

# 시맨틱 웹 기반의 비교구매 에이전트를 위한 동적 웹 온톨로지에 대한 연구

김수경

국립 한밭대학교 정보통신전문대학원 컴퓨터공학과  
(kimsk@hanbat.ac.kr)

안기홍

국립 한밭대학교 정보통신전문대학원 컴퓨터공학과  
(khahn@hanbat.ac.kr)

본 논문에서는 전자상거래 상점별로 상이하게 정의되고 표현되어 있는 디지털 캠코더에 대한 HTML 페이지의 상품 정보를 래퍼(Wrapper)기술을 이용하여 획득하고 이를 RDF 문서 변환기를 통해 RDF 트리플(triple)과 RDF 문서로 변환하여 디지털 캠코더에 대한 메타데이터 스키마를 설계한다. 설계된 메타데이터 스키마를 기반으로 OWL 웹 온톨로지기로 변환하고 이를 관계형 데이터베이스로 구현된 디지털 캠코더(DCC: Digital Camcorder) 도메인 온톨로지 저장소(Domain Ontology Repository)에 DCC 지식 베이스 온톨로지(DCCKBO: DCC Knowledge Based Ontology)로 저장한다. 다음 각 상점의 RDF 트리플과 문서를 DCCKBO와 비교, 매핑 그리고 추론 과정을 통해 최적의 상품 구매 정보를 가진 상점의 DCC 정보를 구매자에게 제공하고, DCCKBO에 저장되어 있는 도메인 온톨로지를 최적의 상품 구매 정보의 내용으로 재 정의하는 동적 웹 온톨로지를 제안하고자 한다.

논문접수일 : 2005년 5월

게재 확정일 : 2005년 10월

교신저자 : 김수경

## 1. 서론

기존의 HTML을 기반으로 한 전자상거래 상점은 구매자가 찾고자 하는 정보의 정확한 위치와 경로 파악에 큰 어려움이 있다. 구매자가 상품을 비교 검색하는 데 있어서 한 상점에서 행한 검색 과정을 다른 상점에서도 유사하게 반복적으로 수행하여야 하며, 상점 별로 상품에 대한 정보 표현이 상이할 경우에는 검색과 비교에 더 큰 어려움이 있다. 이런 문제점을 해결하고자 비교 구매 에이전트(Comparison- Shopping Agent, Shopbot)가 개발되었다.

비교 구매 에이전트는 실시간 검색 또는 제품정보 DB 형태로 개발이 되고 있으나 국내에서는 실

시간 검색이 아닌 제품 데이터베이스 기반 검색으로 제공된다. 그리고 현재 대부분의 비교구매 에이전트는 가격을 포함한 상품 사양의 일부 정보만이 수집 가능 할 뿐, 상품에 대한 전체 사양 정보를 완벽히 수집하고 있지 못하여 상품 간 완전한 비교가 어려울 뿐더러 지능형 구매지원을 하지 못하므로 구매자의 다양한 요구를 만족시키지 못하고 있다. 그리고 구매자가 상품에 대한 정보를 정확하게 입력하지 못할 경우에는 그 효용성이 저하된다. 또한 전자상거래 상점에서 상품에 대한 정보 표현이 상이할 경우에는 비교 구매 에이전트의 검색에 있어 많은 문제가 발생되고 있다[7].

본 논문에서는 시맨틱 웹의 제반 기술들을 이용하여 전자상거래 상점별로 상이하게 정의되고 표

현되어 있는 디지털 캠코더에 대한 HTML 페이지의 상품 정보를 래퍼(Wrapper)기술을 이용하여 획득하고 이를 RDF 문서 변환기를 통해 RDF 트리플(triple)과 RDF 문서로 변환하여, 디지털 캠코더에 대한 메타데이터 스키마를 설계한다. 설계된 메타데이터 스키마를 기반으로 OWL 웹 온톨로지로 변환하여 이를 관계형 데이터베이스로 구현된 디지털 캠코더(DCC: Digital Camcorder) 도메인 온톨로지 저장소(Domain Ontology Repository)에 DCC 지식 베이스 온톨로지 (DCCKBO: DCC Knowledge Based Ontology)로 저장하고, 각 상점의 RDF 트리플과 문서를 DCCKBO와 비교, 매핑 그리고 추론 과정을 통해 최적의 상품 구매 정보를 가진 상점의 DCC 정보를 구매자에게 제공하고, 기존에 도메인 온톨로지 저장소에 있는 DCCKBO를 최적의 상품 구매 정보의 내용으로 재 정의하는 동적 웹 온톨로지 시스템을 제안하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서는 전자상거래 비교 구매 에이전트와 시맨틱 웹과 온톨로지 그리고 래퍼 기술 등에 관련된 이론적 배경을 알아보고, 3장에서는 본 논문에서 제안하는 시스템을 살펴보고, 4장에서는 결론 및 향후 연구 방향에 대해 살펴본다.

## 2. 관련기술 분석

### 2.1 전자상거래와 비교 구매 에이전트

현재 대부분의 전자상거래 서비스는 구매자가 물건의 모습이나 사양을 보고 선택하는 상품 카탈로그 수준의 상태에 있다. 구매자는 자신이 원하는 물건의 사양을 간단히 표시하여 그 사양에 맞는

상품을 여러 업체로부터 제시 받기를 원하고 있으나 그것을 지원하는 시스템은 매우 적다. 그리하여 구매자가 특정 상품을 찾고자 할 때 에이전트가 구매자를 대신하여 인터넷에 존재하는 상점을 검색하여 상점별로 해당 상품을 비교하는 시스템인 비교 구매 에이전트가 앤더슨 컨설팅의 Bargain Finder에 의해 시작되었다[5]. 그것은 미리 등록된 9개의 인터넷 상점의 상품에 대한 상품 정보 DB를 구축하여 구매자가 특정 음반 아티스트와 음반 제목을 입력하면, 여러 CD 상점의 판매 가격을 소비자에게 실시간으로 제시하는 기능이었지만 CD 판매가격이 상대적으로 비싼 상점들이 비교구매 에이전트의 정보 수집을 거부(Agent blocking) 하면서 상업적으로 실패하게 되었다. 이러한 거부 문제를 해결하기 위해 워싱턴 대학에서 개발된 Jango라는 에이전트는 구매자의 PC에 클라이언트 소프트웨어를 설치하여 인터넷 상점의 접속을 구매자의 컴퓨터에서 할 수 있도록 하여 구매자가 많은 인터넷 상점의 가격을 비교할 수 있게 하였으나 상점의 분석 성공률이 50% 이하로 나타나는 등의 문제로 서비스가 중단이 되었다. 그리고 호주의 Curtin 대학에서 개발된 다중 스탠드와 다중 연결 구조를 가진 BarginBot가 있다. 그리고 MIT Media Lab에서 연구 중인 에이전트 기반 가상 시장인 Kasbah는 구매자와 에이전트와 판매자 에이전트를 각각 나누어서 서로의 이익을 위해 교섭해서 물건을 사고 팔 수 있도록 제공한다. 이외에도 많은 비교구매 에이전트가 개발되었으나 관련 정보의 검색, 거래 상대의 물색, 개인 적응화 기능, 기존의 전자상거래 시스템 또는 데이터베이스와의 상호 운용을 위한 방법과 상품 온톨로지 문제에 대한 대응 방안 등의 문제점들이 대두 되었다.

현재 운영되는 대부분의 국내의 비교구매 에이전트는 Bargain Finder와 같은 방법으로 제공되고

있으며, 상품 간 비교에 있어 상품 정보가 특정 상점에는 존재하나 다른 상점에는 존재하지 않거나 비교 자체가 불가능해지고, 또한 구매자의 다양한 요구를 만족시켜주는 지능형 구매 지원을 하지 못하는 등의 문제점을 갖고 있으나, 현재 시맨틱 웹의 출현으로 전자상거래 관련 연구에 있어서 시맨틱 웹의 정보 검색분야와 온톨로지를 활용하고자 하는 다양한 연구가 현재 진행 중에 있다.

## 2.2 래퍼(Wrapper)

래퍼는 특정 정보 소스로부터 필요한 정보만을 추출하는 프로시저 (procedure)이다[8]. 래퍼는 정보 소스에서 필요한 정보만을 추출하기 위한 코드나 규칙으로 구성되어 있고 비교 구매 에이전트에 사용되는 래퍼는 각 전자상거래 상점에서 제공하는 하나 이상의 상품 정보 위치와 이를 추출하기 위한 규칙을 담고 있다. 래퍼는 수동 생성되거나 에이전트에 의해 자동 생성될 수 있다. 수동 생성의 경우 단순하고 시간 소비적인 지루한 작업이며 비교 쇼핑 시스템의 확장이 어려울 수 있고 새로운 전자상거래 상점이 생성되면 수동으로 전자상거래 상점에 방문하여 이에 맞는 상품정보 추출 규칙이 만들어 져야 한다.

래퍼 자동 생성의 경우 에이전트가 HTML로 작성된 정보를 분석해야 한다. HTML은 정보를 어떻게 사용자에게 출력해줄 것 인가에 대해서만 정의하고 있기 때문에 HTML에 포함된 정보의 의미는 사용자가 판단해야 한다. 현재 전자상거래 상점들은 상품 정보에 대해 준 구조화(semi-structured)의 구조를 제공하므로 어느 정도 필요한 정보는 추출할 수 있다. 자동 생성의 문제점으로는 전자상거래 상점의 이형질(heterogeneity)성이 있다. 이형질성을 극복하기 위한 방법으로 래퍼

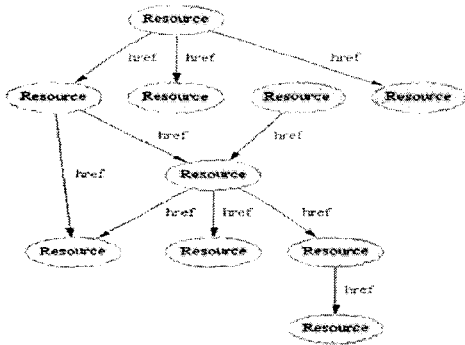
인덕션(induction)이 있으며 ShopBot과 같은 시스템은 이 방법을 이용하였다[3].

## 2.3 시맨틱 웹과 웹 온톨로지

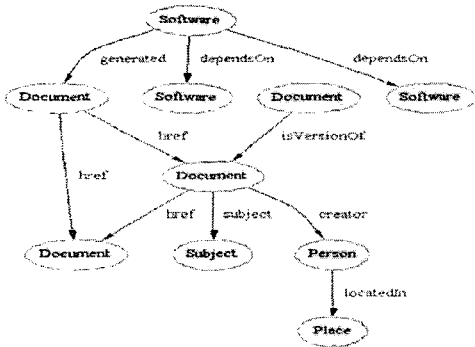
시맨틱 웹은 웹상의 정보에 잘 정의된 의미를 부여하여 사람뿐만 아니라 컴퓨터도 쉽게 문서의 의미를 해석할 수 있도록 함으로서 컴퓨터를 이용한 정보의 검색 및 해석, 통합 등의 업무를 자동화하기 위한 목적으로 제안되었다[11]. 시맨틱 웹이 발전하기 위해 필요한 분야가 온톨로지인데, 온톨로지는 특정 개념(concept)에 대한 의미(semantic)을 표현하기 위한 개념과 개념들 간의 관계를 이용한 지식 표현 방식이다. 기존 온톨로지에 있어 지식 표현이 너무 방대하고 난해하여 웹 상에서 이용하기 위한 웹 온톨로지(Web Ontology)를 개발하였다. 웹 온톨로지는 웹상에 존재하는 자원(Resource)들에 대한 온톨로지로서 W3C 표준인 RDF를 기반으로 정의된 온톨로지 언어를 통해 기술된다. 웹 온톨로지 언어에서는 일반적으로 개념을 클래스로 관계(relation)를 특성(property)로 표현한다[2].

[그림 1]은 현재의 웹과 시맨틱 웹에서의 웹 자원간의 연결 관계를 나타낸다. 기존의 HTML 기반의 웹에서는 원하는 정보를 찾기 위해 키워드를 기반으로 하는 디렉토리(directory) 검색 또는 HTML 페이지의 구문(syntax)에 대한 match-making을 통한 인덱스 검사를 수행하게 되는데 이러한 경우 사용자가 원하는 정확한 의미를 갖는 정보만을 찾아내기 어려우며 동일한 의미를 갖는 다른 키워드를 포함한 정보(synonym)을 찾아낼 수 없는 등의 비효율성이 존재한다.

시맨틱 웹은 웹 온톨로지를 기반으로 웹상에 존재하는 자원(HTML 페이지도 포함)들에 대한 기



a. 현재의 웹



b. 시맨틱 웹

[그림 1] 웹 자원 간의 연결 관계

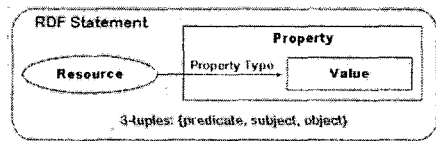
술(description)을 하고 검색 에이전트를 이용하여 의미론적 기술에 대한 추론(reasoning)을 통해 필요로 하는 정보를 보다 정확하게 찾아내는 것을 가능하게 해준다.

## 2.4 웹 온톨로지 언어

### 2.4.1 RDF와 RDFS

RDF(Resource Description Framework)는 W3C의 가장 기본적 시맨틱 웹 언어로서 웹에 있는 자원에 관한 메타 데이터를 표현하기 위한 언어이다. RDF 문서는 URI(Uniform Resource

Identifier)를 네임스페이스로 하는 RDF 모델을 기반으로 XML의 문법을 이용하여 기술하게 된다. RDF 모델은 기본적으로 자원(Resource), 특성(Property), 서술문(Statement)의 개념으로 구성된다. 서술문은 일반 문자의 주어(subject), 동사(predicate), 목적어(object)에 해당하는 것으로서 사람이나 웹 문서 등 특정 대상(object)이 가지고 있는 상태를 표현하며 이것이 RDF 문의 기본 단위가 되며, [그림 2]와 같은 RDF 데이터 모델로 표현된다.



[그림 2] RDF의 데이터 모델

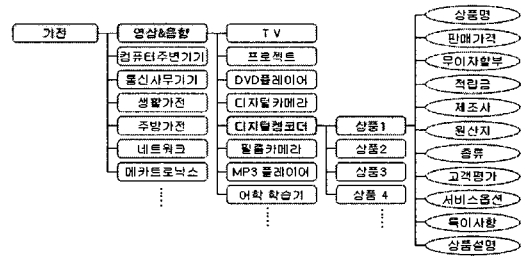
RDF의 새로운 용어를 정의하기 위해서 RDF 스키마(Schema)를 사용한다. RDF 스키마는 특성에 대한 정의나 사용상의 제약 사항을 기술한 것으로 RDF 문을 구성하는 용어(term)을 정의하고 용어들의 세부적인 의미를 기술하고 있으며 이 RDF 스키마는 일반적으로 어휘(vocabulary)로 미리 정의되어 RDF 문서에 사용된다.

### 2.4.2 OWL(Web Ontology Language)

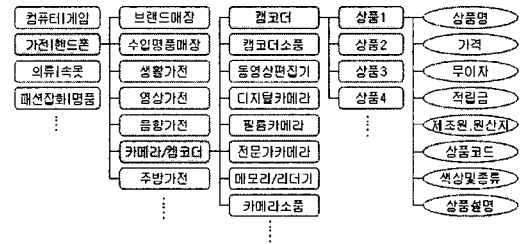
RDF 모델의 부족한 모델링 요소를 확장하고 강화할 필요성과 RDF 스키마에 동의요소나 역관계 등의 다양한 표현력을 부가하여 논리적 추론이 가능한 온톨로지 구축의 필요성에 의해 다양한 웹 온톨로지 언어들이 개발되었으며, 그 중 OWL은 DAML+OIL를 기반으로 발전된 형태로 시맨틱 웹을 위한 W3C의 웹 온톨로지 표준 언어이다[6, 15]. OWL은 DAML+ OIL과 유사한 형식을 가지며,

DAML+OIL의 네임 스페이스와 속성, 클래스 이름 등을 변경하고 RDF/RDFS의 변화를 수용하였다. OWL은 DL (Description Logic)을 기반으로 만들어진 RDF 확장 언어이며, OWL의 axioms은 DAML+OIL의 axioms 보다 더욱 풍부한 표현력을 가지고 있으며, 클래스나 속성 간의 명제(subsumption)나 동치(equivalence) 등의 다양한 성격을 선언하는데 사용된다[4]. OWL은 XML 스키마의 모든 데이터형을 지원하며 RDFS와 밀접하게 연관되어 있다. OWL은 현재 OWL Lite, OWL DL, OWL Full로 나누어져 있다.

인 상품에 대한 텍사노미(taxonomy)를 분석한 결과 [그림 3]과 같이 동일한 상품임에도 불구하고 상점 간에 상이한 텍사노미 구조를 갖고 있다.



a. L사의 상품 카테고리



b. C사의 상품 카테고리

[그림 3] 각 상점별 상품 카테고리

### 2.5 RDQL(RDF Query Language)

RDQL은 RDF 문서를 질의하기 위한 언어로 Jena2에 내장되어 있다. RDQL은 SquishQL과 rdfDB에서 파생된 질의 언어로 SQL과 유사한 구문 형태를 제공하고 데이터 지향(data-oriented)의 질의 모델을 제공한다[1]. 이로 인해 RDQL은 RDF의 그래프 구조를 기반으로 기술된 데이터를 질의할 수 있지만 추론을 위한 방법은 제공하지 않는다. 반면 Jena2의 Reasoner API를 이용하여 지식베이스의 상·하위 클래스(subClassOf) 혹은 속성(subProperty) 간의 관계성을 검색할 수 있다.

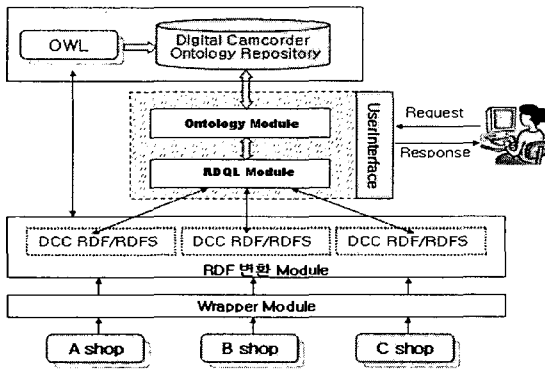
## 3. 시스템 제안

시맨틱 웹을 위한 웹 도메인 온톨로지를 구축하기 위한 첫 번째 단계는 온톨로지의 도메인과 영역을 결정하는 것이다[13]. 본 연구에서는 시맨틱 웹을 기반으로 한 비교구매 에이전트를 지원하기 위한 디지털 캠코더에 대한 웹 도메인 온톨로지를 구축하기 위해 현재 전자상거래 상점에서 판매 중

이러한 텍사노미를 기본으로 비교구매 에이전트가 상품에 대한 정보를 추출하여 비교하는 것은 쉽지 않다. 특히 대부분의 전자상거래 상점들은 비교구매 에이전트가 각 상점의 데이터베이스에 접근하여 정보를 검색하는 것을 막고 있다.

따라서 [그림 4]에 제시된 본 시스템은 ① 전자상거래 상점별로 상이하게 정의된 상품 정보를 래퍼 모듈(Wrapper Module)을 통해 필요한 정보만 추출한 후 ② RDF 모듈 내의 RDF 문서 변환기를 이용해 각 전자상거래 상점의 상품에 대한 정보를 각각의 RDF 트리플과 RDF 문서로 변환한다. ③ 미리 상품에 대한 웹 도메인 온톨로지를 구축하기 위해 상품 정보를 온톨로지로 설계한 뒤 이 설계

를 기반으로 웹 온톨로지로 변환하여 DCC 도메인 온톨로지 저장소에 DCKBO로 저장한다. ④ 각 상점의 RDF 트리플로 정의된 메타데이터를 RDQL을 이용해 검색하여 분류한 뒤 ⑤ DCC 지식 베이스와의 비교 분석을 거쳐 DCC 지식베이스 온톨로지에 정의된 상품에 대한 속성과 값보다 좋은 결과가 나올 경우 ⑥ DCC 지식베이스 온톨로지의 값을 동적으로 재 정의하여 DCC 도메인 온톨로지 저장소에 저장하고, ⑦ 구매자에게는 DCC 지식베이스 온톨로지의 조건에 가장 적합한 상품의 정보를 제시하도록 설계하였다.

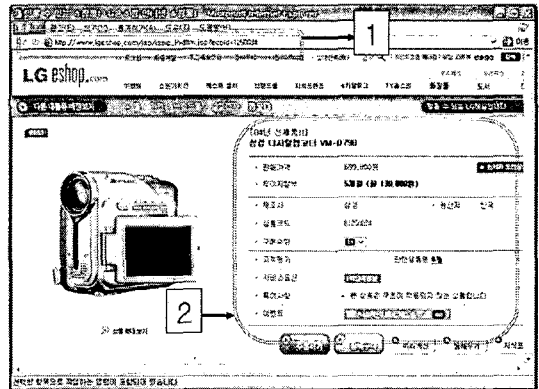


[그림 4] 비교구매 동적 웹 온톨로지 시스템

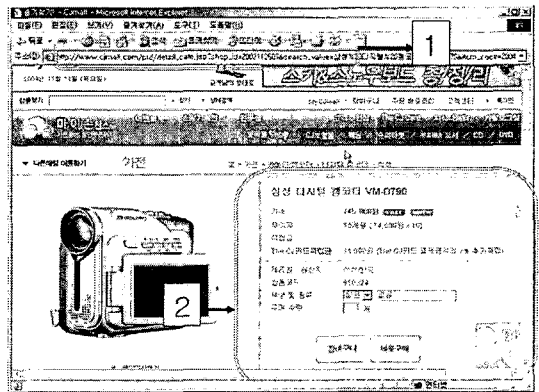
이렇게 동적으로 정의된 온톨로지는 상점별로 메타데이터의 표현이 상이하게 되어 있더라도 메타데이터의 의미적 상호 운용이 가능한 온톨로지를 이용함으로써 상품의 비교와 검색에 효율적으로 이용될 수 있고, 상품에 대한 비교와 검색에 있어서도, DCKBO에 정의된 다양한 속성과 비교하고 검색할 수 있으므로 강력한 비교 구매 에이전트의 연구에 활용되고 DCKBO가 상품을 비교하고 검색할 때 온톨로지가 동적으로 재정의됨으로써 웹 온톨로지를 유연하고 일관성 있게 유지한다.

### 3.1 Wrapper Module 적용

본 제안 시스템의 래퍼 모듈은 동적 생성 방법을 이용한다. [그림 5, 6]과 같이 전자상거래 상점에서 상품 정보를 보여주는 HTML 페이지는 상이하게 나타난다. 먼저 [그림 5, 6]에 있는 (1)번의 내용인 상품 정보가 표현된 페이지의 URL을 래퍼 모듈에 입력한다. 상품에 대한 정보가 표시되어 있는 [그림 5, 6]의 (2) 부분은 각 태그와 표현 내용으로 분리하여 불필요한 태그 부분은 삭제하고 나머지 상품 정보에 해당되는 내용만 추출한다.



[그림 5] L상점의 DCC 정보 HTML 페이지



[그림 6] C상점의 DCC 정보 HTML 페이지

이 래퍼 모듈에 입력될 URL은 <표 1>과 같다.

<표 1> 래퍼 모듈에 입력될 URL

상점	입력 URL
L사	http://www.gseshop.com/jsp/jspip_prdltm.jsp?ecpid=1250034
C사	http://www.cjmall.com/prd/detail_cate.jsp?shop_id=2002112507&search_value="삼성%20디지털%20캠코더%20VMD750&item_code2

입력된 URL을 참조하여 래퍼가 해당 HTML 페이지를 방문하여 읽어온 페이지 내용(content)은 래퍼 생성기를 통해 <표 2>와 같이 추출되어 저장된다. 추출 결과는 RDF로의 변환 처리를 위하여 RDF 모듈로 전송한다.

<표 2> 래퍼 모듈에 의해 추출된 상품 정보

L사 HTML 페이지	C사 HTML 페이지
[04년 신제품]	삼성디지털캠코더VM-D790
삼성디지털캠코더VMD-790	가격
판매가격	745000
790000	무이자
무이자할부	10개월(74,500X10)
5개월(월139,800원)	적립금
제조사	The CJ 카드 적립금
삼성	14,900원
원산지	제조원/원산지
한국	삼성/한국
상품코드	상품코드
81250034	910324
구매수량	색상및종류
고객평가	없음
서비스옵션	구매 수량
특이사항	개
이벤트	

<표 2>의 결과처럼 래퍼 모듈을 통해 추출된 DCC 정보는 동일한 상품임에도 불구하고 전자상거래 상점의 HTML 페이지의 표현(representation) 한계에 의해 같은 DCC 정보로 검색하고 분석하기에는 많은 어려움이 있다. 이러한 문제점을 극복하기 위해 이 추출 결과를 기반으로 다음 절에서 각 상점별로 메타데이터를 설계하여 RDF 트리플과 RDF 문서로 변경한다.

### 3.2 각 상점별 DCC RDF 설계

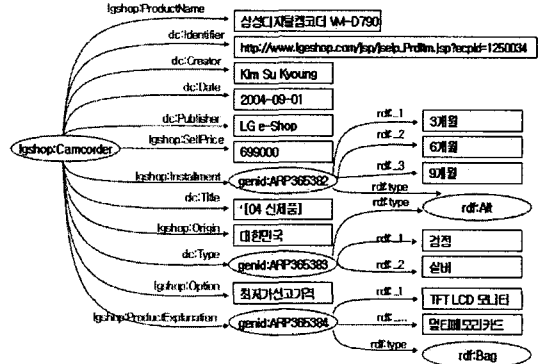
추출된 DCC 정보를 기반으로 상점별로 RDF 엘리먼트(element)와 RDF에서 DCC 정보를 유일하게 구별할 네임스페이스를 RDF 모듈에 있는 RDF 변환 프로세스를 통하여 설계한다. 이 RDF 변환 프로세스에서는 표현된 내용을 카테고리(Category)로 정하고 각 카테고리를 Resource와 Property로 구분한다. 각 카테고리에 적당한 엘리먼트를 부여하게 되는데 본 연구에서는 각 상점에서 한글로 표현된 의미에 맞추어 부여하였다. 그리고 구분에서 Property로 정해진 엘리먼트에는 각각의 Value를 할당하고 각 엘리먼트에 사용될 접두사(prefix)를 정한 뒤 다양한 Attribute(속성)을 갖는 Property 인 경우에는 그에 맞는 속성을 할당한다.

RDF 변환 프로세스에 의해 설계된 <표 3>과 <표 4>의 정보는 RDF 데이터 모델로 변환되어 DCC 상품에 대한 Resource와 Property 그리고 Value에 대한 관련성을 [그림 7, 8]과 같은 RDF 데이터 모델로 보여주었다. <표 5, 6>은 RDF 데이터 모델이 Subject, Predicate, Object의 형태인 RDF 트리플 데이터 모델로 변환된 결과의 일부이다.

<표 3> L사 RDF 엘리먼트

NameSpace(URI)	http://www.lgshop.com/lgshop/#				
Category	구분	Element	Value	prefix	Attribute
캠코더	Resource	Camcorder		rdf	
상품명	Property	ProductName	삼성디지털캠코더 VM-D790	lgshop	
판매가격	Property	SellPrice	699000	lgshop	
판매일	Property	Title	'04년 신제품	lgshop	
방문일	Property	Date	2004.9.1	dc	
무이자할부	Property	Installment	3,6,9,12	lgshop	rdf:Alt
연월금	Property	Save		lgshop	
제조사	Property	Maker	삼성	lgshop	
원산지	Property	Origin	대한민국	lgshop	
종류	Property	Type	결정/일반	dc	rdf:Alt
서비스옵션	Property	Option	최저가신고기여	lgShop	
상품설명	Property	ProductExplanation	▶ 2.5" TFT LCD 모니터 외...	lgShop	rdf:Bag
URL	Property	Identifier	http://www.lgshop.com/jsp/jsp/jeip_PrdItem.jsp?cpid=1250034	dc	

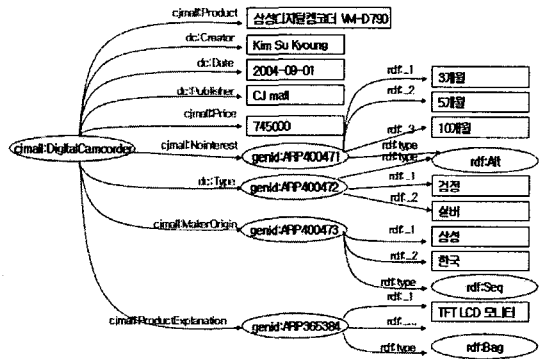
점을 극복하고자 한다.



<표 4> C사 RDF 엘리먼트

NameSpace(URI)	http://www.cjmall.com/cjmall/#				
Category	구분	Element	Value	prefix	Attribute
캠코더	Resource	DigitalCamcorder		rdf	
상품	Property	Product	삼성디지털캠코더 VM-D790	cjmall	
가격	Property	Price	745000	cjmall	
방문일	Property	Date	2004.9.1	dc	
무이자	Property	Nointerest	2개월부터10개월까지	cjmall	rdf:Alt
리월금	Property	Fund		cjmall	
제조사	Property	MakerOrigin	삼성, 한국	cjmall	rdf:Seq
원산지	Property	Origin	대한민국	cjmall	
색상및종류	Property	Type	결정/일반	dc	rdf:Alt
CJ카드리월	Property	Option	3%추가리월	cjmall	
상품설명	Property	ProductDescription	- 2.5" TFT LCD 모니터 외	cjmall	rdf:Bag
주요사항	Property	Spec	- 혁명 Mini-DV 수평형	cjmall	rdf:Bag

[그림 7] L상점의 RDF 데이터 모델



[그림 8] C상점의 RDF 데이터 모델

마지막으로 RDF 모듈에 의해 각 상점의 DCC RDF 문서가 [그림 9, 10]과 같이 생성 된다. 지금까지 생성된 RDF 데이터 모델부터 RDF 문서를 살펴보면 웹상에서 미리 정의되어 공유되는 메타데이터 어휘나 온톨로지를 이용하지 않을 경우 작성자의 임의적인 메타데이터 설계에 따라, 동일한 의미(semantic)를 가진 자원임에도 불구하고 다른 메타 데이터로 표현되는 문서로 작성되어 상품의 비교와 검색에 있어 상품 정보의 공유나 검색이 어려워지게 된다. 따라서 본 연구에서는 정의된 메타데이터가 서로 상이하다고 하더라도 기준이 되는 웹 온톨로지를 미리 정의하고, 이를 분석하고 비교하여 메타데이터 표현이 상이함에 따른 문제

<표 5> L상점의 RDF 트리플 데이터 모델

Subject	Predicate	Object
lgshop:CamCorder	http://www.lgshop.com/lgshop/ProductName	"삼성 디지털 캠코더 VM-D790"
lgshop:CamCorder	http://purl.org/dc/elements/1.0/identifier	"http://www.lgshop.com/jsp/jsep_PrdItem.jsp?cpid=1250034"
lgshop:CamCorder	http://purl.org/dc/elements/1.0/Creator	"Kim Su Kyoung"
lgshop:CamCorder	http://purl.org/dc/elements/1.0/Date	"2004-09-01"
lgshop:CamCorder	http://purl.org/dc/elements/1.0/Publisher	"LG e-Shop"
lgshop:CamCorder	http://www.lgshop.com/lgshop/SellPrice	"699000"
lgshop:CamCorder	http://purl.org/dc/elements/1.0/Title	"[04]년 디지털 캠코더 신제품"
genid:ARP1860374	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/1999/02/22-rdf-syntax-ns#Bag
genid:ARP1860374	http://www.w3.org/1999/02/22-rdf-syntax-ns#1	"3개월"
genid:ARP1860374	http://www.w3.org/1999/02/22-rdf-syntax-ns#2	"6개월"
genid:ARP1860374	http://www.w3.org/1999/02/22-rdf-syntax-ns#3	"9개월"



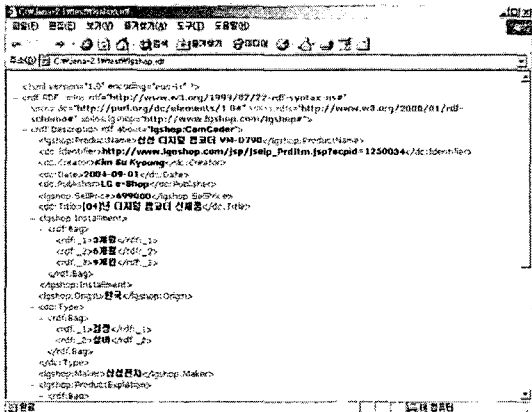
<표 6> C상점의 RDF 트리플 데이터 모델

Subject	Predicate	Object
cjmall:DigitalCamCorder	http://www.cjmall.com/cjmall#Product	*상성 디지털 캠코더 VM-D790*
cjmall:DigitalCamCorder	http://purl.org/dc/elements/1.0#Creator	"Kim Su Ryoung"
cjmall:DigitalCamCorder	http://purl.org/dc/elements/1.0#Date	"2004-09-01"
cjmall:DigitalCamCorder	http://purl.org/dc/elements/1.0#Publisher	"CJ Mall"
cjmall:DigitalCamCorder	http://www.cjmall.com/cjmall#Price	"745000"
genid:ARP1860642	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/1999/02/22-rdf-syntax-ns#Alt
genid:ARP1860642	http://www.w3.org/1999/02/22-rdf-syntax-ns#_1	"3개월"
genid:ARP1860642	http://www.w3.org/1999/02/22-rdf-syntax-ns#_2	"5개월"
genid:ARP1860642	http://www.w3.org/1999/02/22-rdf-syntax-ns#_3	"10개월"
cjmall:DigitalCamCorder	http://www.cjmall.com/cjmall#Nointerest	genid:ARP1860642

각 상점에서 동일한 상품 속성임에도 불구하고 서로 다른 속성으로 정의된 메타데이터를 [표 7]과 같이 다시 정의하였고 이를 기준으로 DCC 지식베이스 온톨로지의 클래스와 속성을 설계하였다.

<표 7> 두 RDF 문서에서 동일한 의미의 메타데이터

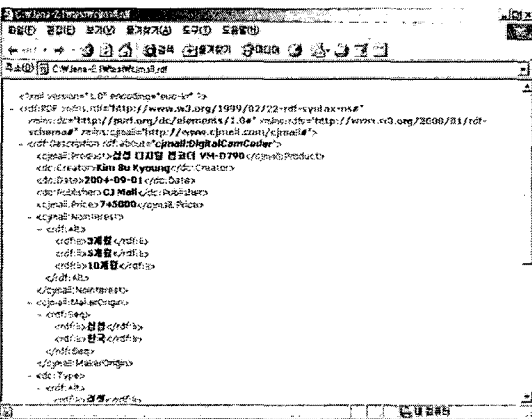
DCC 자원(Resource)에서 동일한 속성(Property)			
속성값	L사 속성	C사 속성	속성값
[04년 신제품]상성 디지털 캠코더	Title	없음	
LG eShop	Publisher	Publisher	CJ mall
상성 디지털 캠코더 VM-D790	ProductName	ProductName	상성 디지털 캠코더 VM-D790
699,000	SellPrice	Price	790,000
3개월	Installment	noInterest	6개월
0	Save	Fund	23,700원
상성	Maker	MakerOrigin	상성/한국
한국	Origin	없음	
경쟁/질비	Kind	ColorKind	경쟁/질비
최저가 신고 대상상품	Option	Option	배송5일이내
상품에 대한 규격	ProductExplanation	ProductDescription	상품에 대한 규격



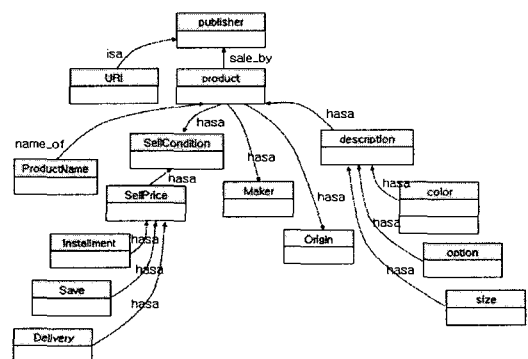
[그림 9] L상점의 상품(자원) RDF 문서

### 3.3 DCC 온톨로지

<표 7>을 기준으로 한 DCC 지식베이스 온톨로지 UML은 최상의 클래스로서 상품에 대한 비교 검색 후 최적의 조건을 가진 상품명을 저장하게 되는 publisher 클래스를 두고 그 아래 제품에 대한 product 클래스와 product 클래스의 속성과 값 들을 [그림 11]과 같이 설계하였다.



[그림 10] C상점의 상품(자원) RDF 문서



[그림 11] DCC 지식베이스 온톨로지 UML

[그림 11]의 UML을 기반으로 DCC에 대한 Textual ontology를 <리스트 1>과 같이 작성한다. 이 온톨로지는 지식 기반의 추론에 사용될 수 있는 것으로 각 클래스들 간의 관계(relation)와 속성간의 관계를 기술할 수 있기 때문에 클래스와 클래스간의 관계를 통해 다른 클래스와 속성 까지도 추론할 수 있으며 개념이라 할 수 있는 클래스를 더욱 명확하게 정의하게 된다.

<리스트 1> DCC Textual ontology 일부

```

1 (defclass
2   » (is-a·User)
3   » (role·concrete)
4   » (single-slot·sales_by
5     » (type·SYMBOL)
6   »+» (allowed-classes·Product)
7   »+» (cardinality·1·1)
8   » (create-accessor·read-write))
9   » (single-slot·part_of
10  » (type·SYMBOL)
11 »+» (allowed-parent·Publisher)
12 »+» (cardinality·0·1)
13 » (create-accessor·read-write))
14 » (single-slot·name_of
15   » (type·SYMBOL)
16 »+» (allowed-parents·Product)
17 » (cardinality·1·1)
18 » (create-accessor·read-write))
19 (defclass·URI
20   » (is-a·Publisher)
21   » (role·concrete)
22   » (single-slot·isa
23     » (type·SYMBOL)
24   »+» (allowed-parents·Publisher)
25   »+» (cardinality·0·1)
26   » (create-accessor·read-write))
27 (defclass·SellCondition
28   » (is-a·Product)
29   » (role·concrete)
    
```

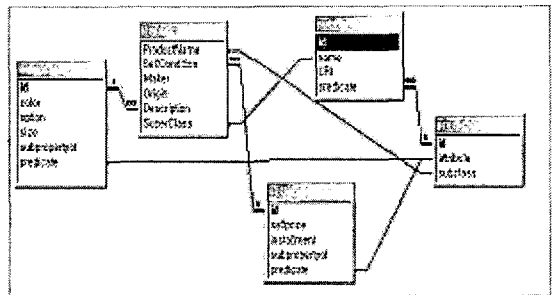
[그림 11]과 <리스트 1>을 웹 온톨로지 언어인 OWL을 이용하여 <리스트 2>와 같이 변환한다. 이 변환된 리스트는 두 가지 타입으로 DCCKBO에 저장이 된다. 하나는 <리스트 2>와 같은 OWL 문서이고 다른 하나는 [그림 11]의 설계를 기반으로 Jena 2.0과 Sesame+Bor 데이터베이스 스키마 [10] 형태를 갖는 [그림 12]의 데이터베이스 스키마를 갖는 데이터베이스 파일로 저장한다. 이 두 가지는 더 나은 조건의 상품 정보가 검색되어질 때 마다 동적으로 재정의 되어 가장 나은 조건의

상품을 검색하는데 주요 인자로 작용하게 된다.

<리스트 2> OWL로 정의된 DCC 지식베이스

```

<owl:Ontology rdf:about="">
  <rdf:comment>An example PRODUCT ontology</rdf:comment>
  <rdf:comment>This ontology is structure of PRODUCT</rdf:comment>
  <rdf:label>Product Ontology</rdf:label>
  </owl:Ontology>
  <owl:Class rdf:ID="Publisher" />
  <owl:Class rdf:ID="Product">
    <rdf:subClassOf rdf:resource="#Publisher" />
  </owl:Class>
  <owl:Class rdf:ID="digitalCamCoder">
    <rdf:subClassOf rdf:resource="#Product" />
  <rdf:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#productName" />
      <owl:cardinality 1</owl:cardinality>
      <owl:hasValue rdf:resource="#VM-D790" />
    </owl:Restriction>
  </rdf:subClassOf>
  <rdf:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#sellCondition" />
    </owl:Restriction>
  </rdf:subClassOf>
  <rdf:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#dccDescription" />
    </owl:Restriction>
  </rdf:subClassOf>
  <owl:Restriction>
    <owl:onProperty rdf:resource="#Mnker" />
    
```



[그림 12] DCC OWL 데이터베이스 스키마

### 3.4 상점별 RDF 문서의 검색 및 공유 방안

각 상점별로 작성된 RDF 문서는 [그림 4] 내부의 RDQL 모듈에 의해 검색 되고 공유된다. RDF 문서의 검색을 위해 RDQL을 사용하였다. RDF 문서에 있는 자원이나 속성은 <리스트 3>과 같은 RDQL 구문을 통하여 검색된다. 이렇게 검색된 결과는 저장된 뒤, Ontology 모듈로 검색 결과인 DCC 상품 자원, 속성, 값들을 전달하여 DCCKBO

와 각 상점별 상품에 대한 검색 조건의 데이터로 사용된다.

<리스트 3> RDQL 구문

- 상품의 Resource 검색  

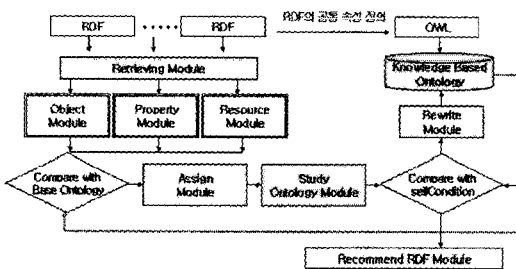
```
SELECT ?resource
WHERE (?resource lgshop:ProductName "삼성 디지털 캠코더 VM-D790")
USING lgshop For <http://www.lgshop.com/igshop/#>
```
- 상품의 가격 Object 검색  

```
SELECT ?resource ?SelfPrice
WHERE (?resource lgshop:ProductName "삼성 디지털 캠코더 VM-D790")
(?resource lgshop:ProductPrice ?z)
(?z lgshop:SelfPrice ?SelfPrice)
USING lgshop For <http://www.lgshop.com/igshop/#>
```
- 두 RDF 문서에서 가격 Object 검색  

```
SELECT ?resource ?SelfPrice ?Price
WHERE (?resource lgshop:SelfPrice ?SelfPrice)
(?resource cjmall:Price ?z), (?z, < cjmall:Price ?Price)
AND ?productName = "삼성 디지털 캠코더 VM-D790"
USING lgshop For <http://www.lgshop.com/igshop/#>
cjmall For <http://www.cjmall.com/cjmall/#>
```

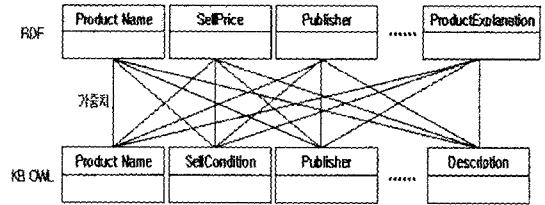
3.5 DCC 온톨로지 재정의와 에이전트 설계

마지막으로 본 논문이 제안하는 비교 구매 에이전트와 온톨로지의 동적 재정의 방법은 [그림 13]과 같다.



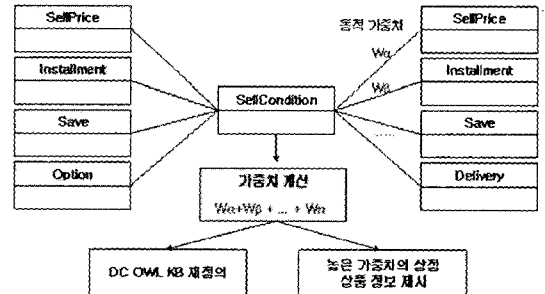
[그림 13] 온톨로지 동적 정의 흐름

먼저 각 상점의 RDF 문서에서 <리스트 3>의 구문을 통해 저장된 결과들은 “Retrieving Module”을 통해 각 자원과 속성 그리고 값에 해당하는 메타데이터로 분리하여 추출되고, 다시 “Value Module”과 “Property Module”, “Resource Module”를 통하여 RDF 문서가 더욱 정확한 메타데이터로 구분되어 분석된다.



[그림 14] Compare module

이 때 상품에 대한 정보인 자원, 속성, 값들을 중요도에 따라 가중치를 부여하고, [그림 14]에서 나타난 방법에 따라 RDF 문서에 생성되었던 메타 데이터 들과 DCCCKBO에 저장되어 있는 DCC OWL과 데이터베이스의 클래스와 속성 그리고 값을 가지고 각각 비교하게 된다. 비교한 결과 일치도가 높은 경우 Assign Module을 통하여 DCCCKBO의 클래스와 속성에 각 값을 할당하여 학습 온톨로지를 임시로 생성한다.



[그림 15] sellCondition을 이용한 비교 모듈

[그림 15] SellCondition을 이용한 비교 모듈에서 DCCCKBO의 “sellCondition” 속성을 이용하여 학습 온톨로지의 값을 가중치를 부여하면서 다시 비교한다. 가중치의 합을 구하여 계산된 결과 가중치가 높으면 “Rewrite Module”을 통하여 DCCCKBO의 클래스와 속성 그리고 값을 재 정의하여 다시 DCC 온톨로지 저장소에 저장한다. 이를 통해 상

품의 판매 정보에 있어서, 가장 좋은 조건을 가진 상점의 상품 정보가 유지될 수 있다. 그 다음 “Recommend RDF Module”을 통해 가장 좋은 조건의 상품 정보를 구매자에게 알려준다.

#### 4. 결론 및 향후 연구 방향

본 논문에서는 현재 인터넷 상점의 HTML 페이지 정보를 래퍼 기술을 이용해 생성한 RDF 문서와 DCC 지식 기반 온톨로지를 이용하여 인터넷 상점들의 상품 정보를 비교 검색하고 좋은 조건의 상품을 구매자에게 제시한 뒤, 동적으로 상품에 대한 지식베이스 온톨로지를 재 정의하는 시스템을 제안하였다. 물론 현재의 인터넷 상점의 정보 환경은 RDF 문서로 이루어져 있지 않기 때문에 제안 시스템이 적용되기에는 HTML에 대한 래퍼 기술과 웹 도메인 온톨로지 구축 등에 대한 더 자세하고 다양한 연구가 이루어져야 할 것이다. 그러나 현재 전자상거래 환경에서 비교 구매 에이전트가 갖는 문제점 중의 하나인 실시간 비교 검색의 어려움이나 거부문제를 본 시스템의 제안으로 해소할 수 있다. 또한 본 논문은 시맨틱 웹 환경을 이용한 전자상거래의 활용 방안으로 의미가 있다. 앞으로 이 연구를 바탕으로 기존 웹 환경인 HTML에서 RDF 문서나 웹 온톨로지로 자동 변경될 수 있는 래퍼 변환기와 더욱 많은 상품 정보에 대한 온톨로지를 구축함으로써 많은 인터넷 상점들에 제시되는 상품 정보를 실시간으로 비교하고 검색할 수 있는 전자상거래와 시맨틱 웹의 접목을 위한 연구가 필요할 것이다.

#### 참고문헌

- [1] 김학래, 김홍기, “시맨틱 웹/온톨로지 기술을 이용한 개인용 문서관리 시스템”, 지능정보학회, 2004 춘계 학술대회,
- [2] 신효필, “지식기반으로서의 온톨로지와 시맨틱 웹”, 정보처리학회지 제11권 제2호, 2004
- [3] 양재영, 최중민, 김중배, “비교쇼핑에이전트 시스템”, HCI 2000 학술대회, 2000
- [4] 오삼균, “Web Ontology Language와 그 활용에 관한 고찰”, 데이터베이스 연구, 18권3호, 2002
- [5] 이정전, “전자상거래 소프트웨어 에이전트”, 한국정보처리학회지 제6권 제1호(1999) 54~62.
- [6] 이재호, “시맨틱 웹의 온톨로지 언어”, 한국정보과학회지, 제21권, 제3호, 2003
- [7] 이호경, 유영훈, 조근식, “시맨틱 웹 기반의 비교 구매 시스템”, 한국정보처리학회 추계 학술발표대회 논문집, 제10권 제2호(2003), 1695~1698.
- [8] 정재목, 김형주, “웹 정보의 추출 및 통합을 위한 래퍼 시스템”, 한국정보처리 학회 논문지, Vol. 09, No. 05(2003), 551~559.
- [9] 최호섭, 옥철영, “정보검색 시스템과 온톨로지”, 정보과학회지 제22권 제4호, 2004
- [10] 탁경현, 김학수, 차현석, 손진현 “효율적인 OWL 시맨틱 정보처리를 위한 데이터베이스 스키마 설계 및 분석”, 데이터베이스연구회(2003)
- [11] John Davies, Dieter Fensel, Frank Van Harmelen, *Towards The Semantic Web*, 2002
- [12] Michael C. Daconta, *The Semantic Web*, Wiley Publishing, 2002,
- [13] Natalya F.Noy, Deborah L. McGuinness, *Ontology Development 101:A Guide to*

- Creating Your First Ontology*, Stanford University, 2002.
- [14] Shelley Powers, *Practical RDF*, O'REILLY, 2002.
- [15] Smith, M, D.McGuinness, R. Volz, C. Welty, "Web Ontology Language (OWL) Guide" <http://www.w3.org/owl-guide>. 2004.
- [16] *Wine Ontology*, <http://www.w3.org/TR/2002/WD-owl-guide-20031218/wine.owl>
- [17] W3C, "Web-Ontology (WebOnt) WG," <http://www.w3c.org/2001/sw/WebOnt/>



Abstract

## A Study of Dynamic Web Ontology for Comparison-shopping Agent based on Semantic Web

Kim, Su Kyoung\* · Ahn, Ki Hong\*

In this paper, convert in RDF triple and a RDF document through RDF document converters and design metadata schema about a digital camcorder after use Wrapper technology, and acquiring commodity information of a HTML page about the digital camcorder which these papers are defined so as to be different by electronic commerce stores, and is expressed. Save in digital camcorder domain ontology storage that implemented to relational database to DCC knowledge base ontology as convert to OWL Web ontology based on designed metadata schema. Through compare with rdf and DCCKBO, mapping, and inference process, provide to buyers by DCC information of the store that had the commodity purchasing information which is the best, and proposed a dynamic Web ontology guessed to contents of the best commodity purchasing information, and to define domain ontology saved in DCCKBO.

**Key words** : 전자상거래, 웹 온톨로지, 시맨틱 웹, 비교구매 에이전트

---

\* Department of Computer Engineering, Hanbat National University

