오디세우스/Parallel-OOSQL: 오디세우스 객체 관계형 데이터베이스 관리 시스템을 사용한 병렬 정보 검색 시스템

유재준[0] 이재길 이민재 황규영
한국과학기술원 전자전산학과 전산학 전공/첨단정보기술연구센터
{jjryu, ariel, mjlee, kywhang}@mozart.kaist.ac.kr

ODYSSEUS/Parallel-OOSQL: A Parallel Information Retrieval System Using the Odysseus Object-Relational Database Management System

Jae-Jun Yoo[0] Jae-Gil Lee Min-Jae Lee Kyu-Young Whang
Department of Electrical Engineering & Computer Science
Division of Computer Science and
Advanced Information Technology Research Center
Korea Advanced Institute of Science and Technology

요 약

인터넷의 성장과 함께 전자적인 형태로 표현되는 정보의 양이 급격하게 증가함에 따라, 문서를 병렬적으로 검색하는 병렬 정보 검색이 많은 양의 문서에 대한 빠른 검색을 지원하는 것에 있어 더욱 중요한 역할을 하고 있다. 병렬 정보 검색 시스템을 구현하기 위해서는 역 색인을 분할하고 분할된 역 색인을 병렬적으로 검색하는 것이 필요하다. 역 색인을 분할하는 방법으로는 다음과 같은 두 가지 방법이 있다: 1) 문서 식별자를 기반으로 하는 분할 방법과 2) 키워드 식별자를 기반으로 하는 분할 방법. 그러나 각 방법은 단점들을 가지고 있다. 본 논문에서는 정보 검색 기능이 밀결합된 데이터베이스 관리 시스템인 오디세우스를 사용하여 병렬 정보 검색 시스템을 설계하고 구현한다. 첫째로, 기존의 역 색인 분할 방법을 분석하고, 각 분할 방법의 단점들을 보완할 수 있는 혼합 분할 방법을 제안한다. 둘째로, 많은 양의 문서에 대해 성능 저하의 원인이 되는 대형 포스팅을 분할 하는 방법을 제안한다. 마지막으로 제안된 시스템의 유용성을 보이기 위해 실험을 수행한다. 예제 데이터베이스로서는 이백만 건의 웹 페이지를 사용한다. 실험 결과, 질의 처리 시간이 역 색인 분할의 블록의 개수에 근사하게 비례하여 줄어들고 시스템이 좋은 확장성을 가짐을 보인다.

1. 서 론

최근에 인터넷이 널리 보급되어 전자 문서로 표현된 정보의 양이 급격히 늘어남에 따라 많은 양의 문서에 대한 정보 검색을 효율적으로 지원해 주는 것이 더욱 중요해 지고 있다[1, 2]. 그러나 기존의 단일 시스템으로는 빠른 속도로 증가하는 대용량의 정보에 대한 효율적인 검색을 지원에 주는 것에 한계가 있으며, 이러한 한계를 극복하기 위해 병렬 정보 검색이 연구되어 왔다[3].

병렬 정보 검색은 정보 검색 시스템에서 주어진 키워드가 발생한 문서들을 찾기 위하여 사용하는 색인을 병렬적으로 검색 하는 방식으로 구현된다[1]. 여러 가지 정보 검색 색인 구조들 중 가장 뛰어난 성능을 가진 것으로 알려져 있는 역 색인(Inverted Index) 구조[4]를 사용하는 경우, 병렬 정보 검색은 역 색인을 분할하고 분할의 각 블록(Block)을 병렬적으로 검색함으로써 가능하다.

병렬 정보 검색을 구현하기 위해 역 색인을 분할하는 방법은 크게 문서 식별자 분할 방법과 키워드 식별자 분할 방법으로 나누어 진다[5]. 문서 식별자 분할 방법은 같은 문서에 대한 역 색인의 내용이 같은 블록에 존재하도록 역 색인을 분할하는 방법이고, 키워드 식별자 분할 방법은 같은 키워드에 대한 역 색인의 내용이 같은 블록에 존재하도록 역 색인을 분할하는 방법이다.

문서 식별자 분할 방법의 경우 키워드 검색을 위하여 모든 역 색인 블록을 검색해야 하므로 디스크 입/출력 횟수가 많아지고, 각 역 색인 블록들이 모든 키워드들을 저장하므로 키워드 저장을 위한 공간을 많이 필요로 한다는 단점이 있으며, 키워드 식별자 분할 방법의 경우 키워드 검색이 하나의 역 색인 블록에서만 일어나게 되므로 질의 처리 시간이 검색하고자 하는 키워드들 중 가장 큰 포스팅을 가지는 키워드의 검색 시간에 의해 결정되고, 키워드에 따라 역 색인 블록이 색인하는 포스팅 수의 불균형이 일어날 수 있어 시스템 성능 저하를 일으킬 수 있다는 단점이 있다. 현재까지의 대부분의 병렬 정보 검색 시스템은 역 색인 분할 방법으로서 문서 식별자 분할 방법 또는 키워드 식별자 분할 방법만을 사용하므로 이와 같은 단점들이 그대로 나타나고 있다.

* 본 연구는 첨단정보기술연구센터를 통하여 한국과학재단의 지원을 받았음.

본 논문에서는 한국과학기술원 데이터베이스 및 멀티미디어 연구실에서 개발한 정보 검색 기능이 밀결합된 객체 관계형 데이터베이스 관리 시스템인 오디세우스[6]를 사용하여 병렬 정보 검색 시스템을 설계하고 구현한다. 역 색인 분할 방법은 문서 식별자 분할 방법, 키워드 식별자 분할 방법을 사용하며 각 분할 방법을 혼합 사용하는 방법을 제안한다. 또한 키워드 식별자 분할 방법에서 가장 큰 포스팅을 가지는 키워드의 검색 시간에 의해 질의 처리 시간이 결정되는 문제를 해결하기 위한 방법을 제안한다. 그리고 구현된 병렬 정보 검색의 유용성을 보이기 위하여 단일 정보 검색 시스템과의 성능 평가를 수행한다. 실험 결과 많은 양의 문서에 대하여 병렬 정보 검색 시스템이 단일 정보 검색 시스템에 비해 역 색인 블록의 개수에 근사하게 비례하여 빠르게 정보 검색을 실행함을 보인다.

본 논문의 구성은 다음과 같다. 제 2 장에서는 연구되어온 정보 검색의 병렬화 방법에 대해서 설명하고, 구현 사례를 소개한다. 제 3 장에서는 오디세우스 데이터베이스 관리 시스템의 정보 검색 기능에 대해서 설명한다. 제 4 장에서는 기존의 역 색인 분할 방법들의 단점을 보완하기 위한 새로운 역 색인 분할 방법을 제안한다. 제 5 장에서는 오디세우스를 이용한 병렬 정보 검색 시스템의 설계 및 구현 방법에 대해서 설명한다. 제 6 장에서는 실험을 통하여 구현된 병렬 정보 검색 시스템의 성능을 측정하고, 결과를 분석한다. 마지막으로 제 7 장에서는 결론을 내린다.

2. 관련 연구

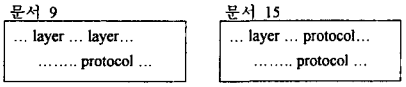
본 장에서는 관련 연구로서 제 2.1 절에서는 정보 검색 시스템에서 가장 널리 쓰이는 역 색인 구조에 대해서 설명한다. 제 2.2 절에서는 역 색인 구조를 분할하는 방법에 대해서 설명하고, 제 2.3 절에서는 병렬 정보 검색 시스템의 구축 사례를 알아 본다.

2.1 역 색인 구조

정보 검색 시스템은 주어진 키워드를 포함하는 문서들을 찾기 위해서 색인을 사용한다[4]. 정보 검색을 위해 사용되는 여러 색인 구조들 중

에서 역 색인 구조가 가장 뛰어난 성능을 가짐이 알려져 있으며[4] 이로 인해 오디세우스를 포함한 대부분의 정보 검색 시스템은 역 색인 구조를 사용한다.

역 색인은 각 문서에 나타난 키워드와 키워드가 나타나는 위치에 대한 정보를 가지며, 크게 사전 파일과 포스팅 파일로 구성된다[4]. 사전 파일은 문서에 나타난 키워드, 키워드의 식별자, 키워드의 포스팅 개수 등을 유지하며, 포스팅 파일은 해당 키워드가 발생한 정보를 포스팅들의 리스트로써 유지한다. 포스팅은 키워드가 발생한 문서의 식별자(document identifier:docID)와 그 문서 내에서의 발생 위치 정보를 유지한다. 예를 들어, 그림 1(a)에서와 같은 문서 9 와 문서 15 에 대해 역 색인을 구축하는 경우 1(b)와 같은 역 색인을 얻는다.



(a) 예제 문서

사전 파일 / 포스팅 파일

키워드	키워드 식별자	포스팅 개수	포스팅 리스트 (DocID, Position)
layer	10	3	(9,2), (9,10), (15,4)
protocol	45	3	(9,8), (15,12), (15,30)

(b) 예제 문서의 역 색인

그림 1. 역 색인 예제.

2.2 역 색인 분할 방법

병렬 정보 검색을 구현하기 위해 역 색인을 분할하는 방법은 크게 문서 식별자 분할 방법과 키워드 식별자 분할 방법으로 나누어 진다

문서 식별자 분할 방법은 같은 문서 식별자에 대한 포스팅 리스트는 같은 역 색인 블록에 포함되도록 분할하는 방법이다[7, 5]. 그러므로 특정 키워드의 검색은 각 역 색인 블록에서 요구된 키워드를 검색하여 얻어진 결과들을 병합함으로써 수행된다. 이 경우, 키워드 검색을 위하여 모든 역 색인 블록을 검색하게 되므로 키워드 식별자 분할 방법에 비해 디스크 입/출력 횟수가 같거나 많으며, 또한 각 역 색인 블록에 대하여 모든 키워드가 저장되므로 키워드 저장을 위한 많은 저장 공간을 필요로 한다는 단점이 있다.

키워드 식별자 분할 방법은 같은 키워드 식별자에 대한 포스팅 리스트는 같은 역 색인 블록에 포함되도록 분할하는 방법이다[7, 5]. 그러므로 특정 키워드의 검색은 특정 키워드에 대한 포스팅을 가지는 역 색인 블록만을 검색함으로써 수행된다. 이 경우, 키워드 검색을 위하여 하나의 역 색은 블록만을 검색하므로 검색 시간이 가장 많은 포스팅을 가지는 키워드의 검색 시간에 의해 결정되며, 또한 키워드에 따라 각 역 색인 블록이 색인하는 포스팅 수의 불균형이 이루어지기 쉬워 시스템의 성능 저하를 일으킬 수 있다는 단점이 있다.

2.3 구현 사례 소개

■ **PLIERS**

PLIERS(Parallel Information Retrieval System using MPI)는 MPI(Message Passing Interface)를 사용하여 무공유(shared nothing)구조로 병렬 정보 검색 시스템을 구현하였다. 역 색인 분할 방법으로서 문서 식별자 분할 방법과 키워드 식별자 분할 방법을 개별적으로 지원할 수 있도록 구현되었다[7].

■ **ORACLE**

Oracle 을 사용한 정보 검색은 ConText 라는 카트리지를 Oracle 에 추가 함으로써 가능하다[8]. 병렬 정보 검색은 Oracle 데이터베이스 시스템을 병렬화 하는데 사용되는 Oracle Parallel Server 를 사용 함으로써 구축 할 수 있다. Oracle 에서는 테이블 분할이라는 개념을 사용하여 문서 식별자 분할 방법을 이용한 병렬 정보 검색 시스템을 구축 할 수 있다[8].

3. 오디세우스의 소개

오디세우스는 한국과학기술원 데이터베이스 및 멀티미디어 연구실에서 개발한 정보 검색이 밀결합된 객체 관계형 데이터베이스 관리 시스템이다[6]. 데이터베이스 관리 시스템(DBMS)과 정보 검색의 밀결합이란 데이터베이스 관리 시스템을 확장하여 시스템의 엔진 수준에서 정보 검색 기능을 가지도록 하는 것을 말한다. 오디세우스는 정보 검색을 위해 많은 양의 문서를 저장하려는 경우, 배치작업을 통해 빠르게 데이터베이스를 구축할 수 있는 벌크 로딩(Bulkloading) 기능을 제공한다[9].

오디세우스는 SQL 에 객체 지향 개념과 정보 검색 기능을 추가한 OOSQL 을 통하여 SELECT 질의와 match 라는 내장 함수를 제공함으로써 문서 검색 기능을 지원한다. OOSQL 에서는 INSERT, DELETE, UPDATE 질의를 통하여 정보 검색 시스템으로의 문서 추가, 삭제, 변경을 위한 기능도 제공하고 있으나, 본 논문에서는 검색 기능에 대해서만 고려하도록 한다.

4. 새로운 역 색인 분할 방법

기존의 역 색인 분할 방법의 단점들을 보완하기 위하여 본 논문에서 제안하는 혼합 분할 방법에서는 문서 식별자 분할 방법에 의해 만들어진 각 역 색인 블록에 대해서 키워드 식별자 분할 방법을 다시 적용한다. 그러므로 각 역 색인 블록은 특정 문서의 특정 키워드들에 대한 포스팅 리스트들을 색인하게 된다

이러한 분할 방법을 사용하면 키워드 검색을 일부의 역 색인 블록만을 검색하므로 문서 식별자 분할 방법에서 보다 디스크 입/출력 횟수를 줄일 수 있으며, 특정 키워드는 일부의 역 색인 블록에서만 색인되므로 키워드 저장을 위해 필요한 공간을 줄일 수 있다. 또한 특정 키워드에 대한 포스팅 리스트들이 여러 역 색인 블록에 나누어져 색인되므로 키워드 식별자 분할 방법에서 보다 질의 처리 시간을 결정하는 키워드의 검색 시간을 줄일 수 있고, 키워드에 따른 역 색인 블록이 색인하는 포스팅 수의 불균형을 감소시킬 수 있다.

혼합 분할 방법을 사용하여도 여전히 남는 문제점은 질의 처리 시간이 "한국"과 같은 대형 포스팅을 가지는 키워드의 검색 시간에 의해 결정된다는 것이다. 본 논문에서는 이러한 문제를 해결하기 위해 특정 문서 블록 내에서의 특정 키워드의 포스팅 리스트가 대형화되는 경우, 해당 키워드를 모든 역 색인 블록에서 병렬적으로 검색할 수 있도록 포스팅들을 모든 역 색인 블록들에 균등하게 분배하도록 한다.

5. 병렬 정보 검색 시스템의 설계 및 구현

본 장에서는 오디세우스를 이용한 병렬 정보 검색 시스템의 설계 및 구현에 대해서 설명한다. 제 5.1 절에서는 구현된 병렬 정보 검색 시스템의 아키텍처를 설명한다. 제 5.2 절에서는 각 역 색인 분할 방법을 사용하여 병렬 정보 검색 시스템을 구축하는 방법을 설명한다. 제 5.3 절에서는 병렬 정보 검색 질의의 처리 과정을 설명한다.

5.1 시스템 아키텍처

본 논문을 통하여 구현된 병렬 정보 검색 시스템은 그림 2 에서와 같이 사용자로부터의 질의를 분석하는 부분인 마스터(master) 오디세우스와 실제로 각 블록을 검색하는 부분인 슬레이브(slave) 오디세우스로 나누어 진다. 마스터 오디세우스는 사용자로부터 받은 질의를 분석하여 검색을 해야 하는 역 색인 블록을 결정하며, 각 슬레이브 오디세우스가 처리해야 하는 질의를 전달하여 해당 슬레이브 오디세우스가 역 색인 블록을 검색하도록 한다. 슬레이브 오디세우스는 마스터 오디세우스로부터 받은 질의를 처리함으로써 할당된 역 색인 블록을 검색하고 검색 결과로서 처리 결과를 마스터 오디세우스에게 돌려준다. 마스터 오디세우스와 슬레이브 오디세우스는 슬레이브 오디세우스가 처리해야 하는 질의와 슬레이브 오디세우스의 질의 실행 결과를 마스터 오디세우스에게 전달하기 위해 RPC(Remote Procedure Call)를 사용하여 통신한다.

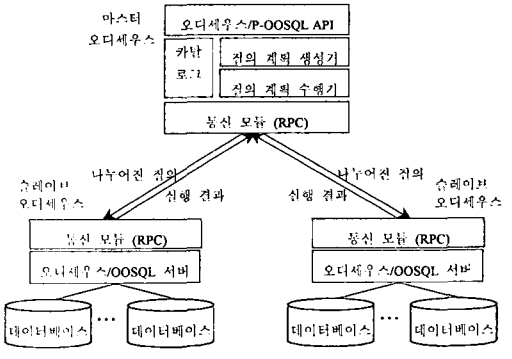


그림 2. 시스템 아키텍처.

5.2 역 색인의 분할 구축

각 역 색인 분할 방법을 사용하여 병렬 정보 검색 시스템을 구축하는 것은 그림 3 과 같이 병렬 검색 시스템에 저장하고자 하는 문서들과 문서들로부터 키워드 포스팅들을 추출한 결과를 각 역 색인 분할 방법에 따라 분배하고, 분배된 문서 및 키워드 포스팅들을 각 슬레이브 오디세우스에 전달하여 오디세우스의 정보 검색 시스템 구축 과정을 통하여 구축함으로써 이루어 질 수 있다.

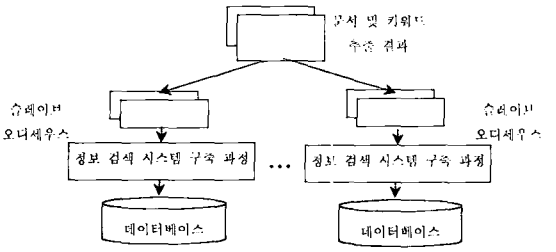


그림 3. 병렬 정보 검색 시스템 구축 과정.

5.3 질의 처리

병렬 질의 처리는 마스터 오디세우스에서 사용자로부터 받은 질의를 분석하고, 역 색인 분할 정보에 따라 질의를 나누어 각 슬레이브 오디세우스에게 전달하고, 슬레이브 오디세우스로부터의 질의 처리 결과를 돌려 받아 최종 결과를 얻기 위한 후처리를 함으로서 이루어지게 된다.

병렬 질의 처리 효율을 높이기 위하여 마스터 오디세우스에서 슬레이브 오디세우스로 전달되는 질의의 수와 슬레이브 오디세우스에서 마스터 오디세우스로 전달되는 결과의 크기가 가능한 작게 되도록 질의 실행 계획을 최적화하는 것이 필요하다. 이를 위해 다음과 같은 최적화 규칙을 사용한다.

■ **최적화 규칙**

검색 조건이 '*조건 1 OR 조건 2 AND 조건 3*'의 형태이고, '*조건 3*'이 '*조건 1*' 또는 '*조건 2*'와 같은 슬레이브 오디세우스에서 실행될 수 있으면, 검색 조건을 동치인 '*(조건 1 AND 조건 3) OR (조건 2 AND 조건 3)*'으로 변형하여 질의 실행 계획을 생성한다.

위와 같이 조건 수식을 변형하면 '*조건 1 AND 조건 3*' 또는 '*조건 2 AND 조건 3*'을 각각 하나의 질의로 처리할 수 있어 슬레이브 오디세우스가 실행해야 하는 질의의 수를 줄일 수 있다.

6. 실 험

본 장에서는 본 논문에서 구현한 병렬 정보 검색 시스템의 성능 평가를 하고 결과를 제시한다. 제 6.1 절에서는 실험 방법에 대해서 설명하고, 제 6.2 절에서는 구축된 병렬 정보 검색 시스템에 대한 실험 결과를 보이고 결과를 분석한다.

6.1 실험 방법

본 실험에서는 단일 정보 검색 시스템과 본 논문을 통하여 구현된 병렬 정보 검색 시스템에서의 질의 처리 시간을 측정하여 비교한다. 실험은 실험 데이터에서 가장 많은 수의 포스팅을 가지는 키워드들로 구성된 질의를 이용한다. 실험을 위한 데이터는 웹 로봇을 통해 얻은 이백만 건의 웹 페이지이며, 사용된 데이터의 스키마는 웹 페이지의 제목과 내용을 나타내는 두 개의 컬럼으로 구성된다. 실험에서 사용한 질의와 질의 결과 건수는 표 1 과 같다.

번호	예제 질의	결과수
1	"사람"이 포함된 문서를 검색	350094
2	"copyright"가 포함된 문서를 검색	323534
3	"경우"가 포함된 문서를 검색	299659
4	"사람"과 "copyright"가 포함된 문서를 검색	48339
5	"사람" 또는 "copyright"가 포함된 문서를 검색	625289
6	"사람"과 "copyright" 또는 "경우"가 포함된 문서를 검색	333276

표 1. 예제 질의.

실험은 600MHz ~ 800MHz 의 중앙 처리 장치 속도와 128M 의 메모리, 60G(54 00 RPM)의 하드 디스크를 갖는 다섯 대의 PC 들에서 Red Hat Linux (Version 7.1)를 운영체제로 하여 수행한다.

6.2 실험 결과

본 절에서는 본 논문을 통하여 구현된 구축한 병렬 정보 검색 시스템과 단일 정보 검색 시스템의 성능을 비교한다.

그림 4 는 역 색인 블록 수의 증가에 따른 질의 처리 시간의 변화를 나타낸 것이다. 그래프의 가로축은 역 색인 블록의 수를 나타내며, 세로축은 단일 정보 검색 시스템에서와 병렬 정보 검색 시스템에서의 질의 처리 시간의 비율로써, 단일 정보 검색 시스템에서의 질의 처리 시간에 비하여 병렬 정보 검색 시스템에서의 질의 처리 시간이 향상된 정도를 나타낸다. 병렬 정보 검색 시스템에서의 질의 처리 시간에는 마스터 오디세우스와 슬레이브 오디세우스 사이의 통신 시간이 포함되어 있다. 그래프는 역 색인 블록의 수를 증가시킬 수록 질의 처리 시간이 역 색인 블록의 수에 근사하게 비례하여 향상됨을 보여준다. 예를 들어 예제 질의 6 에 대하여, 역 색인 블록의 수가 한 개인 경우 처리 시간이 1695ms 이고, 역 색인 블록의 수가 세 개인 경우 처리 시간이 625ms 로써 단일 정보 검색 시스템에서보다 약 2.7 배 빠르게 처리된다. 따라서, 최적화된 분할을 사용하는 병렬 정보 검색 시스템에서 역 색인 블록의 수, 즉 슬레이브 개수를 증가시키면 질의 처리 시간을 더욱 단축 시킬 수 있다고 예상할 수 있다.

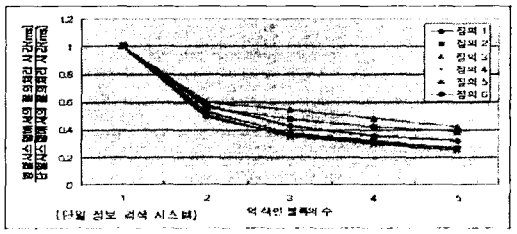


그림 4. 역 색인 블록 수의 변화에 따른 질의 처리 시간의 변화.

7. 결 론

본 논문에서는 대용량의 문서에 대하여 효율적인 정보 검색을 할 수 있도록 오디세우스를 사용하여 병렬 정보 검색 시스템을 구현하였다. 본 논문의 공헌은 다음과 같이 요약될 수 있다. 첫째, 병렬 정보 검색 시스템을 구축하기 위한 기존의 역 색인 분할 방법을 분석하고, 이를 이용하여 병렬 정보 검색 시스템을 설계하고 구현하였다. 둘째, 기존의 역 색인 분할 방법인 문서 식별자 분할 방법과 키워드 식별자 분할 방법을 혼합 사용하는 새로운 분할 방법을 제안하고 구현하였다. 셋째, 단일 정보 검색 시스템과의 질의 처리 시간 비교를 통하여 병렬 정보 검색 시스템의 유용성을 보였다. 실험 결과 동일한 질의를 병렬 정보 검색 시스템이 단일 정보 검색 시스템에 비해 역 색인의 블록의 개수에 근사하게 비례하여 빠르게 질의를 처리하는 것으로 나타났다.

참고 문헌

[1] Tomasic, A., and Garcia-Molina, H., "Issues in Parallel Information Retrieval," *Data Engineering Bulletin*, Vol. 17, No. 3, pp. 41-49, 1994.

[2] Frakes, W. B., and Baeze-Yates, R., *Information Retrieval -- Data Structures and Algorithms*, Prentice Hall, Englewood Cliffs, New Jersey, 1992.

[3] Tomasic, A., and Garcia-Molina, H., "Query Processing and Inverted Indices in Shared-Nothing Text Document Information Retrieval Systems," In *Proc. Int'l Conf. on Very Large Data Bases*, Vol 2. pp. 243-275, 1993.

[4] Grossman, D. A., Frieder, O., *Information Retrieval : Algorithms and Heuristics*, Kluwer Publishers, 1998.

[5] MacFarlane, A., McCann, J. A., and Robertson, S. E., "PLIERS : A Parallel Information Retrieval System using MPI," In *Proc. Euro PVM/MPI'99*, Barcelona, 1999.

[6] 한 욱신, 이 민재, 이 재길, 박 상영, 황 규영, "오디세우스 객체관계형 멀티미디어 DBMS 의 아키텍처," *한국정보과학회 추계학술발표회 논문집*, 2000 년 10 월.

[7] Jeong, B., and Omiecinski, E., "Inverted File Partitioning Schemes in Multiple Disk Systems," *IEEE Trans. on Parallel and Distributed Systems*, Vol. 6, No. 2, pp. 142-153, 1995.

[8] Oracle Corp., interMedia Text, *http://otn.oracle.co.kr/docs/Oracle817/index.htm*, 1999.

[9] 임 효상, 오디세우스/코스모스 객체 저장 시스템을 위한 벌크 로드 기능의 설계 및 구현, 석사 학위 논문, KAIST 전산학과, 1999.