

Enterprise Architecture를 위한 시맨틱 웹 기반의 온톨로지 설계 및 구현*

김왕석** · 변영태***

Design and Implementation of Semantic Web Ontology for Enterprise Architecture*

Wang-Suck Kim** · Young-Tae Byun***

■ Abstract ■

Since EA includes huge information of a company, it takes long time and high cost for company's employees to search for what they need. We try to make the foundation to solve this problem by using ontology technology based on semantic web. In this paper, we try to verify efficiency of EA ontology by developing ontology for Business Enterprise Architecture(BEA). The purpose of this paper is to develop BEA ontology to provide new information by reasoner and to discover new relations between matadata by using extracted information and data. The EA ontology we developed will provide the new way of access and use for companies. The experience of ontology development will help EA ontology development in various domains. In the future, the development of other EAs which has more information resources will help to solve problems for interoperability between different EAs.

Keyword : EA, BEA, Semantic Web, Ontology, Matadata

논문투고일 : 2008년 04월 18일 논문수정완료일 : 2008년 09월 03일 논문게재확정일 : 2008년 09월 18일

* 이 논문은 2007학년도 홍익대학교 학술연구진흥비에 의하여 지원되었음.

** 국방연구원 IT건설팀부 EA팀

*** 홍익대학교 공과대학 컴퓨터공학과 교수

1. 서 론

기업과 조직에서는 정보 자원의 체계적인 관리와 표준화된 정보 기술의 도입 그리고, 조직의 목표인 비즈니스와 정보기술의 정렬 달성에 대한 노력을 기울이게 되었다. 이런 노력에 부응하여 비즈니스와 IT의 연계를 용이하게 하는 EA란 새로운 아키텍처가 나타나게 되었고, 여러 조직과 기업에서는 전사적 관점에서 비즈니스와 정보 기술 전략의 연결 고리 기능을 제공하는 EA의 도입이 활발해졌다.

EA는 기업이나 조직의 광대한 정보 자원을 잘 정리한 일종의 프레임 워크, 형식화된 문서의 한 형태라 생각할 수 있다. 즉, EA에서 사용자가 원하는 정보에 대한 검색을 지원하는데 있어 단순 검색 뿐 만 아니라 사용자의 높은 참여를 요구하고 있다. 기존의 검색은 “누가 무엇을 하기위해 어떤 비즈니스 시스템을 사용하고, 어떤 기술과 생 산품을 사용하고 있는가?”와 같은 복합적인 검색을 지원하지 않는 한계를 가지고 있었다. 이를 개선하기 위해서 현재 활발히 연구가 진행되고 있는 시맨틱 웹 기반의 온톨로지를 적용함으로써 기업과 조직에서 요구하는 정보를 쉽게 제공할 수 있는 기반을 제공할 수 있을 것이다.

EA 온톨로지의 구현은 실제 EA에 제공되는 메타데이터들 간의 관계를 이용한 정보 뿐 만 아니라 추론을 통하여 감추어진 관계를 이용한 새로운 정보까지 제공할 수 있는 장점이 있다. 본 논문에서는 대표적인 EA 중에서 미국방부의 비즈니스 영역에 대한 EA인 BEA에 대한 온톨로지를 구현하여 EA에 대한 시맨틱 웹을 이용한 접근 방법을 제시한다.

2. 관련 연구

2.1 Enterprise Architecture

EA의 개념은 “조직 및 업무 활동과 정보 기술

간의 관계에 대해 현재의 모습과 향후 추구해나갈 모습을 별도로 정의해둔 청사진”이다[2]. 여기서 ‘Enterprise’의 의미는 적용 조직의 범위를 넘어서 정보화에 담겨질 영역의 범위라고 말할 수 있다[1]. 즉, 정보화의 청사진에 담겨질 영역이 조직이 보유한 정보 기술 인프라, 애플리케이션, 업무를 수행하는데 필요한 데이터와 정보뿐만 아니라 조직 목표를 달성을 위해 수행해야 하는 업무(조직 목표 및 전략을 포함) 까지도 포함하는 확장된 개념임을 강조한 것이다[1].

EA의 장점은 EA가 기존의 정보화의 문제점으로 지적되던 자원의 중복이나 낭비를 방지할 수 있고, 비즈니스와 정보 기술의 연계가 가능하게 한다는데 가장 큰 이점을 가지고 있다. 유사한 업무를 수행하는 기업이나 부서가 존재할 경우에 한쪽에서 EA가 존재할 때, 다른 쪽에서 기존의 EA를 활용하여 새로운 EA 구축을 하거나 그대로 사용가능하기 때문에 중복 투자나 낭비를 방지할 수 있다.

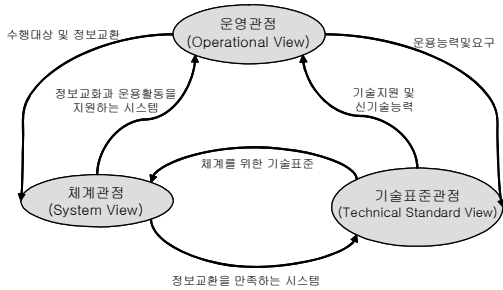
EA의 구성 요소는 대상에 따라서 다양한 형태를 갖는다. EA는 일반적으로 몇 개의 하부 구조가 존재하며 그 하부 구조간의 관계성이 존재한다. 예를 들어 미국방부의 경우에는 3개의 하부 구조인 운영, 시스템, 기술 아키텍처로 구성되고[6], 미 연방정부의 경우에는 5개의 하부 구조인 업무, 응용, 데이터, 기술, 정보 아키텍처로 구성된다[7].

EA를 구축하는 과정은 먼저 조직이 보유한 모든 정보자원을 조직의 전략적 목표와 목표 달성을 위한 관련 업무, 업무 수행에 필요한 정보의 흐름, 정보 획득을 위한 지원도구로써 응용프로그램, 응용프로그램 구현에 필요한 IT 기술로 구분한다. 그 후, 각 요소별 엘리먼트 간 연관관계를 식별하는 전사적 정렬(Enterprise Alignment)단계로 진행된다. 아키텍처 관점에서 정렬이란, ‘어떤 연관된 것들 상호간의 배열(arrangement)’이라고 정의한다[5].

2.2 DoDAF

미국방부는 미래정보전과 체계통합전 수행을 목

적으로 국방 업무에 대한 EA인 DoDAF를 개발하였다[6]. 즉 국방업무에 미래 합동 전장 운용 개념에 부합할 수 있는 아키텍처를 적용함으로써 미래 전의 효과적인 임무 수행을 보장하려는 노력을 수행하고 있다. DoDAF는 아키텍처를 모델링하고 설계하는데 대한 지침 원리를 제공하며, 이러한 아키텍처를 기술하기 위하여 운용(Operations), 체계(Systems), 기술 표준(Technical Standards)의 3가지 표준 관점을 정의한다.



[그림 2-1] DoDAF 관점간의 관계도

운용 관점(Operational View)은 임무와 비즈니스 프로세스를 포함한 중요한 운용 개념과 관련된 능력들이 내부적으로 어떻게 밀접한 관련이 있는지 설명한다. 체계 관점(Systems View)은 기능을 제공하거나 지원하기 위한 체계와 상호 연결을 그래픽이나 서술식으로 설명한다. 기술 표준 관점(Technical Standards View)은 특정한 기술 표준 요구에 부합되는 체계 개발을 보장하기 위해서 체계 구성요소의 배열, 상호 작용 및 상호 의존성을 설명한다.

2.3 BEA 3.1

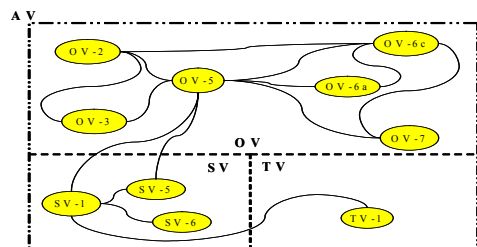
DoD는 DoD의 업무를 4가지 영역으로 분류하고 각각의 EA를 개발하였다. 이중에 BEA는 DoD의 비즈니스 정보 조직을 위해서 개발된 EA이다[5]. BEA의 산출물들은 DoDAF의 산출물 기반으로 만들어졌지만 그보다 작은 수의 산출물로 구성되어 있다. BEA 산출물은 문서와 그래프의 형식으로

프로세스, 데이터, 기술 표준, 비즈니스 규칙, 운용 활동과 정보 교환 등의 DoDAF의 비즈니스 업무와 관련된 모든 정보 및 자료를 포함한다. BEA의 산출물은 다음 [그림 2-2]와 같다.

• All views	
- AV-01	Overview and Summary Information
- AV-02	Integrated Dictionary
• Operational views	
- OV-02	Operational Node Connectivity Description
- OV-03	Operational Information Exchange Matrix
- OV-05	Activity Model
- OV-06a	Operational Rules Model (Business Rules)
- OV-06c	Business Process Model
- OV-07	Logical Data Model System Views
• System views	
- SV-01	Systems Interface Description
- SV-05	Operational Activity to Systems Function Traceability Matrix
- SV-06	Systems Data Exchange Matrix Technical Standards Views
• Technical Standards Views	
- TV-01	Technical Standards Profile

[그림 2-2] BEA 산출물(AV, OV, SV, TV)

AV(All View)는 BEA에 대한 소개(AV-1)와 용어 사전(AV-2)의 정보를 가지고 있다. BEA의 OV, SV, TV는 DoDAF의 OV, SV, TV와 기본적으로 유사한 의미를 갖는다. OV는 운영노드와 노드간의 연결성(OV-2)과 운영활동에 필요한 정보 교환(OV-3)과 운영 활동 모델(OV-5)과 운영에 필요한 규칙과 프로세스(OV-6)와 논리적인 데이터 모델시스템(OV-7)의 정보를 가지고 있다. SV는 시스템 구조(SV-1)와 운영활동과 시스템간의 관계(SV-5)와 시스템간의 데이터 교환(SV-6)에 대한 정보를 가지고 있다. TV는 현재 시스템에서 요구되는 표준 기술에 대한 정보를 가지고 있다. 이러한 BEA의 산출물들은 서로간의 관계성이 존재한다. 이들 간의 관계는 다음의 [그림 2-3]과 같다.



[그림 2-3] BEA 산출물간의 관계성

BEA는 현재 공개된 EA 중에 가장 작은 양의 정

보량을 가지고 있고 산출물간의 관계가 명료하기 때문에 본 논문에서 EA에 대한 온톨로지 구현에 적합하다. BEA의 형태를 보면 일반적인 웹과 문서의 형태로 되어 있다. 즉, 각각의 산출물은 PDF 또는 엑셀 문서형식과 일반 웹페이지의 형태로 제공된다. BEA로부터 정보를 검색하고자 한다면 웹 페이지간의 링크나 문서로 부터 원하는 정보를 검색 가능하다. 이러한 단순 정보 검색을 발전시키기 위해서 BEA에 온톨로지를 적용함으로써 단순 검색이 아닌 복합적 정보에 대한 검색이 가능하게 된다.

2.4 시맨틱 웹과 온톨로지

시맨틱 웹은 기존의 웹과 다른 웹이 아니라 현재의 웹을 확장하여 웹상의 정보들에 잘 정의된 의미를 부여함으로써 사람과 컴퓨터가 함께 웹의 정보와 지식을 처리할 수 있는 웹의 환경을 제공한다[4]. 즉 문서의 의미를 명확하고 컴퓨터가 이해할 수 있는 형태로 표현하여 즉 문서에 의미정보를 부여하여 컴퓨터가 웹 자원들을 효율적으로 관리함으로써 인간의 손을 거치지 않고 컴퓨터가 사용자가 원하는 정보를 찾을 수 있는 기능을 제공하게 하려는 것이다. 이러한 시맨틱 웹은 현재 연구 분야로 큰 이슈가 되고 있으며 현재 발전중인 개념이다. 시맨틱 웹에서 온톨로지는 특정 도메인에 맞게 지식을 개념화하고 이를 명세화한다.

온톨로지란 “해당영역의 공유된 개념화에 대한 형식적이고 명시적 기술이다”라고 설명할 수 있다 [12]. 형식적이라는 것은 기계가 읽고 처리할 수 있는 술어 논리와 같은 인공지능의 지식표현방법을 말하고, 명시적이라는 것은 개념들의 유형과 개념 사용에 대한 규칙을 명시적으로 정의 내린다는 것을 의미한다. 그리고 공유된다는 말은 용어의 사용이 관련 사람들의 합의에 의한 것이라는 것이고 개념화라는 말은 실제 세계에 대한 모형을 의미하며, 해당영역이라는 것은 특정 영역 중심이라는 뜻이다. 또한 온톨로지는 용어 사이의 관계를 정

의하고 있는 일종의 사전과 같은 것이라 할 수 있다. 온톨로지의 표현기술과 논리적 추론 기술을 통해 복잡한 검색이 가능하기 때문에 인공지능 분야와 웹과 관련된 많은 분야에서 연구되고 있다.

온톨로지는 개념들 사이의 상속적인 계층구조와 각 개념의 속성 값의 관계로 표현된다. 온톨로지는 사물이나 행위의 상속, 계층 구조로 개념을 분류하며 개념들 사이를 is-a, part_of, have_a 등 다양한 관계들로 연결해준다. 즉 개념들 사이의 상속적인 계층구조와 각 개념의 속성 값의 관계로 표현된다. 효율적인 온톨로지 언어는 이용자에게 높은 직관성을 제공해야 한다. 또한 추론 특성으로 설계된 형식적 의미를 갖추어 완결성, 정확성, 효율성을 보증하여야 한다. 웹 표준 언어와의 적절한 연결을 보유하여 상호운용성을 보증해야 한다.

온톨로지는 다음의 절차에 따라서 온톨로지 설계가 이루어지게 된다. 먼저 영역과 온톨로지 범위를 설정한다. 어떤 온톨로지를 만들 것인지를 결정하는 것이다. 본 논문에서는 EA라는 영역에 대해 BEA 범위의 온톨로지를 설계하고자 한다. 둘째, 기존의 이미 구축된 온톨로지들 중에 재이용할 수 있는 것을 고려한다. BEA에 대한 온톨로지가 없기 때문에 본 논문에서는 다른 온톨로지의 재사용성은 제외한다. 셋째, 해당 온톨로지 내 중요한 용어들을 열거한다. BEA의 중요 용어들은 AV-2로부터 정보를 추출할 수 있다. 넷째, 클래스와 클래스 계층관계를 정의한다. BEA 온톨로지의 클래스는 필수 메타 데이터들이다. 다섯째, 클래스의 특성을 정의한다. BEA의 필수 메타 데이터들의 특성에 따라 정의한다. 마지막으로, 인스턴스를 생성한다. BEA의 실데이터를 입력한다. 이러한 과정을 통해 온톨로지를 구현할 수 있다.

2.5 OWL(Web Ontology Language)

OWL은 XML과 RDF, 그리고 RDFS 뒤를 이어 차세대 웹 언어로 W3C에 의해 표준화된 언어이다. OWL은 표준 온톨로지 언어로서 의미 있는 검

색을 지원하기 위해 온톨로지에 근거하여 웹 자원의 의미를 기술하는 온톨로지 언어이다[15]. OWL은 클래스의 구성원들에 관한 사실과 구성원들 간의 관계를 기술하며, 이러한 클래스 정보를 모아 만들어진 온톨로지는 구문적으로 정의되지 않은 사실의 논리적 유추를 가능하게 한다.

2.6 PROTÉGÉ

Protégé은 지식 기반의 구조를 작성하기 위한 시스템으로 개발되었다. Protégé는 다른 지식 표현 언어와 호환이 가능하며, 온톨로지의 생성 및 수정을 위한 확장이 용이하여 현재 가장 광범위하게 활용되고 있는 온톨로지 개발 및 관리도구이다. 그리고 온톨로지 재사용 기능, 지식기반 데이터베이스 구축을 통한 질의기능, 추론 기능 등의 풍부한 플러그인을 통한 확장성을 제공하고 있다[13]. 이 연구에서는 Protégé를 EA에 대한 온톨로지를 구현하는데 사용한다.

2.7 DL(Description Logic)

DL은 객체의 정의에 필요한 정의 개념들로 객체 사이의 관계를 표현할 수 있는 정의언어를 지원하며, 기본적인 개념들과 관계들로부터 복잡한 형태의 개념들과 관계들을 생성한다. DL은 개별적인 것들에 대한 정보를 표현하기 위한 논리적 수식으로, 데이터와 스키마의 서술표현에 적합하다. 본 논문에서는 온톨로지의 일관성을 확인하기 위해 DL기반의 Racer를 사용한다.

2.8 Semantic Web Rule Language

SWRL은 시맨틱 웹의 규칙 언어로 만들어졌다. SWRL은 사용자에게 OWL의 용어로 표현될 수 있고 OWL의 individual을 추론할 수 있는 규칙을 작성할 수 있다. 존재하는 OWL 지식 기반으로부터 새로운 지식을 추론할 수 있게 하는 연역적 추론 명세를 제공한다[16]. SWRL은 가정 부분을 포

함하고, 그것은 몸통에 적용되고, 결과 부분은 머리에 포함된다. 몸통과 머리 모두 원자들의 결합으로 이루어진다. SWRL 규칙은 OWL의 individual에 대해 추론한다.

다음은 “사람이 남자 형제가 있다면 그 사람은 남동생을 갖는다”라는 것을 SWRL 규칙으로 표현한 것이다. 여기서 Person과 Man은 class이고 Man은 Person의 Subclass이며, hasSibling과 hasBrother는 Person의 Property이다.

```
Person(?p) ^ hasSibling(?p, ?s) ^ Man(?s) ->
    hasBrother(?p, ?s)
```

[예제 1] SWRL 규칙 1

이 규칙을 수행하면 hasBrother Property의 모든 OWL individual들에게 하나 이상의 남자 형제가 배치되는 결과를 가져올 것이다.

2.9 EA와 온톨로지

미연방정부는 연방정부 업무에 대한 EA인 FEA-RM을 개발하였다[7]. FEA-RM의 목적은 연방정부의 사업과 부서들 간의 연합된 업무와 업무과정에 대해 단순하게 식별하는 것이었다. 그 결과로 더 시민 중심적이고, 고객에 초점을 둔 더 나은 업무 결과 달성을 위해 기술 투자를 극대화하는 정부가 가능하게 하려는 것이다. FEA-RM은 미연방정부와 관련한 업무 성과, 사업, 서비스 요소, 기술적, 데이터, 5개의 참조모델(Reference Model)로 표현하였다. FEA-RM은 연방부서 간의 협력을 위해 연관된 부서간의 차이를 분석하고, 중복된 투자 식별하여, 연관 부서 분석을 촉진하기 위해 개발되었다. 참조모델은 EA의 수단으로 제공된다. FEA-RMO는 OWL을 사용하여 FEA-RM의 5개의 모델(PRM, BRM, SRM, TRM, DRM)을 온톨로지로서 표현하였다. 뿐만 아니라 5개의 모델의 개념(Concept)과 관계성(Property)을 가진 핵심 온톨로지(Core Ontology)와 모델간의 연결 관계를 나타내는 BRM2PRM과 같은 중개 모델(Bridge

Model)을 포함한다[8].

3. BEA 온톨로지 구현

3.1 DoDAF/BEA 분석

이 논문에서는 EA에 대한 온톨로지를 구현하기 위해 대표적인 EA 중 하나인 미국방부의 비즈니스 영역만을 대상으로 개발된 BEA 3.1을 도메인으로 선정하여 이에 대한 온톨로지 구현을 실시하였다.

EA를 온톨로지로서 구현하는데 있어서 먼저 고려되어야 할 사항이 'EA를 온톨로지로서 구현하기 위해 어떻게 EA로부터 온톨로지를 구성하는데 필요한 정보 즉 메타 데이터와 실데이터를 구분하고 추출할 것인가?'라는 문제이다. 메타데이터는 일반적으로 데이터를 위한 데이터로 정의한다. EA는 구조화되고 계층화되어 있으며, EA의 산출물들을 살펴보면 각각의 객체 또는 엔티티 간의 링크 또는 관계가 잘 표현되어 있기 때문에, 이로부터 온톨로지를 구현하기 위한 정보를 쉽게 추출해 낼 수 있는 이점이 있다.

대표적 EA인 DoDAF의 산출물은 CADM(Core Architecture Data Model)으로 표현 되는데, CADM은 필수 메타 데이터와 보조 메타 데이터를 갖는다. 메타 데이터는 데이터의 속성과 데이터간의 관계로 표현된다. 예를 들어 DoDAF의 산출물인 OV-2는 필수 메타 데이터로 Operational Node가 존재하고 보조 메타 데이터로 External Operational Node가 있다. Operational Node의 속성은 이름과 설명 등이 있다. 이러한 CADM으로부터 메타 정보를 추출하면 용이하게 온톨로지를 구현할 수 있다.

EA에 대한 온톨로지를 구현하는데 있어서 고려해야 할 또 다른 사항은 EA와 온톨로지의 공통적인 특성이라고 할 수 있는 재사용성이다. 다른 EA에 대한 온톨로지가 만들어졌을 때의 기존의 EA 온톨로지를 재사용하여 새로운 정보를 추가 또는 삭제함으로써 EA 온톨로지의 정보 공유와 재사용

성을 높일 수 있다.

따라서 본 논문에서 BEA가 DoDAF의 일부분이기 때문에 BEA에 대한 온톨로지를 구현하는데 있어서 BEA를 기준으로 DoDAF의 산출물 중에서 BEA의 산출물과 연관이 있는 산출물들 비교 분석하였다. 먼저 DoDAF의 산출물 중에 BEA와 동일한 산출물을 분류하고 분류된 DoDAF 산출물의 CADM에 있는 데이터 중에 필수 메타 데이터와 보조 메타 데이터를 구분하였다. 이중에 BEA를 위한 온톨로지를 구현하는데 필요한 필수 메타 데이터 정보와 실데이터 정보를 선정하고 이를 기반으로 BEA에 있는 필수 산출물에 적용시켜 BEA에 적합한 온톨로지 정보를 추출함과 동시에 DoDAF의 온톨로지에도 사용될 수 있는 정보를 추출하였다.

3.2 BEA 온톨로지 구현 및 추론 사례

BEA 온톨로지 구현 및 추론의 기본 구조는 PROTEGE에 BEA 정보와 SWRL 규칙을 입력 및 결합하여 JESS를 통해 정보를 추론하는 것이다. 이 실험의 목적은 주어진 질의를 SWRL형태로 작성하여 질의에 대한 정보 검색을 JESS를 이용한 추론을 통해 새로운 PROPERTY가 생성됨으로써 보다 다양한 정보 검색이 가능하게 하는 것이다. BEA 온톨로지 개발에 사용한 언어는 OWL DL이다[15]. 이 온톨로지를 개발하는데 사용한 툴은 PROTEGE 3.2이다[13]. 온톨로지의 일관성(consistency)을 확인하기 위해 Racer Pro를 사용하고 [14], 온톨로지의 규칙 엔진은 DL 기반의 Jess를 사용한다[16]. Jess는 룰 기반의 추론 시스템으로서 프로그램 자체의 크기가 작고 가벼우며, 규칙(rule)과 사실(fact)의 매칭에는 RETE 알고리즘을 이용하여 빠른 추론 기능을 제공하기 때문에 이 실험에 적합한 엔진이다. BEA 온톨로지는 Class와 Property, 그리고 Instance로 구성된다. BEA 온톨로지의 Class는 다음의 표에 나와 있는것 처럼 총 21개의 Class와 System Node와 Operational Node

의 상위 개념인 Node Class까지 22개가 존재한다. 이들 Class들은 서로 관계를 가지고 있다.

〈표 3-1〉 BEA class

	Class		
OV-2	Operational Node	Need Line	
OV-3	Information Exchange	Information Element	
OV-5	Operational activity	Flow Connector	InputAnd Output
SV-1	System Node	System Fuction	Interface
SV-6	SystemData Exchange	SystemData Element	
TV-1	Service	Standard	

Property는 다음의 Object Property로 총 37개와 이들의 Inverse-Property 36개가 존재하며, 각각의 Domain과 Range로서 Class를 갖는다. 이중에 hasParent는 Functional한, hasSibling은 Symmetric한 관계성을 갖는다.

Instance는 각각의 Class에 따라서 다르게 존재하며, 다음의 <표 3-2>와 같다. SystemData의 경

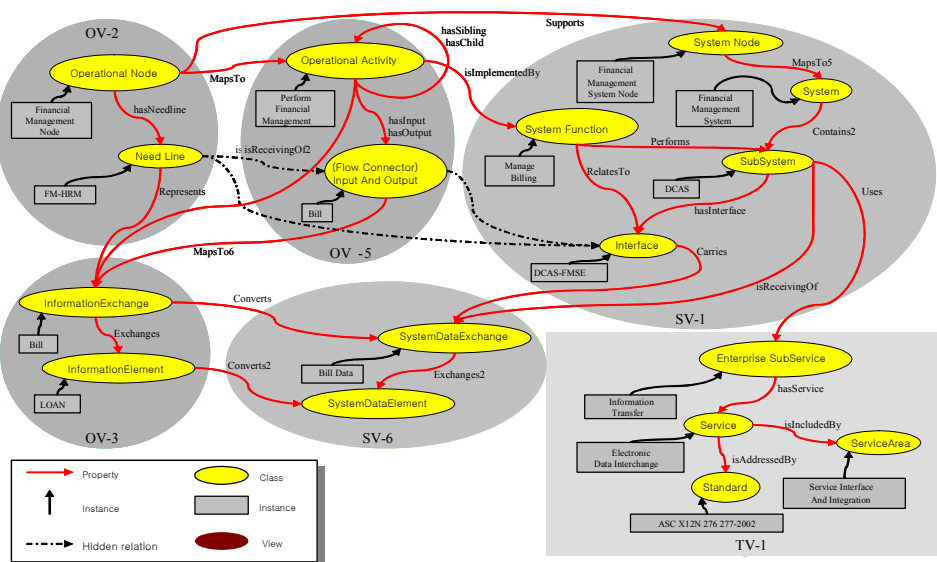
우 실제 System을 구현할 때 발생하기 때문에 BEA의 경우 instance가 존재하지 않는다.

〈표 3-2〉 BEA Instance

	Class(Instance Number)		
OV-2	Operational Node(8)	Need Line(45)	
OV-3	Information Exchange(206)	Information Element(744)	
OV-5	Operational activity(160)	Flow Connector(206)	InputAnd Output(206)
SV-1	System Node(8)	System Fuction(67)	Interface(135)
SV-6	SystemData Exchange(215)	SystemData Element(0)	
TV-1	Service(23)	Standard(334)	

BEA에 Class와 Property간의 관계는 다음의 [그림 3-1]과 같다.

아래의 [그림 3-1]에서 보이는 Hidden Relation은 BEA 온톨로지를 만드는 과정에서 발견된 BEA에 있는 메타 데이터간의 정보에서 유출된 관계를 의미한다.



[그림 3-1] BEA 온톨로지

<표 3-3> BEA Hidden Property

Hidden Property			
Name	Domain	Range	특성
Transfer1	NeedLine	Flowconnector	
Transfer2	NeedLine	Interface	
Transfer3	Interface	Flowconnector	

이제 BEA에 SWRL을 적용하여 Rule을 만들어 추론 사례를 적용시켜보도록 하겠다. 먼저 각각의 관점간의 연계성을 통한 검색을 가정하였을 경우에 다음과 같은 총 6개의 경우가 발생하게 된다.

<표 3-4> 산출물 간 가능한 연계성

	A	B	C	D	E	F
OV	OV-2	OV-2	OV-3	OV-3	OV-5	OV-5
SV	SV-1	SV-6	SV-1	SV-6	SV-1	SV-6
TV	TV-1	TV-1	TV-1	TV-1	TV-1	TV-1

BEA 온톨로지는 위의 <표 3-4>의 6개의 경우에 대한 검색이 가능해야 한다.

이를 바탕으로 추론 사례를 만들어보면 다음과 같다.

- Case A : (OV-2) → (SV-1) → (TV-1)
 Ex) 어떤 Operational Node와 관련된 SubSystem과 그 SubSystem에 연관된 Standard의 관계
 추론 방향 : (OV-2) → (SV-1) → (TV-1)
 Operational Node → System Node → System → SubSystem → Enterprise Service → Service → Standard
- Case B : (OV-2) → (SV-6) → (TV-1)
 Ex) 어떤 Need Line과 관련된 System data Exchange와 그 System data Exchange에 연관된 ServiceArea 의 관계
 추론 방향 : (OV-2) → (OV-3) → (SV-6) → (SV-1) → (TV-1)
 Need Line → Information Exchange → System

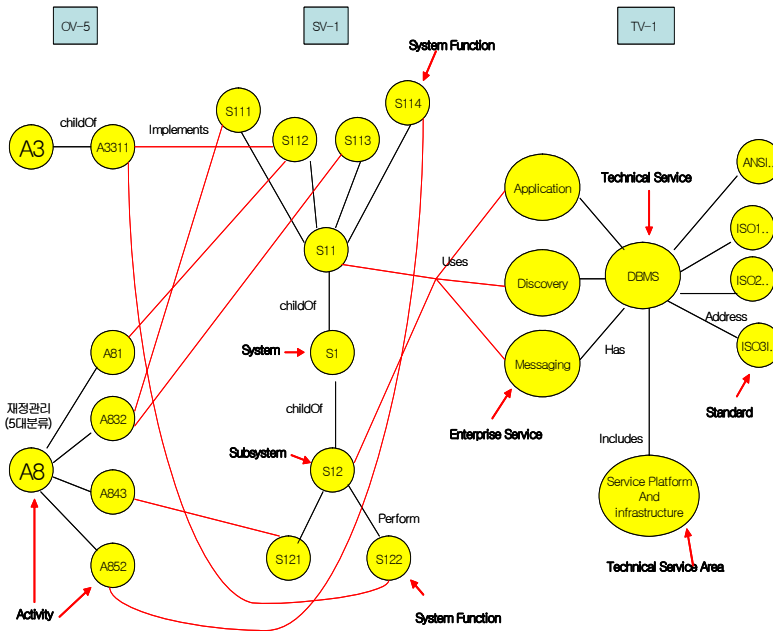
data Exchange → Interface → SubSystem → Enterprise Service → Service → ServiceArea

- Case C : (OV-3) → (SV-1) → (TV-1)
 Ex) 어떤 Information Exchange와 관련된 SubSystem과 그 SubSystem와 연관 있는 Standard의 관계
 추론 방향 : (OV-3) → (OV-5) → (SV-1) → (TV-1)
 Information Exchange → Operational Activity → SystemFunction → SubSystem → Enterprise Service → Service → Standard
- Case D : (OV-3) → (SV-6) → (TV-1)
 Ex) 어떤 Information Exchange와 관련된 System Exchange와 그 System Exchange와 연관 있는 Standard의 관계
 추론 방향 : (OV-3) → (SV-6) → (SV-1) → (TV-1)
 Information Exchange → System data Exchange → Interface → SubSystem → Enterprise Service → Service → Standard
- Case E : (OV-5) → (SV-1) → (TV-1)
 Ex) 어떤 Operational Activity와 관련된 SubSystem과 그 SubSystem에 연관된 Standard의 관계
 추론 방향 : (OV-5) → (SV-1) → (TV-1)
 Operational Activity → System Function → SubSystem → Enterprise Service → Service → Standard
- Case F : (OV-5) → (SV-6) → (TV-1)
 Ex) 어떤 Operational Activity와 관련된 System Exchange와 그 System Exchange와 연관 있는 Standard의 관계
 추론 방향 : (OV-5) → (SV-1) → (SV-6) → (SV-1) → (TV-1)
 Operational Activity → System Function → System data Exchange → Interface → SubSystem → Enterprise Service → Service → Standard

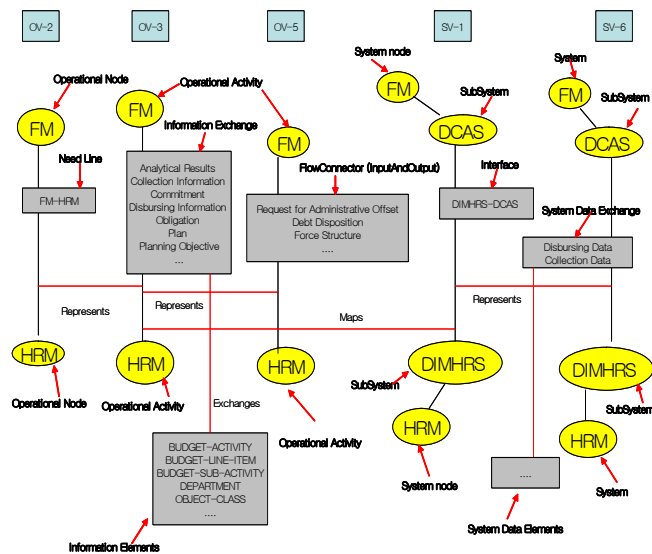
[그림 3-2]는 6개 경우 중에 Case E의 검색을 가능하게 하는 예제이다. 즉 재정 관리라는 운용 활동과 관련된 시스템이나 서비스 또는 기술에 대한 정보를 알고 싶을 때, [그림 3-2]의 재정 관리라

는 운용 활동에서 시작하여 관계성을 나타내는 선 (Property)을 따라 원하는 정보에 도달할 수 있다.

또한 Hidden Property를 통해서 추론 가능한 사례는 다음 [그림 3-3]과 같다.



[그림 3-2] Case E

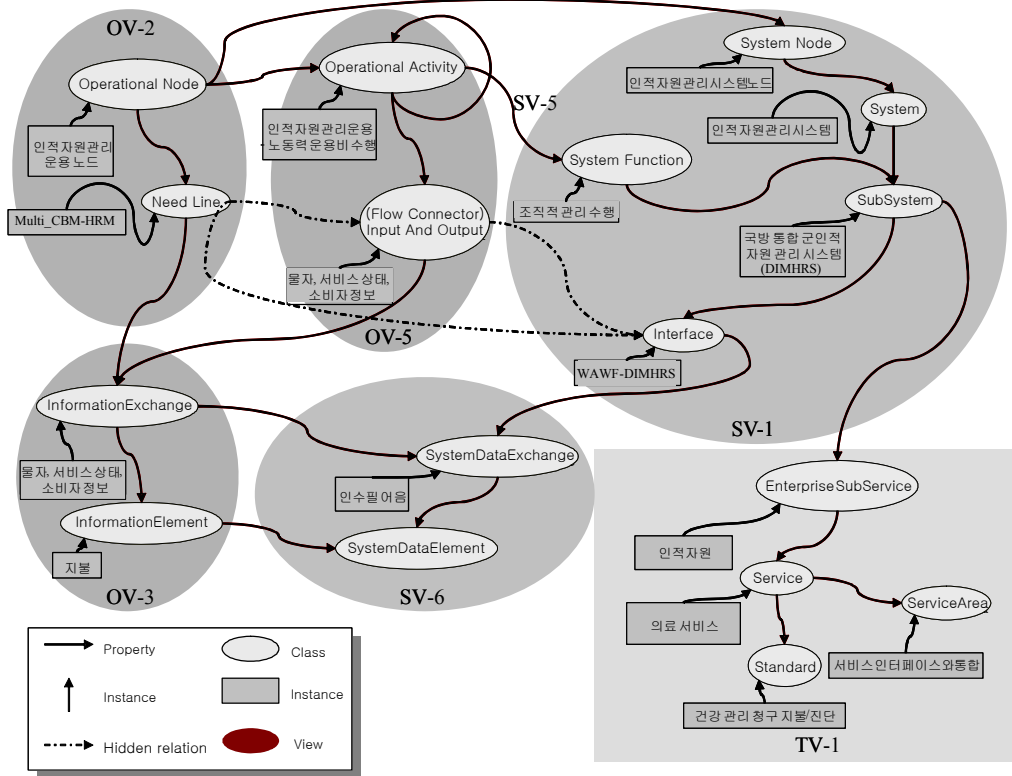


[그림 3-3] Using Hidden Property

[그림 3-3]은 다음과 같은 내용을 담고 있다. 만약 사용자가 재정 관리와 인적 자원 관리 간 주고 받는 정보나 데이터와 그리고 이 데이터를 이용하는 시스템에 관한 검색을 원한다면, 그림의 재정 관리 노드와 인적 자원 관리 노드의 관계를 나타내는 Need Line에서 시작하여 관계성을 나타내는

라인을 따라 원하는 정보에 도달할 수 있다. 이제 실제 예를 만들어 SWRL을 이용한 추론 실험을 실시할 것이다. 다음의 [그림 3-4]를 참조하여 질의 예문을 구성한다.

SWRL에서 사용된 축약어들은 <표 3-5>와 같다. 질의 예문은 다음과 같다.



[그림 3-4] 인적 자원과 관련된 정보 검색

<표 3-5> 축약어

약 어	Class	약 어	Class	약 어	Class
ON	Operational Node	SN	System Node	FC	Flow Connector
NL	Need Line	SF	System Function	SV	Service
IEX	Information Exchange	S	System	ST	Technical Standard
IE	Information Element	SS	SubSystem	ES	Enterprise SubService
OA	Operational Activity	SDE	System Data Exchange	SA	Service Area
I	Interface	M	Mechanism	C	Control
IAO	InputAndOutput	BEP	BusinessEnterprisePriorities		

- Q1 : 미국방부는 노동력 운용비 수행(OA)을 하기 위해 어떤 시스템(SS)을 사용하는가?
 Q2 : 미국방부의 인적자원 관리(ON)와 관련된 시스템(SS)과 서비스(SV)는 어떤 것이 있는가?
 Q3 : 만약 미국방부가 건강관리 청구 지불 방법의 기준(ST)을 변환한다면 인적자원 관리(ON) 외에 어떤 운용 노드(ON)가 영향을 받는가?
 Q4 : 미국방부의 노동력 운영비 수행(OA)과 관련하여 필요한 정보(IEX)는 무엇이며 그 정보가 시스템(SS)에서 어떤 데이터(SDE)로 변형되는가?
 Q5 : 미국방부의 의료서비스(SV)를 향상시키기 위해 어떤 시스템(SS)과 그 기능들(SF)에 대해 살펴볼 필요가 있는가?
 Q6 : 미국방부의 인적 가시성(BEP)은 어떤 서비스 영역(SA)와 관련이 있는가?
- 질의를문(Q1~Q6)들에 대한 응답을 얻기 위해서 다음의 Property(P1~P7)들이 필요하다.

〈표 3-6〉 질의를문을 위한 Property

Question	Property	Name	Domain	Range
Q1	P1	hasInference1	OA	SS
Q2	P2	hasInference2	ON	SS
	P3	hasInference3	SS	SV
Q3	P4	hasInference4	ST	ON
Q4	P5	hasInference5	OA	SDE
Q5	P6	hasInference6	SV	SF
Q6	P7	hasInference7	BEP	SA

추론을 위한 Property(P1~P7)의 연계성은 다음과 같다.

- P1 : hasInference1(?OA, ?SS) R1 : (OA → SF → SS)
 P2 : hasInference2(?ON, ?SS) R2 : (ON → SN → S → SS)
 P3 : hasInference3(?SS, ?SV) R3 : (SS → ES → SV)

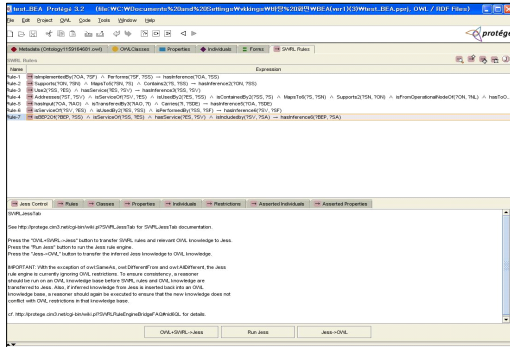
- P4 : hasInference4(?ST, ?ON) R4 : (ST → SV → ES → SS → S → SN → ON → NL → ON)
 P5 : hasInference5(?OA, ?SDE) R5 : (OA → I → AO → I → SDE)
 P6 : hasInference6(?SV, ?SF) R6 : (SV → ES → SS → SF)
 P7 : hasInference7(?BEP, ?SA) R7 : (BEP → SS → ES → SV → SA)

이를 기반으로 Rule을 생성하면 다음과 같다.

- R1 : isImplementedBy(?OA, ?SF) ^ Performs(?SF, ?SS) → **hasInference1(?OA, ?SS)**
- R2 : Supports(?ON, ?SN) ^ MapsTo5(?SN, ?S) ^ Contains2(?S, ?SS) → **hasInference2(?ON, ?SS)**
- R3 : Use2(?SS, ?ES) ^ hasService(?ES, ?SV) → **hasInference3(?SS, ?SV)**
- R4 : Addresses(?ST, ?SV) ^ isServiceOf(?SV, ?ES) ^ isUsedBy2(?ES, ?SS) ^ isContainedBy2(?SS, ?S) ^ MapsTo6(?S, ?SN) ^ Supports2(?SN, ?ON) ^ isFromOperationalNodeOf(?ON, ?NL) ^ hasToOperationalNode(?NL, ?ON) → **hasInference4(?ST, ?ON)**
- R5 : hasInput(?OA, ?IAO) ^ isTransferredBy3(?IAO, ?I) ^ Carries(?I, ?SDE) → **hasInference5(?OA, ?SDE)**
- R6 : isServiceOf(?SV, ?ES) ^ isUsedBy2(?ES, ?SS) ^ isPerformedBy(?SS, ?SF) → **hasInference6(?SV, ?SF)**
- R7 : isBEP2Of(?BEP, ?SS) ^ isServiceOf(?SS, ?ES) ^ hasService(?ES, ?SV) ^ isIncludedBy(?SV, ?SA) → **hasInference7(?BEP, ?SA)**

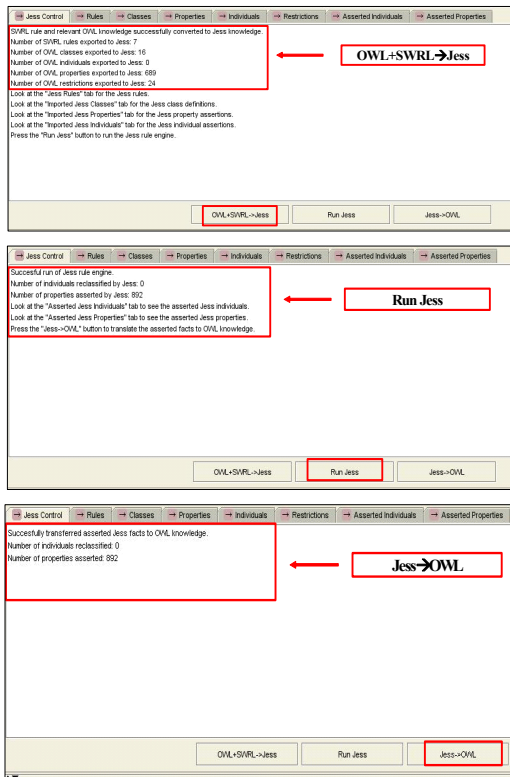
4. 실험 결과

구현한 BEA 온톨로지로부터 질의를문(Q1~Q6)에 실험 결과는 다음과 같다.



[그림 4-1] 규칙 생성

[그림 4-1]에서 먼저 생성한 규칙을 입력한 후, Jess 엔진을 이용한 추론을 실시한다.



[그림 4-2] 추론 과정

[그림 4-2]의 과정을 살펴보면, 먼저 생성한 7개의 Rule과 이와 연관된 16개의 Class와 689개의 Property와 24개의 Restriction이 Jess엔진으로 입

력이 되었다. 그 후 다음 과정에서는 Jess 엔진이 입력된 Rule를 통하여 892개의 새로운 Property를 생성하였다. 마지막으로 새롭게 생성된 892개의 Property가 OWL의 형태로 변환되어 BEA 온톨로지에 입력되었다.

<표 4-1> 시험 결과

	입력	생성	출력
Rule	7	-	7
Class	16	-	16
Property	689	892	1581
Restriction	24	-	24

<표 4-1>의 실험결과와 같이 이 실험을 통해서 새로운 Property가 892개가 생성되었고 이는 목표했던 수준(최초 입력된 Property만큼의 새로운 값)을 만족시키는 수치이다.

5. 결 론

본 논문은 현재 발전 중인 EA와 시맨틱 웹의 만남을 통해서 추론을 이용한 복합적인 정보검색 지원할 수 있는 EA 온톨로지를 구현을 통해 EA로부터 연관성 있는 정보에 대한 다양한 검색이 가능하게 하는 EA를 위한 시맨틱 웹 기반의 온톨로지 구현에 대하여 기술하였다. 논문의 실험을 통해서 약 900개의 새로운 Property를 생성함으로써 복합적인 정보검색의 지원을 검증하였다. 이렇게 개발된 EA 온톨로지는 기존의 EA에 대한 단순한 검색과 달리 추론을 통한 관계 생성을 이용한 EA 활용의 필수적 요소인 쉽고 다양한 정보 제공을 가능하게 함으로써 기업과 조직의 EA 활용과 접근에 새로운 방향을 제시한다고 판단된다. 또한 본 논문에서 개발된 EA 온톨로지는 현재 개발이 완료되었거나 개발 중인 EA에 대한 온톨로지를 이용한 접근에 많은 이점을 제공할 수 있을 것이다. 앞으로는 다른 EA를 위한 온톨로지 구현을 위해 기존의 EA 온톨로지의 재사용과 서로 다른 EA

의 온톨로지에 대한 온톨로지 통합에 관한 연구가 가능할 것이다.

참 고 문 헌

- [1] 행정자치부, 「전자정부아키텍처(EA) 도입가이드」.
- [2] Armour, Frank J., Stephen H. Kaisler and Simon Y. Liu, "Building an Enterprise Architecture Step By Step", *IT Pro.*, 1999.
- [3] *BEA 3.1*, www.dod.gov/acq/ebusiness/products/architecture/BEA_3_1_March_2006/bea_3_1.html.
- [4] Berners, Lee, Tim, WWW Past, present and future.
- [5] Boar, Bernard, "Constructing Blueprint for Enterprise IT Architecture", John Wiley and Sons, Inc., 1998.
- [6] Department of Defense, *DoDAF Version 1.0-Deskbook, Volume I, II*.
- [7] *FEA Reference Models*, <http://www.whitehouse.gov/omb/egov/a-2-EAModelsNEW2.html>.
- [8] *FEA RMO ver 1.1*, www.topquadrant.com/documents/TQFEARMO.pdf.
- [9] H. Knublauch and R. W. Ferguson, "The Protégée OWL Plugin : An Open Development Environment for Semantic Web Applications", Third International Semantic Web Conference, 2004.
- [10] *Jess 7.0 Language Reference*, <http://herzberg.ca.sandia.gov/je>.
- [11] Martin O'Connor and Holger Knublauch, "Supporting Rule System Interoperability on the Semantic Web with SWRL."
- [12] N. Noy and D. L. McGuinness, "Ontology Development 101 : A Guide to Creating Your First Ontology", Technical Report KSL-01-05, Knowledge Systems Laboratory, 2001.
- [13] *PROTEGE*, <http://protege.stanford.edu/>.
- [14] *Racer Pro*, <http://www.sts.tu-harburg.de/~r.f.moeller/racer/>.
- [15] W3C, "OWL Web Ontology Language Guide", <http://www.w3.org/TR/2004/REC-owl-guide-20040210/>.
- [16] W3C, "SWRL : A Semantic Web Rule Language Combining OWL and RuleML", <http://www.w3.org/Submission/2004/SUBM-SWRL-20040521/>.

◆ 저 자 소 개 ◆

**김 왕 석** (vkkings@naver.com)

홍익대학교에서 컴퓨터공학을 전공하고, 홍익대학교에서 컴퓨터공학 석사 학위를 받았다. 주요 관심분야는 인공지능, EA 등이다.

**변 영 태** (ytbyun@hongik.ac.kr)

서울공대 학사, Indiana 대학 석사, Texas Austin 대학 박사, 홍익대학교 컴퓨터 공학대학 컴퓨터공학과 교수로 재직 중이며 주요 연구관심 분야는 인공지능(지식 표현 및 추론) 및 응용이며, 인공지능연구회, 인지과학회, 데이터마이닝 학회, 지능정보시스템 학회 회원으로 활동 중이다.