

XBRL 기반의 의사결정 모형 표현과 공유

김형도* · 박찬권** · 엄지환** · 이성훈**

XBRL-Based Representation and Sharing of Decision Models

Hyoung Do Kim* · Chankwon Park** · Jihwan Yum** · Sung Hoon Lee**

Abstract

Using an exchange standard, we can design an open architecture for the interchange of decision models and data. XML (eXtensible Markup Language) provides a general framework for creating such a standard. Although XML-based model representation languages such as OOSML were proposed, they are partly limited in expression capability, flexibility, generality, etc. This paper proposes a new method for expressing and sharing decision models and data based on XBRL (eXtensible Business Reporting Language), which is a XML language specialized in business reporting. We have developed a XBRL taxonomy for decision models with the concepts and relationships of a representative modeling framework, SM (Structured Modeling). The method allows for expressing data as well as decision models in a consistent and flexible manner. Diverse dependencies between components of SM models can also be affluently expressed.

Keywords : XBRL, Business Reporting, Structured Modeling, DSS, XML

1. 서 론

오늘날 모델링 도구나 환경은 모델과 데이터의 공유를 주장하는 고객들의 의견을 반영하여 시장에 존재하는 다른 도구나 환경과의 연결통로를 제공해야 한다. 만약 폐쇄적인 아키텍처를 채택하고 있는 도구나 환경이라면 다수의 입력/출력 변환기가 필요할 것이다. 다수의 변환기와 사적인 형식들이 모델과 데이터의 효과적인 관리를 어렵게 한다. 교환 표준이 있다면, 모델과 데이터의 상호교환을 위한 개방형 아키텍처를 설계함으로써 이런 문제를 해결할 수 있다.

XML[W3C, 2006b]은 데이터 구조를 생성하고 교환하기 위한 문법을 기술할 수 있는 일반적인 틀을 제공한다. OOSML[Kim, 2001]은 의사결정 모형을 교환하기 위한 목적으로 작성된 최초의 XML기반 모델링 언어로서 DTD(Document Type Definition)로 정의되었다. DTD는 XML에서 사용되는 기본적인 스키마 언어로서, XML문서의 구조 및 내용에 관한 정보를 정의하는데 사용된다. 그런데, DTD는 표현능력이 제한적이어서 응용에 따라서 필요한 보다 많은 정보의 표현에 있어서 제약이 많다. 그리고 모델과 함께 데이터를 표현하는 것이 불가능하여, 모델링 환경에서의 특별한 검증 처리가 필요하다. DTD의 단점을 개선한 XSD(XML Schema Definition)[Roy and Ramanujan, 2001; W3C, 2006a] 등의 스키마 언어가 있으나, 직접적으로 이를 이용해서 모델링 언어를 정의하게 되면 DTD에서와 유사한 한계점을 가지게 된다.

최근 기업의 투명성과 사회적 책임이 강조되면서, 비즈니스 보고(Business Reporting)에 특화된 XML 언어인 XBRL(eXtensible Business Reporting Language)[한국전자거래진흥원, 2006; Cohen and Hannon, 2000; Debreceeny and Gray, 2001; XBRL International, 2005a; 2005b]

의 활용도가 크게 증가하고 있다. 여기서 기업 보고란 일반적으로 재무제표, 재정 정보, 원장 처리, 연간 및 분기간 재무보고서와 같은 재무정보를 기본으로 하고, 그 외의 비재무적 정보까지 포괄하는 기업의 대내외적인 모든 정보 보고를 의미한다. XBRL은 기업 보고 문서에 필요한 개념들을 XSD로 정의하고, 여기에 이들과의 다양한 관계들을 추가하여 의미가 풍부한 모델을 작성하고, 이를 준수하여 사례 문서의 작성을 지원하고 검증할 수 있다는 점에서, 의사결정 모델링 언어의 제정에도 새로운 의미를 부여하고 있다.

이 논문에서는 XBRL을 기반으로 의사결정 모형을 표현하고 공유하는 방법에 대하여 제안한다. 본 논문의 구성은 다음과 같다. 먼저 제 2장에서는 XBRL과 함께 대표적인 의사결정 모형화 프레임워크인 구조적 모형화(Structured Modeling, 이하 SM으로 약칭)[Geoffrion, 1987]에 대하여 간략히 소개한다. 그리고 제 3장에서는 SM을 준수하는 의사결정 모형을 XBRL로 표현하는 방법을 제시하고, 제 4장에서는 그 장단점을 논한다. 마지막으로 제 5장에서는 결론 및 향후 연구방향을 제시한다.

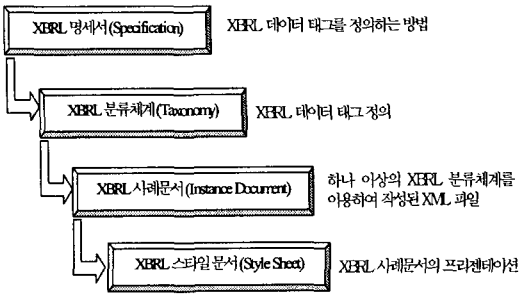
2. 관련연구

2.1 XBRL

XBRL문서의 구조는 <그림 1>과 같이 4단계로 도식화할 수 있다. 이들 각각의 기술적 특징은 다음과 같다.

XBRL 명세서(Specification)는 XBRL 프레임워크(Framework)와 문법(Syntax)을 기술하여 XBRL로 기업정보 문서를 작성하려는 사람들에게 기준을 제공한다.

XBRL에서 기업보고는 XBRL 분류체계(Ta-



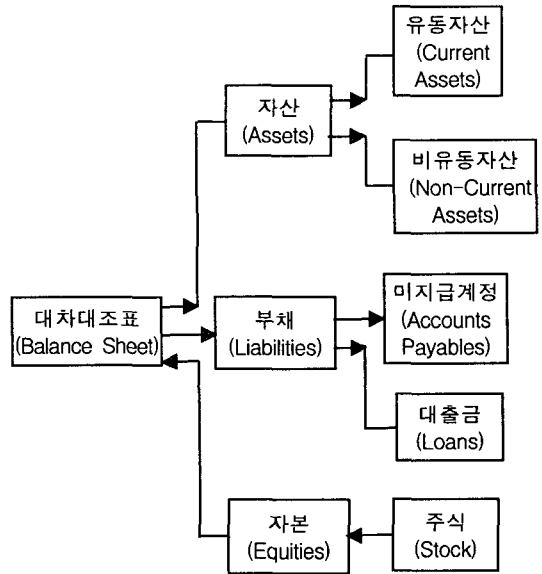
<그림 1> XBRL 문서의 구조

onomy)와 XBRL 사례문서라고 하는 두 종류의 문서들로 구성된다. XBRL 분류체계는 다시 개념과 관계라는 두 부분으로 구성된다. 개념에 대한 정의만이 XBRL 분류체계에서 정의되며, 이에 해당되는 구체적인 값은 XBRL 사례문서에 포함된다. 관계 부분은 개념들간의 관계와 함께 개념과 관련된 자원과의 관계를 포함한다. 개념 부분은 XSD를 사용하여 정의되며, 관계 부분은 W3C의 XLink와 XPointer를 사용하여 정의된다.

개념은 XSD의 element 요소를 사용해서 정의되는데, 아이템(item)과 튜플(tuple)이라는 두 종류가 있다. 아이터민 개념은 기업보고를 위한 구체적인 값을 포함할 수 있는 반면에, 튜플인 개념은 다른 개념들을 묶는 역할을 한다. <그림 2>에서 유동자산은 그 종류가 아이터민의 개념인 반면, 자산은 그 종류가 튜플인 개념이다. 개념에는 데이터 타입(type) 이 설정되어, 포함되는 값에 대한 제약을 명확히 하게 된다. 종류가 튜플인 개념에는 tupleType이라는 단일한 데이터 타입만 있는 반면, 종류가 아이터민 개념에는 monetaryItemType, sharesItemType, decinaItemType, stringItemType, uriItemType, dataTimeItemType 등이 있다.

이렇게 정의된 개념들을 상호간에 관계로 연결하거나 특정한 자원과 관계로 연결하여 분류체계를 완성하게 된다. XBRL 분류체계에서 사

용되는 관계에는 정의(Definition), 계산(Calculation), 프리젠테이션(Presentation), 라벨(Label), 참조(Reference)라고 하는 5가지 종류가 있다. 앞의 3가지 종류는 개념들을 연결하기 위한 것이고, 나머지 두 종류는 개념을 특정한 자원과 연결하기 위한 것이다.



<그림 2> 간단한 대차대조표의 주요 개념들(김형도, 2005)

이상과 같은 관계의 표현은 로케이터(Locator)와 아크(Arc)를 이용하여 독립적으로 표현되며, 개념 정의 부분에서는 확인이 불가능하다. 여기서 로케이터는 특정한 개념을 URI로 지칭하며, 아크는 관계에 참여하는 로케이터를 지칭한다. 아크는 로케이터를 지칭하면서 출발점(from)과 종착점(to)을 설정하여 단일한 방향성을 갖는다

기업보고 데이터가 포함되어 있는 XBRL 문서를 XBRL 사례 문서라고 한다. 이 사례 문서는 하나 이상의 XBRL 분류체계에서 정의된 개념들과 관계들을 참조하여 사실들을 값으로 기술한다. 그리고 이러한 사실에 관한 문맥 정보도 추가적으로 제공된다. 이렇게 XBRL 보고서가 분류체계와 사례 문서라고 하는 두 종류로

구분되어 있는 이유는 보고용어를 재사용하기 위한 것이다. 새로운 대차대조표를 작성할 때마다 변화되는 내용인 사례문서만 작성하면 되기 때문이다. 사례문서들이 동일한 개념들을 사용한다면, 사례 문서의 사실들을 상호 비교하는 것도 가능하다.

XBRL 사례 문서는 XML의 모든 규칙을 준수하는 XML 문서로서 XSD를 사용하여 검증될 수 있다. 따라서 XSD로 정의된 XBRL 분류 체계에 의해서 검증될 수 있다. XBRL 사례 문서에서 핵심은 실제 사실에 대한 보고 내용인 아이템이다. 이것은 XBRL 분류체계에서 아이템인 개념으로 정의되었다. 각각의 아이템은 해당되는 개념의 이름을 사용하여 표현된다.

XBRL 스타일 문서는 XBRL 사례 문서를 사용자가 선호하는 양식으로 변환하기 위한 문서다. XML에서는 이러한 변환을 수행하기 위한 목적으로 W3C 국제표준인 XSL(eXtensible Stylesheet Language) 이 제공된다. 이를 이용해서 XBRL 사례 문서를 HTML 등 다양한 형태로 변환이 가능하다.

2.2 구조적 모델링 (Structured Modeling)

구조적 모델링은 비반복 속성 그래프(Acyclic Attributed Graph)기반의 통합 모델링 틀로서, 모델의 구성요소간의 상호 참조를 표현하고, 이들의 추상화 수준을 표현할 수 있는 계층구조를 제공한다. 모델의 구조는 지너스(Genus)들을 이용하여 정의되며, 기본개체(Primitive Entity), 복합개체(Compound Entity), 속성(Attribute), 함수(Function), 테스트(Test)라고 하는 5가지 유형이 있다. 기본개체는 존재하는 어떤 것을 표현하며, 복합개체는 또 다른 개체를 참조하는 것으로서 값을 가질 수 없는 요소이다. 속성은 개체 또는 개체들의 어떤 특성과 값을 연결한

다. 속성 중에서 변수속성(Variable Attribute)은 모델에 의해서 그 값이 결정되어야 하는 속성을 말하는데, 이러한 속성들은 따로 변수속성 지너스 유형으로 분류된다. 함수는 속성과 또 다른 함수에 의존해서 결정되는 요소이다. 이러한 함수의 값은 일반 규칙에 의해서 결정된다. 테스트는 함수와 유사하나 그 값이 참 혹은 거짓만을 가질 수 있다. 기본개체 지너스를 제외한 모든 지너스는 호출순서(Calling Sequence)들을 갖게 되는데, 이들은 정의관계에서 지너스 간의 의존성을 표현하는데, 반복적인 의존성은 허용되지 않는다. 이러한 지너스들이 모여서 모듈(Module)을 이루며, 이들이 모여서 또 다른 모듈이 되는 나무구조를 이루게 된다. 이러한 구조는 모형을 다양한 추상화 수준에서 사용자가 검토할 수 있도록 허용한다.

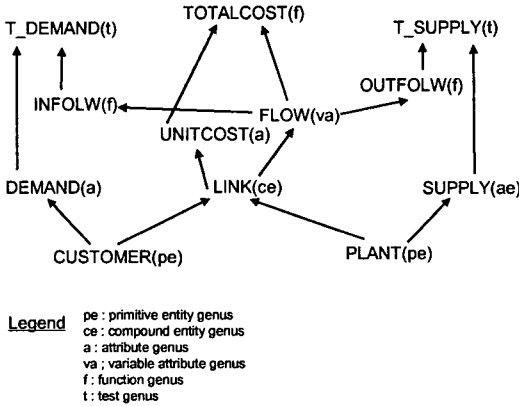
히치콕-쿠프만(Hitchcock-Koopmans)[Dolk, 1988; Geoffrion, 1987; Lenard, 1993]의 수송 문제를 사례로 해서 SM으로 표현해보면 <그림3>과 같은 지너스 그래프로 표현된다. 여기에는 제품의 고객과 공장에 해당되는 CUSTOMER와 PLANT라고 하는 두 개의 기본개체 지너스가 있다. 각 고객은 최소 수요량인 DEMAND를 가지고 있고, 각 공장은 최대 공급량인 SUPPLY를 가진다. 여기에는 수송로에 해당되는 LINK라고 하는 하나의 복합개체 지너스가 있다. 각 LINK는 한 PLANT로부터 한 CUSTOMER로 한 단위의 제품을 운송할 때 소요되는 비용을 나타내는 UNITCOST와 모델에 의해서 결정될 흐름량인 FLOW를 가지고 있다.

3. XBRL 기반의 구조적 모형화 방법

XBRL 기반으로 구조적 모델링을 지원하기 위해서는, <그림 4>와 같이 XBRL 스키마 기반 위에서 SM의 개념들을 정의하고, 이를 바탕으로

로 모델을 표현하며, 최종적으로 모델을 준수하여 모델 사례를 기술하는 3단계가 필요하다.

심볼로서 표현함으로써 제약이나 추론을 직접적으로 처리할 수 있도록 지원한다는 것이다. 구



〈그림 3〉 SM 지너스 그래프에 의한 수송문제의 표현

Model Instances
Model Taxonomies
SM taxonomy
XBRL 스키마

〈그림 4〉 XBRL 기반의 구조적 모델링 지원 체계

첫 번째 단계로 SM 택사노미에서는 SM의 개념들을 표현하게 된다. 6가지의 지너스 유형과 모듈, 그리고 모델링 대상 전체를 표현하는 모델(Model)을 〈그림 5〉와 같이 아이탬으로서 표현할 수 있다. 함수 지너스에서 사용하기 위한 fRule 속성, 테스트 지너스에서 사용하기 위한 tRule 속성을 추가적으로 정의하고 있다. 수리적 모형화 작업에서 대수적 표현 방법은 간결하고 명확하다는 장점이 있으나, 이해하기 어렵고 컴퓨터로 처리하기 어려운 단점이 있다. 이를 개선하기 위해 함수 지너스와 테스트 지너스의 일반 규칙을 요소 함수(Factorable Function) [Lenard, 1986]로서 개념화하는 방법을 채택하였다. 여기에서의 기본적인 원칙은 일반 규칙을

```

<element name="peGenus" type="xbri:stringItemType"
substitutionGroup="xbri:item" nillable="false"
id="id_peGenus" xbrli:periodType="instant" />
<element name="ceGenus" type="xbri:stringItemType"
substitutionGroup="xbri:item" nillable="false"
id="id_ceGenus" xbrli:periodType="instant" />
<element name="fGenus" type="xbri:stringItemType"
substitutionGroup="xbri:item" nillable="false"
id="id_fGenus" xbrli:periodType="instant" />
<element name="tGenus" type="xbri:stringItemType"
substitutionGroup="xbri:item" nillable="false"
id="id_tGenus" xbrli:periodType="instant" />
<element name="aGenus" type="xbri:stringItemType"
substitutionGroup="xbri:item" nillable="false"
id="id_aGenus" xbrli:periodType="instant" />
<element name="vaGenus" type="xbri:stringItemType"
substitutionGroup="xbri:item" nillable="false"
id="id_vaGenus" xbrli:periodType="instant" />
<element name="module" type="xbri:stringItemType"
substitutionGroup="xbri:item" nillable="false"
id="id_module" xbrli:periodType="instant" />
<element name="model" type="xbri:stringItemType"
substitutionGroup="xbri:item" nillable="false"
id="id_model" xbrli:periodType="instant" />
<attribute name="fRule">
<simpleType>
<restriction base="token">
<enumeration value="SCALE"/>
<enumeration value="SHIFT"/>
<enumeration value="POWER"/>
<enumeration value="MIN"/>
<enumeration value="MAX"/>
<enumeration value="LN"/>
<enumeration value="EXP"/>
<enumeration value="VSUM"/>
<enumeration value="VPROD"/>
<enumeration value="SUM"/>
<enumeration value="PROD"/>
<enumeration value="DOT"/>
</restriction>
</simpleType>
</attribute>
<attribute name="tRule">
<simpleType>
<restriction base="token">
<enumeration value="LE"/>
<enumeration value="LT"/>
<enumeration value="EQ"/>
<enumeration value="GT"/>
<enumeration value="GE"/>
</restriction>
</simpleType>
</attribute>
    
```

〈그림 5〉 SM Taxonomy

체적으로 fRule 속성에서는 'SCALE', 'SHIFT', 'POWER', 'MIN', 'MAX', 'LN', 'EXP', 'VSUM', 'VPROD', 'SUM', 'PROD', 'DOT' 등을 제공하며, tRule에서는 'LE', 'LT', 'GE', 'GT', 'EQ' 등을 제공한다. 속성 지너스나 함수 지너스가 가질 수 있는 값의 영역은 XBRL의 아이템 타입을 이용하여 정의될 수 있기 때문에, SM 텍사노미에서 따로 속성으로 정의할 필요는 없다.

두 번째 단계에서는 모델에서 필요한 개념들을 정의하여 모델 텍사노미를 구축하고, 이들을 SM 텍사노미의 개념들과 관계로서 연결하면 된다. 수송 문제를 모델로서 정의하고자 한다면, 이에 필요한 개념들로는 CUSTOMER, PLANT, LINK, DEMAND, SUPPLY, FLOW, UNITCOST 등이 있다. CUSTOMER의 경우, <그림 6>과 같이, 그 값이 stringItemType을 따르는 아이템으로서 정의된다. 기본개체 지너스임을 표현하기 위하여 정의 관계를 이용하는데, 정의 링크의 역할은 'GenusTyping'이다. 모든 지너스 유형의 지정에는 이 역할을 사용하는 정의 관계가 이용되어야 한다. 로케이터는 지칭하는 개념을 명시하기 위하여 href속성을 이용한다. definitionArc 엘리먼트에서는 from과 to 속성을 이용하여 출발점과 종착점인 로케이터를 지칭하고, arcrole속성에 의해서 그 역할을 "http://www.xbrl.org/linkprops/arc/parent-child"로 지정한다.

```
<element name="CUSTOMER" type="xbrli:stringItemType"
substitutionGroup="xbrli:item" nillable="false" id="id_CUSTOMER"
xbrli:periodType="instant" />
<definitionLink xlink:type="extended"
xlink:role="http://www.xbrl.org/2003/role/GenusTyping">
<loc xlink:type="locator" xlink:href="#id_CUSTOMER"
xlink:label="loc_CUSTOMER_01" />
<loc xlink:type="locator" xlink:href="#sm.xsd#id_peGenus"
xlink:label="loc_peGenus_01" />
<definitionArc xlink:type="arc" xlink:from="loc_CUSTOMER_01"
link:to="loc_peGenus_01"
xlink:arcrole="http://www.xbrl.org/linkprops/arc/parent-child" />
</definitionLink>
```

<그림 6> CUSTOMER 지너스의 정의

여기서 from 속성 쪽이 'parent'이고, to속성 쪽이 'child'이다.

함수와 테스트 지너스의 경우, 규칙에 대한 지정이 필요하다. <그림 7>과 같이 INFLOW는 함수 지너스로서 fRule이라는 속성을 이용하여 'SUM'을 사용함을 규정하고 있다. 지너스를 분류하기 위한 정의 링크와 함께, 규칙에 따른 계산에서 사용되는 지너스를 지정하기 위한 정의 링크를 사용하고 있다. 이 때 사용되는 링크의 역할은 'Calling'이다. 모든 호출 관계의 지정에는 이 역할을 사용하는 정의관계가 이용되어야 한다. 속성 지너스인 FLOW와 함수 지너스인 INFLOW의 값은 음이 아닌 정수이므로, XBRL의 nonNegativeIntegerItemType을 type 속성의 값으로 설정하고 있다.

모듈과 모델에 의한 그룹화는 <그림 8>과 같이 튜플을 이용하여 표현된다. 각 모듈이나 모델에서는 구성 요소들을 참조하여 사용하며, 몇 번이나 반복하여 사용할 수 있는지에 유의해야 한다. 'ModelTrans'에 그룹화되는 M_CUSTOMER, M_PLANT, M_LINK의 경우, 무한대의 반복사용이 가능하다. 이 모델에서 사용된 그룹화는 OOSM에서 제시된 방법을 따르고 있다. 모듈 구조를 객체 지향적인 방식으로 구성하고, 각각의 모듈과 전체 모델을 주요 관리대상으로 하는 방법이 OOSM[Kim, 2001]이다. <그림 9>는 <그림 3>의 SM 지너스 그래프를 OOSM의 지너스 그래프로 형상화한 것이다. 여기서 DEMAND, INFLOW, T_DEMAND는 CUSTOMER에 귀속되는 것으로 이들과 CUSTOMER가 모여서 하나의 모듈이 된다. 마찬가지로 PLANT와 LINK에 해당되는 모듈도 구성된다. 최종적으로 이들 모듈들과 TOTALCOST로 이루어지는 것이 전체 모델이 되는 것으로 표현되는 것이다.

세 번째 단계로 모델 사례를 작성하는 단계에서는 이제까지 작성된 XML 스키마를 준수하여

사례를 생성하면 된다. <그림 10>과 같이 ‘Trans Model’을 구성하는 M_CUSTOMER, M_PLANT, M_LINK, 그리고 TATALCOST를 기술하면 된다. 유의할 점은 LINK를 구성하는 PLANT와 CUSTOMER에 대한 기술이 문장으로 되어 있어서, 컴퓨터 처리에 문제가 발생할 수 있다는 점이다. 따라서 XBRL 사례 문서에서의 링크를 이용하여 이들을 명시적으로 표현한다.

```

<element name="INFLOW" type="xbri:
nonNegativeIntegerItemType" substitutionGroup="xbri:item"
nillable="false" id="id_INFLOW" xbrli:periodType="instant"
sm:fRule="SUM" />
<element name="FLOW" type="xbri:nonNegativeIntegerItemType"
substitutionGroup="xbri:item" nillable="false" id="id_FLOW"
xbrli:periodType="instant" />
...
<definitionLink xlink:type="extended"
xlink:role="http://www.xbrl.org/2003/role/GenusTyping">
<loc xlink:type="locator" xlink:href="id_INFLOW"
xlink:label="loc_INFLOW_01" />
<loc xlink:type="locator" xlink:href="sm.xsd#id_fGenus"
xlink:label="loc_fGenus_01" />
<definitionArc xlink:type="arc" xlink:to="loc_INFLOW_01"
link:from="loc_fGenus_01"
xlink:arcrole="http://www.xbrl.org/linkprops/arc/parent-child" />
</definitionLink>
<definitionLink xlink:type="extended"
xlink:role="http://www.xbrl.org/2003/role/Calling">
<loc xlink:type="locator" xlink:href="id_INFLOW"
xlink:label="loc_INFLOW_02" />
<loc xlink:type="locator" xlink:href="id_FLOW"
xlink:label="loc_FLOW_01" />
<definitionArc xlink:type="arc" xlink:to="loc_INFLOW_02"
link:from="loc_FLOW_01"
xlink:arcrole="http://www.xbrl.org/linkprops/arc/parent-child" />
</definitionLink>
...

```

<그림 7> 규칙에 대한 지정

4. 토 의

이 논문에서 제시하고 있는 XBRL 기반의 구조적 모델링 방법은 SM 프레임워크를 구체화

한 대표적 언어인 SML이나 OOSML 등과 비교하여 많은 장점을 가지고 있다. 무엇보다도 구조적 모델링의 핵심 개념인 호출 관계를 관계 정의와 거의 동일한 체제를 가지고 있어서 구조적 모델링이 보다 자연스럽게 진행될 수 있다는

```

<element name="M_CUSTOMER" substitutionGroup="xbri:tuple"
nillable="false" id="id_M_CUSTOMER"
xbrli:periodType="instant">
<complexType>
<sequence>
<element ref="id_CUSTOMER" />
<element ref="id_DEMAND" />
<element ref="id_INFLOW" />
<element ref="id_T_DEMAND" />
</sequence>
</complexType>
</element>
<element name="M_PLANT" substitutionGroup="xbri:tuple"
nillable="false" id="id_M_PLANT" xbrli:periodType="instant">
<complexType>
<sequence>
<element ref="id_PLANT" />
<element ref="id_SUPPLY" />
<element ref="id_OUTFLOW" />
<element ref="id_T_SUPPLY" />
</sequence>
</complexType>
</element>
<element name="M_LINK" substitutionGroup="xbri:tuple"
nillable="false" id="id_M_LINK" xbrli:periodType="instant">
<complexType>
<sequence>
<element ref="id_LINK" />
<element ref="id_UNITCOST" />
<element ref="id_FLOW" />
</sequence>
</complexType>
</element>
<element name="TransModel" substitutionGroup="xbri:tuple"
nillable="false" id="id_TransModel" xbrli:periodType="instant">
<complexType>
<sequence>
<element ref="id_M_CUSTOMER" maxOccurs="unbounded" />
<element ref="id_M_PLANT" maxOccurs="unbounded" />
<element ref="id_M_LINK" maxOccurs="unbounded" />
<element ref="id_TOTALCOST" />
</sequence>
</complexType>
</element>

```

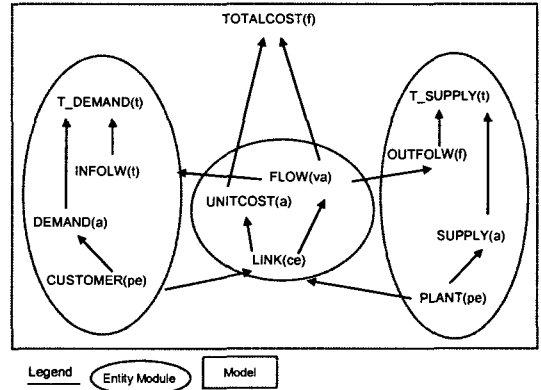
<그림 8> 모듈에 대한 정의

것이다. 이를 바탕으로 모델 사례를 자연스럽게 정의할 수 있는 체제를 제공하고 있다. XML DTD를 사용하여 정의된 OOSML의 경우 DTD의 단점과 더불어서, 모델과 모델 사례를 동일 차원에서 다루고 있다는 점이 문제다. 이 언어에서 사례의 구조는 'ANY'를 이용하여 임의적인 표현을 할 수 있다. 그러나 이렇게 사례를 표현하면, 모델의 개념들을 준수하여 제대로 표현하는지의 여부는 체크하기 어렵다. 이와 달리 이 논문에서 제시하고 있는 표현 방법은 스키마 수준에서 SM 텍사노미와 모델 텍사노미를 표현하고, 이를 기반으로 모델 사례를 모델링 한다는 점에서 장점을 가지고 있다. 모델 사례를 구성하는 대부분의 내용을 XML 스키마를 이용하여 검증할 수 있고, 부족한 부분은 링크를 이용하여 표현할 수 있다.

링크로서 표현된 여러 가지의 관계들은 다양한 분석에 이용될 수 있으며, 특히나 함수 지너스 또는 테스트 지너스와 이 지너스가 호출하는 또 다른 지너스간의 의존성은 XBRL의 계산 관계에 의해서 표현되는 것이 자연스럽다. 그러나 현재 계산 관계에서 지원되는 역할이 summation-item뿐이며, 개개의 아크에 이런 역할이 기술되는 관계로 비효율적이다. 이 논문에서는 함수 지너스나 테스트 지너스에 속성값으로 규칙을 선언하고, 계산에 필요한 의존성은 정의 관계로 표현하도록 하였다. 이러한 방법의 장점은 요소함수로서 표현된 일반 규칙을 선언적으로 정의하는 데 있어서 경제적이다.

모델과 모델에 의한 그룹화는 프리젠테이션 링크를 이용하는 것이 자연스럽다고 판단되나, 모델 사례에 대한 제약을 XML 스키마를 이용하여 바로 적용할 수 있도록 하기 위하여 튜플을 이용하여 정의하도록 하였다. 예를 들어서 M_CUSTOMER 모듈에 대한 그룹화를 프리젠테이션 링크로 정의하고자 한다면, <그림 11>

과 같이 많은 모델링이 필요할 뿐만 아니라, 이러한 그룹화 관계가 모델 사례에서 어떻게 성립하는가를 표현하는 작업이 추가적으로 사례 수준에서 필요하게 된다.



<그림 9> OOSM 지너스 그래프에 의한 수송문제 표현

5. 결 론

이 논문에서는 비즈니스 보고에 특화된 범용 XML 언어인 XBRL을 이용하여 의사결정 모형을 공유하기 위한 SM 텍사노미를 제시하였다. SM 텍사노미는 시맨틱 웹의 온톨로지와 유사하게 구조적 모델링 분야의 개념을 체계적으로 정립한 것으로서 의의가 크다고 하겠다. 이를 기반으로 SM 모델과 사례를 단계적으로 표현할 수 있도록 사례 문제와 함께 구체적인 방법을 제시하였다. XBRL을 이용하면, 의사결정 모형의 개념들과 이들간의 다양한 관계를 유연하게 그리고 직접적으로 표현할 수 있다. 모델과 사례에 대한 체계적인 모델링이 가능하며, SM에서 필요한 다양한 의존성을 풍부하게 표현하고 활용할 수 있다.

향후 연구해야 할 사항으로는 무엇보다도 먼저 MSOR/DSS 커뮤니티의 논의를 통해서 합의된 SM 텍사노미를 표준화하는 것이다. 이를

위해서는 XBRL 등의 표준화 기관과의 협조도

필요하다. 이러한 표준화를 통해서 의사결정 모형과 데이터의 공유에 필요한 시간과 비용을 대폭 절감할 수 있고, 오류를 줄일 수 있다. 둘째로는 이러한 표준이 모델링 도구와 환경에 도입되어 광범위하게 활용될 수 있도록 지원하는 것이다. 무엇보다도, XBRL에서는 다양한 관계를 링크로서 표현하고 있기 때문에, 이를 구체적으로 해석하고 검증하기 위한 지원이 필요하다. 한 가지 예로서, 모델 수준의 복잡한 관계들을 해석하여 모델 사례 수준의 데이터에 적용함으로써, 분석이나 검증, 모델링 안내를 자동화할 수도 있다. 이를 위해서 프로토타입 도구나 환경을 구축하여 그 가능성을 데모하고, 이를 통해서 의사결정 모형의 거래와 통합을 용이하게 하는 접근 라이브러리를 확보하는 작업이 필요하다.

```

<xbrl xmlns="http://www.xbrl.org/2003/instance"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:link="http://www.xbrl.org/2003/linkbase"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.hycu.ac.kr/xbrl/TransModel
TransModel.xsd"
xmlns:trans="http://www.hycu.ac.kr/xbrl/TransModel">
<context id="Context-2007">
  <entity>
    <identifier scheme="http://www.hycu.ac.kr/xbrl">
      Hyoung Do Kim
    </identifier>
  </entity>
  <period>
    <instant>2007-05-01</instant>
  </period>
</context>
<unit id="won"><measure>iso4217:KRW</measure></unit>
<trans:TransModel>
  <trans:M_CUSTOMER>
    <trans:CUSTOMER id="C1"
contextRef="Context-2007">NEWYORK
  </trans:CUSTOMER>
  <trans:DEMAND
contextRef="Context-2007">25000</trans:DEMAND>
  <trans:INFLOW contextRef="Context-2007">0</trans:INFLOW>
  <trans:T_DEMAND
contextRef="Context-2007">False</trans:T_DEMAND>
  </trans:M_CUSTOMER>
  ...
  <trans:M_PLANT>
    <trans:PLANT id="P1" contextRef="Context-2007">DALAS
  </trans:PLANT>
  <trans:SUPPLY
contextRef="Context-2007">20000</trans:SUPPLY>
  <trans:OUTFLOW
contextRef="Context-2007">0</trans:OUTFLOW>
  <trans:T_SUPPLY
contextRef="Context-2007">True</trans:T_SUPPLY>
  </trans:M_PLANT>
  ...
  <trans:M_LINK>
    <trans:LINK>DALAS-NEWYORK</ trans:LINK>
    <trans:UNITCOST
contextRef="Context-2007">23</trans:UNITCOST>
  < trans:FLOW contextRef="Context-2007">0</ trans:UNITCOST>
  </trans:M_LINK>
  ...
  <trans:TOTALCOST>0</trans:TOTALCOST>
  ...
</xbrl>

```

〈그림 10〉 모델 사례에 대한 정의

```

<presentationLink xlink:type="extended"
xlink:role="http://www.xbrl.org/2003/role/Modularization">
  <loc xlink:type="locator" xlink:href="#id_M_CUSTOMER"
xlink:label="loc_M_CUSTOMER_01" />
  <loc xlink:type="locator" xlink:href="#id_CUSTOMER"
xlink:label="loc_CUSTOMER_03" />
  <presentationArc xlink:type="arc"
xlink:to="#loc_M_CUSTOMER_01"
link:from="#loc_CUSTOMER_03"
xlink:arcrole="http://www.xbrl.org/linkprops/arc/parent-child" />
  <loc xlink:type="locator" xlink:href="#id_DEMAND"
xlink:label="loc_DEMAND_03" />
  <presentationArc xlink:type="arc"
xlink:to="#loc_M_CUSTOMER_01"
link:from="#loc_DEMAND_03"
xlink:arcrole="http://www.xbrl.org/linkprops/arc/parent-child" />
  <loc xlink:type="locator" xlink:href="#id_INFLOW"
xlink:label="loc_INFLOW_03" />
  <presentationArc xlink:type="arc"
xlink:to="#loc_M_CUSTOMER_01" link:from="#loc_INFLOW_03"
xlink:arcrole="http://www.xbrl.org/linkprops/arc/parent-child" />
  <loc xlink:type="locator" xlink:href="#id_T_DEMAND"
xlink:label="loc_T_DEMAND_03" />
  <presentationArc xlink:type="arc"
xlink:to="#loc_M_CUSTOMER_01"
link:from="#loc_T_DEMAND_03"
xlink:arcrole="http://www.xbrl.org/linkprops/arc/parent-child" />
</presentationLink>

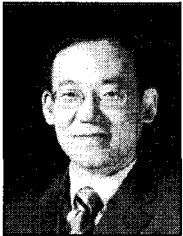
```

〈그림 11〉 프리젠테이션 링크에 의한 모듈화

참 고 문 헌

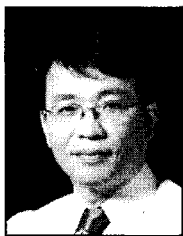
- [1] 김형도, “XML과 전자상거래”, *IE Magazine*, pp. 33-45, 2001.
- [2] 김형도, “중소기업 생산성 향상을 위한 XBRL기반의 지속가능성 보고 방안 연구”, *생산성논집*, 제19권 제4호, 2005, pp. 147-169.
- [3] 한국전자거래진흥원, *2006 e-Biz 표준화 백서*, 2006, pp. 388-402.
- [4] Cohen, E. E. and Hannon, N., “How XBRL Will Change Your Practice”, *The CPA Journal*, Vol. 70, No. 11, 2000, pp. 36-41.
- [5] Debreceeny, R. and Gray, G. L., “The Production and Use of Semantically Rich Accounting Reports on the Internet: XML and XBRL”, *Int'l Journal of Accounting Information Systems*, Vol. 2, 2001, pp. 47-74.
- [6] Dolk, D. R., “Model Management and Structured Modeling : The Role of an Information Resource Dictionary System”, *Communications of the ACM*, Vol. 31, No. 6, 1988, pp. 704-718.
- [7] Geoffrion, A. M., “An Introduction of Structured Modeling”, *Management Science*, Vol. 33, No. 5, 1987, pp. 547-588.
- [8] Kim, H.D., “An XML-Based Modeling Language for the Open Interchange of Decision Models”, *Decision Support Systems*, Vol. 31, No. 4, 2001, pp. 429-441.
- [9] Lenard, M. L., “An Object-oriented Approach to Model Management”, *Decision Support Systems*, Vol. 9, 1993, pp. 67-73.
- [10] Lenard, M. L., “Representing Models as Data”, *Journal of Management Information Systems*, Vol. 2, No. 4, 1986, pp. 36-48.
- [11] Roy, J. and Ramanujan, A., “XML Schema Language : Taking XML to the Next Level”, *IT Professional*, Vol. 3, No. 2, 2001, pp. 37-40.
- [12] W3C, *XML Schema*, <http://www.w3.org/XML/Schema>, 2006a.
- [13] W3C, *Extensible Markup Language (XML) 1.0 (Fourth Edition)*, <http://www.w3.org/TR/xml>, 2006b.
- [14] W3C *XML Pointer Language (X Pointer)*, <http://www.w3.org/TR/xptr/>, 2002.
- [15] W3C, *XML Linking Language (X Link) Version 1.0*, <http://www.w3.org/TR/xlink/>, 2001
- [16] XBRL International, *XBRL 2.1*, <http://www.xbrl.org/SpecRecommendations/>, 2005a.
- [17] XBRL International, *XBRL 2.1 Conformance Suite*, <http://www.xbrl.org/SpecRecommendations/>, 2005b.

■ 저자소개



김 형 도

서울대학교 산업공학과에서 학사, 그리고 한국과학기술원 경영과학과에서 석사와 박사 학위를 취득하였다. 현재 한양사이버대학교 부교수로 재직하고 있으며, ECIF 전자문서기술위원회 부위원장, 전자거래진흥원 비즈니스 프로세스 워킹그룹 위원장 등을 맡고 있다. ebXML 전문위원회 위원장을 역임한 바 있다. 새로운 전자거래 기반을 중심으로 연구개발, 교육 및 표준화 활동을 활발히 전개하고 있으며, Decision Support Systems, ACM SIGMOD Record, IEICE Transactions on Communications, Expert Systems with Applications, Int'l Journal of Management Science 등에 다수의 논문을 게재하였다. 저서로는 "B2B 전자상거래@XML", "전자상거래 원론", "전자문서 용어사전", "데이터마이닝의 이해" 등이 있다. 주요 관심분야는 XML, 전자상거래, 비즈니스 프로세스, 디지털 워터마킹, IT 아키텍처, 데이터마이닝 응용 등이다.



박 찬 권

서울대학교에서 산업공학을 전공하였고, 동 대학원에서 석사 및 박사학위를 취득하였다. 서울대학교 자동화시스템 공동연구실에서 Post Doc을 거친 후, 영산대학교 정보경영학부에 재직하였으며, 현재는 한양사이버대학교 경영학부에 재직 중이다. 현재 한국전자거래진흥원의 전자문서워킹그룹위원장을 맡고 있으며, 주요 관심분야는 ERP/ SCM, 정보시스템 개발 방법론, XML, 전자문서 개발방법론 및 표준화 활동 등이다.



염 지 환

한양대학교에서 무역학을 전공하였고, University of Nebraska에서 경영학 석사 (Master of Art in management)를, US International University에서 전략경영 전공으로 경영학 박사(Doctor of Business Administration) 학위를 취득하였다. 현재 한양 사이버 대학교 경영학부 교수로 재직 중이다. 주요 관심분야는 정보기술전략과 기업성과, e-비즈니스 전략, e-learning, 전략 유형별 조직 성과 등이다.



이 성 훈

연세대학교에서 정치외교학을 전공하였고, University of Pittsburgh에서 경영학 석사(MBA)를, State University of New York에서 재무학(Finance) 전공으로 경영학 박사(Ph.D.) 학위를 취득하였다. 대우실업, 한미은행을 거쳐 1985년부터 2002년까지 LG투자증권에서 근무하면서 Research Center장, 자산운용부문장 및 기획/인사부문장 등으로 재직하였고, 2002년부터 2006년까지 델타투자자문(주)의 대표이사/사장으로 금융시장 실무 경험을 쌓았다. 현재 한양사이버대학교 경영학부 교수로 재직중이며 주요 관심분야는 자산운용 시스템 설계, Mutual Fund 성과 평가, 금융기관 경영전략 등이다.