멀티데이타베이스 시스템에서 정보공유를 위한 개념-기반 의미망의 구축

(A Concept-based Semantic Network for Information Sharing in Multidatabase Systems)

이 정 욱 † 백 두 권 ††
(JeongOog Lee) (DooKwon Baik)

요 약 멀티데이타베이스 시스템(multidatabase system)에서 여러 요소 데이타베이스(component database)에 대한 통합된 접근을 제공하기 위해서는 의미 이질성(semantic heterogeneity)이 탐색되고 해결되어져야 한다. 즉, 멀티데이타베이스 시스템은 각 요소 데이타베이스가 가지고 있는 정보의 의미를 이해하고 의미적으로 동등한 또는 유사한 정보들을 식별할 수 있어야 한다. 또한, 멀티데이타베이스 시스템은 사용자로 하여금 실세계의 동일한 정보를 가지고 있는 여러 다른 데이타베이스로부터 원하는 정보를 용이하게 획득할 수 있도록 해야 한다.

본 논문에서는, 요소 데이타베이스간의 의미 이질성을 탐색하고 해결하기 위하여 정보가 가지고 있는 개념간 의미관계에 기반한 의미망(semantic network)을 구축한다. 또한 의미질의어(semantic query language)를 제공하여 사용자가 스키마에 대한 사전 지식이 없이도 여러 자율적인 데이타베이스로부터 원하는 정보를 용이하게 획득할 수 있도록 한다.

Abstract A multidatabase system provides integrated access to heterogeneous, autonomous component databases in a distributed system. In order to gain integrated access to a multidatabase system, semantic heterogeneities have to be detected and resolved. That is, the multidatabase system must interpret and integrate the meaning of the information and identify semantically equivalent or related objects. Another problem in multidatabase systems is allowing users to handle information from different databases that refer to the same real-world entity.

In this paper, we provide semantic networks so that multidatabase systems can detect and resolve semantic heterogeneities among component databases. And we provide a semantic query language, SemQL, to capture the concepts about what users want. It enables users to issue queries to a large number of autonomous databases without prior knowledge of their schemas.

1. 서 론

기존의 데이타베이스 시스템에서 정보는 특정한 요구사항에 따라 독립적으로 생성, 저장 그리고 사용되어져 왔다. 정보의 양이 빠르게 증가함에 따라, 많은 정보를 하나의 데이타베이스 시스템에서 유지관리하는 것은 매우 어려운 일이 되었다. 또한, 웹(web)과 같은 거대한 지식베이스에서는 많은 질의들이 단일 정보소스가 아닌 통합된 정보로부터 해결되어지는 경우가 상당히 많다. 그러므로, 분산된 데이타베이스를 통합하는 필요성은 증가하고 있다. 멀티데이타베이스 시스템(multidatabase system)은 분산된 환경에서 이질적이고 자율적인 요소 데이타베이스(component database; CDB)들에 대한 통합된 접근을 제공해야 한다. 멀티데이타베이스 시스템에서 상호운용성을 달성하기 위한 가장 기본적이고 중요한 것은 요소 데이타베이스들[1]이 가지고 있는 정보에 대하여 의미적으로 동등하거나 유사한 데이타 항목들을 식별하는 것이다[1].

† 비 회 원 : 고려대학교 컴퓨터학과
ljo@swsys2.korea.ac.kr
†† 종신회원 : 고려대학교 컴퓨터학과 교수
baik@swsys2.korea.ac.kr
논문접수 : 2000년 11월 20일
심사완료 : 2001년 4월 17일

1) 본 논문에서는 요소 데이타베이스(component database)와 정보소스(information source)를 혼용하여 사용한다.

멀티데이타베이스 시스템에서의 또 다른 문제는 사용자로 하여금 여러 다양한 요소데이타베이스로부터 동일한 실세계 정보를 가리키는 정보들을 다룰 수 있도록 하는 것이다. 즉, 멀티데이타베이스 시스템은 사용자가 질의에 적합한 결과를 어떻게 획득할 것인지 는 문제보다는 찾고자 하는 정보를 어떻게 명세할 것인지에만 초점을 맞추도록 해야 한다. 그럼으로써, 사용자는 찾고자 하는 정보에 관련된 요소 데이타베이스를 식별하고, 식별된 요소 데이타베이스에서 특정 인터페이스를 통해 질의를 수행하고, 여러 데이타베이스로부터의 결과를 취합하는 방대한 작업으로부터 해방될 수 있다[2].

본 논문에서는, 요소 데이타베이스간의 의미 이질성(semantic heterogeneity)을 탐색하고 해결하기 위하여 정보가 가지고 있는 개념(concept)과 스키마간의 관계를 표현하는 의미망(semantic network; SN)을 구축한다. 또한 의미질의어(semantic query language; SemQL)를 제공하여 사용자가 스키마에 대한 사전 지식이 없이도 여러 자율적인 데이타베이스로부터 원하는 정보를 용이하게 획득할 수 있도록 한다. 본 논문에서는 의미망을 구축하고 의미질의를 처리하기 위하여 WordNet을 이용한다. WordNet은 온-라인 어휘 사전이며 synonymy, antonymy, hyponymy, 그리고 meronymy와 같은 의미관계(semantic relation)로 조직되어 있다. WordNet 의 명사부분은 어휘의 의미가 매우 연관된 동의어 집합으로 구성되어 있다[3].

본 논문의 구성은 다음과 같다. 2장에서는 정보통합을 위해 해결되어야 하는 의미 이질성의 분류 등 기본개념에 대하여 설명한다. 3장에서는 각 요소 데이타베이스에 대한 의미망의 구축 방법을 소개하고, 4장에서는 의미 이질성의 탐색과정과 전역 의미망(global semantic network; GSN)의 역할을 설명한다. 의미질의어와 의미질의 처리과정은 5장에서 보여주며, 6장에서는 본 논문에서 제안한 의미 이질성 해결 방법에 대한 실험과 그 결과를 보여준다. 7장에서는 관련 연구에 관하여 논의하고, 8장에서 요약 및 결론을 맺는다.

2. 기본 개념

이 장에서는 정보통합을 위하여 탐색되고 해결되어야 할 의미 이질성에 대하여 2.1 절에서 설명하고 본 논문의 접근방법에서 지식베이스로 사용하는 WordNet 에 대하여 2.2절에서 간략히 소개한다. 2.3 절에서는 정보공유를 위한 고려사항을 알아보고 멀티데이타베이스 시스템에서의 효과적이고 효율적인 정보공유를 위한 본 논문의 접근방법을 2.4절에서 소개한다.

2.1 의미 이질성의 분류

다수의 데이타베이스 시스템이 효율적으로 상호 연동되도록 하기 위해서는 많은 해결해야할 문제들이 있다. 가장 기본적인 문제는 이질성(heterogeneity)이다[4] [5]. 이질성은 하드웨어, 운영체제, DBMS 와 같은 플랫폼 이질성(platform heterogeneity)과 의미 이질성(semantic heterogeneity)으로 구분된다[4]. 의미 이질성은 동일한 실세계 정보가 데이타베이스 스키마로 모델링되는 과정에서 여러 데이타베이스에서 다르게 표현되는 것이다[6]. 그림 1은 의미 이질성을 보여주기 위한 예제이다. 플랫폼 이질성에 대해서는 중요한 많은 진전이 있었지만, 의미 이질성에 대한 해결책은 아직 어려운 문제로 남아 있다[5].

데이타베이스는 스키마와 데이타에 의해 정의되기 때문에, 의미 이질성은 스키마 이질성과 데이타 이질성으로 분류될 수 있다[7]. 스키마 이질성(schema hetero-

Component Database 1 (CDB_1)

Undergraduate (sid, name, sex, address, advisor#)
Graduate (sid, name, sex, address, advisor#)
FullProfessor (pid, name, sex, office)
AssociateProfessor (pid, name, sex, office)
AssistantProfessor (pid, name, sex, office)

Component Database 2 (CDB_2)

Student (sid, nm, gender, advisor#)
Address (sid, street, city, state)
Professor (pid, nm, gender, position, salary, office)

Component Database 3 (CDB_3)

FemaleStudent (sid, name, street, city, state, advisor#)
MaleStudent (sid, name, street, city, state, advisor#)
FemaleProfessor (pid, name, salary, office)
MaleProfessor (pid, name, salary, office)

Component Database 4 (CDB_4)

Student (pid, nm, female, male, advisor#)
Faculty (fid, nm, office)

그림 1 예제 데이타베이스 스키마

geneity)은 주로 동일한 정보에 대해 다른 구조를 사용하고 동일한 구조에 대해서 서로 다른 개체명(entity name)이나 속성명(attribute name)을 사용함으로서 야기된다. 예를 들면, 그림 1에서, 학생에 대한 정보가 CDB_2에서는 Student 하나의 테이블에서 표현되고 CDB_1에서는 Undergraduate 와 Graduate 두 개의 테이블로 표현된다. 성(性)에 대한 속성명도 CDB_1에서는 sex를 CDB_2에서는 gender를 사용한다. 데이타 이질성(data heterogeneity)은 스키마 이질성이 해결된 상태에서 데이타 포맷의 불일치로 야기된다.

본 논문에서 해결하려고 하는 주된 이질성은 스키마 이질성이므로 동일한 데이타에 대해 다른 포맷을 가짐으로서 야기되는 데이타 이질성은 없는 것으로 가정한다. 스키마 이질성에 초점을 맞추어, 본 논문에서는 의미 충돌(semantic conflict)의 형태를 다음과 같이 7가지로 정의한다[8][9][10].

- 개체간 구조 충돌(EESC: Entity versus Entity Structure Conflict)
  이러한 충돌은 서로 다른 요소 데이타베이스가 동일한 정보를 표현하는데 있어 서로 다른 개수의 개체(또는 테이블)를 사용할 때 일어난다.
- 개체와 속성간 구조 충돌(EASC: Entity versus Attribute Structure Conflict)
  이러한 형태의 충돌은 어떤 데이타베이스의 속성이 다른 데이타베이스에서는 개체로 표현될 때 일어난다.
- 개체와 값간 구조 충돌(EVSC: Entity versus Value Structure Conflict)
  이 충돌은 어떤 데이타베이스의 속성값(attribute value)이 다른 요소 데이타베이스에서 개체와 의미적으로 연관될 때 일어난다. 예를 들어, CDB_2의 gender 속성값은 male 또는 female 인데 이 값은 CDB_3의 Female Student 테이블과 MaleStudent 테이블 이름속에 그 의미가 내포되어 있다.
- 속성간 구조 충돌(AASC: Attribute versus Attribute Structure Conflict)
  이 충돌은 서로 다른 요소 데이타베이스가 동일한 정보를 표현하는데 있어 서로 다른 개수의 속성을 사용할 때 일어난다.
- 속성과 값간 구조 충돌(AVSC: Attribute versus Value Structure Conflict)
  이러한 형태의 충돌은 어떤 데이타베이스의 속성값이 다른 데이타베이스에서 속성과 의미적으로 연관될 때 일어난다.
- 개체간 이름 충돌(EENC: Entity versus Entity Name Conflict)
  이 충돌은 서로 다른 데이타베이스에서 동일한 정보를 표현하는 개체에 대해 다른 이름을 사용할 때 발생한다.
- 속성간 이름 충돌(AANC: Attribute versus Attribute Name Conflict)
  이 충돌은 EENC와 유사하게 동일한 정보를 표현하는 속성에 대해 다른 이름을 사용할 때 발생한다.

2.2 WordNet

일반적으로 단어는 발음과 개념을 동시에 나타내므로, 모호성을 줄이기 위하여, "단어 형태"는 물리적 발음을 나타내고 "단어뜻"은 어휘개념을 나타내는데 사용한다고 하였을 때, 어휘의 의미 파악은 형태와 뜻 사이의 사상(mapping)으로부터 출발한다[3].

WordNet은 영어 어휘 지식을 모델링하기 위하여 시도된 프린스턴 대학의 연구 프로젝트의 산물이다. 이 시스템은 온라인 시소러스와 온라인 사전의 기능과 더불어 그 이상의 기능을 가지고 있다. WordNet 은 synset이라고 불리우는 논리적 그룹으로 구성된다. 각 synset은 같은 뜻의 단어형태와 현재 synset과 다른 synset들간의 관계성을 가리키는 의미 포인터(semantic pointer)로 구성된다. 의미 포인터는 synonymy, antonymy, hyponymy, 그리고 meronymy를 포함하여 여러 형태가 있다.

단어형태와 단어뜻간의 사상은 many:many 가 될 수 있다. 어떤 단어형태는 여러 다른 의미를 가지고, 어떤 단어뜻은 여러 다양한 단어형태로 표현될 수 있다. 전자를 polysemy라 하고 후자를 synonymy라고 하는데, 멀티데이타베이스 시스템에서 정보를 접근하는 과정에서 야기되는 문제들이다.

customer와 client는 동일한 의미를 가지는 synonymy의 예이고, 반면, client는 많은 다른 뜻을 가지고 있는데 이는 polysemy의 예이다. 그림 2는 client에 대한 여러 다른 의미를 보여준다.

The noun client has 3 senses (first 2 from tagged texts)

1. client -- (a person who seeks the advice of a lawyer)
2. customer, client -- (someone who pays for goods or services)
3. node, client, guest -- ((computer science) any computer that is hooked up to a computer network)

그림 2 'client'에 대한 여러 다른 단어의미

2.3 정보공유를 위한 고려사항

웹과 같은 동적이고 개방된 환경에서는, 많은 수의 정

보소스가 존재하고 새로운 정보소스가 정해진 규칙없이 자동적으로 그리고 끊임없이 생성된다. 멀티데이타베이스 시스템이 그러한 환경에 적응하기 위해서는 정보가 공유되고 교환되도록 하는 메카니즘이 필요하다. 이는 각 정보소스가 다른 정보 소스가 의도하는 것과 정확히 같은 방법으로 정보를 이해할 수 있어야 한다. 또한 사용자가 원하는 정보를 보유한 정보소스가 쉽게 찾아져야 되고 그 정보소스로부터 정보가 쉽게 검색되어져야 한다. 이는 정보의 의미와 표현방법이 정보소스들간에 알려져 있고 또 일치되어야 가능하다. 멀티데이타베이스 환경에서 정보가 효율적이고 효과적으로 공유되기 위해서 필요한 고려사항을 정리하면 다음과 같다[9].

- 각 요소 데이타베이스에서의 정보의 의미는 통합된 방법으로 표현되어져야 한다(의미 표현: semantic representation).
- 멀티데이타베이스 시스템은 각 요소 데이타베이스에 있는 정보의 의미를 이해할 수 있어야 한다(의미 해석: semantic interpretation).
- 멀티데이타베이스 시스템은 모든 요소 데이타베이스에 있는 동등한 또는 유사한 정보를 통합할 수 있어야 한다(정보 통합: information integration).
- 통합된 정보로부터 원하는 정보를 검색하기 위한 효율적이고 효과적인 접근 메카니즘이 제공되어야 한다(정보 접근: information access).

2.4 정보공유를 위한 본 논문의 접근방법 개요

그림 3은 멀티데이타베이스 시스템에서의 상호운영성을 위한 본 논문의 접근방법을 개략적으로 보여주고 있다. 본 논문에서는 각 요소 데이타베이스가 가지고 있는 정보의 통일된 의미 표현과 정확한 의미 해석 그리고 정보통합과 효과적이고 효율적인 정보접근을 위한 지식베이스로서 WordNet을 활용한다. 우선, 각 요소 데이타베이스에서 스키마에 대한 설명(description)을 WordNet의 개념들을 이용하여 표현하고 이로부터 의미망을

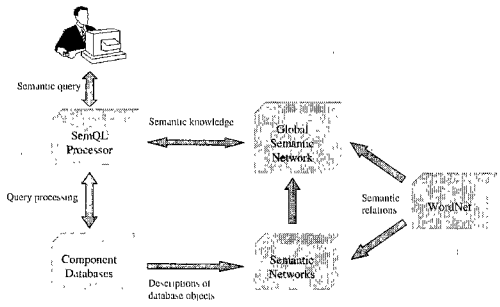


그림 3 정보공유를 위한 본 논문의 접근방법 개요

구축한다. 의미망(semantic network: SN)은 WordNet의 개념(concept)들과 데이타베이스 스키마 정보간의 사상(mapping)을 제공한다[8]. 다음에, 이러한 개별적인 의미망들을 이용하여 의미이질성을 탐색하고(의미해석) 정보통합을 위한 전역 의미망을 생성한다. 전역 의미망(global semantic network; GSN)은 분산된 환경에 대한 여러 가지 의미 지식(semantic knowledge)을 제공한다. 이러한 전역 의미망을 통하여 사용자에게 효과적이고 효율적인 정보접근 방법을 제공하며 사용자의 의미질의(semantic query)를 처리하게 된다.

3. 의미망(Semantic Network)

여러 요소 데이타베이스로부터 정보를 통합하기 위해서, 멀티데이타베이스 시스템은 각 요소 데이타베이스에 있는 정보의 의미를 이해해야 하고, 각 요소 데이타베이스에서는 통일된 방법으로 정보를 표현해야 한다. 이를 위하여, 요소 데이타베이스 관리자는 자신의 데이타베이스 스키마에 대한 서술(description)을 작성한다. 작성된 서술과 WordNet의 개념을 이용하여 표현 테이블(representation table)을 작성한다. 그리고 나서, 표현 테이블과 WordNet 의 의미관계에 따라 요소 데이타베이스에 대한 의미망을 생성한다. 각 요소 데이타베이스를 위한 모든 의미망은 멀티데이타베이스 시스템을 위한 전역 의미망으로 통합되어 진다.

3.1 스키마 정보의 서술

다음은 ISO/IEC 11179를 참조하여[11], 데이타베이스에 있는 스키마 정보의 서술을 작성하기 위하여 준수해야 하는 본 논문에서 규정한 일반적인 규칙에 대한 설명이다. 서술은 구문 구조와 의미 규정에 의해 작성되어진다. 의미 규정(semantic rule)은 서술에 포함되는 어휘의 내용에 관한 약속이며 의미가 전달될 수 있도록 한다. 구문 구조(syntactic principle)는 서술내의 어휘간의 정렬을 명세한다.

3.1.1 의미 규정

의미는 서술에 포함된 구성 요소(어휘)의 뜻에 관여된다. 구성 단어는 개체어, 속성어, 한정어이다. 개체어(entity term)는 실세계의 행위나 개념, 객체를 표현하는 서술의 구성 요소이다. 예를 들면, Student Last Name 이라는 서술에서, 구성 요소인 Student는 개체어이다. 속성어(attribute term)는 개체의 속성을 나타내는 서술의 구성 요소이다. 예를 들면, Student Last Name에서 Name 이 속성어이다. 한정어(qualifier term)는 서술을 유일하게 식별하기 위하여 개체어와 속성어에 첨가되어지는 구성 요소이다. Student Last Name에서

Last 는 Name을 한정하는 한정어이다.

3.1.2 구문 구조

개체어는 서술에서 가장 처음에 나타나고 속성어는 서술내에서 마지막 위치를 차지한다. 한정어는 한정하는 용어 앞에 위치한다. 축약은 허용되지 않는다. 예를 들면, Student ID는 허용되지 않고 Student Identification Number 와 같이 작성한다. 게다가, 모든 서술은 한 요소 데이타베이스내에서는 유일해야 한다.

3.1.3 데이타 타입

모든 속성값(attribute value)은 여러 데이타 타입중 하나에 속하게 된다. 다양한 데이타베이스 시스템이 서로 다른 데이타 타입을 지원할지라도, 대부분의 시스템은 최소한 숫자, 문자, 그리고 불리언 데이타 타입을 제공한다. 어떤 데이타베이스 시스템은 또한 속성을 나타내기 위하여 코드(code)를 사용할 수도 있다. 예를 들면, 그림 2에서 CDB_2는 position 속성에 대해, full professor는 1, associate professor는 2, 그리고 assis- tant professor는 3이라는 코드값으로 표현되어진다. 그러한 경우에는, 각 코드의 의미에 대한 서술도 작성해야 한다.

3.2 의미망의 생성

3.1절의 규칙에 의해 서술이 작성된 후에, 서술은 단위용어로 분해되고 표현 테이블이 작성된다. 단위용어(unit term)는 WordNet에서 표현되어 있는 한 단어나 구를 의미한다. 예를 들면, 복합명사 'phone number'는 WordNet에 표현되어 있으므로, 'phone number'는 단위용어로서 취급된다. 이러한 분해 과정의 산물이 표현 테이블이다. 표현 테이블(representation table)은 객체타입(object type), 객체명(object name), 데이타 타입(data type), 서술(description), 그리고 단위용어 집합(a set of unit terms) 등으로 구성된다. 그림 4는 표현 테이블의 한 예이다. 본 논문에서 객체는 개별적으로 취급되는 개체(entity), 속성(attribute), 그리고 값(value)을 가리킨다. 객체타입에서 E는 개체를, A는 속성을 의미한다. 그리고 데이타 타입에서, n은 숫자(numeric), s는 문자열(string), b는 불리언(boolean), c는 코드 스킴(code scheme)을 각각 가리킨다.

Rec #	Object_Type	Object_Name	Data_Type	Description	Unit_Terms
1	E	Professor		professor	<professor>
2	A	pid	n	professor identification number	<professor> <identification number>
3	A	nm	s	professor name	<professor> <name>
4	A	*gender*	*b*	*professor gender* /M=male I F=female	<*professor*> <*gender*> /<male> I <female>
5	A	position	c	professor position /1=full professor I 2=associate professor I 3=assistant professor	<professor> <position> /<full professor> I <associate professor> I <assistant professor>
6	A	salary	n	professor salary	<professor> <salary>
7	A	office	n	office room number	<office> <room> <number>

그림 4 표현 테이블(representation table)의 예

표현 테이블이 작성되는 동안, 요소 데이타베이스 관리자는 synonymy와 polysemy 문제를 해결해야 한다. synonymy 문제는 WordNet의 synset을 이용하여 자동으로 식별되어진다. 그러나, 단위용어의 정확한 뜻을 획득하기 위해서는, 관리자가 수동으로 polysemy 문제를 다루어야 한다. 예를 들면, 요소 데이타베이스 관리자가 'client'에 대해 디스플레이된 여러 뜻 중에서 하나를 수작업으로 선택해야 한다.

한 요소 데이타베이스의 단위용어들이 추출된후, 각 단위용어들은 WordNet의 개념들과 연결된다. 이 과정의 산물이 의미망(semantic network)이다. 그림 5는 그림 1의 CDB_2에 대한 의미망을 보여준다. 의미망은 Word Net의 개념들과 요소 데이타베이스 스키마간의 사상을 제공한다.

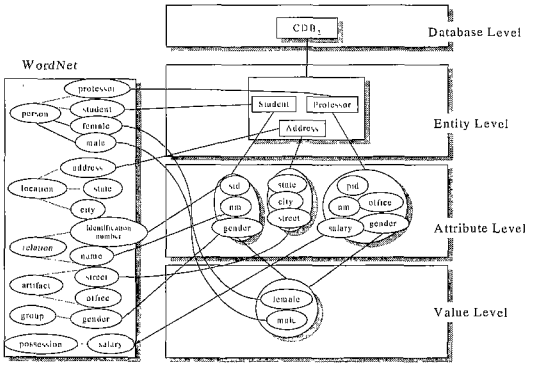


그림 5 CDB_2에 대한 의미망의 일부

4. 의미 이질성 탐색과 전역 의미망

멀티데이타베이스 시스템은 정보의 뜻을 이해하고 의미적으로 동등하거나 또는 유사한 객체를 식별할 수 있어야 한다. 각 요소 데이타베이스에 대한 의미망이 생성되면, 이들은 전역 의미망으로 통합되어진다. 의미망들을 통합하는 과정에서, 멀티데이타베이스 시스템은 의미망들과 WordNet의 의미관계를 이용하여 의미 이질성을 탐색할 수 있다[8] [9].

4.1 의미망과 WordNet을 이용한 의미 이질성 탐색

다음은 각 요소 데이타베이스에 대한 의미망들과 WordNet의 의미관계를 이용하여 어떻게 의미 이질성을 탐색할 수 있는지를 예를 통하여 보여준다. 예제들은 그림 1의 스키마를 이용하여 설명되어진다. 의미 이질성 탐색 과정의 결과들은 정보 검색에서 의미 이질성을 해

결하는데에 사용되어진다.

EESC의 탐색: WordNet의 hyponymy 의미관계에 의해, professor는 full professor, associate professor, 그리고 assistant professor 들의 상위개념(hypernym)임을 알수 있다(그림 6). 그러므로, CDB_2의 Professor 개체는 CDB_1의 개체집합 {FullProfessor, Associate Professor, AssistantProfessor}과 의미적으로 동등하다.

EASC의 탐색: 그림 6에서, CDB_1의 속성 address와 CDB_2의 개체 Address 모두 WordNet 의 address라는 개념과 연결되어 있다. 그러므로, CDB_1의 속성 address는 CDB_2의 개체 Address와 의미적으로 관련이 있다.

EVSC의 탐색: CDB_2는 속성 position에 대해 코드를 사용한다(full professor=1, associate professor=2, assistant professor=3). 이러한 코드값들은 CDB_1의 개체들과 각각 연결되어 있다(그림 6). 그러므로, CDB_2의 속성 position의 코드값들은 CDB_1의 개체 Full Professor, AssociateProfessor, 그리고 Assistant Professor 들과 의미적으로 관련이 있다.

AASC의 탐색: 위에서 CDB_2의 개체 Address는 CDB_1의 속성 address와 의미적으로 동등하다고 설명하였다. 그리고 CDB_2의 개체 Address의 모든 속성들은 CDB_3의 개체 Student의 속성집합 {state,city,street}과 의미적으로 관련이 있다(그림 7). 그러므로, CDB_1의 속성 address는 CDB_3의 속성집합 {state,city,street}와 의미적으로 관련이 있다.

AANC의 탐색: 그림 7에서와 같이, WordNet에서 sex와 gender는 동의어 관계에 있으므로, CDB_1의 속성 sex와 CDB_2의 속성 gender의 의미는 같다고 해석할 수 있다.

AVSC의 탐색: CDB_2는 속성 gender에 대하여, female은 F, male은 M이라는 코드값을 사용한다. 이

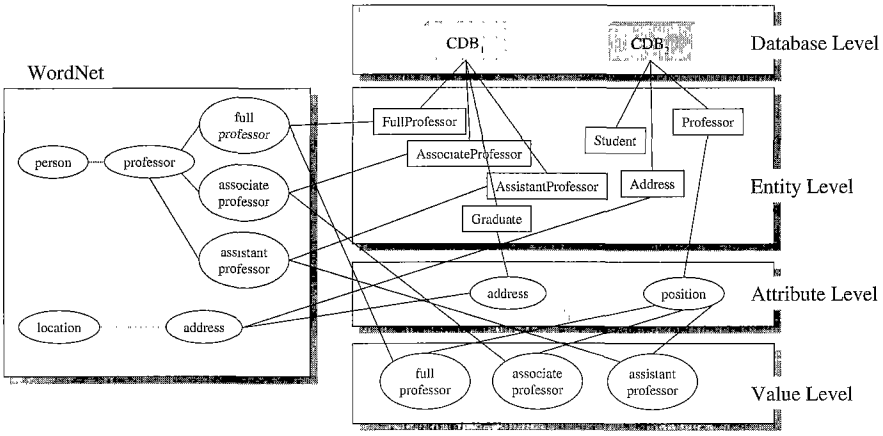


그림 6 두 요소 데이타베이스 CDB_1과 CDB_2의 통합과정의 부분상태

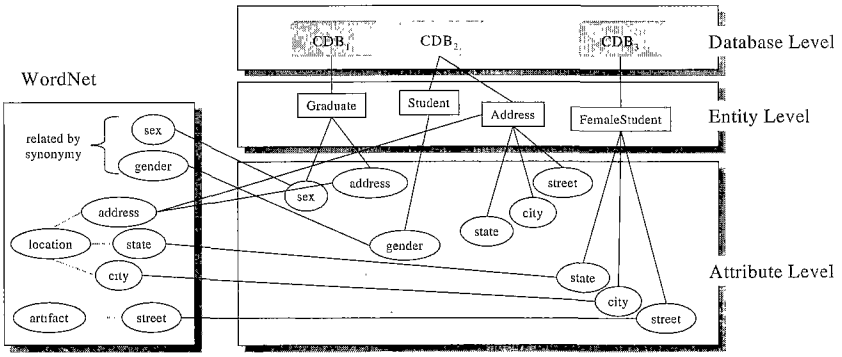


그림 7 세 요소 데이타베이스, CDB_1, CDB_2, 그리고 CDB_3의 통합과정의 부분상태

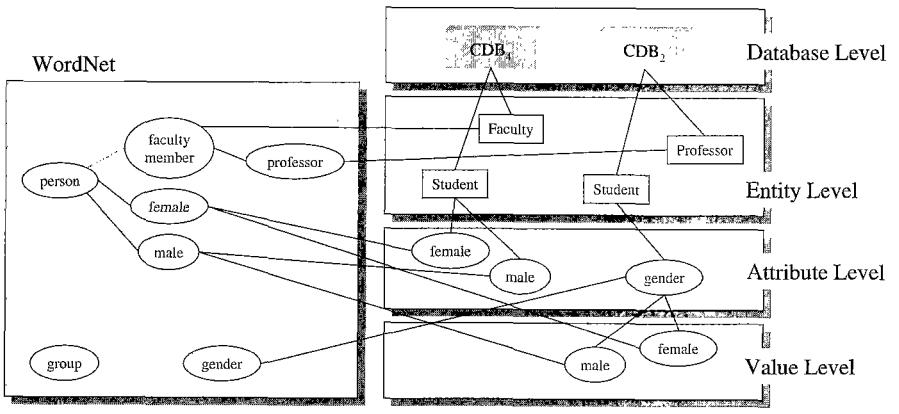


그림 8 두 요소 데이타베이스, CDB_2와 CDB_3의 통합과정의 부분상태

코드값들은 CDB_4의 속성 female, male 과 각각 연결되어 있다(그림 8). 그러므로, CDB_2의 속성 gender의 코드값들은 CDB_4의 두 속성 female, male과 의미적으로 동등하다.

EENC의 탐색: WordNet의 hyponymy 의미관계에 따라, faculty member는 professor의 상위개념(hypernym)임을 알 수 있다(그림 8). 그러므로, CDB_2의 개체 Professor는 CDB_4의 개체 Faculty와 의미적으로 연관이 있다.

이와 같이 개별 의미망들을 모두 통합하면 전역 의미망이 생성된다. 전역 의미망은 요소 데이타베이스들에 대한 통합된 접근에 필요한 지식들을 멀티데이타베이스 시스템에 제공한다.

4.2 전역 의미망의 역할

멀티데이타베이스 시스템은 여러 요소 데이타베이스에 대해 단일의 통합된 인터베이스를 제공해야 한다. 다음과 같은 질의에 응답하기 위하여 멀티데이타베이스 시스템이 필요로 하는 지식에 대해 고려해보자.

"Find those professors whose salary is over $50,000."

이와 같은 질의에 응답하기 위해서, 멀티데이타베이스 시스템은 다음과 같은 지식들을 가지고 있어야 한다.

- 멀티데이타베이스 시스템은 질의에 연관되는 정보를 가지고 있는 요소 데이타베이스들을 식별할 수 있어야 한다(접근 지식: access knowledge)
- 멀티데이타베이스 시스템은 각 요소 데이타베이스의 어떤 개체, 속성, 또는 속성값이 질의에 내포된 의미를 만족하는지를 식별할 수 있어야 한다(의미 지식: semantic knowledge)

이러한 지식들을 획득하기 위하여, 요소 데이타베이스의 개체들, 속성들, 값영역 간의 의미 관계를 명세하는 전역 의미망을 개발하였다. 전역 의미망을 통하여 의미 이질성이 해결되고 사용자의 의미 질의를 처리할 수 있다.

전역 의미망에 관련된 중요한 문제중의 하나는 각 정보소스가 어떻게 접근 지식과 의미 지식을 획득하고 유지하는가 하는 것이다. 전역 의미망을 유지하고 관리하는 두가지 기본적인 접근 방법이 있다. 중앙집중식과 분산 방식이 그것이다. 중앙집중식 전역 의미망 관리 스킴에서는, 각 정보소스에 대한 접근 지식과 의미 지식이 중앙 호스트로부터 얻어진다. 중앙집중식 전역 의미망 관리 시스템은 새로운 정보소스가 추가 또는 삭제되거나 정보소스에 수정사항이 발생될 때, 전역 의미망을 최신의 내용으로 유지하려고 노력한다. 분산 방식 전역 의미망 관리 스킴에서, 모든 정보소스는 각 정보소스가 보유하고 있는 접근 지식과 의미 지식이 최신의 그리고 일관성 있게 유지되도록 하기 위하여 분산 방식으로 협조한다. 전역 의미망은 모든 개별 정보소스에 중복되어 유지된다.

중앙집중식 방식은 전역 의미망을 관리하는 중앙 호스트가 있고, 모든 정보소스는 변경사항이 발생할 때마다 최신의 정보를 중앙 호스트에게 전달한다. 이러한 최신의 정보에 근거하여 중앙 호스트는 모든 정보소스에 대한 정확한 접근 지식과 의미 지식을 유지한다. 이 방식은 항상 최신의 정보를 유지한다는 장점이 있는 반면, 다음과 같은 문제점도 있다.

- 최신의 정보가 전달되는 과정에서 트래픽이 중앙 호스트에 집중되므로 그 주위가 매우 혼잡하다.
- 각 개별 정보소스에서 수행되는 사용자의 모든 질의

는 중앙 호스트에서 관리되는 전역 의미망을 통하여 접근지식과 의미지식을 획득해야 하므로 전체 네트워크 성능이 저하될 우려가 있다.

- 중앙 호스트의 고장이 멀티미디어 시스템 전체를 불통시킬 위험성이 있다.

분산 방식에서 각 정보소스는 변경사항이 발생할 때마다 최신의 정보를 주변 이웃 정보소스와 주고받아, 자신의 전역 의미망 정보를 갱신한다. 분산 방식의 장점은 정보교환이 이웃 정보소스에만 국한되므로 트래픽의 양이 많지 않으며 골고루 분포된다는 것이다. 또한 중앙집중식에서의 중앙 호스트와 같은 것이 없어서, 어느 한 정보소스의 고장이 전체 시스템에 큰 피해를 입히는 경우가 발생하지 않는다. 그러나, 한 정보소스의 최신 변경사항은, 많은 전달과정을 반복해야 모든 정보소스에게 인식될 수 있다는 문제점을 가진다. 더구나 이러한 전달과정중에 추가적인 상황변화가 발생한다면, 전체 시스템이 불안정해질 가능성이 있다. 또한, 각 정보소스마다 전역 의미망을 보유함으로써 중복성 문제가 생기고, 일관성 문제 또한 신중히 고려되어야 한다.

정보소스의 메타정보(스키마)는 일반적으로 자주 변경되지는 않는다. 따라서, 규모가 큰 멀티데이타베이스 시스템에서는 효율적인 질의처리 능력을 고려하여 분산 방식이 적합하고 규모가 작은 멀티데이타베이스 시스템에서는 구현이 비교적 쉽고 관리가 용이한 중앙집중식이 적합하다. 본 논문에서 구현된 시스템은 적은 수의 정보소스를 통합하는 시스템으로써 중앙집중식의 전역 의미망 관리 방식을 채택하였다. 추후에는 많은 수의 정보소스를 통합하기 위하여 분산 방식의 전역 의미망 관리 시스템을 구현할 예정이다.

5. 의미질의 처리

의미적 관점에서 보면, 데이타베이스 설계자는 실세계에 대한 설계자 고유의 개념화(conceptualization)를 수행하고 이를 데이타베이스 설계에 반영한다. 이것은 서로 다른 설계자들로 하여금, 같은 실세계 정보에 대하여 서로 상이한 스키마, 경우에 따라서는 모순된 스키마를 설계하도록 한다. 따라서, 여러 데이타베이스로부터 정보를 획득하기를 원하는 사용자는 찾고자 하는 정보를 보유하고 있는 정보소스를 식별하고 이들로부터 원하는 정보를 통합 검색해야 하는 문제를 해결해야 한다.

이러한 문제에 대한 한가지 가능한 해결책은 간단히 사용자로 하여금 모든 관련있는 정보소스에 직접 접근하게 하는 것이다. 그러나, 종종 사용자는 관련있는 정보소스가 존재하는지조차 알지 못할 수도 있다. 사용자가 관련있는 정보소스를 식별하였다고 하더라도, 사용자는 아마도 모든 관련 있는 정보소스의 스키마를 알고 있지는 않을 것이다. 그러므로, 이와같은 해결방법은 사용자가 모든 정보소스의 스키마를 알고 있어야만 가능한데, 이는 사용자에게 엄청남 부담을 지우게 된다.

또 다른 접근방법은 단일의 통합된 전역 뷰를 사용자에게 제공하는 전역 스키마 멀티데이타베이스 시스템을 구축하는 것이다. 그러나, 전역 스키마를 생성하고 유지하는 것은 매우 어렵고 사용자가 전역 스키마에 대해서 알고 있어야 한다는 부담이 있다.

보다 효과적이고 효율적인 접근방법은 사용자가 많은 수의 데이타베이스에 대한 질의를 사용자가 가지고 있는 자신의 개념에 기반하여 질의할 수 있게 하는 것이다. 이는 사용자가 개별 데이타베이스 스키마에 대하여 알지 못해도 질의를 가능하게 한다. 본 논문에서는 사용자가 스키마 정보를 알지 못하더라도 자신이 알고 있는 개념을 이용하여 질의를 할 수 있게 하기 위하여 의미질의어인 SemQL(SEMantic Query Language)을 제안한다[9][10].

5.1 SemQL: 의미질의어

SemQL은 FROM 절이 없다는 것을 제외하고는 SQL과 유사하다. SemQL의 기본형태는 두 개의 절 -SELECT절과 WHERE절-로 구성되고 다음과 같은 형태를 가진다:

SELECT <concept list>
WHERE <condition>

여기서 <concept list>는 질의에 의해 검색되어야 하는 값의 개념에 대한 리스트이다. <condition>은 질의에 의해 검색되어야 하는 튜플을 식별하기 위한 조건식이다. SQL의 FROM절에 기술되는 테이블 정보는 질의처리기에서 질의처리시 관련있는 정보소스를 식별한 후 자동으로 추가된다.

SemQL의 SELECT절과 WHERE절은 데이타베이스 스키마에 있는 개체명과 속성명을 명세하는 것이 아니라 사용자가 원하는 정보에 대한 개념을 명세한다. 예를 들면, SQL에는 익숙하지만, 접근되어져야 하는 데이타베이스들의 스키마 정보는 알지 못하는 사용자가, 월급여가 $50,000 이상인 교수에 대한 정보를 원하다고 가정하자. 그러면, 사용자는 자신의 개념에 기반하여 SemQL에서 다음과 같이 질의를 할 수 있다.

SELECT professor.name
WHERE professor.pay > $50,000

또 다른 사용자는 다음과 같이 질의를 할 수도 있다.

SELECT professor.name

WHERE professor.earnings > $50,000

위의 두 질의는 질의에 사용된 단어형태가 'pay'와 'earnings'로 다르지만 의미적으로는 같은 질의를 수행한 것이다. 다음 절에서는 이러한 사용자 자신의 개념에 의해 명세된 의미질의를 처리하는 과정에 대하여 소개한다.

5.2 의미질의 처리 과정

그림 9는 의미질의 처리과정을 보여준다. SemQL 처리기는 Query Parser, Resource Finder, Mapping Generator, Sub-query Generator, Query Distributor, 그리고 Integrator로 구성된다. 질의처리과정에서 SemQL 처리기의 각 구성요소는 4장에서 언급한 전역 의미망을 이용하여 임무를 수행한다.

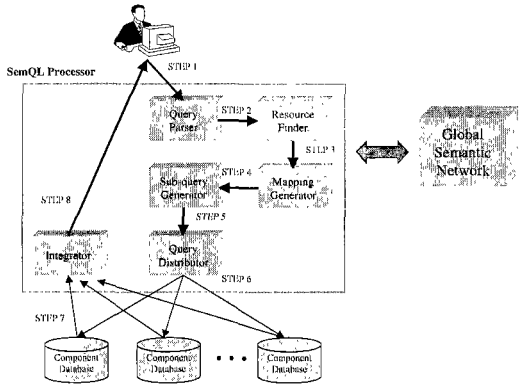


그림 9 의미질의 처리 절차

의미질의의 단계별 처리과정은 다음과 같다.

단계 1: 사용자는 찾고자 하는 정보를 검색하기 위하여 자신의 개념을 이용하여 의미질의를 작성한다.

단계 2: Query Parser는 질의를 분석하여 질의로부터 개체, 속성, 값에 해당하는 개념을 추출한다.

단계 3: Resource Finder는 전역 의미망을 이용하여 단계 2에서 추출된 모든 개념들을 포함하고 있는 정보소스를 식별한다.

단계 4: Mapping Generator는 원질의(original query)에 포함되어 있는 개념과 식별된 정보소스 스키마간의 사상(mapping)을 생성한다.

단계 5: Sub-query Generator는 원질의를 단계 4에서 생성된 사상에 따라 각 정보소스 스키마에 맞는 부질의(sub-query)들로 재구성한다. 이 단계에서, 전역 의미망을 참조하여, Sub-query Generator는 각 부질의에 FROM 절을 추가한다.

단계 6: Query Distributor는 생성된 부질의들을 해당 정보소스로 전송한다.

단계 7: 각 정보소스는 부질의를 받아서 실행하고 결과를 SemQL 처리기로 보낸다.

단계 8: Integrator는 다양한 정보소스로부터의 결과를 취합하여 통합된 결과를 사용자에게 보여준다.

다음은 예제 질의 시나리오를 이용하여 위에서 설명한 의미질의 처리과정을 보여준다. 또한, 예제 질의 시나리오를 통하여, 의미 이질성이 전역 의미망을 통하여 어떻게 해결되는지를 설명한다. 예제 질의는 서울에 살고있는 여학생을 찾는 것이다. 질의를 하는 사용자는 자신이 알고있는 개념을 이용하여 질의를 한다고 가정한다. 즉, 사용자는 각 요소 데이타베이스의 상세한 스키마 구조는 모른다고 가정한다.

질의: Find those female students who live in Seoul.

질의는 다음과 같이 SemQL로 표현되었다고 가정한다(단계 1).

```
SELECT student.name
WHERE student.sex="female"
    AND student.city="Seoul"
```

Query Parser는 질의를 분석하여 질의로부터 다음과 같은 개념들을 추출한다(단계 2).

{student, name, sex, female, city}

그리고 나서, Resource Finder는 추출된 개념들을 모두 포함하고 있는 관련된 요소 데이타베이스, CDB_1, CDB_2, 그리고 CDB_3를 전역 의미망을 통하여 식별한다(단계 3). 원질의에 포함된 개념과 관련된 요소 데이타베이스 스키마간의 사상은 Mapping Generator에 의해 생성되고(단계 4), 그림 10에서 이 사상을 보여준다.

이 예제 시나리오에서, CDB_3는 주소에 대한 정보를 위해, 세 속성 street, city, 그리고 state를 사용하는 반면, CDB_1은 하나의 속성 address를 사용한다. 이는 AASC인 경우이다. EASC 타입은 CDB_1이 학생의 주소를 나타내기 위하여 개체 Student의 속성 address를 사용하고, CDB_2는 같은 정보에 대해 개체 Address를 사용하는데서 볼 수 있다. CDB_3는 여학생에 대하여 개체 FemaleStudent를 사용하고, 성(性)에 대한 정보를 CDB_1에서는 속성 sex의 값으로 표현하기 때문에, EVSC 타입 역시 예제 시나리오에서 발생한다.

이제, Sub-query Generator는 원질의를, 생성된 사상에 따라, CDB_1, CDB_2, 그리고 CDB_3에 대한 세 개의 부질의로 재구성한다(단계 5). 그러므로, CDB_1에 대한 부질의는 다음과 같다;

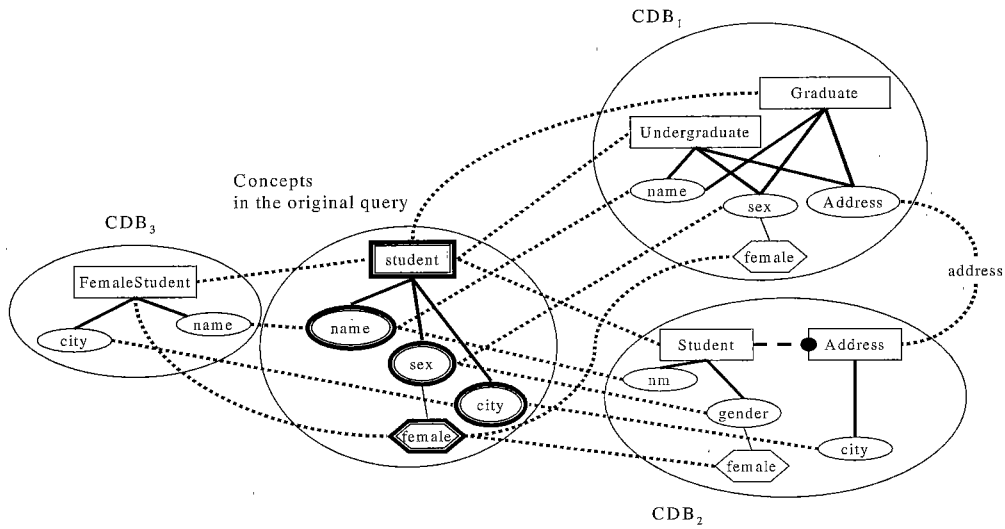


그림 10 예제 시나리오에 대한 데이타베이스 객체와 개념간 사상(mapping)

```
SELECT name
FROM Undergraduate
WHERE sex="female"
   AND address LIKE "%Seoul%"
UNION
SELECT name
FROM Graduate
WHERE sex="female"
   AND address LIKE "%Seoul%"
```

원질의는 CDB_2에 대하여 다음과 같이 재구성된다;

```
SELECT Student.nm
FROM Student, Address
WHERE Student.sid=Address.sid
   AND Address.city="Seoul"
   AND Student.gender="female"
```

CDB_3에 대한 부질의는 다음과 같다;

```
SELECT name
FROM FemaleStudent
WHERE city="Seoul"
```

Sub-query Genarator가 원질의를 세 개의 부질의로 재구성한 후에, Query Distributor는 이 부질의들을 각각 CDB_1, CDB_2, 그리고 CDB_3로 전송한다(단계 6). 세 요소 데이타베이스, CDB_1, CDB_2, 그리고 CDB_3는 부질의에 대한 결과를 SemQL 처리기로 보낸다(단계 7). 마지막으로, Integrator는 세 요소 데이타베이스로부터의 결과를 취합하여 통합된 결과를 사용자에게 보여준다(단계 8).

6. 실 험

지금까지 각 정보소스에 대한 의미망을 구축하는 방법, 이들 의미망을 통하여 의미 이질성을 탐색하고 전역 의미망을 생성하는 과정, 그리고 전역 의미망을 이용하여 의미 이질성을 해결하고 사용자의 의미질의를 처리하는 과정을 설명하였다. 이러한 멀티데이타베이스 시스템에서의 효율적이고 효과적인 정보공유를 위하여 본 논문에서 제안한 방법의 성능을 평가하기 위하여 웹 DB를 대상으로 실험을 수행하였다. 웹 DB를 정보소스로 선택한 이유는, 웹 DB는 웹을 통하여 누구나 쉽게 접근할 수 있고, 다양한 실세계의 일반적인 정보들을 많이 보유하고 있으며, 그리고 웹 인터페이스를 통하여 편리한 접근방법을 제공하기 때문이다.

일반적인 웹문서 정보검색 시스템에서는 질의에 대한 검색결과가 얼마나 정확한지에 대한 평가를 필요로 한다. 이러한 종류의 평가를 검색성능평가(retrieval performance evaluation)라고 한다. 검색성능평가는 일반적으로 테스트에 필요한 자료집합들과 성능 척도에 기반하여 수행된다. 테스트에 필요한 자료집합들은 웹문서 집합, 정보질의 집합, 그리고 전문가에 의해 제공된 각 정보질의에 대한 관련문서집합으로 구성된다.

주어진 검색전략 S에 대해, 성능 척도는 각 정보질의에 대해 S에 의해 검색되어진 문서집합과 전문가에 의

해 제공된 관련문서집합 사이의 유사도에 의해 양적으로 평가된다. 이것은 검색 전략 S의 우수성에 대한 평가를 제공한다[12].

위와 같은 웹문서를 대상으로 한 검색성능평가 방법을 응용하여, 본 논문에서는 사용자의 의미질의가 제안된 방법을 통하여 얼마나 효과적으로 처리되는지 하는 검색성능 측면에서 실험을 수행하였다.

6.1 실험내용

실험을 위하여 Windows 2000 운영체제를 기반으로 ASP2.0을 이용하여 시스템을 구축하였다. 구축된 시스템은 정보소스 스키마에 대한 서술(description)을 작성하는 모듈, 작성된 서술로부터 표현 테이블을 만들고 의미망을 생성하는 모듈, 모든 의미망을 통합하여 전역 의미망을 생성하는 모듈, 그리고 사용자의 의미질의를 처리하는 SemQL 처리기 모듈 등으로 구성된다.

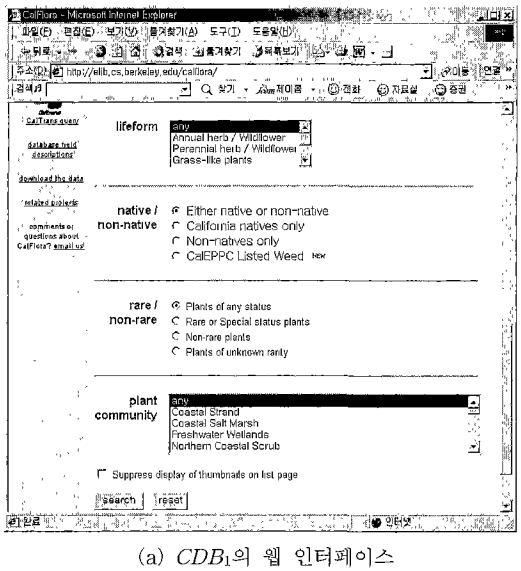
정보소스로는 4개의 웹 DB를 선택하였으며 실험의 용이성을 위하여 모두 식물관련 정보를 제공하는 웹 DB를 선택하였다. 실험에 사용된 웹 DB들의 목록은 다음과 같다.

- 요소 데이타베이스 1(CDB_1): 버클리 대학의 California Flora 데이타베이스
- 요소 데이타베이스 2(CDB_2): San Jose 대학의 Sharsmith Herbarium 데이타베이스
- 요소 데이타베이스 3(CDB_3): 미국 농무부의 Plants 데이타베이스
- 요소 데이타베이스 4(CDB_4): 영국의 Plants For A Future 데이타베이스

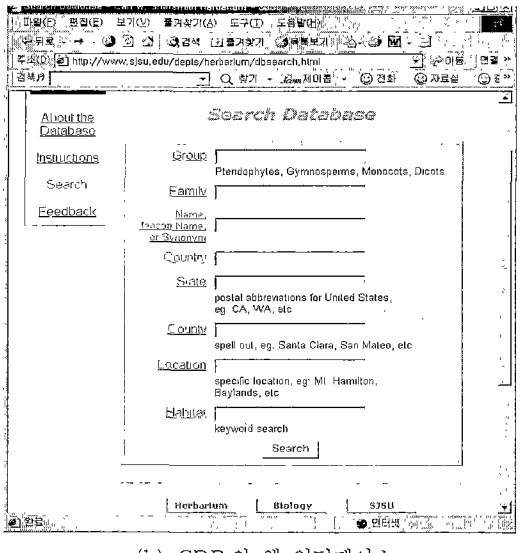
일반적으로 DB를 통해 정보를 제공하는 웹 사이트의 구현은 두 단계로 수행된다. 우선, 웹 사이트의 실제 컨텐트에 대한 데이타베이스를 설계한다. 다음에, 웹 사이트의 컨텐트를 검색할 수 있게 해주는 웹 인터페이스(템플릿)의 설계이다. 사용자의 질의는 웹 인터페이스를 통해 DB에 전달된다. 본 논문의 실험에서는 각 CDB의 실제 데이타베이스 스키마 정보대신 웹 인터페이스를 하나의 논리적 개체(테이블)로 간주하고 웹 인터페이스에 있는 필드들을 속성으로 간주한다. 그림 11의 (a)는 CDB_1의 웹 인터페이스를, (b)는 CDB_2의 인터페이스를 보여준다.

구축된 시스템을 이용하여 전문가의 도움을 받아 4개의 데이타베이스를 통합한 전역 의미망을 생성하였다. 그리고 나서, 생성된 전역 의미망에 있는 모든 개념들을 추출하고 이를 토대로 가능한 모든 예상되는 질의집합을 생성하였다. 다음에, 각 질의를 SemQL 처리기를 통하여 수행하여 그 결과를 분석하고 검색성능을 평가하였다.

그림 12의 (a)는 사용자가 "서식지가 '습지대'인 식물의 이름"을 찾기 위해 질의를 입력하는 예이고, (b)는 입력된 질의에 대한 처리결과를 보여준다. (b)는 주어진 질의와 관련된 웹 사이트와 해당 웹 사이트의 인터페이스에 맞게 질의가 재구성된 결과를 보여준다.

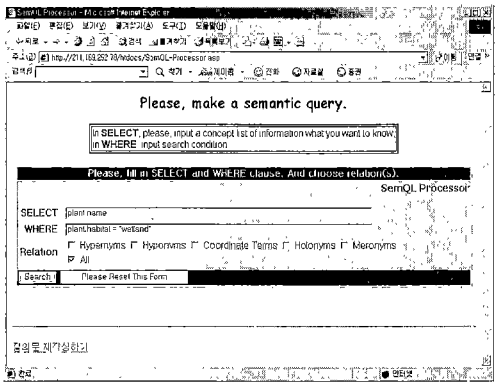


(a) CDB_1의 웹 인터페이스

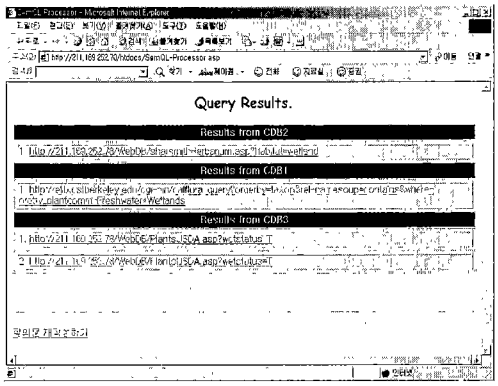


(b) CDB_2의 웹 인터페이스

그림 11 실험에 사용된 웹 DB 인터페이스의 예



(a) 의미질의 작성



(b) 처리결과

그림 12 의미질의 작성과 SemQL 처리기를 통한 처리결과

6.2 실험에 사용된 검색성능 평가방법

본 실험에서는 주어진 질의집합에 대하여 각 요소 데이타베이스에서의 질의처리결과를 토대로 성능을 평가한다. 다음은 평가방법에 사용된 기호의 의미와 수식이다.

- Q: 질의집합
- R_{CDB_i}(단, i=1,2,3,4): CDB_i와 관련된, CDB_i에서 처리되어져야 하는 질의집합
- P_{CDB_i}(단, i=1,2,3,4): CDB_i에 의해 실제 처리된 질의집합
- R_{CDB_i}, $P_{CDB_i} \subseteq Q$
- $Ra_{CDB_i} = R_{CDB_i} \cap P_{CDB_i}$
- $|Q|$: Q에 속하는 질의개수
- $|R_{CDB_i}|$: R_{CDB_i}에 속하는 질의 개수
- $|P_{CDB_i}|$: P_{CDB_i}에 속하는 질의 개수
- $|Rp_{CDB_i}|$: Rp_{CDB_i}에 속하는 질의 개수
- $Recall_{CDB_i} = \frac{|Rp_{CDB_i}|}{|R_{CDB_i}|}$
- $\Pr ecision_{CDB_i} = \frac{|Rp_{CDB_i}|}{|P_{CDB_i}|}$
- $Error_{CDB_i} = 1 - \Pr ecision_{CDB_i}$

R_{CDB_i}는 전문가의 도움을 얻어 전체 질의집합 Q에서 추출하였으며 CDB_i에서 처리되어져야 하는 질의들의 집합이다. P_{CDB_i}는 Q의 각 질의를 실제 구축된 SemQL 처리기에서 수행한 결과, 실제 CDB_i에서 처리가 되어진 질의들의 집합이다. Rp_{CDB_i}는 R_{CDB_i}와 P_{CDB_i}의 교집합에 속하는 질의들의 집합을 의미한다. 그림 10은 이들 집합간의 관계를 보여준다.

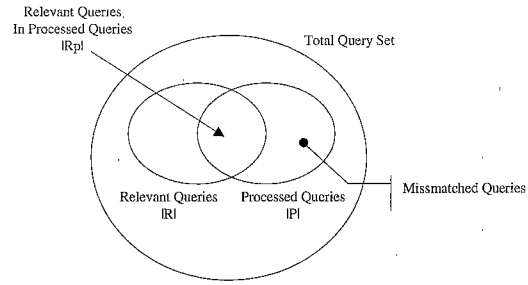


그림 13 R(관련 질의), P(처리된 질의), Rp(처리된 질의 중 관련질의) 집합간의 관계

$Recall_{CDB_i}$는 CDB_i에서 처리되어져야 하는 질의중 실제 처리된 질의의 비율을 나타내고, $\Pr ecision_{CDB_i}$은 CDB_i에서 SemQL 처리기 에 의해 실제 처리된 질의중 CDB_i와 관련된 질의의 비율이다. $Error_{CDB_i}$는 CDB_i에서 실제 처리된 질의중 CDB_i와 관련이 없는 질의의 비율로 SemQL Processor의 질의처리시 CDB_i에 대해 오류가 발생할 확률이다.

6.3 결과 및 평가

실험에 사용된 전체질의개수($|Q|$)는 134개이며, 실험결과 다음과 같은 기초 데이타를 얻었다.

$|R_{CDB_1}| = 56$, $|P_{CDB_1}| = 53$, $|Rp_{CDB_1}| = 53$

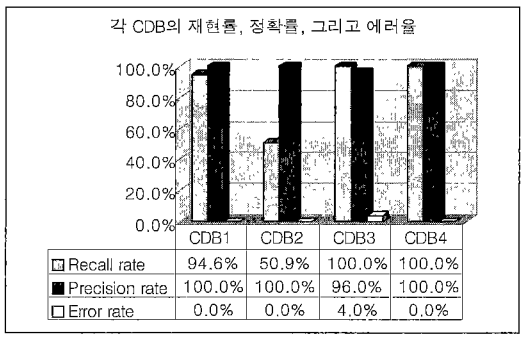
$|R_{CDB_2}| = 53$, $|P_{CDB_2}| = 27$, $|Rp_{CDB_2}| = 27$

$|R_{CDB_3}| = 20$, $|P_{CDB_3}| = 23$, $|Rp_{CDB_3}| = 22$

$|R_{CDB_4}| = 85$, $|P_{CDB_4}| = 85$, $|Rp_{CDB_4}| = 85$

획득된 기초 데이타와 6.2절에서 언급한 수식에 따라 각 요소데이타베이스의 재현률(recall rate), 정확률

(precision rate) 그리고 에러율(error rate)을 계산한 결과를 그림 11에서 그래프로 표현하였다.



	CDB1	CDB2	CDB3	CDB4
Recall rate	94.6%	50.9%	100.0%	100.0%
Precision rate	100.0%	100.0%	96.0%	100.0%
Error rate	0.0%	0.0%	4.0%	0.0%

그림 14 각 요소 데이타베이스의 재현률(recall rate), 정확률(precision rate) 그리고 에러율(error rate)

CDB_1의 재현률(recall rate)은 94.6%로서 매우 우수한 편인 반면, CDB_2의 재현률은 50.9%로서 그다지 높지 않다. 이는 CDB_1의 웹 인터페이스의 입력필드들의 대부분이 사용자가 값을 선택하는 방식으로 구성되어 있는 반면(그림 11 (a) 참조), CDB_2의 웹 인터페이스의 입력필드들은 모두 사용자가 값을 입력하는 방식으로 구성되어 있다(그림 11 (b) 참조). 즉, CDB_1이 CDB_2보다 더 많은 스키마 정보를 제공한다. 실험결과 스키마 정보를 보다 많이 그리고 상세히 제공하는 웹 DB인 경우 재현률이 매우 높다는 것을 알 수 있었다. CDB_3도 웹 인터페이스의 대부분의 입력필드가 사용자가 값을 선택하는 방식으로 구성되어 있기 때문에, 재현률이 매우 높으며, CDB_4인 경우에는 모든 입력필드가 사용자가 값을 선택하는 방식으로 되어 있어 재현률이 거의 완벽에 가까운 것을 볼 수 있다. 평균적으로 SemQL 처리기는 사용자의 질의에 부합되는 관련 정보소스를 식별하는 능력이 매우 우수함을 알 수 있었다.

정확률(precision rate)의 경우에는 모든 CDB에서 매우 우수하다는 것을 알 수 있었다. 이는 SemQL 처리기가 질의처리과정에서 식별한 관련 CDB에 대해, 원질의를 각 CDB에 맞는 부질의로 정확히 재구성한다는 것을 보여준다.

본 실험은 실험실 수준의 실험으로, 4개의 실제 웹 DB만을 대상으로 그리고 4개의 웹 DB에 있는 개념들의 조합으로 구성된 질의집합에 대해서만 수행되었기 때문에, 실제 불특정 다수의 웹 DB에 적용하였을 경우에는 재현률이나 정확률이 조금은 낮아지리라고 예상된다. 그러나, 각 웹 DB의 스키마에 대한 정보가 충분히 제공되고 이에 대한 서술(description)이 본 논문에서 제안한 규칙에 따라 제대로 작성되었다고 가정할 때, 본 논문에서 제안한 방법은 효과적이고 효율적인 정보공유를 위한 한 방법으로서 우수한 성능을 보인다고 할 수 있다.

7. 관련 연구

멀티데이타베이스 시스템에서의 의미 이질성에 대한 초창기 연구는 주로 개별 요소 데이타베이스 스키마를 단일의 전역 스키마로 통합하는 절차에 초점이 맞추어졌다. 전역 스키마 멀티데이타베이스는 사용자에게 단일의, 통합된 전역 뷰(view)를 지원하고 간단하고 효과적인 패러다임을 제공한다. 그러나, 전역 스키마를 생성하고 유지하는 것은 매우 어렵다. 지역 스키마에 대해 요구되는 지식의 양과 지역 스키마간의 이질성을 식별하고 해결하는 것은 이 방법에서 주된 문제가 된다. 또한, 지역 스키마에 대한 작은 변경도 전역 스키마에 커다란 영향을 미친다. 그러므로, 이러한 정적인 접근방법은 작은 그리고 정적인 시스템에서는 만족스러울지 모르지만, 큰 규모의 상호운영 가능한 데이타베이스 시스템과 데이타베이스가 수시로 변하고 새로운 데이타베이스가 자율적으로 추가되는 동적인 환경에는 부적합하다.

본 논문에서 제안한 방법은 전역 뷰를 구축할 필요가 없으므로 정보통합 시스템 구축이 용이하다. 또한, 각 개별 정보소스는 다른 정보소스들과 독립적으로 정보통합 시스템에 통합되므로, 각 요소 데이타베이스의 스키마 변경이 전체 정보통합 시스템 관리에 영향을 미치지 않는다. 이와 같은 접근방법은 확장성이 용이하므로, 독립적이고 자율적인 많은 수의 정보소스가 연동되는 개방적이고 동적인 환경에 적합하다.

멀티데이타베이스 언어는 전역 스키마와 연관된 일부 문제들을 해결하기 위하여 시도되었다. 멀티데이타베이스 언어를 지원하는 시스템에서는 전역 스키마를 유지하지 않는다. 이 접근방법은 사용자에게 표준 SQL 이상의 기능을 제공하여 사용자가 질의의 일부로서 통합정보를 명세하게 한다. 멀티데이타베이스 언어를 이용한 방법은 전역 스키마의 생성과 유지의 문제를 제거하지만, 사용자는 보다 복잡한 전역 인터페이스를 이용할 수 있어야 한다.

본 논문에서 제안한 방법은 의미질의어인 SemQL을 지원한다. SemQL은 사용자가 자신의 개념에 기반하여 질의를 할 수 있게 함으로써, 많은 수의 요소 데이타베

이스에 대한 질의를 손쉽게 할 수 있는 사용자 인터페이스를 제공한다. 이러한 접근 방법을 통하여 사용자는 각 개별 요소 데이타베이스 스키마에 대하여 알지 못해도 자신이 알고 있는 개념을 이용하여 질의를 용이하게 할 수 있다.

이후 여러 연구에서[2][13][14], 웹과 같은 동적이고 개방 환경에서, 에이전트, 도메인 온톨로지, 지능형 중재기, 그리고 고수준 질의어와 같은 새로운 기술을 이용하여 정보를 통합하려는 새로운 방법들이 개발되고 있다. MCC의 InfoSleuth 프로젝트는 Carnot의 기능을 동적으로 변화하는 환경에 적합하도록 확장하였다[14]. SIMS 프로젝트에서, 각 SIMS 에이전트는 도메인 지식의 상세 모델과 이 모델에 이용 가능한 정보 소스의 모델을 포함한다. 주어진 정보 요청에 대하여 에이전트는 적절한 정보소스의 집합을 선택하고 질의 플랜을 생성하고 이를 실행한다. 이러한 접근방법들은 유연성과 개방성을 지원하기 위하여 설계되었다[13]. 이러한 동적인 접근방법들의 공통적인 가정은 사용자가 정보를 통합하기 위하여 필요한 도메인에 대한 지식을 알고 있다는 것이다. 이는 정보를 통합하려는 사용자에게 큰 부담이 된다.

본 논문에서는 WordNet을 지식베이스로 활용하여 공통의 도메인 지식을 제공함으로서 각 개별 정보소스는 다른 정보소스의 특정 도메인 지식을 알지 못하더라도 쉽게 통합되어진다. 즉, 각 개별 정보소스는 다른 정보소스들에 독립적으로 자신의 도메인 지식을 서술하고 의미망을 구축함으로서, 정보통합시스템에 통합되어진다. 이러한 접근 방법은 각 개별 정보소스의 독립성을 보장하며, 정보를 통합하려는 사용자에게 용이한 정보통합 방법을 제공한다. 또한, 많은 수의 정보소스를 쉽고 자연스러운 방법으로 통합할 수 있게 함으로써, 우수한 확장성을 제공한다.

최근의 온라인 사전과 시소러스의 기술발전은 사용자가 언어 이론들을 토대로 좀더 편리하게 정보를 통합할 수 있도록 지원한다. [15]에서, Summary Schemas Model(SSM)은 의미 식별을 지원하기 위하여 기존 멀티데이타베이스에 대한 확장으로 제안되었다. SSM은 멀티데이타베이스 시스템에서 이용 가능한 정보를 요약하기 위하여 전역 데이타 구조를 사용한다. 요약된 형태는 사용자로 하여금 정보를 접근할 때 시스템-특정적인 용어가 아닌 사용자 자신의 용어를 이용하는 것을 허용한다. 이 시스템은 사용자의 용어를 의미적으로 가까운 이용 가능한 시스템 용어와 일치시키기 위하여 전역 데이타 구조를 사용한다. 그러나, 이 접근 방법은 질의를 단일의 논리적 인덱스에 한정하기 때문에 규모가 큰 멀티데이타베이스 시스템에서는 성능의 한계를 가진다.

본 논문에서는 특정 시스템 용어를 사용하는 것이 아니라, WordNet에서 제공하는 일반적인 용어, 동의어 관계에 있는 synset이라는 논리적 그룹, 그리고 이들 synset간의 의미관계를 활용하여 정보통합 시스템에서 필요로 하는 스키마 정보를 표현한다. 이러한 접근 방법은 특정 시스템 용어에 한정되지 않으므로, 확장성이 용이하고 규모가 큰 멀티데이타베이스 시스템에 적합하다. 최근 수십 년간 발전된 언어 이론들을 토대로 언어학자들은 음운체계, 구문론 등을 위해 어휘가 포함해야 하는 정보들에 대하여 보다 명확하게 이해할 수 있게 되었다. 그러한 노력중의 하나인 WordNet은 프린스턴 대학에서 개발된 전자 어휘 시스템이다[3]. 여러 접근방법들이 검색 효율을 향상시키기 위하여 이미지의 의미 내용에 대한 지식베이스로서 WordNet을 사용한다[16][17]. 특히, [16]은 질의와 데이타베이스 확장을 위하여 WordNet을 사용한다.

8. 요약 및 결론

멀티데이타베이스 시스템에 대한 전형적인 접근방법은 개별 요소 데이타베이스 스키마들을 단일의 전역 스키마로 병합하는 절차에 초점을 맞추었다. 이 방법의 단점은 정보소스의 일부가 변경되거나 새로운 정보소스가 추가될 때마다 전역 스키마를 새로 재구성해야 하는 비용이 매우 크다는 것이다. 멀티데이타베이스 언어를 이용한 접근방법은 전역 스키마를 생성하고 유지하는 문제는 제거하였지만, 사용자에게 복잡한 전역 인터페이스를 제시한다. 여러 연구에서, 정보 중재기(information mediator)는 동적이고 개방된 환경에서 정보를 통합하기 위하여 개발되어 왔다. 정보 중재기를 사용하는 이러한 접근방법들에 공통적인 한가지 문제점은 쉽고 자연스러운 방법으로 많은 수의 정보소스로 시스템을 어떻게 확장할 것인가 하는 것이다.

지식베이스로서 WordNet을 사용하여, 본 논문에서는 여러 데이타베이스간의 정보 공유를 위한 방법을 제안하였다. 스키마의 서술로부터, 각 요소 데이타베이스가 보유한 정보의 의미를 표현하기 위해 의미망을 구축하였다. 각 요소 데이타베이스에 대한 의미망들을 전역 의미망으로 병합하면서, 각 요소 데이타베이스가 의도한바와 같은 방법으로 정보를 해석할 수 있었다. 전역 의미망은 요소 데이타베이스들에 대한 통합된 접근을 위해 필요한 지식인, 접근지식과 의미지식을 멀티데이타베이스 시스템에게 제공한다. 전역 의미망과 의미질의어인

SemQL을 통하여, 사용자에게 통합된 정보에 대한 효율적이고 효과적인 접근 메카니즘을 제공한다. SemQL은 사용자가 원하는 정보에 대한 개념을 취하여, 사용자로 하여금 개별 요소 데이타베이스의 스키마 정보 없이 많은 수의 자율적인 데이타베이스에 대한 질의를 수행할 수 있도록 한다.

사람의 개입 없이 시스템이 정보의 의미를 정확하게 획득한다는 것은 어렵기 때문에, 의미망의 생성시, 사용자는 초기정보인 스키마에 대한 서술을 제공해야 한다. 그러므로, 데이타베이스 스키마에 대한 사용자의 서술은 요소 데이타베이스들을 통합하는데 있어 아주 중요하다. 부족하고 잘못된 스키마 서술은 멀티데이타베이스 시스템의 성능을 저하시킬 수 있으며, 본 논문이 제안하는 방법의 단점이 될 수 있다. 그러나, 사용자가 본 논문에서 제안하는 규칙에 따라 서술을 작성한다면, 멀티데이타베이스 시스템은 사용자로 하여금 관련된 요소 데이타베이스를 찾고 각 개별 데이타베이스에 적합한 특정 인터페이스로 상호작용해야 하는 굉장히 어려운 작업으로부터 해방시킬 수 있다. 더욱이, 본 논문의 접근방법은 일반적인 어휘개념들을 지식베이스로 사용함으로서, 개별적인 정보소스에 독립성을 보장한다. 즉, 개별적인 정보소스는 다른 정보소스들에 독립적으로 정보 의미를 서술할 수 있다. 이는 멀티데이타베이스 시스템이 요소 데이타베이스가 변경되거나, 추가되거나, 또는 삭제되더라도 전역 의미망을 쉽고 효율적으로 관리할 수 있고 시스템 확장이 매우 용이하다는 것을 의미한다. 결론적으로, 본 논문의 접근방법은 독립적이고, 이질적인 정보소스간의 정보공유를 위한 간단하고, 효율적이며, 그리고 효과적인 메카니즘을 제공한다.

참 고 문 헌

[1] Amit Sheth, Vipul Kashyap, *"So Far(Schematically) yet So Near(Semantically),"* Proceedings, IFIP WG 2.6 Conference on Semantics of Interoperable Database Systems(Data Semantics 5), pp. 283-312, 1993.

[2] Marti A. Hearst, *"Trends & Controversies Information Integration,"* IEEE INTELLIGENT SYSTEMS Vol. 13, No. 5, pp. 12-24, SEPTEMBER/OCTOBER 1998.

[3] G. A. Miller, R. Beckwith, C. Fellbaum, D.Gross, and K. Miller, *"Five Papers on WordNet,"* CSL Reort 43, Cognitive Systems Laboratory, Priceton Univ., 1990.

[4] Richard Hull, *"Managing semantic heterogeneity in databases: a theoretical prospective,"* Proceedings of the sixteenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems, pp. 51-61, 1997.

[5] Aris M. Ouksel, Amit P. Sheth, *"Semantic Interoperability in Global Information Systems*: A Brief Introduction to the Research Area and the Special Section". SIGMOD Record 28(1), pp. 5-12, 1999.

[6] M. Garcia-Solaco, F. Saltor, and M. Castellanos, *Semantic Heterogeneity in Multidatabase Systems, Object-Oriented Multidatabase Systems: A Solution for Advanced Applications*, Edited by Orman A. Bukhres, Ahmed K. Elmagarmid, Prentice Hall Inc., pp. 129-202, 1996.

[7] W. Kim, J. Seo, *"Classifying Schematic and Data Heterogeneity in Multidatabase Systems,"* IEEE Computer, Vol. 24, No. 12, pp. 12-18, 1992.

[8] Jeong-Oog Lee, Doo-Kwon Baik, *"SemNet: A Semantic Network for Integration of Databases,"* Lecture Notes in Artificial Intelligence, LNAI 1747, Springer-Verlag, 1999.

[9] Jeong-Oog Lee, Doo-Kwon Baik, *"Semantic Integration of Databases using Linguistic Knowledge,"* Lecture Notes in Computer Science, LNCS 1749, Springer-Verlag, 1999.

[10] Jeong-Oog Lee, Doo-Kwon Baik, *"SemQL: A Semantic Query Language for Multidatabase Systems,"* In Proceedings of the Eighth International Conference on Information and Knowledge Management (CIKM-99), 1999.

[11] *Specification and standardization of data elements*, ISO/IEC 11179

[12] R. Baeza-Yates, B. Ribeiro-Neto, *Modern Information Retrieval*, 1st Ed., pp. 73-97, Addison-Wesley, 1999.

[13] Craig A. Knoblock, Yigal Arens, and Chun-Nan Hsu, *"Cooperating Agents for Information Retrieval,"* Proceedings of the second International Conference on Cooperative Information Systems, 1994.

[14] R. Bayardo, W. Bohrer, et al: InfoSleuth, *"agent-based semantic integration of information in open and dynamic environments,"* ACM SIGMOD Record, Vol. 26, No. 2, 1997.

[15] A. R. Hurson, M. W. Bright, *"Global Information Access for Microcomputers,"* Journal of Mini and Micro Computer Applications, 1991.

[16] Aslandogan, Y. A., C. Thier, C. T. Yu, J. Zou and N. Rishe, *"Using semantic contents and WordNet in image retrieval,"* Proceedings of the 20th Annual ACM SIGIR Conference on Research and Development in Information Retrieval, 1997.

[17] Eugene J. G. And Neil C. R., *"Natural-Language Retrieval of Images Based on Descriptive Captions,"* In ACM Transactions on Information Systems, 14(3), 1996.

이 정 욱
1992년 고려대학교 컴퓨터학과 학사. 1994년 고려대학교 컴퓨터학과 석사. 1994년 ~ 현재 고려대학교 컴퓨터학과 박사과정. 관심분야는 데이터베이스, 인공지능

백 두 권
1970년 ~ 1973년 고려대학교 수학과. 1974년 ~ 1976년 고려대학교 산업공학과(석사). 1981년 ~ 1983년 Wayne State Univ. 전산학 석사. 1984년 ~ 1985년 Wayne State Univ. 전산학 박사. 1986년 ~ 현재 고려대학교 컴퓨터학과 교수. 1989년 ~ 현재 한국정보과학회 평의원/이사. 1991년 ~ 현재 ISO/IEC JTC1/SC32 국내위원회 위원장. 관심분야는 소프트웨어 공학, 데이타 공학, 메타데이타, 컴포넌트 시스템, 데이타베이스