
XMDR 기반의 통합 검색을 위한 데이터 그리드 Wrapper 설계

황치곤* · 정계동* · 최영근*

The Design of Data Grid Wrapper for Integrated Retrieve based on XMDR

Chi gon Hwang* · Kye dong Jung* · Young-keun Choi*

이 논문은 2008년도 광운대학교 연구비를 지원받았음

요 약

최근 데이터 통합을 위한 방안으로 데이터 이질성을 해결하기 위한 많은 연구가 진행되고 있다. 우리가 제안하는 시스템의 구성요소는 XMDR 래퍼와 XMDR 저장소이다. XMDR 래퍼는 XMDR의 표준 정보를 기반으로 인터페이스를 생성하고, 표준 정보와 로컬스키마 간의 매핑정보를 이용하여 글로벌 XMDR 쿼리와 로컬 쿼리 간의 상호 변환을 수행함으로써 기존 시스템의 이질성을 해결한다. XMDR 저장소는 표준 정보와 로컬 간의 매핑정보를 관리하는 XMDR과 수행된 결과를 저장하는 Proxy DB로 구성되어 있다.

사용자는 동일한 인터페이스를 사용하고, XMDR 래퍼가 XMDR의 메타 시맨틱 온톨로지를 이용하여 스키마의 이질성을 해결뿐만 아니라 인스턴스 시맨틱 온톨로지를 통한 값의 의미에 따른 이질성도 고려함으로써 중복된 질의를 수행하지 않아도 된다. 따라서 본 논문에서는 이러한 데이터 이질성을 해결하고 효율적인 데이터 통합을 위한 데이터 그리드 래퍼를 제안한다.

ABSTRACT

Recently, many researches have been conducted to solve data heterogeneity as a way for data integration. The elements of the system that we suggest are an XMDR wrapper and XMDR Repository. XMDR wrapper solves the heterogeneity of the existing system by creating the interface based on the standard information of XMDR, and performing the inter-conversion between global XMDR query and local query using mapping data on standard information and local schema. XMDR Repository are composed of XMDR which manages the mapping data on standard information and local schema, and of Proxy DB which saves the accomplished results.

With XMDR wrapper and XMDR Repository, users can use the same interface, and they need not conduct repeated queries since XMDR wrapper not only solves the heterogeneity of the schema using the meta-semantic ontology of XMDR, but also considers the heterogeneity accompanying the meaning of the value through instance semantic ontology. Therefore, in this paper we suggest the grid wrapper for the solution of data heterogeneity and efficient data integration.

키워드

XMDR(eXtended Meta-Data Registry), MDR(MetaData Registry), XML, Distributed DataBase, Data-Grid

I. 서 론

그리드 컴퓨팅 기술은 많은 분야에서 이용되고 있다 [1]. 데이터를 저장하는 데이터베이스 또한 한 곳에 집중되어 있는 것이 아니라 분산되어 있으며, 분산된 데이터베이스는 다양한 DBMS로 구성되고, 구조와 의미상 이질성을 가지고 있다.

DAIS-WG(Database Access and Integration Service Working Group)의 명세[2][3][4]는 DB 소유자들이 그들의 메타데이터를 공유함으로써 통합이 가능하다. OGSA(Open Grid Services Architecture)를 이용하여 통합하는 방안들이 나오고 있다[5]. 따라서 분산 데이터의 상호운용은 기존의 시스템들의 협업을 위한 기본적인 조건으로, 데이터의 가용성과 투명성 문제, 협업에 참여하는 시스템의 이질성에 따른 문제들을 해결해야 한다. 또한 기존의 시스템의 변경을 최소화하고 전체 시스템에 독립적이며 자율적인 환경을 제공해야 한다[6]. 분산된 데이터베이스나 멀티 데이터베이스를 접근하는 방법은 미디어이터/래퍼를 이용한 접근법과 제휴/분산 접근법이 있다[7].

본 논문에서 제안하는 시스템은 DB 통합을 위해 XMDR과 XMDR을 기반으로 동일 인터페이스를 제공하고 그리드에 참여하는 DB를 통합하기 위한 래퍼 시스템이다.

XMDR은 분산된 데이터의 통합 또는 그리드에서 메타 데이터의 이질성을 해결하기 위한 메타 시맨틱 온톨로지와 메타 로케이션, 실제 데이터 값에 의한 인스턴스 간의 이질성을 해결하기 위해 인스턴스 시맨틱 온톨로지 구성된다.

XMDR 래퍼는 XMDR의 정보를 application에 제공하여 공통 인터페이스 생성하고, XMDR 관리와 노드의 상태정보를 관리하고, 검색결과를 Proxy DB에 집중시키는 역할을 제공한다.

래퍼는 XMDR을 이용하여 글로벌 XMDR 쿼리를 생성하고, 결과를 위한 XML Schema를 생성하고 글로벌 XMDR 쿼리를 로컬 쿼리로 변환하는 방안을 제안한다. 논문의 2장은 XMDR의 정의와 그에 대한 관련 연구에 대해서 기술한다. 3장은 제안 시스템의 구성과 운영에 대해서 기술한다. 4장은 타 시스템과의 비교 평가 및 시스템 수행 시간테스트를 통한 성능 평가한다. 5장은 시스템 구현에 대해 기술한다. 마지막 장은 본 논문의 결론

과 향후연구에 대해 기술한다.

II. XMDR의 정의

XMDR은 데이터 통합에 따른 데이터 이질성을 해결하기 위하여, XML[8] 기반의 관계형 데이터베이스 메타 데이터를 객체지향 데이터베이스에 저장하는 기술과 분산된 데이터의 이질성을 해결하고자 MDR(Metadata Registry)과 온톨로지를 결합하여 데이터를 통합하는 시스템이다[9][10].

본 논문에서는 분산된 데이터 통합을 위한 또는 그리드에서 메타 데이터의 이질성을 해결하기 위한 메타 시맨틱 온톨로지와 메타 로케이션을 결합하고 실제 값들 사이의 이질성을 해결하기 위해 인스턴스 시맨틱 온톨로지를 결합한 것을 XMDR이라 한다.

각 구성 요소들은 ISO/IEC 11179-3에서 제안한 데이터의 속성 명세를 따른다[11].

- 식별속성 : 데이터 요소의 식별을 위한 속성
- 정의속성 : 데이터의 표준요소와 그리드 노드의 데이터베이스 요소의 의미 관계를 표현하기 위한 속성
- 표현속성 : 데이터를 표현하기 위한 속성
- 관계속성 : 메타 시맨틱 온톨로지의 영역과 메타 로케이션을 연결하는 속성

정의된 속성들에 의해 메타 시맨틱 온톨로지에 표현된 데이터의 표준과 관계성을 고려한 매핑으로 각 노드의 XMDR에 접근할 수 있는 방법을 제공한다.

2.1 메타 시맨틱 온톨로지 (MSO:Meta Semantic Ontology)

그리드를 구성하는 각 노드의 데이터베이스 스키마 정보를 표준에 맞추어 매핑한 것으로, 메타데이터의 관계성과 DB 표현의 이질성 해결을 목적으로 한다.

- 의미적 이질성 : 의미는 같으나 표현이 다름으로 인해 발생하는 이질성
- 구조적 이질성 : 두 개 이상의 필드로 분할됨으로서 발생하는 이질성
- 데이터 타입 이질성 : 데이터의 타입이나 크기의 차이로 발생하는 이질성.

이러한 방법으로 이질적인 문제를 해결하고 데이터 이동의 일관성을 위해 표준항목을 결정한다. 결정된 표준항목과 그리드 노드의 데이터베이스 스키마 정보를 표준항목에 매핑시키고 이질적인 문제를 해결하기 위해 요소 사이의 변환, 데이터 타입간의 변환정보를 생성한다. 이를 통해 표준 항목으로 그리드의 모든 노드를 접근할 수 있도록 한다. 그림 1의 ①에 해당한다.

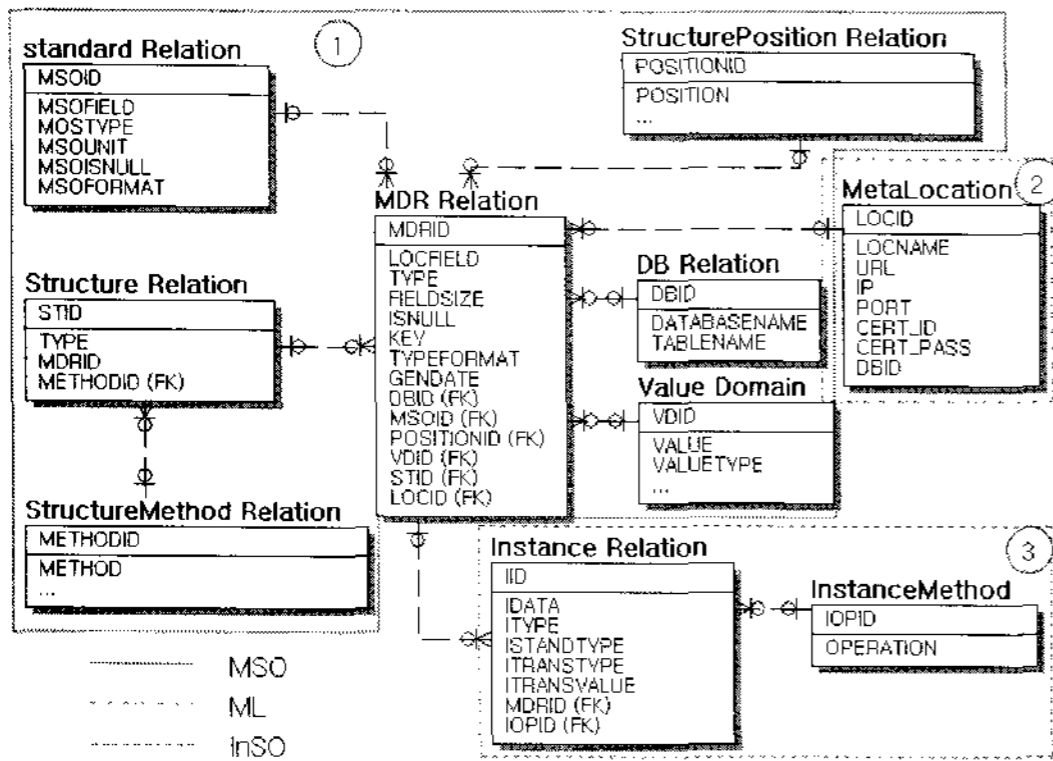


그림 1. XMDR 표현 모델
Fig 1. XMDR Presentation Model

2.2 메타 로케이션(ML:Meta Location)

MSO와 연계하여 각 노드의 DB 위치정보, 접근 권한 정보를 관리한다. 이는 XMDR 래퍼가 노드에 접근하기 위해 사용되는 정보로 글로벌 XMDR 쿼리 생성시 로케이션 정보로 추가되어 노드를 접근할 수 있도록 한다. 그림 1의 ②에 해당한다.

2.3 인스턴스 시멘틱 온톨로지 (InSO:Instance Semantic Ontology)

XMDR의 MSO에서 정의한 개념적 관계성에 대한 값들의 관계성을 정의한 것이다. 단순한 검색이 아니라 검색 키워드에 대한 유사성과 단위의 이질적인 표현을 동일하게 표현함으로써 검색에 대한 결과를 정확하도록 한다. ‘서울’을 검색한다면 서울의 고어나 다른 표현인 ‘한양’, ‘경성’ 등과 같은 데이터도 같이 검색하여 서울에 대한 정보를 더욱 확대하여 검색할 수 있도록 한다. 그림 1의 ③에 해당한다.

III. 제안시스템

3.1 개요

본 논문은 그리드의 각 노드에게 동일한 인터페이스와 동일한 질의를 제공하기 위해 XMDR을 이용한 XMDR 래퍼 시스템을 제안한다. 이질적인 데이터를 통합하기 위해 MDR을 확장시킨 XMDR을 기반으로 한다. 각 노드의 어플리케이션이 XMDR에 접근하여 정보를 제공하고 관리하기 위해서 XMDR Resource Manager (XRM)를 이용한다. 래퍼는 XRM에서 전송된 쿼리 정보와 표준항목을 이용하여 글로벌 XMDR 쿼리를 생성하고 이를 각 노드에 전송한다. 전송받은 노드의 래퍼는 이를 XMDR의 정보를 이용하여 로컬 쿼리로 변환하여 데이터베이스를 검색하고, 검색한 결과를 요구한 래퍼에게 전송함으로써 이질적인 환경에서 데이터를 통합할 수 있도록 하는 XMDR과 XMDR 래퍼를 제안한다.

본 시스템의 구성은 그림 2와 같이 크게 XMDR, XMDR 래퍼로 이루어진다. XMDR은 2장과 같이 데이터 통합을 위한 이질성 매핑 정보를 가진다. XMDR 래퍼는 쿼리 생성·변환 및 수행, XMDR의 Proxy DB에 결과 저장을 수행한다.

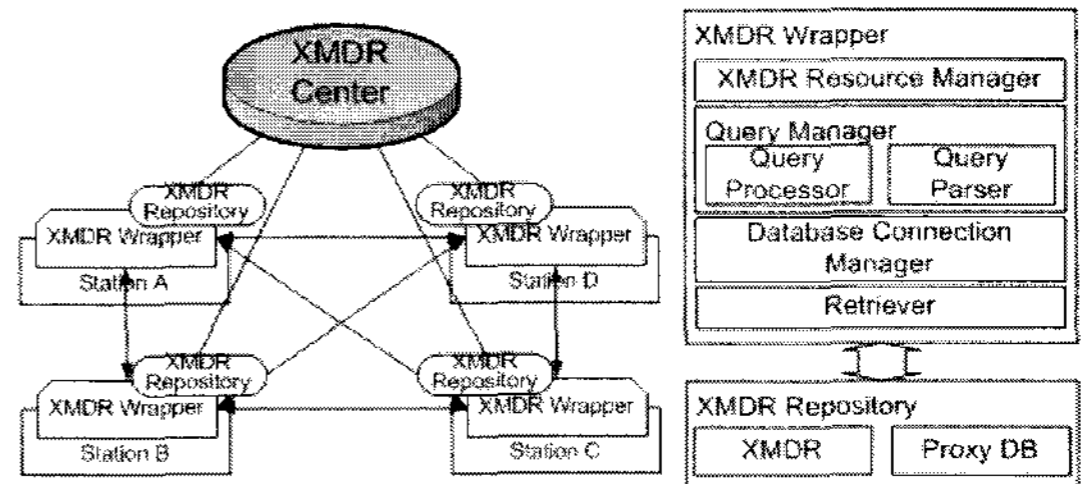


그림 2. XMDR 기반의 데이터그리드 래퍼 시스템
Fig 2. Data-Grid Wrapper System based on XMDR

3.1.1 XMDR 센터

XMDR 센터는 2장의 XMDR을 생성하고 생성된 XMDR을 각 노드의 래퍼에 전송해주는 역할을 수행한다. 주된 내용은 다음과 같다.

먼저 표준항목과 각 노드의 스키마 정보와 매핑하고 그 정보를 보관한다. 두 번째 각 노드의 로케이션 정보 및 접근 권한정보를 관리한다. 세 번째 각 노드의 데이터베이스에 저장된 데이터 즉, 인스턴스들의 의미적 관계

성을 추출하여 이를 InSO를 생성한다. 네 번째 각 노드의 상태정보를 파악하여 래퍼의 XRM의 요구에 따른 노드의 상태정보를 제공한다.

3.1.2 XMDR 래퍼

XMDR 래퍼는 크게 두가지 기능을 제공한다. 먼저 어플리케이션의 인터페이스 생성에 필요한 정보를 제공하고 사용자의 요구에 따른 글로벌 XMDR 쿼리를 생성하고, 이 쿼리를 지정된 노드에 전송하고 수신된 쿼리를 로컬 쿼리로 변환하여 그 결과를 요구된 노드로 전송하는 역할을 담당한다. 세부적인 각 요소의 역할은 다음과 같다.

- XMDR Resource Manager(XRM)

XRM은 서비스를 제공하기 위해 어플리케이션의 인터페이스 생성에 필요한 정보를 제공하고 각 노드의 상태정보를 XMDR 센터를 통해 각 노드의 상태를 파악하는 끊임없는 서비스이다.

- Query Manager(QM)

글로벌 XMDR 쿼리를 생성하고 다른 노드에서 전송된 글로벌 XMDR 쿼리를 로컬 쿼리로 변환하는 역할을 담당한다. 전자는 XRM에서 전송된 사용자 요구사항과 검색 노드에 접근 정보를 이용하여 표준 쿼리를 생성한다. 후자는 다른 노드에서 글로벌 XMDR 쿼리를 전송하면 이를 로컬 데이터베이스를 검색하기 위해 XMDR의 정보를 이용하여 로컬 쿼리로 변환하는 역할을 수행한다. 이에 대한 상세한 과정은 본장 4절에서 다룬다.

- Database Connection Manager(DCM)

수집된 결과를 XMDR을 PROXY 릴레이션에 저장하여 사용자에게 결과를 제공할 수 있도록 한다. 다른 노드에서 요구된 쿼리의 처리를 처리하고 처리된 결과

를 수집 및 변환을 수행하여 Retriever에 전달하는 역할을 담당한다.

- Retriever

생성된 글로벌 XMDR 쿼리를 지정된 노드에 전송하는 역할을 수행하고 다른 노드에서 요구된 쿼리를 변환할 수 있도록 QM에 전달하는 역할을 수행하고 처리된 결과를 요구된 노드에 실제적으로 전송하는 역할을 수행한다.

3.2 XMDR 래퍼의 운용

XMDR 래퍼는 클라이언트 측면과 서버 측면으로 운용된다. 래퍼는 쿼리 변환 생성 및 변환, 결과 형식 스키마 생성, DCM을 통한 스키마에 적합한 결과 생성이 주된 역할이다.

3.2.1 클라이언트 측면

클라이언트 측면이란 다른 노드를 검색하는 주체가 된다. 다른 노드를 검색하기 위한 글로벌 XMDR 쿼리를 생성하고 이를 Retriever를 통해 전송하고 검색된 결과를 DCM에서 XMDR의 Proxy 릴레이션에 저장하여 사용자에게 제공될 수 있도록 한다. 이를 상세하게 기술하면 아래와 같다. XRM에서 제공되는 표준정보, 사용자 요구조건, 노드 정보를 이용하여 글로벌 XMDR 쿼리를 생성한다. 생성된 글로벌 XMDR 쿼리와 함께 전송될 결과에 대한 XML Schema를 QM에서 생성한다. Retriever에서 글로벌 XMDR 쿼리를 각 노드에 전송한다. 각 노드에서 전송된 결과를 DCM를 통해 XMDR의 Proxy DB에 저장한다.

XRM을 통해 결과를 어플리케이션에 제공한다. 이러한 XMDR 래퍼의 클라이언트 측면의 수행과정은 그림 3과 같다.

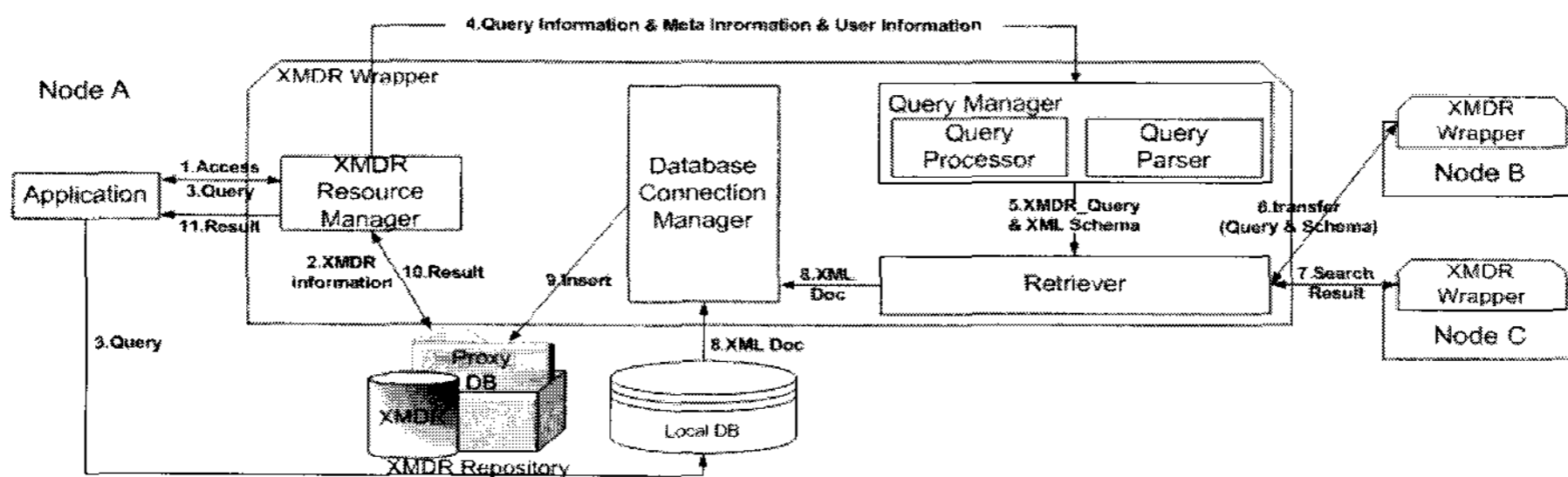


그림 3. 클라이언트 측면의 XMDR 래퍼의 작업 수행 과정
Fig 3. Work performs process of the client aspect of the XMDR wrapper

3.2.2 서버 측면일 경우

서버 측면이란 다른 노드에서 요구된 쿼리를 처리하는 것이 주체가 된다. 다른 노드에서 전송한 글로벌 XMDR 쿼리를 QM에서 로컬 쿼리로 변환하고 이를 수행하기 위해 DCM에 전달하고 DCM은 로컬 데이터베이스를 검색하여 수신된 XML Schema의 형식에 따른 결과를 생성하여 요구된 노드에 전송한다. 이를 상세하게 기술하면 아래와 같다.

수신된 글로벌 XMDR 쿼리를 로컬 쿼리로 변환한다. 변환은 XML 문서를 파싱하여 자신의 데이터베이스의 실제 스키마 구조와 제공되는 XMDR의 매핑 정보를 기반으로 로컬 데이터베이스를 검색하기 위한 쿼리를 생성한다. 생성된 쿼리는 DCM에 의해 로컬 데이터베이스를 검색하고 결과를 요구한 노드에 반환한다. 결과는 요구한 노드에서 제공한 XML Schema의 형식에 적합하게 생성된다. 결과 반환은 Retriever를 통해서 수행된다.

이러한 XMDR 래퍼의 서버 측면의 수행과정은 그림 4와 같다.

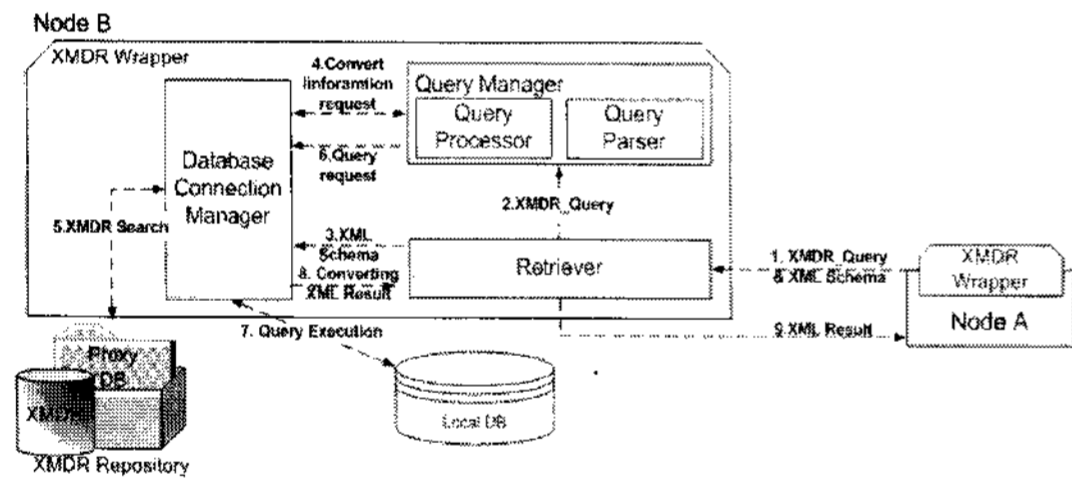


그림 4. 서버 측면의 XMDR 래퍼의 작업 수행
Fig 4. Work performs process of the server aspect of the XMDR wrapper

3.3 시스템 운용

글로벌 XMDR 쿼리는 사용자의 요구에 따른 표준 항목, 조건, 그리드 노드의 물리적 위치 및 접근권한을 포함하여 생성된다. 생성은 MSO의 표준 항목 정보, 접근 노드의 정보인 ML의 위치와 접근 권한을 통해 이루어진다.

생성되는 과정은 MSO의 표준 항목이 사용자 인터페이스에 제공되고 사용자의 선택에 의해 검색 요구 항목인 CONTENTS와 검색조건인 CONDITIONS이 생성된다. 또한 검색하고자 하는 노드도 사용자의 요구에 의한 선택으로 LOCATIONS에 접근 위치 정보와 접근 권한정

보가 포함된다. 이 일련의 작업은 XRM에 의해 수행되며 그림 5에 의해 확인할 수 있다.

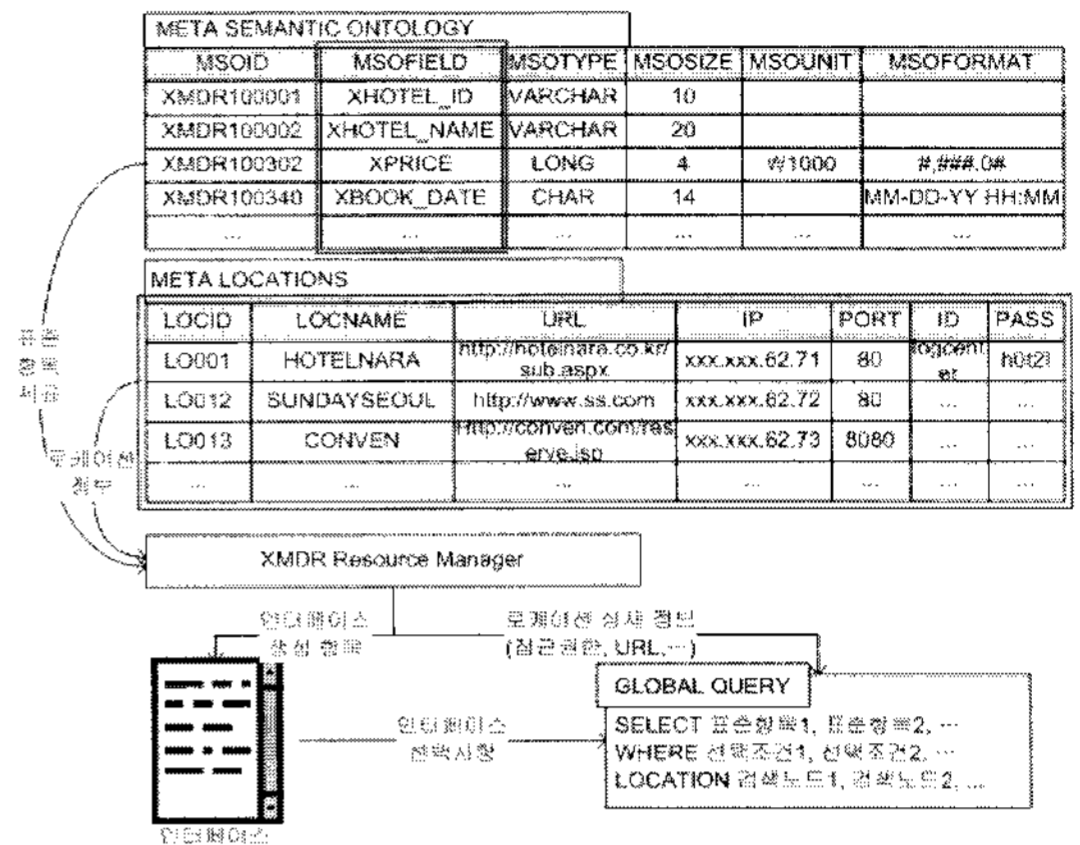


그림 5. 글로벌 XMDR 쿼리의 생성
Fig 5. Creation of Global XMDR Query

테이블과 관련된 정보는 노드에서 표준 항목에 대응되는 테이블과 연결되므로, 글로벌 XMDR 쿼리에서 테이블 정보는 포함되지 않는다. 이 테이블 정보는 로컬 쿼리로 변환될 때 표준 항목에 대응되는 노드의 항목과 테이블 명을 포함한다.

이렇게 생성되는 글로벌 XML 쿼리는 표 1과 같이 표현할 수 있으며 각 요소는 다음과 같다.

표 1. 생성된 글로벌 XMDR 쿼리
Table 1. Created Global XMDR Query

```
<?xml version="1.0" encoding="euc-kr"?>
<GLOBAL>
  <SQL>SELECT</SQL>
  <CONTENTS>
    <FIELD msoid="XMDR100002">XHOTEL_NAME</FIELD>
    <FIELD msoid="XMDR100003">XPRICE</FIELD>
    <FIELD msoid="XMDR100302">XSIZE</FIELD>
  </CONTENTS>
  <CONDITIONS>
    <WHERE msoid="XMDR100003">XPRICE=100,000</WHERE>
    <WHERE msoid="XMDR100003"
logical="AND">XPRICE<=150,000</WHERE>
  </CONDITIONS>
  <LOCATIONS>
    <NODE locid="LO001" url="http://hotelnara.co.kr/sub.aspx"
id="logcenter" pass="h0t2l">Hotelnara</NODE>
    <NODE locid="LO013" url="Http://conven.com/reserve.jsp" id="..."
pass="...">conven</NODE>
  </LOCATIONS>
</GLOBAL>
```

<GLOBAL>요소는 최상위 요소로 글로벌 XMDR 쿼리의 시작을 알리며, 이는 <SQL>요소, <CONTENTS>요소, <CONDITIONS>요소, <LOCATIONS>요소를 포함한다. <SQL>요소는 현재는 검색을 수행하기 때문에 SELECT로 사용하고 추후에 갱신, 삭제, 삽입을 위한 방안을 추가할 예정이다. <CONTENTS>요소는 검색된 결과에 해당하는 표준 항목들을 포함한다. 하위 요소로 <FIELD>요소를 포함하고 이 요소의 속성인 msoid는 MSO의 표준 항목의 식별자이다. 요소의 값은 해당 필드의 표준 항목명으로 실제 쿼리 변환에서 로컬 쿼리로 변환하기 위한 값이다.

<CONDITIONS>요소는 쿼리 조건 및 조인 조건을 표현한다. 이 요소의 하위 요소로 <WHERE>요소를 가진다. <WHERE>요소는 <FIELD>요소와 유사하지만 요소의 값으로 쿼리 조건을 가진다는 차이가 있다. <LOCATIONS>요소는 검색할 각 노드의 정보를 가지는 요소로, 그 하위요소로 <NODE>요소를 포함하고 노드에 대한 접근정보를 가진다. <NODE>요소는 여러 개가 정의 되지만 Retriever에서 각 노드로 분할되어 전송되게 하기 위한 정보가 된다.

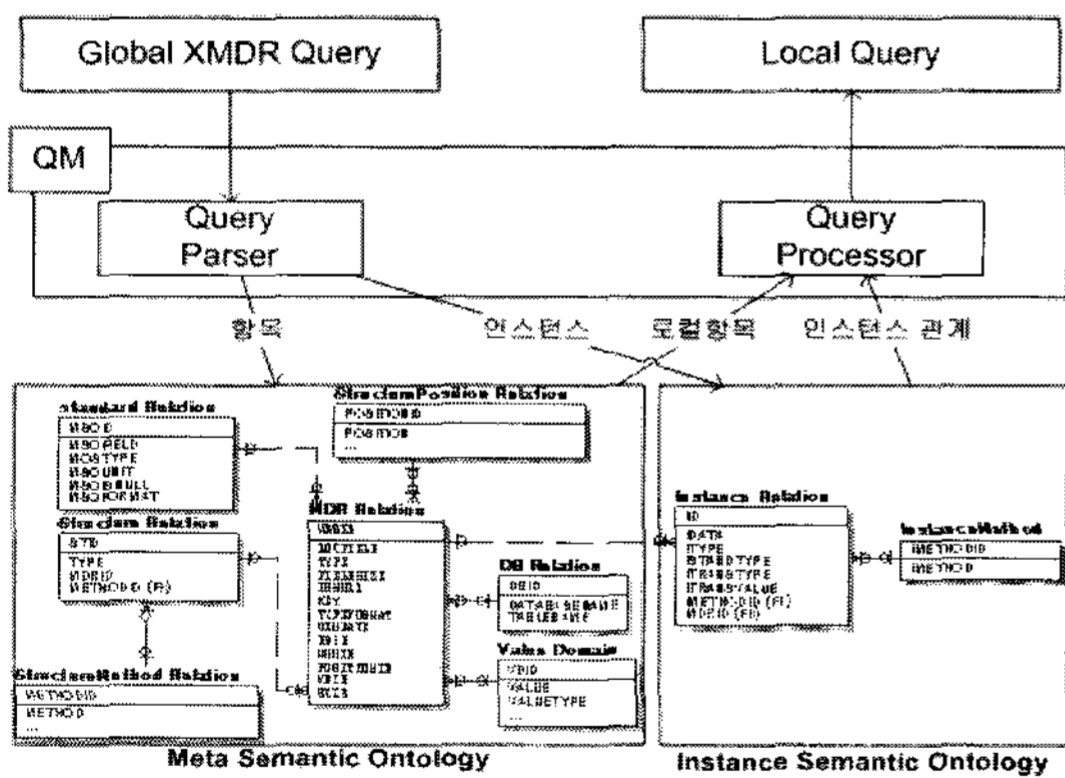


그림 6. 글로벌 XMDR 쿼리의 로컬 쿼리 변환
Fig 6. Global XMDR Query convert to Local Query

그림 6은 서버 측면에서 XMDR 래퍼가 글로벌 XMDR 쿼리를 로컬쿼리로 변환 과정이다. 변환은 QM과 XMDR 저장소의 정보를 이용한다. 쿼리 파서는 글로벌 XMDR 쿼리의 표준 항목을 XMDR의 매핑정보를 이용하여 로컬 항목으로 변환하고, 글로벌 XMDR 쿼리의 조건은 조건 수식을 그대로 이양되지만 조건 값은 InSO

를 통해 인스턴스 관계성을 이용하여 로컬 쿼리를 생성한다.

IV. 비교 및 시험평가

본 시스템은 데이터 교환 및 공유의 표준인 XMDR을 이용하여 데이터의 구문적, 구조적, 의미적 이질성을 해결하고, 데이터 교환은 XML 메시지를 이용함으로써 그리드상의 데이터 교환을 표준화 했다. 이에 따라 관련 시스템인 Grid-DBMS[12], 오라클 그리드[13]와 비교분석한 내용은 표 2와 같다[10].

표 2. 기존 그리드 시스템과 비교
Table 2. Comparison with the existing grid system

		Grid-DBMS	Oracle-Grid	본시스템
데이터 가용성	연속성	×	○	○
	접근성	○	×	○
	유연성	○	×	○
데이터 투명성	이질성극복	△	△	○
	데이터교환	△	○	○
데이터 활용	검색효율성	△	△	○

각 노드에서 가지고 있는 데이터는 XMDR 래퍼를 통해서 요구에 의해 수집되므로 데이터의 변경이 발생할 때 마다 갱신할 필요없이 XMDR 래퍼의 요구에 따라 조건에 적합한 데이터를 전송함으로써 데이터의 연속성을 보장한다. XMDR 래퍼의 설치와 노드에서 가지고 있는 스키마 정보와 XMDR과의 매핑만으로 노드를 추가할 수 있어 접근성과 유연성이 보장된다. 이질성 문제는 XMDR의 MSO를 이용하여 극복한다. 데이터의 교환은 모든 시스템이 가능한 XML 메시지를 전송한다. 검색의 효율성을 위해서 InSO를 이용하여 연관성을 가지는 데이터를 포함하여 검색 질의를 수행한다.

본 시스템에서 여행정보 사이트 정보를 바탕으로 구축한 시스템을 기준으로 처리 시간을 평가한다. 테스트는 4개의 DB서버에 테스트용 데이터베이스를 구축하고 약 20,000건 결과를 가지는 질의문을 적용하였고, 데이터 이동을 위해 생성된 XML 크기는 3,663KB의 크기를 가지고 있다.

각 DB서버에서 처리된 시간은 표 3과 같다.

표 3. 각 DB서버에서 처리시간
table 3. processing time at each DB server

시간 \ ip	71	72	73	74
그리드노드	45ms	65ms	85ms	33ms
래퍼	2057ms	1356ms	1698ms	1972ms

쿼리 처리시간은 래퍼에서 소요되는 시간과 그리드 상의 각 노드에서 처리되는 시간으로 구분하여 테스트 한다. 래퍼에서 처리되는 시간은 클라이언트의 래퍼가 사용자의 요구를 입력받아 글로벌 XMDR 쿼리를 생성하고, 서버의 래퍼가 글로벌 XMDR 쿼리를 로컬쿼리로 변환하는데 소요되는 시간이다. 그리드의 각 노드에서 처리되는 시간은 네트워크에서 소요되는 시간과 각 노드에서 처리하는데 필요한 시간이다. 즉, 그리드 상에서 소요되는 시간은 네트워크와 각 노드의 DBMS에 의해 이루어진다. 래퍼에서 소요되는 시간은 XMDR을 접근하는데 소요되는 시간으로 처리에 걸리는 시간은 무시될 정도로 적은 시간이 소요된다는 것을 테스트로 확인 된다. 이 테스트의 결과는 그림 7과 같다.

테스트한 결과 래퍼에서 쿼리 처리에 소요되는 시간은 거의 일정하고, 래퍼의 처리 시간은 전체 처리 시간에 비해 미약하다. 그러므로 XMDR을 이용한 통합 질의 방식을 이용하는 것이 기존 시스템 보다 처리시간이나 사용방법에 비해 효율적이라 할 수 있다.

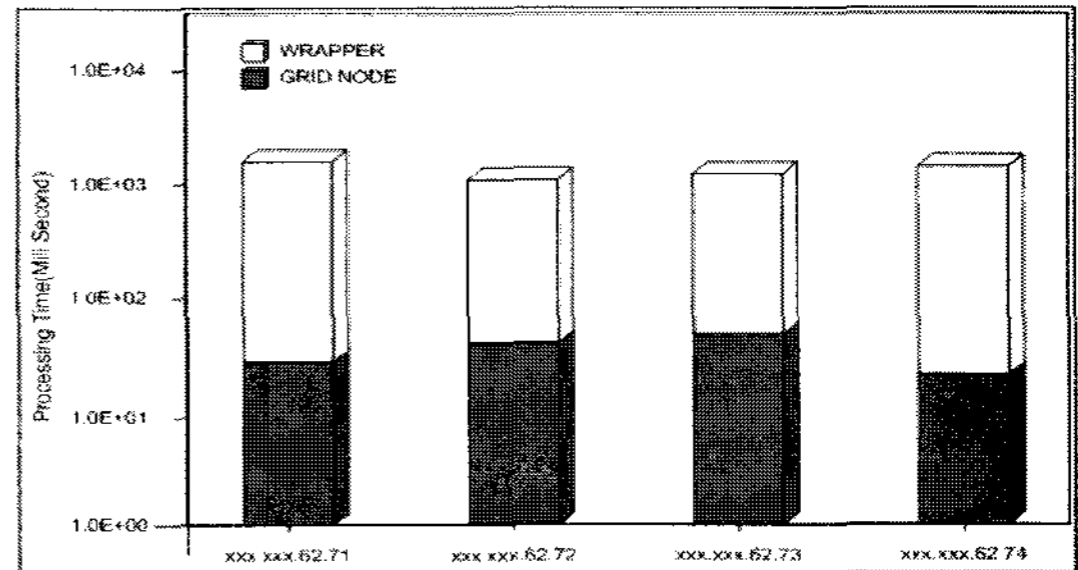


그림 7. 질의 처리 시간 비교
Fig 7. Comparison of query processing time

V. 시스템수행

본 시스템 수행은 Visual Basic 6.0, MSSQL 2000 Server, XML, Windows 2003 Server 환경에서 수행되었으며, 여행정보 검색을 적용사례로 한다.

그림 8은 사용자에게 제공되는 클라이언트 인터페이스이다. 사용자가 요구조건, 요구항목, 검색노드를 선택하여 글로벌 XMDR 쿼리가 생성되고, 이에 검색을 수행하면 각 노드에 생성된 글로벌 XMDR 쿼리가 전송된다. 선택된 노드는 로컬쿼리로 쿼리를 변환하여 로컬 데이터 베이스를 검색하여 요구조건에 따른 결과가 XML 형식으로 산출되며, 산출된 XML 문서는 클라이언트가 되는 노드의 Proxy DB에 누적되게 된다. Proxy DB에 누적된 결과는 사용자의 클라이언트 인터페이스에 출력되며, 각 노드의 상태에 따라 추가적으로 Proxy DB에 삽입된다. 추가된 결과는 새로고침을 통해서 사용자가 확인할

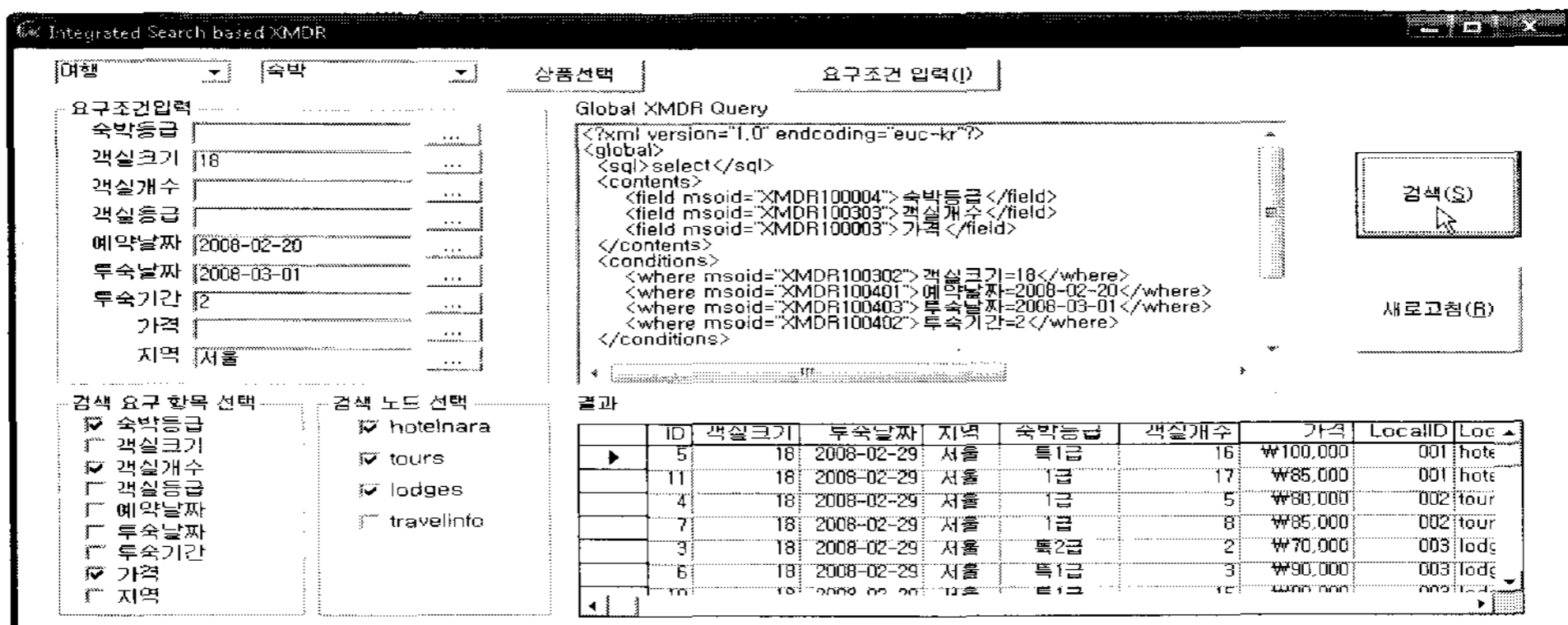


그림 8. 통합 인터페이스 사례
Fig 8. Example of integrated interface

수 있다.

Proxy DB에 쌓인 결과는 사용자의 식별자에 따른 검색정보로 사용자의 한번 검색으로 추가적인 작업없이 확인할 수 있다. 추가된 정보를 확인하기 위해서 새로고침을 통해 다른 추가적인 결과를 확인할 수 있다.

VI. 결과 및 향후연구

본 논문은 분산 데이터의 상호운용을 위한 이질성 극복을 위해 데이터 교환과 공유를 위한 표준 명세인 MDR을 바탕으로 확장된 XMDR을 설계 및 구축하고, 분산 시스템 프레임워크의 이질성 극복을 위해 XML기반의 메시지 교환을 위한 XMDR 래퍼 시스템의 설계 방안을 제안하였다.

특히 XMDR을 통한 표준 항목을 사용자에게 제시함으로써 실제 데이터들의 이질적 환경과는 관계없이 통합된 환경을 제시할 수 있다.

제안하는 본 논문의 특징은 다음과 같다. DBMS의 이질성 문제를 해결하기 위한 XMDR을 제안했다. 둘째, XMDR 래퍼가 XMDR의 MSO를 이용하여 스키마의 이질성을 해결뿐만 아니라 InSO를 통한 값의 의미에 따른 이질성도 고려함으로써 중복된 질의를 수행하지 않아도 된다. 셋째, XMDR을 통해 단일 인터페이스를 사용할 수 있으며, 래퍼의 처리시간이 미약하여 기존 시스템 보다 효율적이고 편리하다. 넷째, 각 노드는 ML을 통해 자신의 위치 투명성을 제공함으로써 사용자는 노드의 위치정보를 알지 못하더라도 접근할 수 있게 한다.

이후는 현재 시스템을 검색뿐만 아니라 상호운용이 가능하도록 확장해야 하며, 인스턴스의 관계성을 확장하여 단순 검색에서 대체안을 제공할 수 있는 지식관리 부분으로의 확장 연구가 필요하다. 또한 데이터 웨어하우스와 같이 결과를 분석하여 의사결정 지원에 확장을 위한 연구가 필요하다.

참고문헌

- [1] Ian Foster, Carl Kesselman, "The Grid2:Blueprint for a New Computing Infrastructure", Morgan Kaufmann, 2004.
- [2] Open Grid Services Architecture Data Access and Integration (OGSA-DAI). <http://www.ogsa-dai.org/>.
- [3] N. P. Chue et al. Grid data service specification: The relational realisation. The GGF9, 2003.
- [4] N. P. Chue et al. Grid data service specification: The XML realisation. The GGF9, 2003.
- [5] Said Mirza Pahlevi, Isao Kojima, "OGSA-WebDB: An OGSA-Based System for Bringing Web Databases into the Grid", IEEE Proceedings of the International Conference on Information Technology: Coding and Computing, 2004.
- [6] Ronald R. Yager, Frederick E. Petry, "A Multicriteria Approach to Data Summarization Using Concept Ontologies", IEEE TRANSACTIONS ON FUZZY SYSTEMS, VOL. 14, NO. 6, DECEMBER 2006.
- [7] An Experience Report on Designing and Building OGSA-DQP. A Service based Distributed Query Processor for the Grid]
- [8] Extensible Markup Language, <http://www.w3.org/XML>.
- [9] 정계동, 황치곤, 최영근, "분산 환경에서 XMDR을 이용한 예약 정보 시스템", 한국해양정보통신학회 논문지 Vol.11 No.10 pp.1957-1967, 2007
- [10] 국윤규, 하이브리드 에이전트에 의한 효율적인 데이터 그리드 시스템, 광운대학교 박사학위 논문, 2006.
- [11] Kevin D. Keck and John L. McCarthy, "XMDR: Proposed Prototype Architecture Version 1.01", <http://www.xmdr.org>, February 3, 2005
- [12] G Aloisio, M Cafaro, S Fiore, M Mirto, "The Grid-DBMS:Towards Dynamic Data Management in Grid Environments", Information Technology: Coding and Computing(ITCC'05),IEEE, Vol. 2, pp.199-204, April 2005.
- [13] Matthew Hart, Scott Jesse, "Oracle Database 10g:High Availability with RAC, Flashback & Data Guard", Oracle Press, 2005.



황 치 곤(Chi-gon Hwang)

1995년 창원대학교 경영학과(학사)
2004년 광운대학교 정보통신학과
(공학석사)

2008년 ~ 현재 광운대학교 컴퓨터과학 박사과정
2006년 ~ 현재 전자넷 연구원
※관심분야: 웹서비스, XMDR, 그리드컴퓨팅, 이동에이전트, 상호운용



정 계 동(Kye-dong Jung)

1985년 광운대학교 전자계산학
(이학사)
1992년 광운대학교 산업정보학
(이학석사)

2000년 광운대학교 컴퓨터과학 (이학박사)
1993년 ~ 2004년 광운대학교 정보과학원 교수
2005년 ~ 현재 광운대학교 교양학부 교수
※관심분야: XML 분산시스템, 분산 컴퓨팅기술, 이동에이전트



최 영 근(Young-keun Choi)

1980년 서울대학교 수학교육과
(이학사)
1982년 서울대학교 계산통계학과
(이학석사)

1989년 서울대학교 계산통계학과 (이학박사)
1982년 ~ 현재 광운대학교 컴퓨터과학과 교수
1992년 ~ 2000 광운대학교 전산정보원 원장
2002년 ~ 2005 광운대학교 교무연구처장
※관심분야: 객체지향 설계, 분산시스템, 이동에이전트, 상호운용