

액티브 엔터프라이즈 워크플로우 그리드 아키텍처*

백 수 기**

An Active Enactment Architecture for Enterprise Workflow Grid*

Suki Paik**

■ Abstract ■

This paper addresses the issue of workflow on Grid and P2P, and proposes a layered workflow architecture and its related workflow models that are used for not only distributing workflows' information onto Grid or P2P resources but also scheduling the enactment of workflows. Especially, the most critical rationale of this paper is on the fact that the nature of Grid computing environment is fitted very well into building a platform for the maximally parallel and very large scale workflows that are frequently found in very large scale enterprises. The layered architecture proposed in this paper, which we call Enterprise Workflow Grid Architecture, is targeting on maximizing the usability of computing facilities in the enterprise as well as the scalability of its underlined workflow management system in coping with massively parallel and very large scale workflow applications.

Keyword : Enterprise Workflow Grid/P2P Architecture, Workflow Models, Massively Parallel and Very Large-Scale Workflow

1. 서 론

현재까지의 그리드 기술의 주요 응용분야는 주로 물리학, 생물학이나 기계공학 등과 같은 초고속 데이터 처리 분야에 편중되어 왔다. 하지만, 최근에 전통적인 정보처리분야에도 전자상거래를 비롯한 고객관리기술(CRM), 공급망 및 가치사슬망 관리기술(SCM), 데이터 및 응용 프로그램 통합기술(EAI), 기업자원계획기술(ERP) 등의 활성화에 따른 초대형 데이터 처리 능력의 필요성이 매우 중요하게 인식되고 있다. 마찬가지로 비즈니스 프로세스 관리(BPM) 및 워크플로우 분야에서도 매우 중요한 영역이 되어 가고 있다.

일반적인 그리드 컴퓨팅의 개념은 네트워크를 통해서 분산된 자원의 공유를 통해 응용과 서비스를 제공하는 것을 말한다[4]. 그리드의 개념적 정의는 네트워크상에서 자원의 나눔과 공유이다. 이러한 개념적 관점에서 보면 거의 모든 IT 시스템은 모두 그리드라고 할 수 있다. 잘 알려진 그리드 컴퓨팅의 예로 수 천 명의 사람들이 자신의 PC에서 사용되지 않는 프로세서를 공유함으로써, 외계로부터의 가치 있는 신호 조짐들을 광범위하게 검색하고 있는 SETI@Home 프로젝트[6]가 대표적이라고 할 수 있다. 이러한 초기 그리드 모델의 적용 사례들이 알려지면서 그리드를 단순히 PC자원들을 연결해서 슈퍼컴퓨터처럼 활용하는 것으로만 인식 하는 경우도 많다. 따라서 그동안 그리드 컴퓨팅 기술은 일부 연구소나 학계를 중심으로 발전되어 오고 있으며 아직도 계산지향적인 자연과학 그리드에 주로 사용되어 오고 있는 것이 사실이다.

반면 엔터프라이즈 그리드 컴퓨팅(Enterprise Grid Computing)은 엄밀히 말하면 그리드 컴퓨팅을 학술 또는 연구 목적 보다는 기업이나 조직 환경에서 활용하는 것을 말한다[3]. 조직에서 엔터프라이즈 그리드 컴퓨팅을 도입하는 것과 연관된 요구사항과 해결 과제들은 특히 운영적인 측면에서 조직마다 다르다.

본 논문에서는 비즈니스 프로세스 관리 및 워크

플로우 분야에서도 매우 중요한 영역이 되어가고 있는 그리드 기술을 적용하는 방안으로 엔터프라이즈 그리드의 개념을 도입하여, 액티브 엔터프라이즈 워크플로우 그리드 아키텍처를 제안하고자 한다.

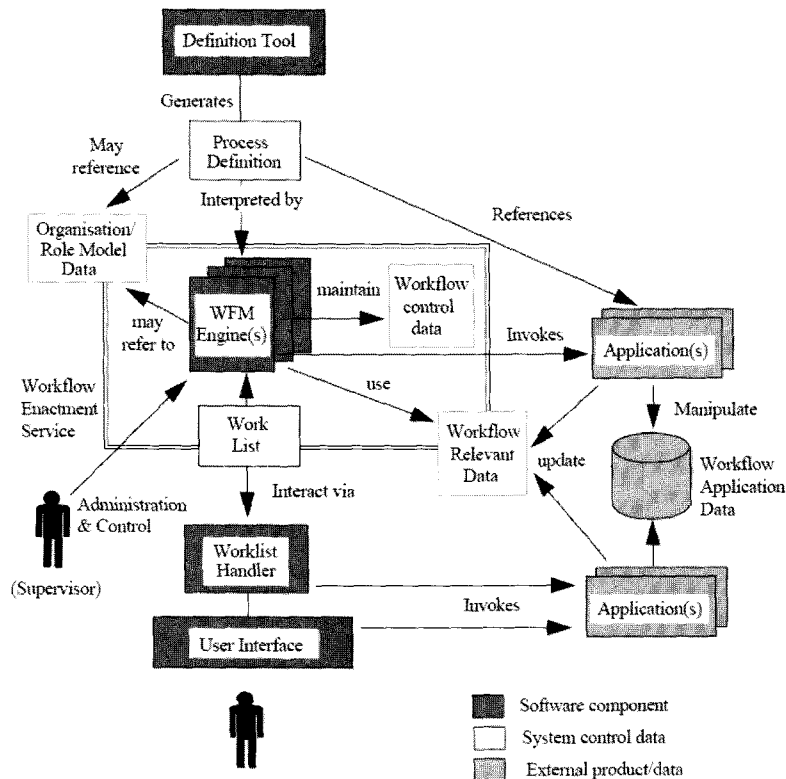
본 연구를 통해 제안된 아키텍처는 궁극적으로 현재 초대형 워크플로우 응용분야에서 이슈화 되고 있는 워크플로우의 구조적인 문제와 대용량의 워크케이스 실행문제를 해결해 내기 위한 초대형 워크플로우 아키텍처가 될 수 있다.

본 논문은 워크플로우 모델과 밀접한 관련이 있다. 워크플로우 모델의 변화를 기존의 업무 중심의 모델에서 사람 중심의 모델로 변화시켜, 엔터프라이즈 그리드를 적용하는 것을 의미한다. 따라서, 정형적 표현이 가능한 워크플로우 모델인 ICN 워크플로우 모델링 방법을 기반으로 워크플로우 모델을 구성하고, 워크플로우 모델의 조직과 관련된 역할 기반 모델, 수행자 기반 워크플로우 모델을 구성하고자 하는 알고리즘을 제안하고 적용하고자 한다.

2. 이론적 배경 및 관련연구

WfMC에서 제안한 참조 모델[7]에서의 워크플로우 관리 시스템은 하나 이상의 워크플로우 엔진을 포함한 워크플로우 제정서비스(Workflow Enactment Service)를 기준으로 한 중앙 집중형(Centralized) 워크플로우 모델이다. 프로세스 정의 시점에 정의한 워크플로우 모델을 중앙에 집중되어 있는 워크플로우 제정서비스에 전달하게 되면 각 클라이언트들은 수행이 완료 된 후 전이정보를 얻기 위하여 모든 정보를 갖고 있는 제정서비스에 요청을 하여 다음 업무의 수행 경로를 결정한다.

네트워크의 발달과 인터넷을 기반으로 하는 환경의 출현은 워크플로우를 사용할 조직의 거대화와 조직에서 사용되는 프로세스, 즉 워크플로우의 거대화를 이루게 되며, 이러한 워크플로우의 거대화를 통하여 나타난 새로운 동향이 초대형 워크플



[그림 1] 중앙 집중형 워크플로우 관리 시스템

로우(Very Large Scale Workflow)[5]이다. 초대형 워크플로우는 컴퓨팅 환경과 기업 환경의 발전으로 인하여 발생하는 대량의 워크플로우 인스턴스를 관리하는 능력을 가진 워크플로우를 말하며 워크플로우 시스템을 사용하는 조직이 점차 거대화되어 조직에서 사용되는 워크플로우의 종류가 많아지고 대량의 작업이 일어남에 따라서 발생하는 대량의 워크플로우 인스턴스를 처리하는 것에 목적을 두는 워크플로우이다.

이러한 초대형 워크플로우는 최소한의 자원을 사용하면서도 최대한의 효율성과 성능을 내기 위해 분산 컴퓨팅 환경을 선택하게 되며 분산 컴퓨팅 방식의 그리드 워크플로우의 형태가 등장하게 된다. 현재까지 여러 가지의 분산 컴퓨팅 방식의 그리드 컴퓨팅이 등장하였는데, 각각의 특징을 정리해 보면, 다음 <표 1>과 같이 정리해 볼 수 있

다. 각각은 그리드 워크플로우 또는 워크플로우 그리드 관련 프로젝트의 이름, 조직, 그리드를 구성하기 위한 기반 기술, 어떻게 그리드 환경에 통합시킬 것인지, 어떤 분야에 적용하는지, 어떤 허가가 있어야 하는지를 정리한다.

3. 액티브 엔터프라이즈 워크플로우 그리드 아키텍처

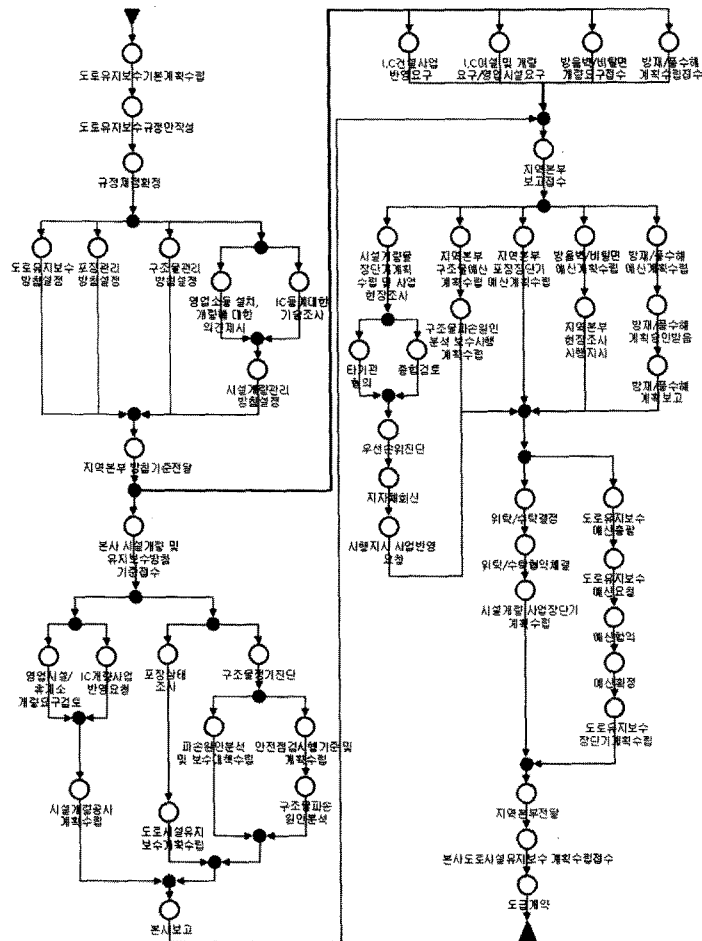
[그림 2]는 한국도로공사내 존재하는 프로세스를 도식화 해 본 것이다. 프로세스의 모양이 매우 복잡한 것을 볼 수 있다. 보이는 부분의 프로세스 뿐 아니라, 업무 내 존재하는 프로세스까지 포함하면, 100여 개의 업무가 하나의 프로세스 내에 존재할 수 있음을 'e-비즈니스 프로세스 기반의 차세대 전자결재시스템 발전방안[1]'이라는 연구를

〈표 1〉 그리드 관련 워크플로우 관리 시스템 프로젝트

Name	Organization	Prerequisite	GRID Integration	Applications	Availability
DAGMan [condor1]	University of Wisconsin-Madison, USA. http://www.cs.wisc.edu/condor/dagman	Condor	Condor which can run on top of Globus Toolkit version 2 (GT2)	Compute-intensive	GPL (General Public License)
Pegasus [pegasus2]	University of Southern California, USA. http://pegasus.isi.edu	Condor DAGMan, Globus RLS.	Condor and Globus.	Targeted for data-intensive, but supports other types.	GTPL (Globus Toolkit Public License)
Triana [triana]	Cardiff University, UK. http://www.trianacode.org/	Grid Application Toolkit (GAT)	GAT (JXTA, Webservices, Globus)	Compute-intensive	the Apache Software License
ICENI [iceni]	London e-Science Centre, UK. http://www.lesc.ic.ac.uk/iceni/	Globus Toolkit	Jini, JXTA, Globus	Compute-intensive	ICENI Open Source Code Licence
Taverna [taverna]	Collaboration between several European Institutes and industries. http://taverna.sourceforge.net/	Java 1.4+	Web services, Soaplab, local processor, BioMoby, etc.	Service Grids	GNU Lesser General Public License (LGPL)
GridAnt [gridant]	Argonne National Laboratory, USA. http://www.cogkit.org/	Apache Ant, Globus Toolkit	GT2, GT3, GT4	Client controllable workflow applications	GTPL
GrADS [grads]	Collaboration between several American Universities. http://www.hipersoft.riace.edu/grads/	Globus Toolkit, Autopilot, NWS	Globus, Parallel Systems (e.g. MPI)	Computeintensive and communication-intensive applications with MPI components	Not yet available in public
GridFlow [gridflow]	University of Warwick, UK http://www.dcs.warwick.ac.uk/research/hpsg/workflow/workflow.html	Agent-based Resource Management System, PACE Toolkit, Titan	Parallel Systems (e.g. MPI and PVM)	MPI and PVM based components	Not yet available in public
Unicore [unicore]	Collaboration between German research institutions and industries http://www.unicore.org	Unicore middleware	Unicore	Computational-intensive and MPI components	Community Source License
Gridbus workflow [gridbus]	The University of Melbourne, Australia. http://www.gridbus.org	Globus Toolkit	GT2	Computationaland Data-intensive	GPL

〈표 1〉 그리드 관련 워크플로우 관리 시스템 프로젝트(계속)

Name	Organization	Prerequisite	GRID Integration	Applications	Availability
Askalon [askalon]	University of Innsbruck http://dps.uibk.ac.at/askalon	Globus Toolkit	GT2, GT4, WSRF, Web services	Performance oriented applications	GTPL
Karajan [karajan]	Argonne National Laboratory http://www.cogkit.org	Java 1.4	GT2, GT3, GT4, Condor, runtime exec, ssh, WebDAV	Those required to access Grid middleware	GTPL
Kepler [kepler]	A cross-project collaboration. http://keplerproject.org	Java	Globus, Storage Resource Broker(SRB), EcoGrid, Web services	Scientific workflow applications	UC Berkeley License



[그림 2] 한국도로공사 프로세스 예제

통해 알려졌다. 이렇게 많은 업무를 하나의 프로세스, 그리고 각 프로세스가 발생시키는 업무의 반복은 전통적인 방식의 아키텍처를 따르고 있는 워크플로우 관리 시스템이 처리하기에는 어려움이 많을 것이다. 그리고, 시스템 자원을 분배하고, 효율적으로 관리하기 위한 고전 그리드 기술을 적용하기에도 어려움은 있어 보인다. 따라서, 기존의 워크플로우 관리 시스템과 엔터프라이즈 그리드 기술의 접목을 적절히 고려한 워크플로우 그리드 아키텍처가 필요하다.

위와 같은 요구로, 본 논문에서는 [그림 3]과 같은 엔터프라이즈 워크플로우 그리드 아키텍처를 제시하고자 한다. 아키텍처의 큰 모습은 전통적인 워크플로우 관리 시스템의 모습을 띄고 있지만, 실제 노드를 분산시키는 환경은 그리드 기술을 기반으로 하고 있다. 워크플로우 관리 시스템을 이용하고자 하는 워크플로우 사용자 및 관리자는 워크플로우 관리 시스템내 존재하는 하나의 노드를 이용하고 있으며, 이용하는 각 노드들이 모여, 하

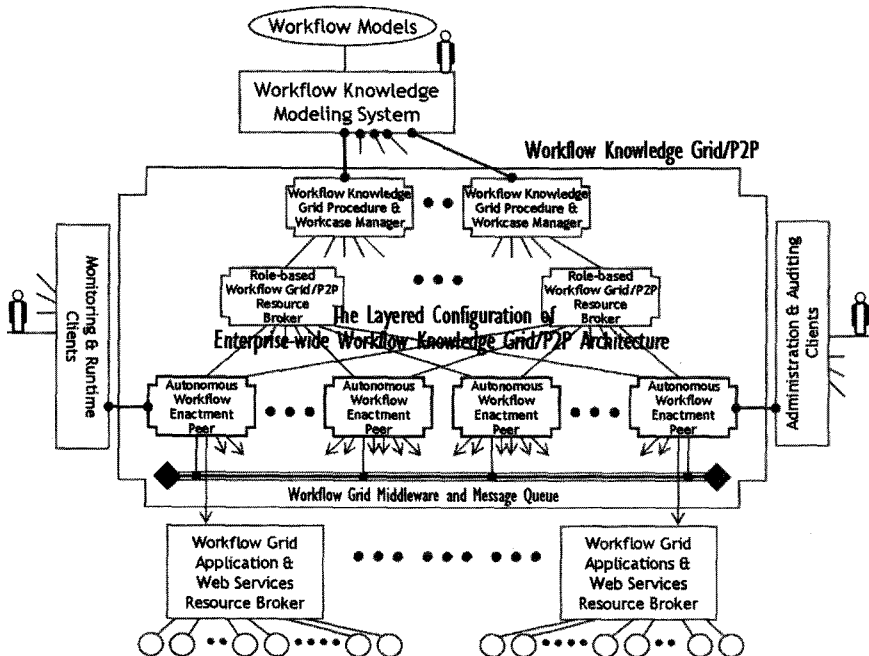
나의 워크플로우 관리 시스템을 구성하고 있다. 또한 각 노드의 자원을 적절히 분배해 줄수 있는 브로커와 각 브로커를 관리해주는 워크케이스(프로세스 인스턴스) 매니저가 존재하게 된다.

3.1 ICN(Information Control Nets)

본 연구에서 액티브 엔터프라이즈 워크플로우 그리드 아키텍처를 제안하기 위해, 기반이 될 수 있는 워크플로우 모델링 방법으로 ICN(Information Control Net) 워크플로우 모델링 방법을 이용하고자 한다. 본 연구에서 기본적으로 사용하는 ICN 모델[2]은 워크플로우를 표현할 수 있는 기호들의 집합으로 기본적인 ICN의 7튜플은 $\Gamma = (\delta, \gamma, \varepsilon, \pi, \kappa, I, O)$ 과 같이 표현되며, 각각의 정의는 다음과 같다.

- $\delta = \delta_i \cup \delta_o$

여기서, $\delta_o : A \rightarrow \mathcal{P}(A)$ 은 하나의 액티비티를 후행하는 액티비티들의 집합에 연결하는



[그림 3] 엔터프라이즈 워크플로우 그리드 아키텍처

관계를 나타내며 $\delta_i : A \rightarrow \mathcal{P}(A)$ 은 하나의 액티비티를 선행하는 액티비티들의 집합에 연결하는 관계를 나타내는 관계이다.

- $\gamma = \gamma_i \cup \gamma_o$

여기서, $\gamma_o : R \rightarrow \mathcal{P}(A)$ 은 하나의 액티비티를 후속하는 액티비티 집합들을 출력 저장소들의 집합과 연결하는 것 중 하나이며, $\gamma_i : R \rightarrow \mathcal{P}(A)$ 은 하나의 액티비티를 선행하는 액티비티 집합들을 입력 자료저장소들의 집합과 연결하는 관계를 나타내는 것 중 하나이다.

- $\varepsilon = \varepsilon_a \cup \varepsilon_p$

여기서, $\varepsilon_a : P \rightarrow \mathcal{P}(A)$ 는 하나의 액티비티를 조합된 역할들의 집합에 액티비티를 단일 값으로 연결하는 것이고, $\varepsilon_p : A \rightarrow \mathcal{P}(P)$ 는 조합된 액티비티들의 집합을 역할에 단일 값으로 연결하는 것이다.

- $\pi = \pi_p \cup \pi_c$

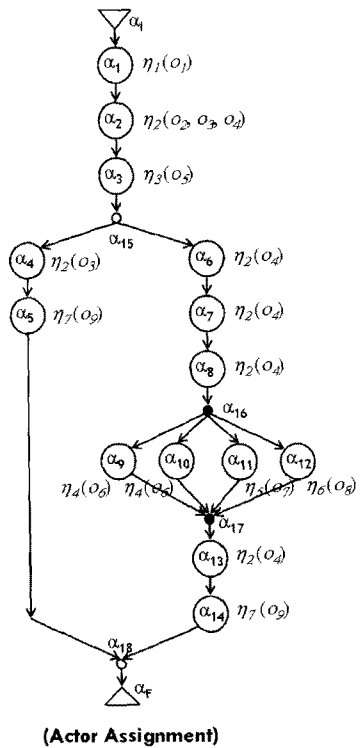
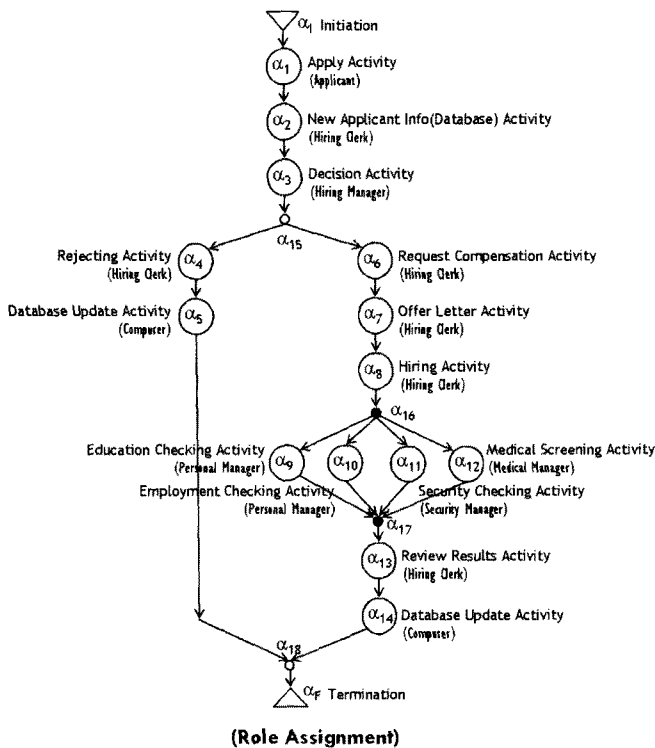
여기서, $\pi_p : C \rightarrow \mathcal{P}(P)$ 조합된 수행자들의 집합을 역할에 단일 값으로 연결하는 것이고, $\pi_c : P \rightarrow \mathcal{P}(C)$ 는 조합된 역할들을 수행자에 단일 값으로 연결하는 것이다.

- $\kappa = \kappa_i \cup \kappa_o$

여기서, κ_i 는 $a \in A$ 일 때, $(\delta_i(a), a)$ 사이에서, 제어흐름(T) 조건들의 집합이고, κ_o 는 $a \in A$ 일 때, $(a, \delta_o(a))$ 사이에서, 제어흐름(T) 조건들의 집합이다. 이때 집합 $T = \{\text{default}, \text{or}(\text{conditions}), \text{and}(\text{conditions})\}$.

- I는 초기 입력되는 자료 저장소들의 유한 집합이며, ICN의 실행 전에 어떠한 외부의 프로세스에 의해서 로드 되어야 한다.

- O는 마지막으로 출력되는 자료저장소들의 유한 집합이며 ICN의 실행 후에 어떠한 외부의 프로세스에 의해서 이용되는 정보를 포함하고 있다고 가정한다.



[그림 4] ICN 기반 워크플로우 모델(고용 프로세스)

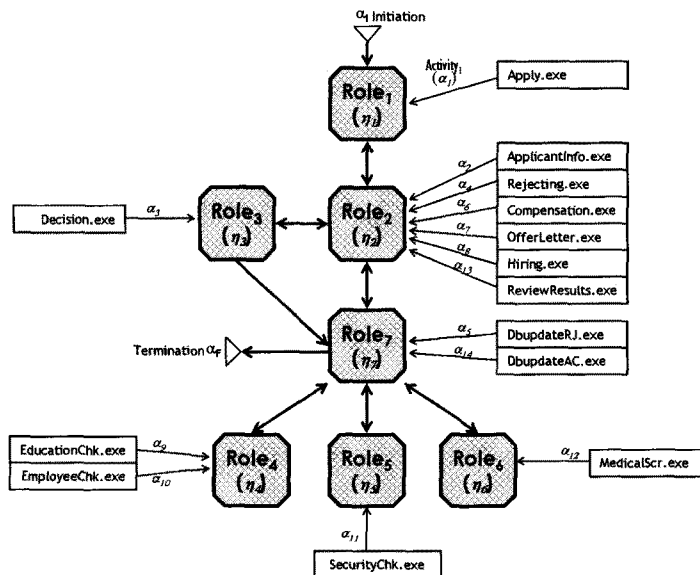
3.2 역할기반 워크플로우 모델

액티브 엔터프라이즈 워크플로우 그리드 아키텍처를 구성하기 위해 가장 중요한 요소 중 하나는 역할기반 워크플로우 모델이다. 역할기반 워크플로우 모델은 기존 액티비티 기반의 워크플로우 모델로 볼 수 있는 ICN 워크플로우 모델링 방법으로부터 알고리즘을 통해 추출해 낼 수 있으며, 기본적인 역할기반 워크플로우 모델의 4 튜플은 $R = (\xi, \Theta, S, E)$ 과 같이 표현되며, 각각의 정의는 다음과 같다.

- $\xi = \xi_i \cup \xi_o$
 $\xi_o : P \rightarrow \mathcal{P}(a \in P)$ 조합된 역할들의 집합을 후행자 역할에 다중 값으로 연결, $\xi_i : P \rightarrow \mathcal{P}(a \in P)$ 조합된 역할들의 집합을 선행자 역할에 다중 값으로 연결하는 것이다.
- $\Theta = \Theta_i \cup \Theta_o$
 Θ_o 은 액티비티들의 집합이며, 수행자의 선행자와 수행자를 관계를 의미하고, Θ_i 은 액티비티들의 집합이며, 수행자의 후행자와 수행자의 관계를 의미한다.
- S 는 초기 액티비티들의 유한집합이며, 외부

〈표 2〉 역할기반 워크플로우 모델 생성 알고리즘

Input An ICN, $\Gamma = (\delta, \chi, \chi, \varepsilon, \pi, \kappa, I, O)$;
 Output A Role-based Workflow Net, $R = (\xi, \Theta, S, E)$
 Begin Procedure
 For ($\forall \in A$) Do
 Begin
 Add all members of $\varepsilon_p(a)$ To $\xi_i (e_p(\delta_o(a)))$;
 Add all members of $\varepsilon_p(\delta_o(a))$ To $\xi_i (e_p(a))$;
 /*Getting an activity on the arc between two roles.*
 Add To $\Theta_i (e_p(\delta_o(a)))$;
 Add To $\Theta_o (e_p(a))$;
 End
 End Procedure



[그림 5] 역할기반 워크플로우 모델(고용 프로세스)

프로세스에 의해 로드된다.

- E는 마지막 액티비티들의 유한집합이며, 외부 프로세스에 의해 로드된다.

역할기반 워크플로우 모델은 위의 정의를 통해 정의되며, <표 2>와 같이 역할기반 워크플로우 모델 생성 알고리즘을 통해 생성해 낼 수 있다.

[그림 5]는 ICN 기반 고용 프로세스로부터 역할기반 워크플로우 모델 정의와 알고리즘으로부터 유도할 수 있는 고용 프로세스의 역할기반 워크플로우 모델이다.

3.3 수행자 기반 워크플로우 모델

본 논문에서 제안하고 있는 액티브 엔터프라이즈 워크플로우 그리드 아키텍처를 구성하기 위해 가장 중요한 요소 중 또 다른 하나는 수행자 기반 워크플로우 모델이다. 수행자 기반 워크플로우 모델도 역할기반 워크플로우 모델과 마찬가지로 ICN 워크플로우 모델링 방법으로부터 알고리즘을 통해 추출해 낼 수 있다. 기본적인 수행자 기반 워크플로우 모델의 4 튜플은 $\Lambda = \langle \sigma, \psi, \zeta^a, I \rangle$ 과 같이 표현되며, 각각의 정의는 다음과 같다.

- I는 ICN의 프로세스 실행 절차에 따라 수행자 기반 워크플로우 모델로 초기화 시키고, 인스턴스를 발생시키도록 도움을 주는 객체 또는 도움을 주는 객체 그룹이 될 수 있다.
- $\sigma = \sigma_i \cup \sigma_o$
 $\sigma_o : C \rightarrow \mathcal{P}(O \in C)$ 은 수행자와 후행자를 다중 값으로 연결하는 것을 의미하고, $\sigma_i : C \rightarrow \mathcal{P}(O \in C)$ 은 수행자와 선행자를 다중 값으로 연결하는 것을 의미한다.
- $\psi = \psi_i \cup \psi_o$
 ψ_o 는 액티비티에서 수행자로의 관계를 의미하고, ψ_i 는 수행자에서 액티비티로의 관계를 의미한다.
- $\zeta^a = \zeta_i^a \cup \zeta_o^a$
 ζ^a 는 수행자들 간의 노드 전이를 의미한다.
 ζ_o^a 수행자가 일을 처리한 후 후행시키는 수행자를 의미하고, ζ_i^a 는 수행자가 일을 처리하기 전 선행했던 수행자를 의미한다.

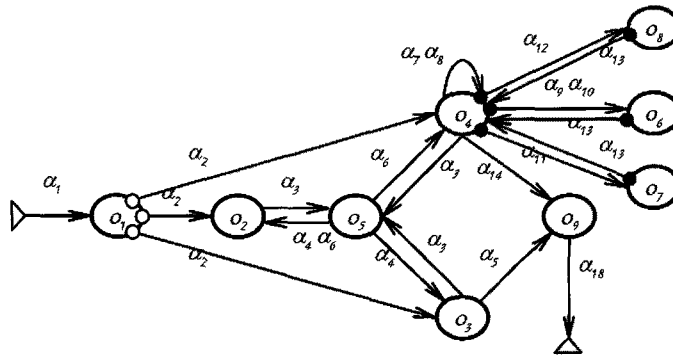
수행자 기반 워크플로우 모델은 위의 정의를 통해 정의되며, <표 3>과 같이 표현할 수 있는 수행자 기반 워크플로우 모델 생성 알고리즘을 통해 생성해 낼 수 있다.

<표 3> 수행자 기반 워크플로우 생성 알고리즘

```

Input An ICN,  $\Gamma = \langle \delta, \gamma, \chi, \varepsilon, \pi, \kappa, I, O \rangle$ ;
Output An Actor-based Workflow Net,  $\Lambda = \langle \sigma, \psi, \zeta^a, I \rangle$ ;
Begin Procedure
  For (  $\forall a \in A$  ) Do
    Begin
      Add all members of  $\pi_c(e_p(a))$  To  $\sigma_i$  (each member of  $\pi_c(e_p(a))$ );
      Add all members of  $\pi_c(e_p(\delta_o(a)))$  To  $\sigma_o$  (each member of  $\pi_c(e_p(a))$ );
      /*Getting all arcs labeled by (activity, relationship) between two actors.*/
      Add (a, relationship of  $e_p(a)$ ) To  $\omega_i$  (all members of  $\pi_c(e_p(\delta_o(a)))$ );
      Add (a, relationship of  $e_p(\delta_o(a))$ ) To  $\omega_o$  (all members of  $\pi_c(e_p(a))$ );
    End
    /*Getting actor-transition conditions.*/
  For (  $\forall o \in C$  ) Do
    Begin
      Assign  $\omega_i(o)$  To  $\zeta_i^a(o)$ ;
      Assign  $\omega_o(o)$  To  $\zeta_o^a(o)$ ;
    End
  End Procedure

```



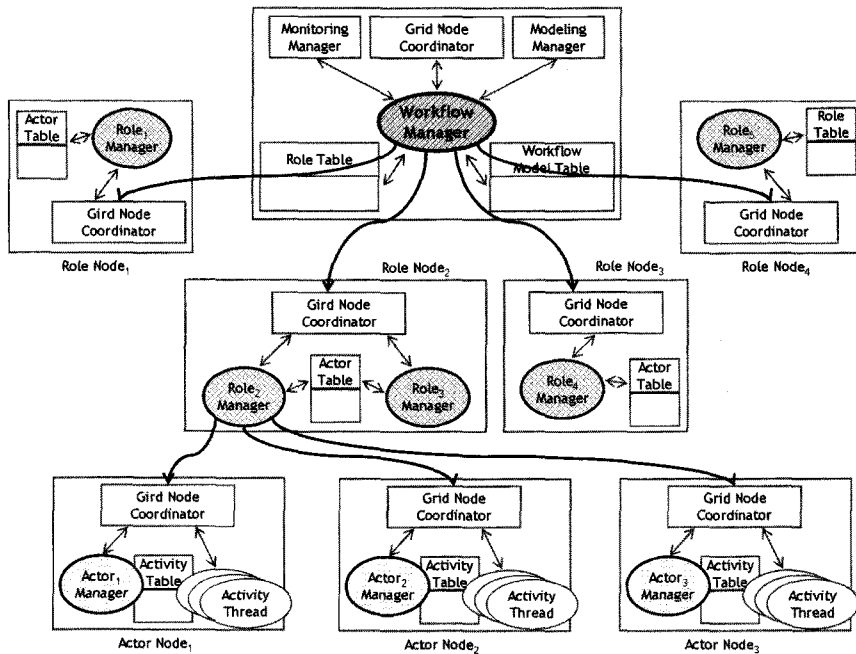
[그림 6] 고용 프로세스의 수행자 기반 워크플로우 모델

[그림 6]은 고용 프로세스의 ICN 모델을 기반으로 하여, 본 연구를 통해 정의된 수행자 기반 워크플로우 모델로 적용된 모습을 보여준다.

ICN을 기반으로 한 워크플로우 모델은 알고리즘을 통해 역할기반 워크플로우 모델과, 수행자 기반 워크플로우 모델을 생성하고, 생성된 모델을 중심으로 워크플로우 프로세스 관리자가 각 노드를 관리함으로써, 액티브 엔터프라이즈 워크플로

우 그리드를 실현할 수 있다. [그림 7]은 엔터프라이즈 워크플로우의 분산을 보여주며, 워크플로우 관리자가 그리드 노드를 관리하면서, 액티브 엔터프라이즈 워크플로우를 실현해 낼 수 있는 모습을 보여준다. 물론 각 노드는 역할기반 워크플로우 모델과 수행자 기반 워크플로우 모델을 통해 생성된 노드간의 관련성을 통해 관리된다.

또한, [그림 7]에서 나타내는 바와 같이 실제 업



[그림 7] 엔터프라이즈 워크플로우의 분산

무들은 수행자가 존재하고 있는 컴퓨팅 노드로 분산되며, 수행자가 처리하고자 하는 업무만이 수행자 노드에 존재하게 된다. 여기서 수행자 기반 워크플로우 모델을 통해, 업무의 흐름을 제어할 수 있고, 이는 역할 기반 워크플로우 모델로 구성될 수 있는 상위 노드로, 그리고 역할 모델을 제어할 수 있는 워크플로우 모델로 구성되어, 분산됨을 알 수 있다.

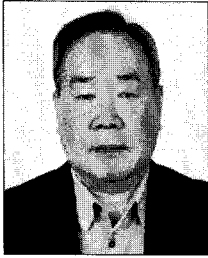
4. 결 론

본 논문에서는 현재 초대형 워크플로우 응용분야에서 이슈화 되고 있는 워크플로우의 구조적인 문제와 대용량의 워크케이스 실행문제를 해결해내기 위한 초대형 워크플로우 아키텍처를 제시하였다. 초대형 워크플로우 아키텍처의 한 모습으로 액티브 엔터프라이즈 워크플로우 그리드 아키텍처를 제시하였으며, 액티브 엔터프라이즈 워크플로우 그리드 아키텍처에서 필요한 세 가지 모델인 ICN 기반 워크플로우 모델, 역할 기반 워크플로우 모델, 그리고 수행자 기반 워크플로우 모델을 제시하였다. 또한, ICN 기반에서 역할 워크플로우 모델과 수행자 기반 워크플로우 모델을 알고리즘을 통해 생성할 수 있는 방법을 제시하였으며, 각 알고리즘을 적용하여 역할 기반 워크플로우 모델과 수행자 기반 워크플로우 모델이 구성 되었을때, 실제 어떻게 노드의 분산 환경을 이룰 수 있는지에 대한 가능성을 보여 주었다.

참 고 문 헌

- [1] 유혁재, 한승혁, 정재우, 이정훈, 안형진, 박민재, 원재강, 김광훈, "한국도로공사의 차세대 정보기술 인프라를 위한 e-비즈니스 프로세스 기반 차세대 전자결재 시스템 적용 및 발전 방안 연구", 「최종연구보고서」, 2004.
- [2] Clarence A. Ellis and Gary J. Nutt, "ICNs Revisited and Revised", *Technical Report*, 1996.
- [3] Global Grid Forum(GGF), <http://www.ggf.org>.
- [4] I. Foster, C. Kesselman, and S. Tuecke, "The Anatomy of the Grid : Enabling Scalable Virtual Organizations", *International J. Supercomputer Applications*, Vol.15, No.3 (2001).
- [5] Kwang-Hoon Kim, "Architecture for Very large scale workflow management systems", Ph.D Thesis, *Computer Science Department, University of Colorado at Boulder*, 1998.
- [6] SETI@home project : <http://setiathome.berkeley.edu/>.
- [7] Workflow Management Coalition Specification Document, "The Workflow Reference Model", Version 1.1, 1994.

◆ 저 자 소 개 ◆



백 수 기 (skpaik@kgu.ac.kr)

연세대학교에서 학사를 동국대학교에서 석사, 박사를 취득했으며, 현재 경기대학교 컴퓨터과학전공 교수로 재직하고 있다. 주요 관심분야는 컴퓨터 네트워크, 데이터구조 및 알고리즘, 데이터베이스, 그리고 워크플로우 관리 기술 등이다.