형적인 CDN 모델과 그래픽적 표현
위에 그림은 정형적인 ICN 모델을 제어 의존성 분석 메커니즘을 이용해서 변환한 CDN 모델이다.

5. 결론 및 향후 계획

현재 버전(Version)의 도구는 컴퓨팅 도구를 이용하는 사용자들이 이 도구를 이용해서 프로세스를 모델링할 수 있고, 이를 이용해서 단위 업무 제어 의존성 분석할 수 있다. 또한 제어 의존성 분석의 결과를 통해서 향후 기업의 BPR 에 필요한 단위 업무의 경로와 경로의 정보를 제공한다. 향후 계획으로 워크플로우 모델을 속성을 분석해서 다양한 단위 업무(Activity)에 대한 속성 및 응용프로그램을 첨가해서 시뮬레이션을 통한 실시간 발생할 수 있는 문제점을 나타내게 할 것이다.

참고문헌

[1] Clarence A. Ellis and Gary J. Nutt, "Office Information System and Computer Science", Computing Surveys, Vol.12, No. 1, March 1980
 [2]Clarence A. Ellis, " Goal Based Models of Groupware" , University of Colorado , Boulder ,Colorado, USA
 [3]Kwang-Hoon Kim and Su-Ki Paik, "Actor-Oriented Workflow Model", The Second Cooperative Database Systems for Advanced Application, Wollongong Australia, March 1999
 [4]Byung-Deuk Yoo, Byung-Ok Jang, Kwang-Hoon Kim,Su-ki Paik, "Data Dependency Analysis in Transactional Workflow"

[5]Hoon Jin, Hak-seiong Kim, Kwang-hoon Kim,Su-ki Paik "A Java-Based ICM Modeling Tool"
 [6]Workflow Management Coalition Specification Document,"The Workflow Reference Model",Document Number : TC-1003 Version 1.1, Jan 1195
 [7]Kwang-Hoon, Kim,Su-Ki Paik "Actor-Oriented Workflow Model"
 [8]Kwang-Hoong Kim "Architectures for Very large Scale Workflow Management Systems"
 [9]Hyeong-Seok Hong, Jak Seong Kim , Kwang-Hoon Kim , Su-Gi Paik "A Transactional Workflow Monitoring Tool"
 [10]Woong Park , Jong-Tae Oh , Kwang-Hoon Kim , Su-Ki Paik "Control Dependency Analysis in Workflow"