

# 규칙 기반(Rule-Based) 비즈니스 응용 개발 및 운영 시스템에 관한 연구

한우용\*, 손주찬\*, \*함호상, 강지훈\*\*

\*한국전자통신연구원 인터넷 컴퓨팅 연구부\*\*충남대학교 컴퓨터 과학과  
{wyhan<sup>o</sup>, jcsohn, hsham }@etri.re.kr, jhkang@cs.cnu.ac.kr

A Study on a System to develop Rule-base applications

Woo-Yong Han\*, Joo-Chan Sohn\*, Ho-Sang Ham\*,Jee-hwoon Kang\*\*

\*Internet Computing Department, ETRI \*\*Dept. of Computer Science,  
Chung-Nam National University

## 요 약

본 논문에서는 미래에 많은 기업들이 경쟁력 향상을 위하여 도입할 것으로 예상되는 규칙 기반의 비즈니스 응용 시스템을 개발하고 운영하는데 필요한 환경을 제공하는 시스템의 구조를 제안하고 그리고 규칙 기반 응용들에게 상호 운영성을 제공하는데 필요한 규칙 표현의 표준화 동향들을 기술하고 있다.

### 1. 서론

미래의 정보 기술 동향을 예측하는 많은 기관(예:가트너)들에서 나오는 연구보고서 자료에 의하면 향후 70% 이상의 많은 기업들이 경쟁력 향상을 위하여 기존의 시스템 개발 방법과는 다른 방법으로 개발되고 도입, 운영되는 규칙 기반의 응용 시스템을 도입하여 자신들의 많은 업무분야에 이용할 것으로 예측되고 있다.

기존의 응용 시스템들은 모든 업무 처리 내용들을 시스템 내부에 코드화함에 따라 회사의 업무와 정책 변경시 비즈니스 업무 분석가가 그 변경된 내용을 프로그래머에게 전달하고 이를 코드에 반영하는 과정에서 오류가 발생할 가능성과 업무 변경시 즉시 대응하지 못하는 문제점을 갖고 있다. 규칙 기반 비즈니스 응용 개발 시스템은 위의 문제점들을 해결할 수 있는 방법을 제공한다.

즉 업무의 처리 내용 중 영업 전략과 같이 비즈니스 상황에 따라 자주 바뀔 수 있는 회사의 중요한 정책 결정 사항들을 처리하는 프로세스는 별도의 규칙 서비스로 구현한다. 이 기술의 이점은 프로그래밍 기술을 갖고 있지 않은 일선 비즈니스 관리자들도 비즈니스 정책들을 규칙으로 손쉽게 작성하고 수정할 수 있어 업무 내용이 변경되더라도 프로그래머의 지원 없이도 변경된 비즈니스 업무를 정보 시스템에 즉시 반영할 수 있다는 점이다.

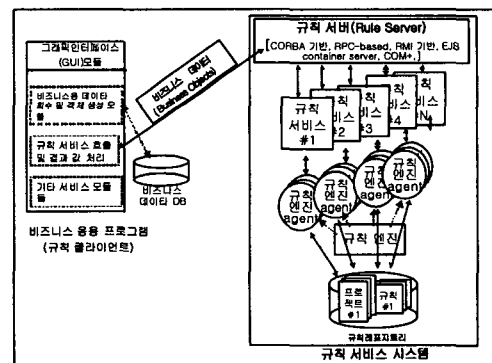
위에서 언급한 것처럼 잦은 비즈니스 업무 변경에 신속하게 대처할 수 있는 규칙 기반의 시스템은 미래에 많은 기업들이 도입할 것으로 쉽게 예상할 수 있다.

### 2. 규칙 기반 비즈니스 응용 개발 및 운영 시스템

규칙 기반의 비즈니스 응용 시스템의 개발 및 운영 환경을 제공하는 시스템은 비즈니스 규칙을 편집, 관리, 도입, 운영 및 시험하는 등의 서비스를 제공하는 개발 환경 시스템과 규칙 기반의 응용 시스템을 운영하는데 필요한 운영 시스템으로 이루어져 있다.

#### 2.1 운영 환경의 구성 요소

아래 <그림 1>에서는 규칙 기반의 응용 시스템을 운영하는데 필요한 환경을 제공하는 규칙 서비스 시스템의 구조와 그 환경에서 비즈니스 규칙 기반 응용 시스템과 규칙 서비스 시스템간의 동작 구조를 보여주고 있다.



<그림 1> 규칙 기반 비즈니스 응용과 규칙 서비스 시스템의 운영 구조

규칙 서비스 시스템은 규칙 기반의 응용 시스템으로부터 규칙 서비스 요청을 받아 해당 비즈니스 규칙 서비스를 호출하는 규칙 서버와 여러 개의 규칙 클라이언트 응용으로부터 오는 서비스 요청들을 동시에(concurrently) 처리하기 위하여 여러 개의 규칙 엔진 에이전트(agent)들을 생성하여 규칙을 처리하는 규칙 서비스가 사용하는 규칙 추론 엔진과 규칙들을 저장하고 있는 규칙 레포지토리로 구성되어 있다.

☞ 규칙 서버

이 서버는 보다 빠른 규칙 처리 서비스를 클라이언트에게 제공하기 위한 것이다. 그리고 다양한 서버 플랫폼에서 규칙 서비스를 제공할 수 있도록 분산 객체 표준 시스템인 CORBA 의 서버로, 원격 프로세스 호출(RPC) 서버로, 원격 메서드 호출(RMI) 서버로, EJB container 서버로 규칙 서비스를 제공한다.

이 규칙 서버는 다음과 같은 유형으로 서비스를 제공한다.

☞ Stateless 서버 유형

클라이언트가 한번 요청 메시지를 보내고 그 결과를 받으면 세션이 종료됨. 대량의 데이터를 동시에 처리하는 배치 프로세싱 처리할 때 이용됨

☞ 비동기식 stateful 서버 유형

동일한 세션의 컨텍스트(context)에서 클라이언트와 여러 번의 상호동작(interaction)을 할 수 있음. 클라이언트 응용이 최종 사용자에게 interactive 한 서비스를 제공할 때 이 서비스를 사용함

☞ 동기식 stateful 서버 유형

세션 기반으로 상호동작을 하는 것은 위와 동일하나 이 유형의 서비스는 여러 클라이언트로부터 서비스 요청을 받는 경우 데이터의 상태를 유지한다.

이 규칙 서버는 런타임시 클라이언트 비즈니스 응용으로부터 오는 비즈니스 객체 데이터들을 규칙 처리 엔진에서 사용하는 데이터 타입으로 자동 변환하고 그와 반대 방향의 데이터 변환도 수행한다. 그리고 서비스 agent dispenser 를 이용하여 규칙 서비스에 규칙 엔진 에이전트들을 배정하는 일을 수행한다.

☞ 규칙 서비스

클라이언트 응용들에게 아래와 같은 유형의 규칙 서비스를 제공한다.

☞ 함수 호출기(function invoker) 유형

클라이언트는 규칙 서비스에 있는 어떤 함수를 호출한다. 사전에 이 함수는 entry point로 configuration 파일에 지정되어 있어야 한다.

☞ 규칙세트 호출기(RuleSet invoker) 유형

클라이언트가 규칙 서비스에 있는 규칙세트를 호출할 때 사용하는 서비스 유형으로 사전에 호출될 ruleset 은 규칙 서비스의 entry point로 configuration 되어 있어야 한다.

☞ 객체 매퍼(Object Mapper) 유형

클라이언트가 규칙 서비스를 호출할 때 비즈니스 객체를 전달한다. 규칙처리는 그 객체 혹은 객체 배열들의 규칙 서비스의 내부 데이터로 매핑이될 때 시작된다.

☞ 이벤트 포스터(Event Poster) 유형

규칙 서비스가 어떤 이벤트를 감지했을 때 그 규칙 서비스가 호출된다. Ruleflow 를 사용할 때 사용하며 entry point method 의 매개 변수 유형은 ruleflow 가 기다리는(listening)하는 이벤트에 상응한다.

☞ 규칙 엔진

이 규칙 엔진은 규칙들과 비즈니스 객체들을 입력 받아 규칙의 조건과 데이터를 비교하여 규칙의 조건에 맞는 데이터가 있으면 규칙의 동작부(action part)에 기술해 놓은 함수를 수행한다. 이 규칙 엔진은 인공 지능 분야에서 수년간 연구 개발을 통하여 나온 많은 추론 엔진들중의 하나인 Jess(Java Expert Sytem Shell)와 같은 규칙 기반의 추론 엔진들을 활용하여 규칙 서비스 시스템을 구현하는 것도 하나의 방법일 수 있다.

☞ 규칙 레포지토리

레포지토리는 파일 기반, JDBC, LDAP 을 이용하여 구현할 수 있다.

☞ 파일 기반의 레포지토리에 저장되는 내용들

☞ 비즈니스 규칙들을 만들 때 필요한 비즈니스 객체 클래스와 규칙들의 그룹(규칙세트)을 정의한 파일들

☞ 규칙 흐름을 표현한 내용을 포함하고 있는 파일들

☞ 위의 파일들의 이름과 그런 파일에 있는 내용을 내부 처리 데이터 구조로 만드는데 필요한 자바 클래스 정보를 포함하고 있는 프로젝트 파일들

☞ 규칙 서버 유형, 서비스 유형들에 관한 서버 구성(configuration) 정보 파일

☞ 규칙 서비스를 실행에 필요한 원천 코드 파일들

2.2 통합형 개발 환경(Integrated Development Environment)

이 도구에서는 다른 개발 도구와 마찬가지로 프로젝트 단위로 개발 내용들을 관리한다. 거기에는 응용이 규칙 엔진을 이용하여 규칙을

처리하는데 관련된 모든 정보들을 포함하고 있다. 규칙들을 정의한 파일의 명과 이를 자원화하는 자바 클래스 정보와 비즈니스 객체들을 규칙 엔진이 처리할 수 있는 객체 모델로 매핑하는데 필요한 클래스 정보들을 담고 있는 파일의 명과 이를 자원화하는 자바 클래스 정보들을 포함하고 있다.

2.2.1 규칙 표현 언어

프로그래밍 기술이 전혀 없는 비즈니스 업무 종사자들도 비즈니스 규칙을 표현할 수 있도록 하기 위해서는 자연어에 가까운 구문과 문법을 제공하는 규칙 표현 언어를 제공하여야 하며 아울러 객체 기반의 프로그래밍 언어와 유사한 표현 방법도 제공하여야 한다. 아래 <그림 2>는 비즈니스 규칙을 표현한 3 가지 형태의 표현 방법을 보여주고 있다.

```

if customer.debt > customer.assets
then set the status of customer application to Declined.

if the debt of the customer is greater than the assets of the customer
then the status of the application of the customer is Declined.

if customer.debt > customer.assets
    
```

<그림 2> 3 가지 유형의 규칙 표현

2.2.2 데이터 매핑 도구(객체 클래스 편집기능 포함)

규칙들은 응용의 비즈니스 객체들의 데이터 값들을 평가(evaluate)하고 수정 한다. 응용에 있는 객체들을 규칙 처리 시스템에서 처리하기 위해서는 규칙 처리 시스템의 class 로 정의를 하여야 한다. 즉 외부 응용 객체들을 규칙 시스템의 내부 객체 class 로 매핑하는데 이용한다. 이외에도 DB 테이블들을 내부 객체 클래스로 매핑하는 기능과 그 밖의 Java class 와 COM 객체들, CORBA interfaces 객체들을 도구로 임포트 한 후 내부 객체 클래스로 매핑한다. 이는 규칙 처리 대상 데이터를 규칙 시스템의 classes 로 별도로 정의하지 않아도 되므로 개발 속도를 높여준다. 이외 내부 객체 클래스를 편집하는 기능도 제공한다.

2.2.3. 규칙흐름(Ruleflow) 편집기

규칙엔진은 어떤 순서로 어떤 규칙을 구동(fire)할 것인지를 자동으로 결정하기 때문에 순서를 지정할 필요가 없으나 비즈니스 프로세스를 개발하는 과정에서 개발자가 규칙의 실행 순서를 제어 할 필요가 있는 비즈니스 프로세스를 구현하여야 할 경우가 많이 생긴다. 이때 규칙의 실행 순서를 제어하기 위하여 규칙 흐름을 작성하게 되는데 이때 그래픽 화면에서

용이하게 규칙의 흐름을 만들수 있도록 지원하는 도구를 말한다.

2.2.4. 규칙 세트(RuleSet) 편집기

어떤 기능을 수행하는데 관련된 규칙들을 하나의 규칙 셸로 그룹화해 놓음으로써 다른 비즈니스 프로세스를 구현할 때 유사 기능을 수행하는 규칙 셸을 신속하게 찾아 업무에 맞게 다시 수정하여 사용할 수 있어 유지관리 하거나 성능 측면에서 중요한 이점을 제공한다.

2.2.5 의사 결정 테이블

많은 기관에서는 업무 수행시 의사 결정 테이블을 사용하여 중요한 의사 결정을 하는 경우가 많다. 그 테이블에는 다수의 업무 규칙들을 정의하고 있어 규칙들을 한눈에 파악할 수 있는 이점을 제공한다. 이 테이블을 이용하여 업무의 규칙들을 정의하고 관리할 수 있다.

2.2.6 도입기(deployer)

이 도구는 개발 환경에서 만든 하나의 규칙 프로젝트를 개발 환경 밖에서 하나의 규칙 서비스로 동작하는데 필요한 형상(configuration) 파일들과 java 파일들을 만든다.

2.2.7 규칙 도입 관리기

시스템 구성요소들이 실행중에도 중지 시키지 않고 새로운 규칙들을 추가하거나 교체하는데 이용되는 도구를 말한다. 이 도구는 규칙 갱신 요청을 받았을 때 현재 규칙을 처리하고 있는 규칙 서버와 협력하여 규칙들을 갱신한다.

3. 표준화 동향

각기 다른 규칙 표현을 처리하는 규칙 엔진들을 이용하여 규칙 기반 응용들이 개발되어 왔다. 이로 인하여 각기 다른 규칙 표현을 사용하는 규칙 기반의 응용들이 상호 운영할 수 없는 문제점을 갖고 있다. 이런 문제 점을 해결하기 위하여 규칙 마크업 언어 이니시에이티브(initiative:http://www.dfki.uni-kl.de/ruleml)에서는 표준 규칙 언어로 RuleML 를 제안하고 있다. 이 언어가 시맨틱웹에서의 규칙 표현 언어로 독립적으로 나아갈지 아니면 시맨틱웹의 온톨로지 표현 언어인 OWL 에 통합될 것인지에 대한 연구가 진행되고 있다.

4. 향후 연구 방향

프로젝트 수행 초기 단계에서는 단일 규칙 언어 기반의 응용 시스템 개발 및 운영 시스템의 개발을 목표로 하고 있으나 향후에는 표준 규칙 마크업 언어 기반을 지원하는 시스템을 개발할 계획이다.

참고 문헌

[1] Ernest J. Friedman-Hill, "Jess, The Expert System Shell for the Java Platform", <http://herzberg.ca.sandia.gov/jess/docs/manual.pdf>, 6.1RC ver., march, 2003.