I. 서 론

1.1 연구 배경 및 목적

80 년대 후반 이후, 컴퓨팅 환경은 네트워크 및 데이터베이스 또는 컴퓨터 시스템 등에 관한 정보 통신 기술의 발전이 크게 이루어 졌다. 이러한 컴퓨팅 환경의 발전은 컴퓨터를 사용하여 모든 문서나 메모를 전달하고자 하는 기본 욕구를 창출하게 되었고, 컴퓨터 네트워크를 사용하는 응용 프로그램이 등장하여 사람들의 기대 수준을 만족시키기도 했다. 그러나 이러한 응용 프로그램은 외우기도 힘든 상대방의 정확한 컴퓨터 주소를 알고 있어야 한다는 어려움이 있었으며, 그나마 알고있는 주소의 정확성 여부도 확신할 수 없는 실정이었다.

엔터프라이즈(Enterprise)로 대표되는 연구 기관 및 대학 또는 기업에서, 관련된 또는 유사한 기관끼리 정보를 공유하고, 점차 거대해지는 엔터프라이즈 기관들의 다양한 정보를 저장하고 표현하기 위해 엔터프라이즈 정보 제공(EIR) 서비스 시스템이 필요 하게 되었다. 그 결과로 80 년대 후반부터 90 년대 초반까지, 한정된 지역의 엔터프라이즈 기관들의 정보를 허가된 사용자에게 제공하는 시스템이 등장 하였으나, 이 시기에 출현된 정보 제공 서비스 시스템들은 정보 저장에 관한 내부적 구조(internal structure)를 정의하지 않아 엔터프라이즈의 한 부분에 해당하는 엔티티들의 정보 저장에는 유리한 반면, 정보 표현과 인터넷 환경 또는 분산 처리 환경에서 요구되는 대 용량 또는 분산된 정보의 수용에는 그 한계를 보이고 있다. 또한 정보 제공 서비스 시스템에 종속적인 클라이언트는 다양한 정보를 원하는 사용자에게, 즉 여러 가지 정보 제공

서비스 시스템을 사용, 즉 사용자에게 여러 가지 EIR 서비스 시스템의 클라이언트 인터페이스에 대한 복잡한 컴퓨팅 지식을 강요하였다.

현재 우리의 엔터프라이즈 환경에서도 다양한 혹은 유용한 정보 제공 서비스를 필요로 할 것이다. 그러므로 우리의 엔터프라이즈 환경에서는 더 화려한 컴퓨팅 기술 기반하의 다양한 정보 구조를 수용하며, 더 많은 용량의 정보를 저장하고, 체계적인 정보 저장 및 관리가 이루어지며, 정보를 이용하는 사용자에게는 보다 간편한 단일의 사용자 인터페이스가 제공되어야 할 것이다. 이 논문에서는 우리의 엔터프라이즈 정보 환경에 적합한 EIR 서비스 시스템 설계 모형을 제시함에 있어 월드 와이드 웹(World-Wide Web 또는 WWW)의 기술과 ISO/ITU-T 의 X.500 디렉토리 국제 표준 안을 하부 구조로 하는 EIR 서비스 시스템을 구현하는데 목적이 있다. 또한 EIR 서비스 시스템은 지역적 또는 공간적으로 분산되며, 사용자들에게 다양한 정보를 제공할 수 있는 상호 운영이 가능한 형태(topology)를 갖추고, OSI(Open Systems Interconnections)의 어플리케이션 서비스를 제공하며 프로세스를 관리할 수 있는 개방형 시스템(Open System)에 걸맞도록 설계되었다.

1.2 글의 체계

이 논문에서 1 장은 논문의 서론으로 연구 배경 및 목적을, 2 장에서는 아키텍처 중심으로 기존의 EIR 서비스 시스템을 고찰하였으며, 3 장에서는 EIR 서비스 시스템에 요구되는 사항들을 정의하였으며, 4 장에서는 개념적인 EIR 서비스 시스템 개념 모델 및 설계 모델을 제시하고 EIR 서비스 시스템의 요구 사항에 부합되는

응용 프로그램들에 대한 기술 요소들과 WWW 를 기반으로 하는 구현 모델을 제시 하였다. 5 장에서는 4 장을 토대로 하여 구축된 포항 공대 EIR 서비스 시스템의 사례를 보이며, 데이터 관리 측면에서 요구되는 제반 사항들에 대해서 기술하고 있다.

II. 정보 제공 서비스에 대한 고찰

사회 구성원들은 자신이 처해있는 사회 구조에 필요한 정보들을 찾아 다닌다. 이 논문에서 다루는 정보는 부가적 가치가 있는 정보를 포함하여, 일반적으로 널리 홍보하여 그 정보의 유용성을 알리고, 전역적으로 같은 공감대를 형성하기 위해 공유할 수 있는 정보를 대상으로 한다. 이 장에서는 기존의 EIR 서비스 시스템을 서비스 아키텍처 중심으로 그 문제점들을 고찰한다.

2.1 국제 표준 프로토콜 및 시스템

ARPAnet 에서 최초의 인터넷 연구가 시작된 이후, 현재의 인터넷 환경은 연구 기관 및 대학 또는 기업 등으로 대표될 수 있는 엔터프라이즈 기관들은 다양한 정보를 보유했으며, 점차로 거대해져 가고 있다. 또한 유사 또는 동일한 엔터프라이즈 기관들은 네트워크상에 광범위하게 분산되어있는 정보를 선택적으로 찾아주기 위한 서치(search) 도구들을 제공하기 시작했다. 초기에는 대량의 정보들을 찾아주는 FTP 에서부터 후기에는 특정한 핵심어

(key word)에 대해서 찾아 주는 WHOIS, WAIS 서버와 같은 도구들도 등장하였다.

2.1.1 Netfind

Netfind[7]는 유닉스의 finger 와 인터넷 프로토콜인 SMTP(Simple Mail Transport Protocol)를 사용하는 white page 서비스의 일종으로, 전자 메일 주소나 다른 일반적인 정보들을 찾아준다. 사용자가 핵심어로 쿼리(query)를 하면, Netfind 데이터 베이스에서 해당 정보를 찾아준다. 이러한 메커니즘이 원할히 수행되기 위해서 Netfind 는 USENET 리스트와 Whois 데이터 베이스와 심지어는 네트워크 서비스 로그 파일(log file)과 같은 정보 자원에서도 정보를 수집한다. Netfind 데이터 베이스는 최신의 정보를 보유하기 위해서 정기적인 정보 수집 활동을 수행하여야 한다.

운영 아키텍처는 두 가지 단계로 나누어, 첫 번째 단계는 정보 자원을 보유한 지역의 데이터 베이스를 돌아다니며 필요한 정보, 즉 name, descriptive 정보 등을 수집하는 단계로 이때 Netfind query를 사용한다. 두 번째 단계는 어플리케이션 서비스를 위한 정보, 즉 전자 메일 주소 등을 로컬 인터넷 서비스로부터 획득한다. 운영 특징상 어려움 없이 간단하게 사용될 것 같으나 Netfind의 동적 서치(search) 절차가 인터넷 변동 특성으로 인하여 같은 서치(search)를 실시한다 하더라도 다른 결과를 초래할 수 있다는 것과, 사용자가 원하는 정보에 대한 충분한 설명을 Netfind 서버에게 주어야 한다는 운영상의 어려움이 있다[7]. 즉 사용자가 도메인 범위를 잘 정의할 수 없다면 예측되지 않은 결과가 도출될 수 도 있다. 결과적으로 Netfind 서치(search)는 사용자의 서치 도메인(search

domain)의 범위 추론 정도에 따라 서치(search) 능력이 좌우된다고 할 수 있다. 그러나 사용자가 서치 도메인 범위를 잘 정의 한다고 하는 것은 상당히 어려운 문제가 아닐 수 없으며, 좋은 시스템이 갖추어야 할 항목은 아니다.

2.1.2 WAIS (Wide Area Information Servers)

WAIS 서버[7]는 WAIS 의 인덱스 정보를 가지고서 인터넷 정보 보관소에 저장되어있는 논문과 같은 정보를 찾아주는 서버로서 1988 년 미국의 Apple Computer 사와 Think Machines(TM)사에 의해 개발되었다. 사용자는 Z39.50 프로토콜[7]을 이용하여 WAIS 서버 디렉토리 내에서 정보를 찾으며, 다른 서버의 정보를 찾고자 할 때는 특정 서버를 선택 할 수 있다. 즉 WAIS 클라이언트는 WAIS 데이터베이스의 핵심어(key word) 인덱스를 사용하여 정보를 찾는다. 정보는 문자 중심(text-based)으로 이루어진 도큐먼트를 저장하며, 그림 1은 WAIS 서버의 서비스 아키텍처를 나타내고 있다.

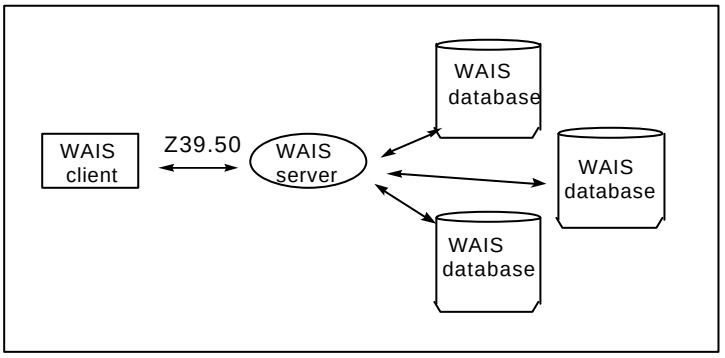

그림 1 : WAIS Architecture

WAIS 서버는 Z39.50 프로토콜에서 정의한 포트에서 사용자의

서비스 요청을 기다리며, WAIS 데이터 베이스에서 정보를 찾아 클라이언트에게 그 결과를 돌려준다. 즉 사용자가 WAIS client를 통하여 서치(search) 요구를 하였다면 클라이언트는 라이브러리(library)를 관리하는 서버와 연결하여, 찾을 단어의 서치(search)를 차례로 요구하게 된다.

서버는 사용자가 정의한 문장이나 단어를 데이터 베이스의 온-라인(On-Line) 도큐먼트에서 인용된 문장을 자동 검색하여 그 결과를 반환한다. 즉 사용자가 정의한 단어들이 도큐먼트에서 몇 번 있는가를 헤아린 후 모두 더한 것을 가중치로 환산하여 점수로 변환시키는 방법을 이용한다. 그러나 이 방법은 필요 없는 단어까지도 가중치에 더하는 문제점이 있다. 즉 묘사하기 위해 쓰여진 “and” 와 같은 접속사 까지도 대상이 되는 것이다. 또한 문맥상의 민감성으로서 정확한 의미가 있는 단어의 순서가 필요로 되며, 사고를 요하는 질문과 특정 분야를 지정하는 질문은 수용할 수 없다 [7].

2.1.3 Gopher

미국의 Minnesota 대학에서 Distributed Campus Information 서비스를 제공할 목적으로 일명 “Golden Gopher”에 대한 연구로 시작하였다. 서버가 보유하고 있는 데이터를 서버 스스로 통제하자는 방안으로, 지역 또는 조직별로 서버를 두고 로컬 데이터 베이스를 스스로 유지/관리 하면서 서버들끼리 데이터를 공유하기 위해 출발하였다. Gopher 디렉토리는 책의 목록이 적혀있는 카드 카탈로그와 사서가 없는 공공 도서관이라 할 수 있다. 즉 도서관의 서지(書誌) 정보를 분류하고 인덱스를 생성하는 도서관 사서 역할을 담당한다.

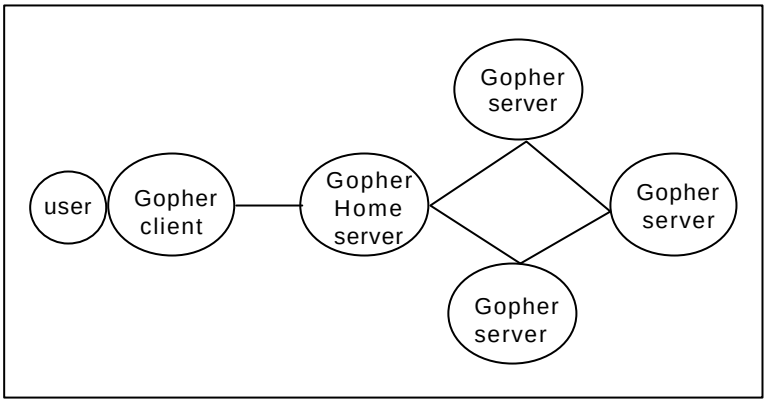

그림 2 : Gopher Operational Model

그림 2 는 Gopher 클라이언트를 자신의 홈 서버에 연결한 상태로서, 문자 중심(text-based)으로 구성된 홈 서버의 메뉴가 최초의 메뉴를 결정하며, 홈 서버에 연결된 포인트에 의해서 공유될 다른 서버들을 결정한다. 사용자는 Gopher 서버를 통하여 주제어가 포함된 리스트를 대충 읽어가면서 필요한 Gopher 서버가 있는 곳으로 옮겨 갈 수 있다[7,9]. 그러나 Gopher 서버가 항상 훌륭한 사서 역할을 담당할 수 만은 없다. 이는 모든 주제가 항상 규격화된 리스트로 제공되지 않기 때문이다. Gopher 정보를 유지/관리하는 곳에서 조금씩은 차이가 나는 제 나름 데로의 서치(search) 방식을 사용하기에, 사용자들은 오픈 마인드(open mind)를 가져야 할 것이다. 즉 Gopher 에서는 ARCHI 또는 WAIS 서버를 이용하여 서치(search)가 이루어지는데 이들 각각의 서버들은 다른 서치(search) 메커니즘을 추구하므로 같은 핵심어에 대해서도 다른 해석을 할 수 있다[7,9]. 한 예로 ARCHI 서치(search)는 단 하나의 단어에 대해서만, WAIS 서치(search)는 핵심어로 된 문장을 대상으로 서치(search) 한다. 즉 핵심어에 대한 서치(search)를 행할 때 인덱스에서 인덱스를 거치면서 핵심어의 의미가 변질될 수 있다는 것으로 이 문제는 사용자에

게 혼돈을 야기 시켰다. Gopher 서버의 인덱스 서치(search)는 매우 다양한 기능을 제공하며, 매우 일반적인 인터페이스를 제공하므로 이상적인 성능면을 갖추고 있다. 그러나 서치(search)를 행할 때 이용자는 어떤 컴퓨터에서 어떤 소프트웨어를 사용해서 서치(search)가 이루어지는지 알 수가 없다.

2.1.4 Whois

Whois[7]서버는 NIC(Network Information Center)의 한정된 도메인에 등록된 소수의 인터넷 사용자에게만 서비스를 제공하는 단일(single) 데이터베이스 이다. 반면에 Whois++는 Whois 모델에 whois++ 인덱스 서비스를 통해서 여러 개의 데이터베이스가 운영 가능하도록 확장한 것으로 분산 환경을 지원할 수 있도록 쿼리(query) 프로토콜 및 논리연산자(boolean operators)를 표현할 수 있는 강력한 서치(search) 메커니즘 등이 강화된 형태이다. Whois++의 정보 모델은 X.500과 유사하나, X.500의 DIT(Directory Information Tree)구조와 같은 계층 이름 구조(hierarchical name space)를 따르지 않고 그물(mesh) 구조로 형성된 인덱스 서비스를 위한 공간을 하나 이상 보유한다[8].

Whois 는 서치(search)가 이루어질 서버의 도메인 이름을 포함하고 있어야 하므로 하나의 거대한 디렉토리일 때는 좋은 방법이었으나, 여러 개의 서버와 데이터 베이스가 존재하는 분산 환경에서는 Whois 클라이언트가 잘못된 데이터를 보유할 확률이 매우 높으며, 같은 서비스를 많은 수의 사용자들에게 제공할 때 트래픽 부하가 발생한다. 또한 인터넷 기반하의 네임서버(name-server)로서의 확장성이 없어 일반 정보를 제공하는 White Page 서비스로는 부적합하고, 전자 메일(E-mail), HTTP, 음성(sound) 등과 같은 응용서비

스를 통합하여 제공할 수 없다[7]. Whois++는 Yellow Page 서비스에는 적합한 것으로 되어있으나 대 용량의 분산 데이터 베이스에는 적합하지않은 것으로 판명되어 있다[8].

2.1.5 World-Wide Web(WWW)

가장 최신에 등장한 인터넷 정보 검색 도구로서, WWW[7]와 관련된 자원들, 즉 이미지(image)와 음성(audio)은 물론 문자 등 모든 자원에 대해서 상호 참조(cross-reference)가 가능한 연결(link) 설정이 허용되는 하이퍼 텍스트(hypertext) 모델 기반하의 인터넷 기술이다.

WWW 의 정보 탐색 도구는 주기적으로 정보를 수집하는 하나의 응용 프로그램인 로보트가 WWW 상에 흩어져 있는 모든 특정 파일의 도큐먼트 또는 이들이 가지고있는 링크에 대해서 자동적으로 서치(search)하여 수집하는 기능을 가지고 있다. 또한 도큐먼트 리스트를 검색하는데 있어서는 다른 전략을 사용할 수 있으나 기본적으로는 URL 을 기본으로 한다. 발견된 URL 은 특정 핵심어(key word)를 사용하여 인덱스를 유지하며, 대량의 인덱스를 저장한 거대한 데이터 베이스는 사용자가 사용하기에 간편하면서도 강력한 능력을 발휘할 수 있는 구조로 이루어져 있다[7].

WWW 의 장점 중의 하나로서 WWW 부라우저를 사용하는 사용자에게 응용 서버가 위치한 하드웨어 또는 운영체제에 독립적인 단일의 인터페이스를 제공하며, WWW 부라우저에서는 전자 메일, 멀티 미디어 데이터 및 HTML(Hyper-Text Markup Language)등에 관해 인터넷에서 제공되는 응용 어플리케이션을 서비스를 가능하게 한다.

2.1.6 X.500 디렉토리

컴퓨터 보급의 급격한 증가와 전산망 기술의 발전 및 일반화 추세에 따라, 컴퓨터 전산망에 연결되는 정보 기기들도 급속히 늘어나고 있다. 따라서 이들의 이름 및 주소를 비롯하여 통신 서비스에 필요한 다량의 정보들을 효율적이고 체계적으로 관리해주는 독립적인 시스템의 필요성이 대두되었다. 이를 위해서 구상된 것이 인터넷 기반하의 X.500 디렉토리 이다. X.500 디렉토리는 OSI(Open Systems Interconnection) 표준 안으로, ISO/CCITT(지금의 ISO/ITU-T)에 의해 연구 개발되어 사람들에 대한 정보를 여러 가지 타입으로 나타내어 저장할 수 있는 하나의 "Single World-Wide Directory" 를 생성하고자 탄생되었다[1,3]. 그림 3 은 X.500 디렉토리 모델을 나타내고 있다.

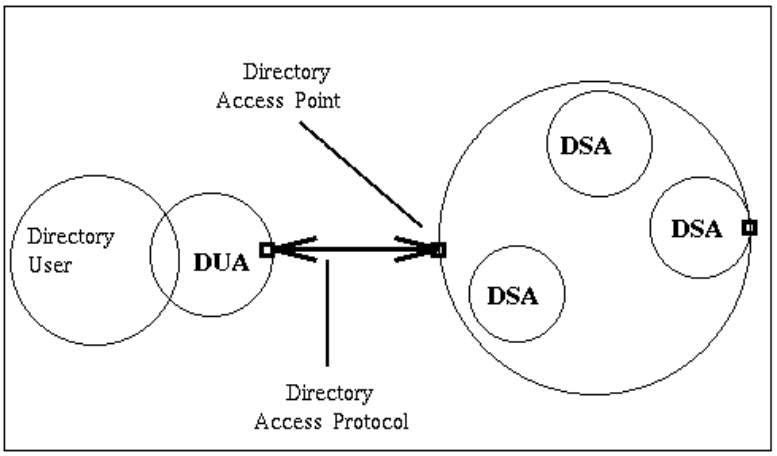

그림 3 : X.500 디렉토리 모델

X.500 디렉토리는 분산 데이터 베이스로서 기본 모델은 Gopher 와 마찬가지로 클라이언트-서버를 근간으로 하나 좀더 복잡한 형태로 구성되어 있다. X.500 디렉토리에서 많이 사용되는 디렉토리의 클라이언트로는 Gopher, WWW 와 Telnet 등이 있다[1,3]. X.500

의 구성요소 중에서 DUA 는 가까운 DSA 에 서비스 요청을 하고, 요청받은 DSA 는 자신의 데이터 정보에서, 또는 연결이 설정된 다른 DSA 에 질문을 재 요청하여, 정보를 제공할 수 있는 DSA 의 주소를 반환한다. 정보는 DIT 라 불리는 계층적 구조 내에 엔트리로 정의되어 있으며, DIB 형태로 구성된 DSA 들이 DIB 내의 한 지역의 DSA 를 구성하며, 각각의 DSA 가 모여 거대한 디렉토리를 형성하기도 한다. 여기서 각각의 DSA 는 정보를 저장하여야 하므로 스키마의 저장 및 관리 규칙(schema rule)과 DIB 내의 다른 DSA 에 대한 지식(Knowledge)에 관한 규칙이 정의되어야 한다.

1) 스키마(schema)

DIT 내에서 엔트리 정보의 위치를 나타내기 위해 데이터를 대상으로 구조화를 실행한 것이다. 만약 그렇지 못하다면 새로이 저장된 정보가 어느 위치에 저장 되었는지 모를 것이고 그 결과로 정보 제공 서비스에 문제가 생긴다. 즉 스키마는 정보의 관리를 위해 정보를 표현하는 애트리뷰트로 구성된 엔트리 규칙과 관련된 제약 사항들의 집합으로, 객체 클래스를 표현하는 애트리뷰트 타입에 따른 값이 잘못 정의되어, 엔트리가 오류 정보를 갖지 않도록 예방하는 기능을 갖는다. 즉 엔트리는 DIT 내에서 엔트리를 식별할 수 있는 상대적인 고유 이름 (Relative Distinguished Name) 과, 객체 클래스에 정의되어야 하는 필수 또는 선택 속성 항목에 관한 사항과, 객체를 구성하고 있는 속성 항목의 타입과 값들에 관한 규약을 정의한다. [1]

2) Knowledge Reference

디렉토리 정보는 두 종류로 구분할 수 있다. 디렉토리 내에 조직(organization) 또는 사람 등과 같은 실질 정보(actual information)와 DIT 내의 주변 DSA 들이 보유한 정보에 대한 지식 정보(knowledge information)이다. 계층구조를 따르는 DSA 는 자신의 바로 위와 아래 DSA 에 대한 지식 정보를 알고 있으면 다른 DSA 의 정보를 공유할 수 있다. 즉 지식(knowledge)은 DIT 에 포함된 다른 DSA 에 어떠한 방법을 통하여 해당 DSA 에 접근할 수 있는지를 알려준다. 이것은 하나의 조직 단위인 DSA 에서 설정되며, DSA 들간의 연관된 지식 정보를 관리할 수 있다. 디렉토리가 유용한 정보를 제공하기 위해서는 지식 및 데이터를 DIT 단계에서 사본(replication)을 운영할 수 있어야 한다. 즉 정보 저장소에서는 사본 저장에 관한 서비스는 DSA 가 지원해야 할 필수 요소로서 디렉토리 객체의 실질적 정보를 어디서 찾아야 하는 것에 대한 지식에 관한 사항들로서, 사본으로 저장된 DSA 의 정보에 대해서는 정보 갱신이 쉽게 이루어질 수 있음을 의미한다. 이러한 요건은 디렉토리 서비스 시스템 성능에 영향을 미치는 요소로 작용된다[1].

X.500 디렉토리는 네트워크상에서 계층적인 트리 구조 형태로 구성된 분산 데이터 베이스로서 DUA 가 디렉토리에 접근할 때 정의된 오퍼레이션은 BIND 이다. 디렉토리와 사용자는 네트워크상에 흩어져 있으므로 네트워크를 이용하는 것은 필수이므로 네트워크 이용에 따라 데이터 보호를 위한 보안(security) 메커니즘은 반드시 제공되어야 한다. 이 논문에서는 1993 년도 ITU-T 에서는 X.509 표준 안의 인증(Authentication)과 X.501 표준 안의 접근 조절(Access Control)에 관한 두 가지 메커니즘에 대해서 고찰해 보고자한다.

3) 인 증(Authentication)

X.500 디렉토리를 이용하는 사용자는 디렉토리에 접근 시 인가된 사용자임을 확인하는 인증 서비스를 받아야만 한다. 인증 서비스는 크게 Simple Authentication 와 Strong Authentication 또는 No Authentication 이 세 가지로 나눌 수 있다.

가) Simple Authentication

디렉토리 엔트리의 DN(Distinguished Name)과 패스워드는 엔트리 안에서 하나밖에 존재하지 않는 유일한 식별성을 갖으므로, DUA 를 이용하는 사용자들의 디렉토리 접근 요청에 사용할 수 있다[1]. 패스워드는 네트워크를 통하여 전송되므로 전송 전에 1 차 해쉬 함수(one-way hash function) 등을 이용하여 엔코딩(Encoding)하여 전송 하여야 한다. 엔코딩 할 때 사용한 객체 식별자(object identifier)를 함께 전송 하여야 한다. 전송된 패스워드 및 DN 은 DUA 의 DAP bind 요구를 전송받은 DSA 가 수신하여 접속을 원하는 사용자가 올바른 사용자인지를 판단하는데 사용되고 그 결과(True 또는 False)를 반환한다. 만약 해당 엔트리가 DIT 내의 올바른 엔트리에 위치하고 있다면, 비교(compare) 오퍼레이션을 사용하여 엔트리 내에 저장되어있는 패스워드 값과 비교하는 것이다. 이처럼 간단한 방법의 보안 메커니즘은 손쉽게 적용될 수 있으므로, 보안이 크게 요구되지 않는 환경하에서 사용될 수 있다. 그림 4 는 Simple Authentication 방안에 대한 개념적 모델을 나타내고 있다. [1]

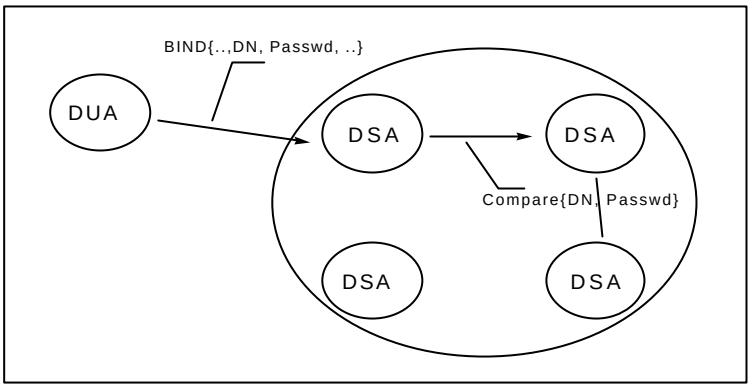

그림 4 : Simple Authentication 모델

나) Strong Authentication

좀더 강력한 보안 기법인 비 대칭적 암호화 테크닉에 기반을 두고있다. X.500 Directory 표준 안은 어떤 특정한 암호화 알고리즘에 대한 암호화 기술을 표방하고 있지않으나 주로 많이 사용되는 암호화 기법으로는 NSA(National Security Agency)의 DES 와 RSA(Rivest, Shamir, Abelson) 알고리즘이 있다. 전자는 같은 키(key)를 사용하여 암호화(encryption) 및 복호화(decryption) 하는데 사용하고, 후자는 서로 다른 키(key)를 암호화 및 복호화에 사용된다. 즉 RSA 기법은 키를 public 키와 private 키로 구분하고, private 키로 암호화 하였다면 수신 측에서는 private 키로 복호화 한다. 이와 반대로 두 가지의 키가 역할을 바꾼다 해도 무방하다. RSA 알고리즘은 효과는 좋은 반면 암호화 및 복호화 실행에 따른 부하가 많이 걸리며, 더욱이 미국의 한 회사가 특허권을 보유하고 있어 임의로 사용할 수 없으므로 다른 방안의 기법을 강구해야 하는 실정이다. 다음 그림 5는 Strong Authentication에 대한 개념적 모델을 나타낸다.

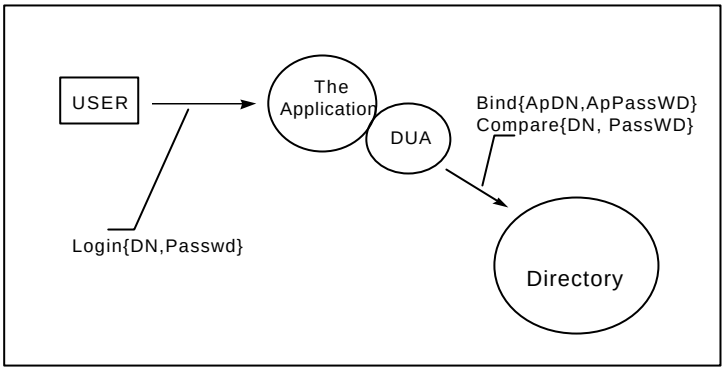

그림 5 : Strong Authentication Model

다) No Authentication

앞에서 이미 언급했듯이 X.500 디렉토리 서비스에 대한 표준안에서는 인증 사용에 대한 요구는 없다. 그러므로 통상적으로 WWW를 통해서 디렉토리 시스템에 접근하는데 적용될 수 있다.

4) 접근 조절(Access Control)

X.500 디렉토리의 DIT 접근을 권한에 따라 제한하고 조절하는 접근 제어 방안을 제시 하였다. 이 방안은 유닉스 파일 시스템과 같이 디렉토리 관리자가 각각의 정보 및 엔트리에 대해서 사용자 그룹의 권한을 정의한 것으로, 접근 제어 조절 목록(access control list)은 보호 대상이 되는 정보 아이템들을 정의한다. 예를 들어 디렉토리 정보를 이용하고자 하는 사용자는 속성을 갖는 항목이나, 엔트리에 설정된 접근 조절 리스트(ACL list)에 정의된 권한에 따라 인식된다. 즉 이미 제공된 인증 기반하에서 엔트리 항목을

읽기 원하는 사용자는 Simple Authentication 을 갖고서 로그 인 하여야 한다면, 항목 변경을 원하는 사용자는 Strong Authentication 을 갖추어야 한다[1]. 그러므로 DUA 는 디렉토리 정보에 선택적 접근 권한을 갖는 기능이 제공되어야 한다.

이장에서 고찰한 기존의 EIR 서비스 시스템은 정보 저장 구조 및 서비스 측면에서 분산 아키텍처를 지원하지 못하여 시스템 개방 및 확장 측면과 서치(search) 문제, 그리고 시스템 서비스에 따르는 부하 문제 등이 거론 되었다. 일례로 Whois 서버는 약 7 만여개의 엔트리를 보유했을 뿐, 수 백만에 이르는 인터넷 사용자 수의 엔트리를 보유 하기는 벅찬 실정이다[2].

III. EIR 서비스 시스템의 요구 사항

이 장에서는 EIR 서비스 시스템의 정의를 통하여, 시스템을 이루는 구성요소에서 필요로 하는 요구 사항들을 정의 한다. 또한 EIR 서비스 시스템을 정보 사용자와, 정보들을 체계적인 방법을 통하여 저장하고 관리하는 시스템으로 구분하여 각각의 구성 요소들의 요구 사항을 정의하기로 한다.

EIR 서비스 시스템은 사용자와 시스템 상호간에 유기체적인 관계를 유지 하여야 한다. 만약 사용자와 EIR 서비스 시스템이 정

의하고 있는 개념이 서로 상이하다면 정보의 가치는 떨어지게 되고, 이러한 시스템은 쉽게 사장될 것이다.

3.1 EIR 서비스 시스템의 정의와 구성 요소

정보는 체계적인 방법을 적용하여 적절한 장소에 저장되어야 한다. EIR 서비스란 정보의 사용을 원하는 사용자에게 신속 정확한 방법으로 어떤 규정된 정보를 제공한다는 것을 의미하는 것으로 EIR 서비스의 본보기로는 전화 번호 안내 서비스가 있다. 전화국에서 전화 가입자 상호간에 정확한 전화 번호를 알려주는 서비스를 위하여 전화 번호부를 발간해서 정기적으로 정보의 갱신을 수행한다. 이 예에서 전화 가입자들의 전화 번호 및 인적 사항은 정보가 되며, 전화국에서 생성한 전화 번호부 책은 정보 저장소가 된다. 물론 현 시점에서는 물리적 저장소인 컴퓨터 장치가 이용될 것이다. 또한 적극적인 해결 방안의 하나로 한정된 지역(local area)의 전화 가입자에 대한 정보를 실시간에 제공하는 것이다. 그렇다면 원활한 서비스 제공을 위해 정보 저장소에서는 체계적으로 정보를 저장하여야 하며, 사용자와의 상호 작용을 위하여 사용자가 원하는 기대 수준에 일치하는 서비스를 제공하여야 한다.

정보 제공 서비스를 위한 체제 구성요소 중에서 정보를 저장하는 정보 저장소와 사용자간에 상호 행위가 적용될 정보 제공 서비스 시스템의 클라이언트 인터페이스는 중요한 요소가 될 것이다.

3.1.1 정보 저장소(Information Repository)

오늘날의 컴퓨팅 환경에서 연구기관 및 대학 또는 기업 등으로 대표되는 엔터프라이즈 기관들은 지역적으로 같은 곳에 위치할 수도, 또는 관련된 조직을 모두 포함하여 네트워크 상에 널리 흩어져 존재할 수도 있다. 정보 저장소는 정보의 저장 및 제공을 위해 일련의 오퍼레이션 및 규약이 정의되어야 하며, 대용량의 정보를 저장할 수 있는 구조 및 체제를 갖추어야 한다. 사용자는 이러한 오퍼레이션을 이용하여 정보의 추가 및 삭제와 변경 등을 실행 할 수 있어야 한다. 또한 정보 저장 및 관리와 연관하여, 우리가 알고 있는 데이터 베이스의 특수한 알고리즘을 사용하든, 않든지 간에 정보를 저장하는데 필요한 데이터 베이스도 역할을 담당하여야 한다.

3.1.2 사용자 인터페이스

정보 사용자는 공동체 사회의 구성원들과 또는 컴퓨터 호스트의 어플리케이션 프로그램과 정보 저장소 관리자로 분류할 수 있다[1]. 그러나 사용자들의 컴퓨팅 환경 및 생활 습관 또는 언어체계 등 사람들마다 처한 환경은 모두 틀린다. 대 부분의 EIR 서비스 시스템의 이용은 EIR 서비스 시스템의 하드웨어 또는 운영체제에 의존적인 시스템 인터페이스에 의해 접근할 수 있었다. 그러나 사용자들은 모든 사용자의 컴퓨팅 환경 및 지역에 무관하게 자기가 처한 환경 내에서 편리하면서도 신속 정확하게 EIR 에 접근할 수 있어야 한다. 즉 EIR 서비스 시스템과는 무관하게 단일의 사용자 인터페이스로 EIR 에 접근하고 정보 제공 서비스 받기를 원하기 때문이다.

3.2 요구 사항

EIR 서비스 시스템은 체계적인 정보 저장 및 관리를 통하여 유효한 엔터프라이즈 정보를 제공하여야 한다. 사용자는 EIR 의 정보를 간편한 방법으로 제공 받기 원한다. 이러한 유기체적인 흐름을 만족시키기 위한 일련의 사항들을 사용자와 시스템 측면에서 정의하고자 한다.

3.2.1 사용자 측면

인터넷 상에서 WWW 부라우저(browser)를 사용하는 사용자는 EIR 에 대한 접근을 대신하는 역할을 담당하는 응용 프로그램이 마련되어야 한다. 즉 WWW 부라우저의 HTTP 프로토콜과 정보 저장소와의 가교 역할을 담당 한다. 이를 위해서는 정보 저장소의 모든 오퍼레이션을 수용하며, 정보 표현 및 저장 구조를 잘 이해할 수 있는 인텔리전트한 인터페이스가 정의된 게이트웨이가 존재하여야 한다. 다음은 EIR 서비스 시스템과 관련하여 사용자 측면에서 요구되는 사항을 나타낸다[2,12,15].

- 다양한 컴퓨팅 환경을 지원할 수 있어야 한다. 즉 트랜스포트 계층 위에서 직접 운영되는 장비, 또는 인터넷 공중 전화 망(Internet Public Switched Telephone Network)과 멀티 미디어 응용 어플리케이션 등을 사용하는 다양한 사용자 환경과 접근 방법 등이 모두 지원되어야 한다. 즉 EIR 서비스가 어떤 종류의 하드웨어 또는 운영체제에 존재한다 할 지라도 유용한 인터페이스만 갖추었다면 서비스를 보장 받을 수 있어야 한다. 이는 EIR 서비스 시스템의 유효성(wide availability)과도 관계된 요소이기도 하다.
- 정보 저장소 위치 및 저장 방식에 무관하게 편리하고 강력한

검색 방법을 사용자에게 제공하여야 한다.

3.2.2 시스템 측면

정보 저장소는 최신의 정보를 유지해야 하므로 변동 사항이 발생하였을 때 즉각적인 정보의 변경이 이루어져야 한다. 이러한 유기체적인 성질을 만족하려면 정보 제공자와 데이터 소스가 밀접한 관계를 유지해야 가능할 것이다. 즉 이 의미는 충분한 자원을 보유한 조직의 고유 정보 저장소에 저장하는 것이다. 즉 분산 디렉토리 서버를 두고 로컬 디렉토리를 스스로 유지 관리하며 데이터의 갱신 및 관리에 체계를 부여하여야 한다. 다음은 하나의 정보 저장소에서 정보의 저장 및 관리를 위해 갖추어야 할 필수적 요건들에 대해 정의한 것이다[1,2,3,4].

- EIR 서비스 시스템이 갖춘 모든 기능적 서비스가 효과적으로 이루어지기 위해서는 다양한 하드웨어 및 운영 체제 플랫폼 모두를 지원할 수 있는 개방형 시스템(open system)을 지향하여야 한다.
- 대 용량의 정보 저장 및 성능과 정보의 유효성이 보장되어야 한다. 즉 최대한의 성능 요건을 만족시키기 위해, 분산 아키텍처를 갖도록 설계 구현되어야 한다.
- 정보에 대한 보호 대책이 강구되어 있어야 한다.
- 산재 되어있는 EIR 의 이용 율을 향상시키기 위해 데이터의 사본(replication)의 보관도 필수적으로 요구되므로 이로 인해 발생하는 데이터 사본에 대한 갱신도 가능하여야 한다.
- 중앙 집중식 데이터 베이스는 저장된 데이터 구조에 특정 메커니즘을 조합한 인덱스를 사용한다. 그러나 EIR 은 분산 환경에 적합한 메커니즘을 갖는 서치(search) 도구가 지원 되어야

한다.

- 인터넷 어플리케이션 정보 서비스를 위한 정보도 저장 가능하여야 하며, EIR 클라이언트는 어플리케이션 서비스도 통합 지원하여야 한다.
- 서비스 결과 및 오류 메세지는 가장 신속한 방법을 통하여 정확히 통보되어야 한다.

이 장에서 언급한 사항들은 EIR 이면 당연히 갖추어야 할 일반적인 것들에 불과하다. 그러나 현재 우리가 추구하는 EIR 은 범용 데이터의 사용뿐 아니라 인터넷 환경의 모든 서비스도 포함하기 위한 것이다.

3.3 EIR 서비스의 의미

전자 또는 컴퓨터 네트워크 등으로 얽혀있는 우리의 경제, 사회 및 기관에서 계층적인 구조 형태로 구성된 정보는 중요한 역할을 담당한다. 만약 우리 사회가 포괄적 의미를 갖는 정보서비스가 가능한 네트워크가 구축되어 있다면, 이미 정보 서비스의 한 부분으로 존재하는 전자 메일 시스템 등을 논리적인 한 부분으로 인식할 수 있으므로 전체 정보에 대한 관리가 쉬어진다[5]. 물론 네트워크 환경과는 무관하게 원하는 정보에 접근할 수 있는 방법이 있어야 한다.

예를 들어 전화 시스템의 경우, 전화 번호만 알고 있다면 일차적인 방법으로 언제든지 연락을 주고 받을 수 있다. 그러나 정보 제공을 위해 정보 저장 및 관리에 있어 동일한 규격으로 정의되지 않은 표준 안이 없다면 제각각 다른 특성이 적용될 수 있음

을 의미하며 그 혼란은 예측하기 힘들다. 우리 생활에서는 전화번호부와 교환원과 같은 표준 안이 제시되어 있으며, 사용자는 동일한 방법으로 접근하도록 제시되어 있다. 그러므로 네트워크와 디렉토리 어플리케이션 서비스에 관한 절차나 규약을 정하는 표준안을 필요로 한다.

IV. EIR 서비스 시스템 설계

이장에서는 오늘날의 다양하고 복잡한 엔터프라이즈 기관에 요구되는 EIR 서비스 시스템 개념 및 설계 모델을 제시하며, 인터넷 상에서 풍부한 정보 제공 서비스를 목표로 하여 사용자 인터페이스와 시스템 확장 및 개방성 등을 만족하는 WWW 기반 하의 EIR 서비스 시스템 구현 모델을 설정하고 EIR 서비스 시스템을 이루는 시스템 구성요소에 대해서 정의한다.

4.1 EIR 서비스 시스템 개념 모델

오늘날의 엔터프라이즈 기관은 그 특성상의 이유로 네트워크상에 산재되어 있으며 다양하면서도 대 용량의 엔터프라이즈 정보를 보유하므로, EIR 서비스 시스템의 정보의 표현 및 저장 메커니즘은 분산 아키텍처를 지원하여야 하며, 시스템의 확장 및 개방성에 대한 요소들을 필수적으로 요구된다. 또한 EIR 서비스 시스템 사용자들은 네트워크상의 어느 위치에서라도 EIR 시스템 클라이언

트에서 제공하는 단일 인터페이스만으로 어떤 정보 저장소라도 접근할 수 있기를 바란다. 또한 정보 저장소끼리는 규정된 통신 프로토콜을 이용하여 정보 저장소에 대한 정보를 상호 교환할 수 있어야 한다[1]. 한편 인터넷의 어플리케이션 서비스를 EIR 서비스 시스템의 클라이언트에서 통합하여 지원할 수 있는 기능도 포함되어야 할 것이며, 이미 3 장에서 정의한 EIR 서비스 시스템의 사용자와 시스템이 지녀야 하는 모든 요구 사항들을 충분히 지원하여야 한다. 그림 6 은 EIR 서비스 시스템이 갖추어야 할 요구 사항들을 적용하여 정립한 EIR 서비스 시스템의 개념적 모델을 나타낸 것이다.

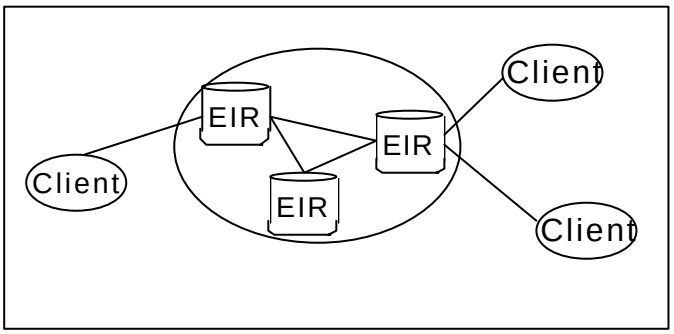

그림 6 : EIR 서비스 시스템의 개념 모델

4.2 WWW 기반하의 EIR 서비스 시스템 설계 모델

EIR 의 정보 제공 서비스 시스템은 하드웨어나 운영체제 등에 독립적인 EIR 과 EIR 서비스 시스템 사용자를 위해서 시스템의 단일 인터페이스를 지원하여야 한다. 또는 EIR 서비스 시스템에서

WWW 응용 서비스 정보를 통합하는 서비스를 제공하기 위해서는, WWW 응용 정보에 대한 표현 및 저장이 가능하여야 한다. 이러한 추세에서 EIR 과 EIR 서비스 시스템 클라이언트와 WWW 의 연계는 필수적이라 할 수 있다.

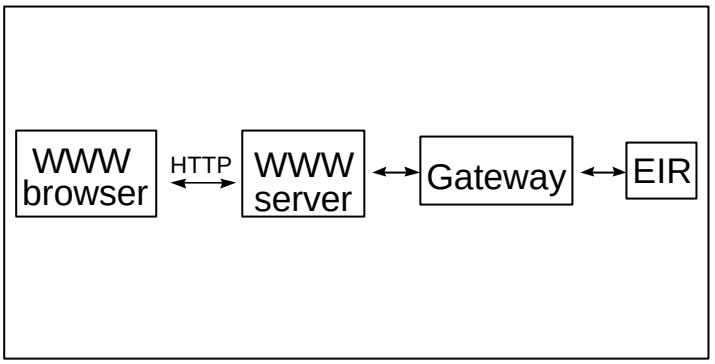

그림 7 : EIR 서비스 시스템 설계모델

그림 7 은 EIR 서비스 시스템 개념 모델을 WWW 기반을 적용하여 EIR 서비스 시스템의 설계 모델로 확장한 것이다.

EIR 은 네트워크상의 한 컴퓨터 호스트에 위치하게되며, EIR 서비스 시스템 클라이언트인 WWW 부라우저를 사용하는 사용자에게 단일의 인터페이스를 제공하며 EIR 에 대한 접근을 허용한다. 여기서 EIR 은 WWW 서버의 HTTP 프로토콜을 번역하여 EIR 에서 정의된 프로토콜로 변환하는 WWW 게이트웨이(gateway)를 필요로 하며 WWW 게이트웨이는 EIR 서비스 시스템이 서비스를 제공하는 것과 관련하여 EIR 의 부하도 고려하여야 한다. 또한 사용자가 기대하는 EIR 의 기능적 기대 수준을 만족시키기 위해 정의된 오퍼레이션은 EIR 서비스 시스템 클라이언트인 WWW 부라우저에서도 지원되어야 하며, EIR 은 동시 다발적인 여러 사용자의 모든 오퍼레이션을 수용할 수 있어야 한다[16].

4.3 EIR 서비스 시스템 구현 모델 및 구성 요소

그림 8 은 EIR 서비스 시스템 설계 모델을 확장하여 WWW 상에서 실질적으로 운영되고 있는 응용 프로그램을 EIR 서비스 시스템에 적용시킨 구성요소를 포함시켜 나타낸 것이다. 사용자는 WWW 부라우저인 Netscape 을 이용하여 WWW 서버에 접근하고 WWW 서버는 ISO/ITU-T 에서 분산 환경에 적합한 분산 데이터 베이스의 내부 저장 구조를 정의한 X.500 디렉토리에 LDAP 게이트웨이를 통하여 EIR 에 접근하는 것이다

EIR 서비스 시스템의 구성은 정보의 표현 및 저장 등이 분산 데이터 베이스 아키텍처를 수용하는 X.500 표준 안과 X.500 의 특성에 맞게 WWW 서버와 EIR 의 통신 프로토콜을 변환하는 LDAP 과 EIR 의 정보 표현 구조를 계층적 트리 구조 방식으로 설정하고 나타낼 수 있는 WWW 클라이언트인 Web500gw[16]로 구성되어 있다.

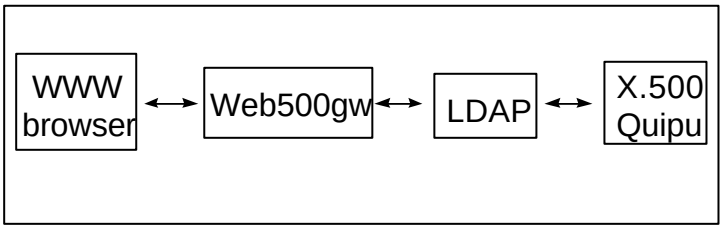

그림 8 : EIR 서비스 시스템 구현 모델

4.3.1 정보 저장소

X.500 디렉토리 표준 안은 디렉토리 서비스를 위해 ISO/ITU-T 에서 정의한 국제 표준 안으로서, 정보의 저장 및 표현에 있어서 분산 구조를 수용하고 대용량의 정보를 저장 할 수 있도록 설계된 일종의 분산 데이터베이스 이다. 그러므로 이 논문의 저장 기반은 정보 객체의 표현 및 저장 측면에서 충분한 자격을 소유한 X.500 디렉토리를 기반으로 하여 EIR 을 구성할 것이다. 다음은 X.500 표준 안이 EIR 서비스 시스템 관리 측면에서 갖는 유리한 장점들이다[1,2,7].

- 정보 객체(object)의 저장 및 표현에 있어서 논리 계층적 트리 구조(hierarchical tree structure)를 이루므로 정보 객체의 표현이 자유롭고, 데이터 관리가 용이하다.
- 정보 객체들은 엔트리(entry)로 표현되며, 엔트리는 애트리뷰트(attribute) 집합으로 표현되므로 객체 클래스 상속을 통하여 애트리뷰트를 공통으로 사용할 수 있다.
- 엔트리 속성을 결정하는 애트리뷰트 타입은 구체적인 구조를 갖는 객체를 간단하게 표현하며, 유일한 엔트리 이름을 갖게 하므로 강력한 서치(search) 메커니즘을 지원할 수 있다.
- 인터넷상의 방대한 조직을 논리적 트리 구조로 표현할 수 있어, 현실 세계를 표현하는 각각의 엔트리들을 하나의 객체 형태로 간단하게 표현할 수 있다. 간단하게 표현할 수 있다는 것은 높은 확장성을 갖는다는 것을 의미한다

그림 9 는 ISO/CCITT (지금의 ISO/ITU-T, International Standard Organization/International Telecommunication-Union Telecommunications Standardization)에서 정의한 X.500 디렉토리의 기능 모델을 나타낸다.

트리 형태로 구성된 DSA 들은 DIB(Directory Information Base)내의 한 지역 DSA 를 구성하며, 각각의 DSA 가 모여 전체 디렉토리를 구성한다. 각각의 DSA 는 정보를 저장하는 데이터베이스 이므로 디렉토리 스키마의 저장 및 관리 규칙(schema rule)과 DIB 내의 다른 DSA 와 정보 교환에 필요한 지식(knowledge)에 관한 규칙이 정의되어 있다[1,2,3,4].

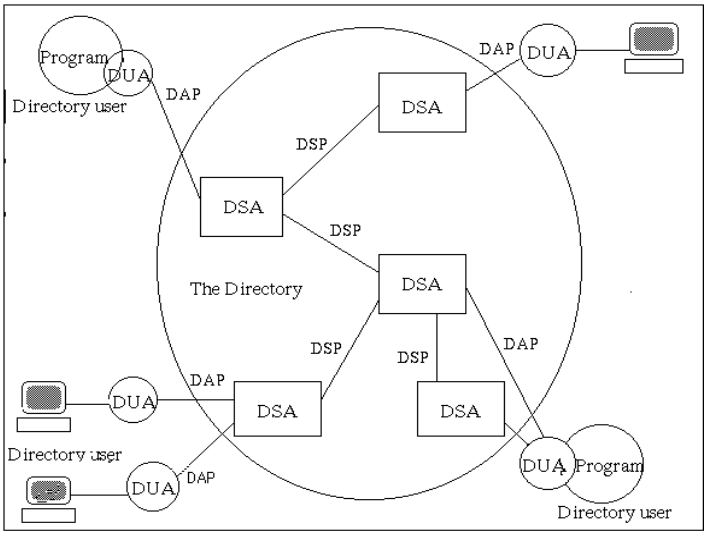

그림 9 : X.500 디렉토리 기능 모델

DSA(Directory System Agent)는 현실 세계에서 발생한 정보를 저장하며, DSA 간 통신 프로토콜(DSP-Directory System Protocol)을 사용하여 다른 도메인의 디렉토리와 상호 연결(chaining) 하는 데이터의 사본 저장(replication)도 가능하게 한다. 또한 어플리케이션 프로세스로서 연결되어있는 다른 DSA 또는 DUA(Directory User Agent)에게 DIT(Directory Information Tree)의 검색 및 엔트리 정보에 대한 추가 및 삭제와 갱신에 관한 오퍼레이션을 허용한다[1].

DUA 는 사용자의 어플리케이션 소프트웨어로서 인접해 있는 DSA(일명 home DSA)에 서비스 요청을 위한 오퍼레이션 및 통신 프로토콜(DAP - Directory Access Protocol) 요소와 DSA 에서 제공하는 서비스를 사용자에게 전달하기 위한 뷰(view) 서비스 기능 및 인터페이스 기능과 또 다른 어플리케이션 서비스를 위한 기능을 포함한다[1].

4.3.2 LDAP (Lightweight Directory Access Protocol)

X.500 디렉토리는 화이트 페이지 서비스와 기타 정보 검색을 위한 응용 서비스 등을 구현하는데 사용된다. X.500 표준 안에서 DUA 가 DSA 에 접속할 때 사용하는 DAP(Directory Access Protocol)는 OSI 상위 계층의 구조를 OSI 응용 계층 프로토콜로서 사용한다[12,13]. 이를 구현하는데는 방대한 양의 프로그램 작업을 필요로 하여, 소규모의 워크스테이션이나 개인용 컴퓨터에서 OSI 디렉토리를 이용하기에는 많은 어려움이 따른다. 또한 오늘날 정보 통신망의 규모가 커지고 인터넷과 같은 전세계적인 정보 통신망이 구성되어 통신망에 연결된 정보 기기와 가입자들이 크게 늘어나, 이를 이용하는 서비스 사용자들도 점차 증가할 것으로 보이므로 서비스 요청에 대한 부하도 가중되었다. 이러한 문제점을 해결하기 위해서 DAP 의 기능을 분리하는 DUA 모델 방법이 연구되어왔다. DAP 기능 분리 모델 중의 하나인 LDAP 모델에서는 DAP 를 용량이 큰 워크스테이션이나 메인 프레임에 위치한 LDAP 서버에 구현하고, 사용자 인터페이스가 위치한 클라이언트는 LDAP 서버를 통하여 X.500 디렉토리에 접근한다[11,14,18].

LDAP 은 인터넷 디렉토리 표준 안(Internet Directory standard)으

로서 X.500 프로토콜인 DAP의 한 부분이므로 X.500 디렉토리 서비스와 상호 연동이 가능하고, LDAP 을 사용하는 모든 어플리케이션 프로세서들이 하드웨어나 운영체제 플랫폼과는 무관하게 X.500 디렉토리에 대한 접근을 가능하게 한다. 이를 위해서는 LDAP 서버를 이용하는 사용자들은 LDAP 시스템 클라이언트의 구현은 필수이다. LDAP 은 온-라인(online) 정보 서비스를 위한 프로토콜로 TCP/IP 위에서 운용된다. 이 의미는 X.500 디렉토리에서 요구하는 OSI 의 transport, session, presentation, association control, remote operation 과 reliable transfer server 등이 LDAP 에서 해결 가능하고, OSI 의 스택(stack)의 유용성을 살리면서 복잡성을 제거한 것이다[15]. LDAP 의 주요 특징은 다음과 같다[12,13,14,15].

- 프로토콜 요소들은 세션/프리젠테이션(session/presentation)에 걸리는 부하를 거치지 않고 TCP 또는 다른 Transport 계층으로 직접 전달된다.
- 데이터 해석을 위한 파싱(parsing)과 조합(compose)에 많은 노력을 필요로 하는 ASN.1 을 사용하는 X.500 에서의 데이터 엔코딩(encoding) 대신에 모든 요소들을 평범한 문자열로 엔코딩하는 문자 기반의 접근 방식의 Lightweight BER(Basic Encoding Rule)을 사용함으로써 어플리케이션 단계가 아닌 커뮤니케이션 단계에서 간단하게 적용시킬 수 있다.

4.3.3 WWW 클라이언트

Web500gw 는 TCP/IP 위에서 LDAP 프로토콜을 이용하여 X.500 디렉토리에 접근한다. LDAP 프로토콜은 X.500 디렉토리에 대해서

항상 같은 기능을 제공하므로 모든 DUA 가 LDAP 프로토콜을 사용하는 것은 당연한 결과로서 WWW 클라이언트로서 사용자를 대신하는 Web500gw 가 등장하였다.

Web500gw 는 WWW 브라우저에 대해서 단일 인터페이스를 제공하며 하드웨어 및 소프트웨어 플랫폼에 독립적인 WWW 클라이언트를 제공하는 장점이 있다. Web500gw 는 사용자 인터페이스를 정의하고, 디렉토리 서비스 요청을 위해 LDAP 서버와 TCP 연결을 위한 오퍼레이션을 설정한다. Web500gw 는 TCP 포트(port) 상에서 WWW 브라우저를 사용하는 사용자에게서 발생한 서비스 요구 사항을 받아들이며, 요청된 URL 과 독립적인 서비스 요청은 X.500 디렉토리에 대해서 읽기 및 서치(search), 갱신 등과 같은 기능적 명령을 수행한다[16]. 즉 Web500gw 는 HTTP 와 LDAP 프로토콜 사이에서 X.500 디렉토리의 구조를 메뉴 형식으로 제공하며, WWW 상에서 서치(search) 및 부라우징(browsing)이 지원되는 WWW 브라우저를 갖는 사용자들을 X.500 디렉토리에 접근할 수 있게 해준다. 또한 Web500gw 를 WWW 부라우저에서 사용한다는 것은, 위의 장점뿐 아니라 분산 멀티미디어 정보들을 통합할 수 있으며, 빠른 발전을 보이는 컴퓨팅 환경에도 적절히 적응할 수 있다. 예를 들어 X.500 디렉토리에 대한 연구가 처음 시작 되었을 때 유럽 서구 나라의 문자 코드를 기반으로 하는 ISO/IEC 10646-1(Unicode)을 채택하였기에 한 문자 당 2 바이트(Byte)를 차지하는 우리의 한글 문자를 제대로 표현할 수 없었다. 임시적으로 DUA 의 문자 처리(national character set) 부분을 수정하고, 새로운 라이브러리를 추가하면 한글 문자를 표현할 수 있으나 기존의 EIR 의 DUA 는 하드웨어 플랫폼에 종속적이므로 모든 DUA 가 수정되어야 한다는 곤란함이 있다[18]. 그러나 현재의 WWW 기술은 이러한 문제를 간단히 해결하고 있다. 또한 모든 WWW 클라이언트는 하드웨어 플랫폼과 문자 부

분에 현저한 적용성을 보이고 있으므로 독립적인 WWW 클라이언트를 사용하여 X.500 디렉토리를 사용하는 것이 현명한 해결 방안이 될 것이다.

V. 구 현

EIR 서비스 시스템의 구현은 엔터프라이즈 기관의 정보 객체 클래스의 형식 선언 및 저장 구조를 정의하고, 이를 바탕으로 실질적인 정보들을 포함하는 EIR 과 EIR 서비스 시스템을 이루는 구성요소들에서 정보 제공 서비스와 관련한 제반 기술 적용에 중점을 두었다. 또한 EIR 의 정보에 대한입력 및 수정, 삭제 등과 관련한 데이터 관리(management) 부분으로 나누어 구현해 볼 수 있다.

5.1 객체 클래스와 애트리뷰트

정보는 엔트리 단위로 저장되며, 최소 하나 이상의 객체 클래스를 포함 한다. 실질적인 정보 엔트리는 애트리뷰트 속성에 의해 결정되며, 애트리뷰트 타입과 이들이 갖는 값들은 객체 클래스(object class)에 정의된 속성을 따르게 된다[6]. 또한 DIT 내 엔트리들은 최소 하나 이상의 애트리뷰트를 사용하게 되며, 각각의 애트리뷰트는 하나 이상의 값들을 갖을 수 있다. 그림 10 은 정보를 표현하고 있는 엔트리와 애트리뷰트와의 관계를 보여주고 있다.

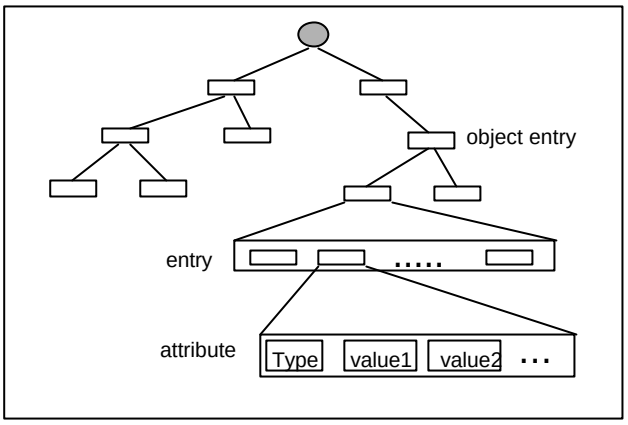

그림 10 : 디렉토리 엔트리

각각의 엔트리들은 DIB 내에서 유일하게 자신의 엔트리를 식별할 수 있는 상대적 고유 이름(Relative Distinguished Name)을 갖는다. 또한 전체 정보 구조에서 고유 엔트리는 DIT 의 루트(root)로부터 자신의 엔트리 RDN 까지의 경로를 설정한 DN 으로 구별할 수 있다[1]. EIR 관리자는 자신의 엔터프라이즈에 적합한 객체 클래스를 미리 정의되어있는 정보 객체 클래스들 중에서 반드시 포함되어야 하는 필수 또는 선택 항목들 중에서 필요한 항목들을 선택하여 구성할 수 있다.

엔트리의 애트리뷰트들은 객체 클래스 상속으로 공동 사용이 가능하다. 이 의미는 상위 클래스에서 정의된 정보 객체를 상위 클래스의 의미론(semantics)적인 별도의 재 정의 없이 하위 클래스에서 상속 받아 똑같이 사용할 수 있음을 의미하며, 다른 속성을 갖는 애트리뷰트들을 추가하여 새로운 하위 클래스를 생성할 수 도 있다[1,6]. 즉 EIR 에서는 필요한 속성들을 상속 받음으로써 애트리뷰트 항목에 대한 관리가 용이하다. 다음은 우리 현실 세계에서 사람이 지닐 수 있는 속성 항목들을 COSINE 에서 정의한 것을 상위 클래스로 하고, 객체 클래스를 상속받아 사용자가 재 정의하는 하위 객체 클래스를 나타낸다.

?? 상위 클래스에서 정의된 person에 대한 객체 클래스

person : standardObjectClass.6 : top : CN ,surname : \
description, seeAlso, telephoneNumber,
userPassword

?? 하위 클래스인 postechPerson 에서 상위 클래스인 person의 속성들을 상속받아 해당 엔터프라이즈에서 필요로 하는 요소들을 추가하여 생성한 하위 클래스

postechPerson : postechpersonObjectClass.3 : person : CN, \
surname : socialCircle, hobby

정보 객체의 특성은 디렉토리를 구성하는 엔트리에 따라서 나타난다. 다음은 엔트리를 구성하고 있는 애트리뷰트의 타입과 값들의 속성을 선언한 것이다.

socialCircle : postechPersonAttributeType.1 : CaseIgnoreString
hobby : postechPersonAttributeType.2 :
CaseIgnoreString

POSTECH 디렉토리는 KOREA 디렉토리를 구성하는 하나의 서브 디렉토리 이다. 즉 하나의 로컬 도메인 범위로 표현되는 하나의 서브 디렉토리가 국가 단위의 디렉토리를 구성하며, 더 나아가 국가와 국가를 연결하는 World 디렉토리의 구성 요소가 된다.

POSTECH 디렉토리는 19 개의 학과로 이루어진 조직과 전체 4000 여 명의 사람들에 대한 정보들을, 조직을 구성하는 그룹과 그룹에 해당하는 사람들의 엔트리 정보들을 포함하고 있다.

그림 11 은 전 세계의 국가들의 디렉토리를 연결한 DANTE 의 World 디렉토리로서 KOREA 의 국내 마스터 디렉토리로 설정된 전자 통신 연구소의 DRAGON 도 연결되어 있다.

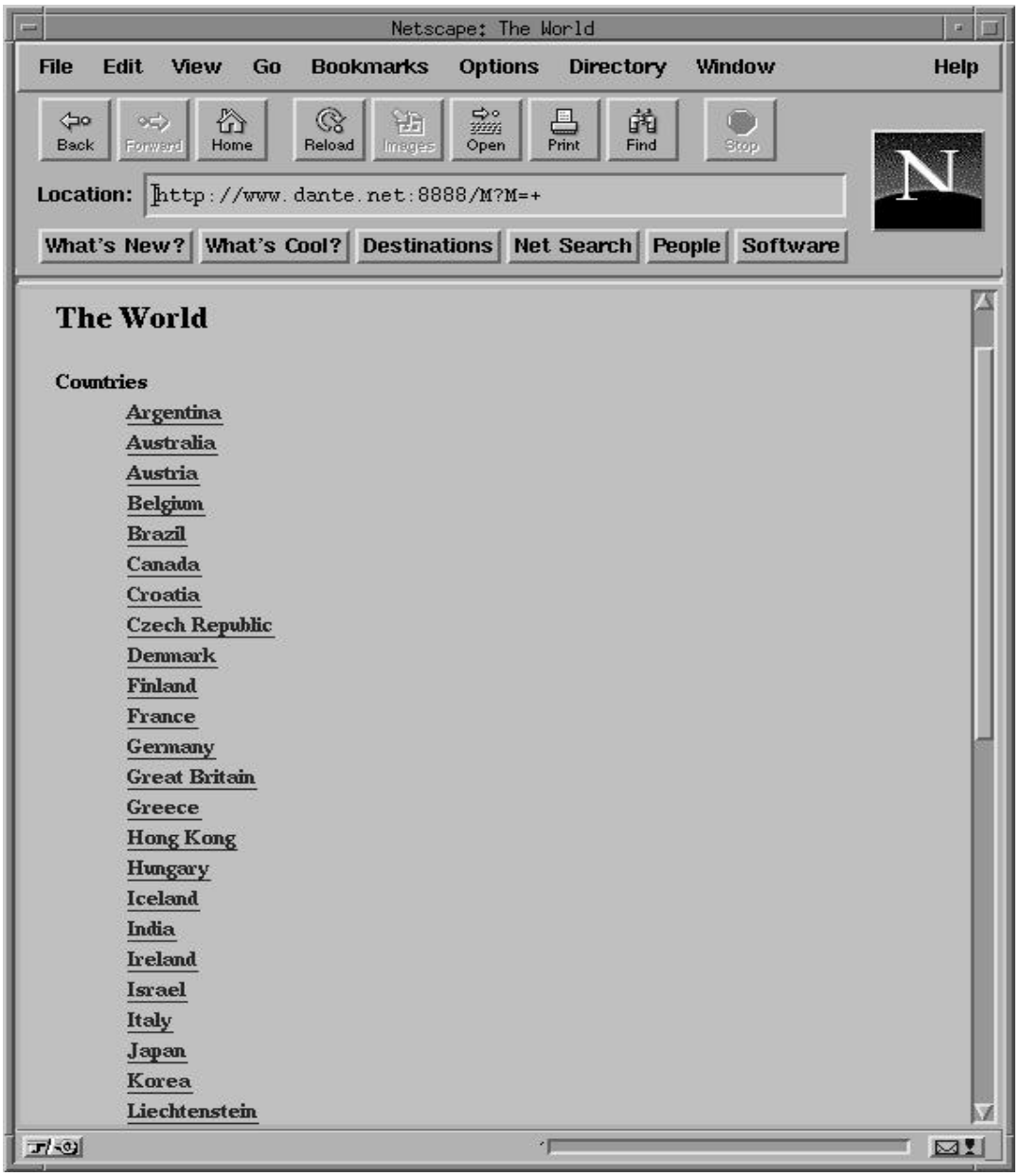

그림 11 : DANTE 의 World 디렉토리

그림 12 는 국내 마스터 디렉토리 역할을 담당하고있는 전자통신 연구소의 DRAGON 으로서, 국내 연구 기관 및 대학을 그 하위 디렉토리로 연결한다.

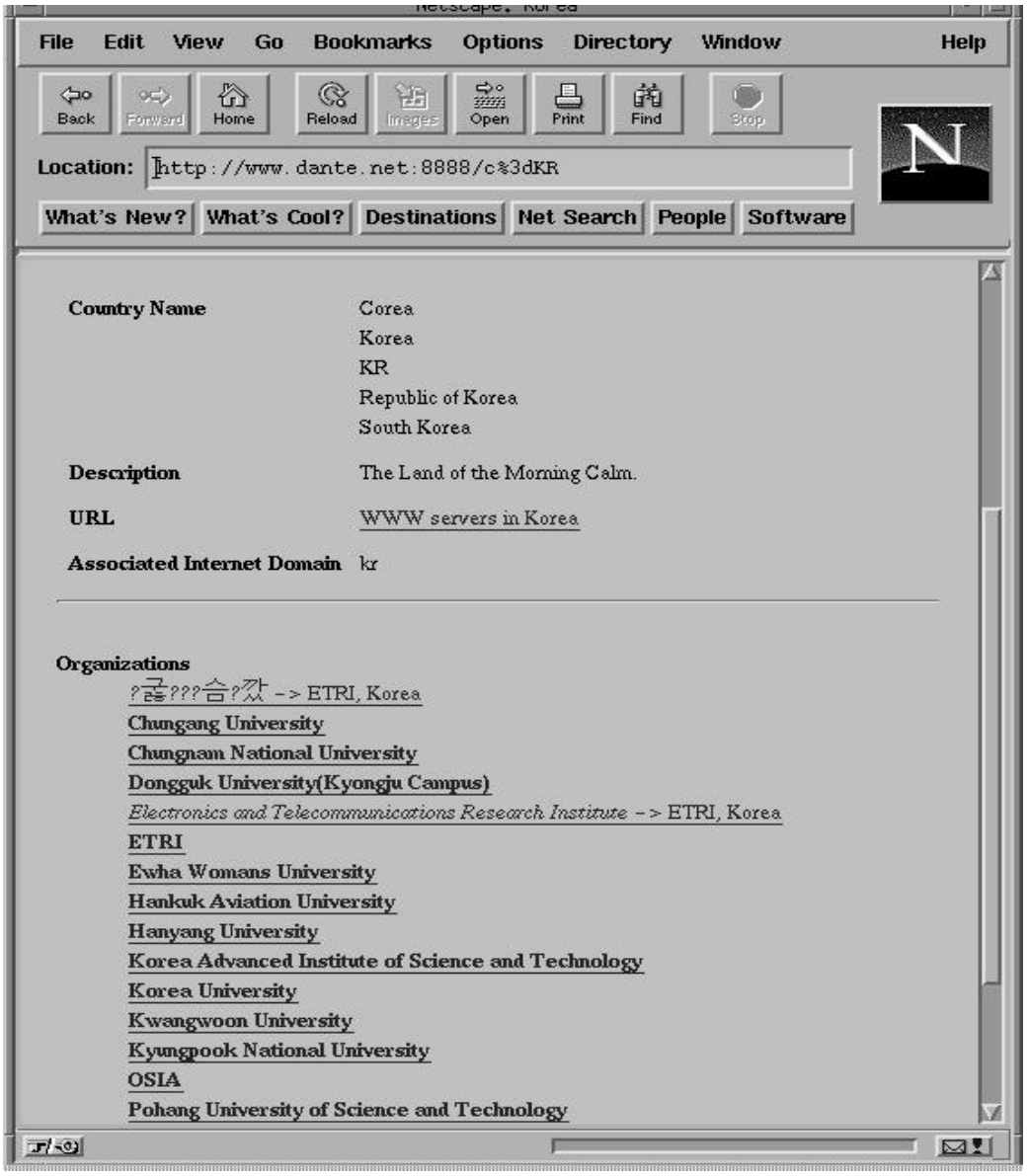

그림 12 : KOREA 마스터 디렉토리

그림 13은 DRAGON을 상위 디렉토리로 설정한 포항 공대 디렉토리인 Pohang Owl을 나타낸다. 그 예하 조직 및 사람들의 엔트리 정보에 대해서는 WWW 클라이언트인 Web500gw 에서 제공하는 부라우저나 서치(search) 도구를 통해서 원하는 조직 및 사람들의

엔트리에 접근할 수 있다.

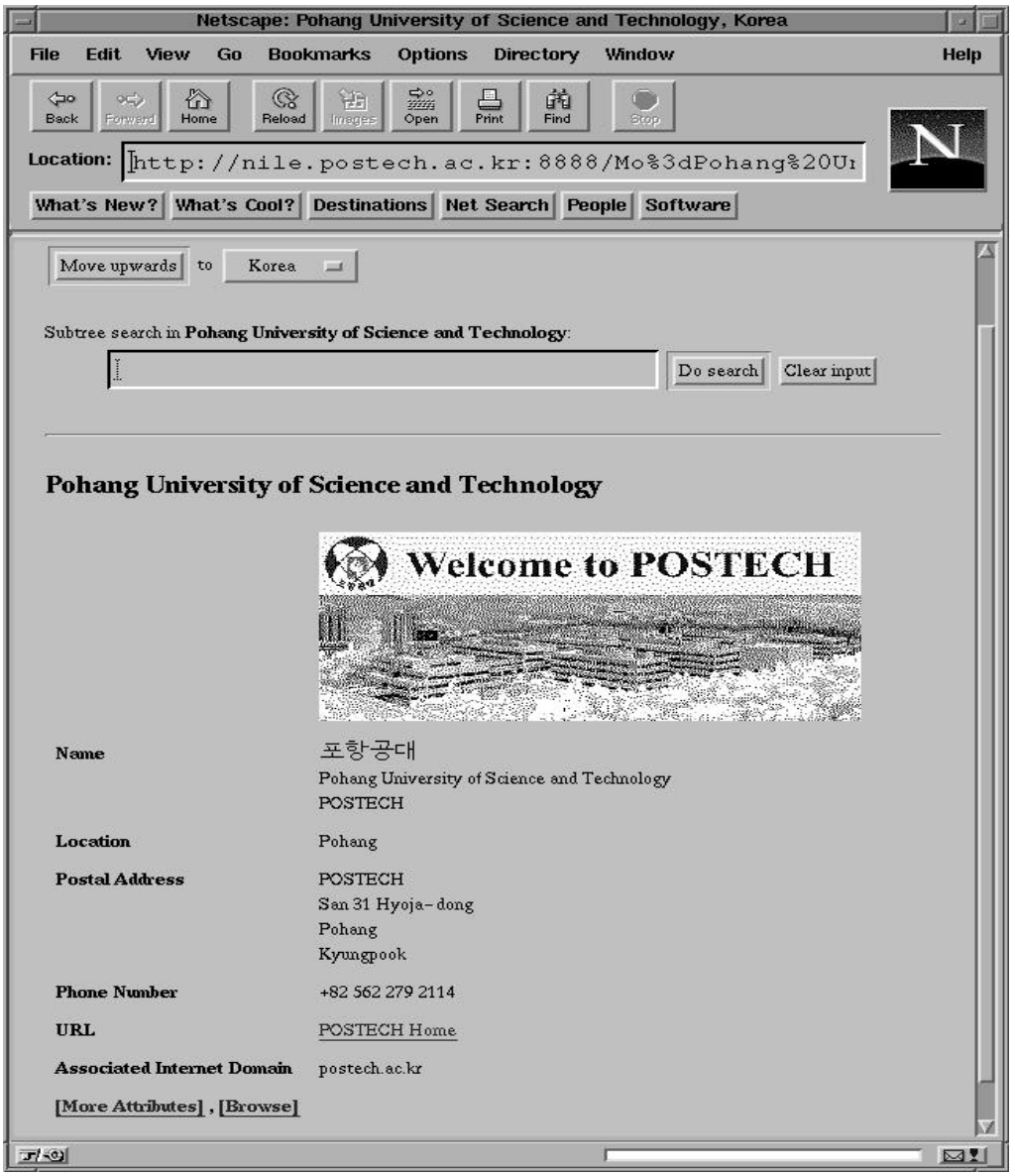

그림 13 : Pohang Owl

그림 14는 학과별로 구성된 POSTECH 조직을 나타내며, 각각의 학과는 해당 학과에 걸 맞는 POSTECH Group으로 세분 되어있다.

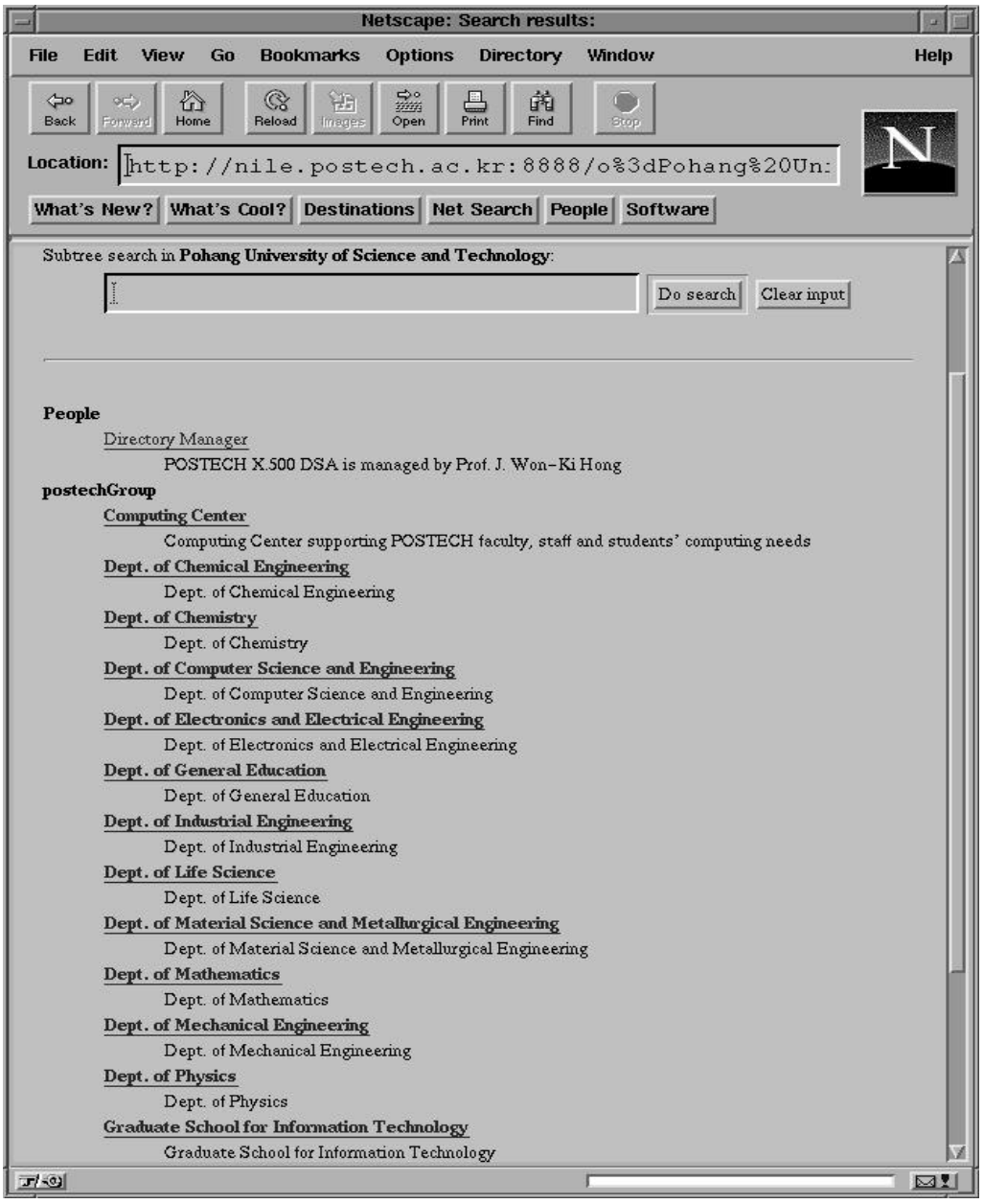

그림 14 : POSTECH Organization

그림 15는 POSTECH Group에서 포함한 사람들의 엔트리 정보를 표현하고 있다. 사람들의 엔트리는 사람을 표현한 객체 클래스를 재 정의하여 성명은 필수 항목으로, 그 밖의 나머지 항목은 선

택 항목으로 정의 하였으며, 조직 및 사람들의 엔트리에 대한 검색 및 서치(search)는 상위 단계에서 와일드 카드(Wild Card Expression) 및 정확한 비교(Exact Match)에 의한 서치(search) 방법을 이용한다[1].

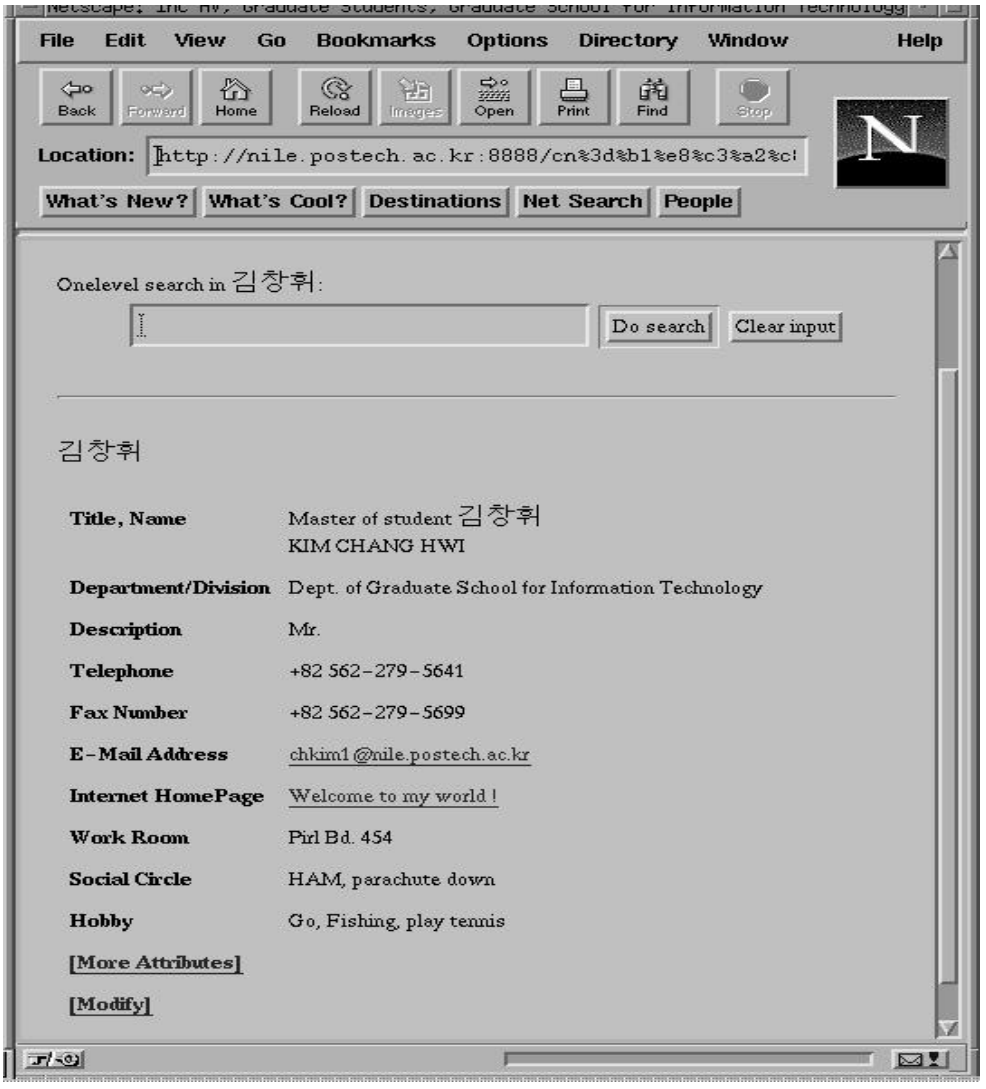

그림 15 : POSTECH 퍼스날 엔트리

그림 16 은 엔트리 정보에 대한 변경 조작을 허용하기 위해서 Simple Authentication 메커니즘[1,3,4]인 엔트리의 DN 과 패스워드

항목을 사용하여, 디렉토리 관리자나 엔트리에 대해서 접근 권한을 보유한 사람들이 엔트리 항목 및 값들에 대한 변경 조작이 가능하도록 하였다.

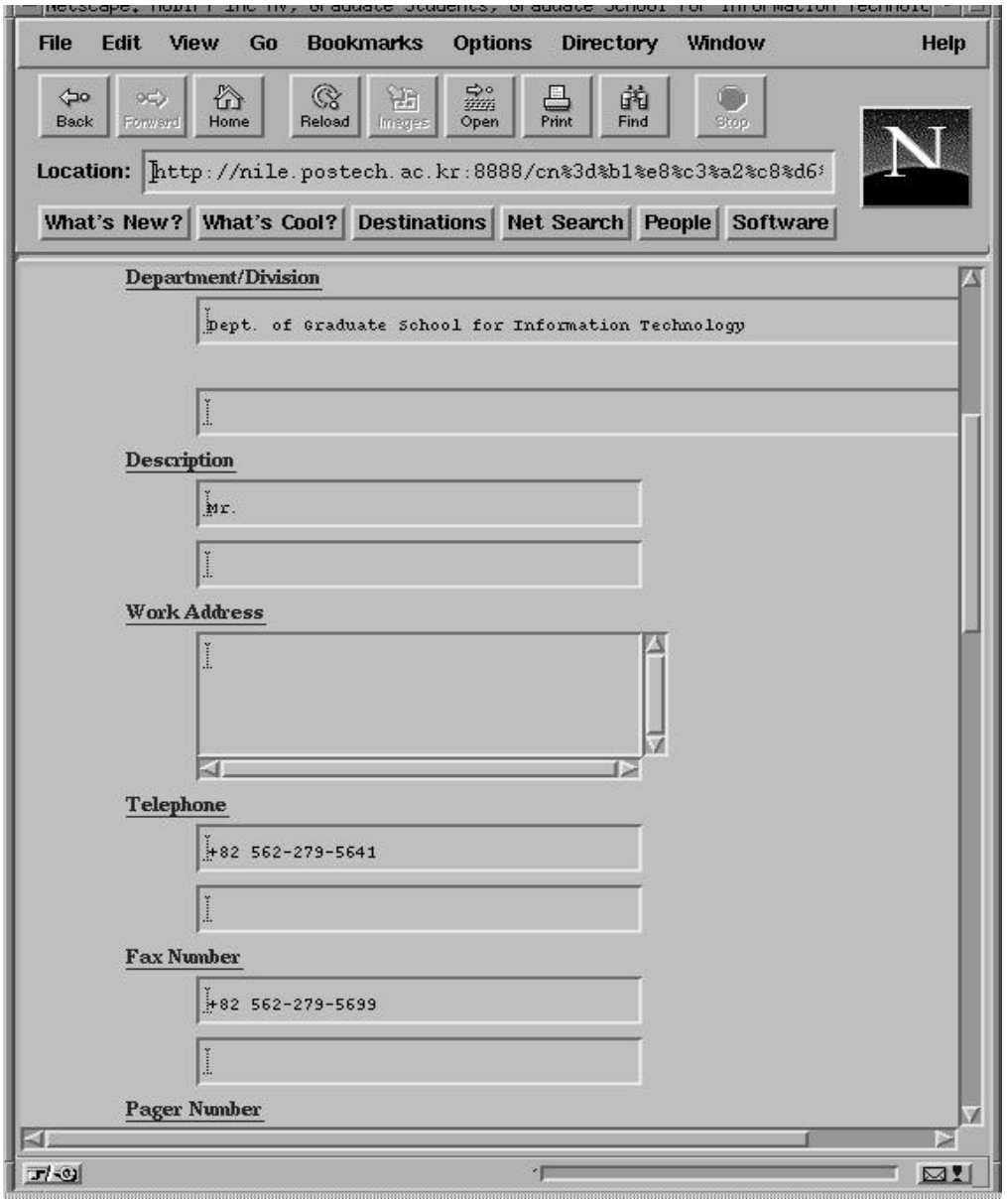

그림 16: 퍼스널 엔트리 변경

5.2 데이터 관리 및 도구

EIR 서비스 시스템에서 데이터 관리가 차지하는 부분은 중요한 역할이 아닐 수 없으며, 분산 아키텍처를 기반으로 하는 EIR에서는 더욱 세심한 주의를 요한다. EIR의 정보의 저장 형태는 종류가 다양하며, 일반적인 문자를 나열 한 것이 아니므로 세심한 주의를 요한다. 데이터 관리의 범위로는 엔트리 정보의 생성 및 입력과 삭제, 엔트리 데이터 항목의 변경 및 삭제 등으로 분류할 수 있으나 데이터 관리 대상 중 먼저 선행 되어야 할 것은 원시 데이터를 EIR의 정보 객체 표현 구성을 따르는 엔트리 정보로 생성하는 것이다.

EIR의 데이터 관리는 EIR 서비스 시스템에 의존적인 시스템 클라이언트 또는 다양한 기능과 인터페이스를 갖춘 WWW 클라이언트에 의하여 이루어 진다[17].

데이터 관리 도구가 갖추어야 할 요구 사항으로는 EIR 정보의 저장 구조 및 특성상 엔트리 정보의 입력에서 변경 삭제까지, 현재 EIR을 구성하여 서비스를 제공하는 엔트리 정보들과 비교하여 새로운 엔트리 정보를 생성하는 기능과 엔트리 정보 로드시 로드 오류에 대한 검증 절차가 마련되어야 할 것이다. 또한 방대한 양의 데이터 처리가 가능하여야 하고 엔트리 정보 조작 시 EIR 서비스 시스템이 일정시간 휴지 상태에 들어가는 것과 같은 영향을 주어서는 안되며, 데이터 조작의 일시적인 오류로 인한 중단도 허용해서는 안된다. 이는 서비스의 운영과 활용성에 문제점을 제기하는 것이 된다[17].

5.2.1 데이터 조작(Manipulation)

포항공대 EIR 서비스 시스템의 정보 저장소는 X.500 표준 안을 따라, 유닉스 파일 시스템과 같이 트리 형태를 이루는 계층 구조로 이루어져 있어 데이터 저장 및 접근에 별다른 제약이 없다는 장점을 이용할 수 있다.

엔트리 정보를 구성하게 될 원시 데이터는 기존의 다른 데이터 베이스또는 파일 시스템에서 엔터프라이즈 엔트리 정보에 필요한 초기 데이터를 추출하며, 추출된 원시 데이터는 유닉스 시스템의 응용 프로그램인 Pirl 스크립트를 활용하여 EIR 의 객체 클래스에 정의된 엔트리 정보를 내용으로 하는 엔트리 정보 리스트 파일로 생성 시킬 수 있다. 다음 코드는 학생들의 원시 데이터로부터 POSTECH EIR 에서 포함하고 있는 POSTECH 그룹 내의 학생 엔트리 정보 리스트를 생성하는 스크립트의 한 부분으로, EIR 디렉토리 스키마 규약에 따라 기술한 것이다[1,3,6].

```
$ENV{'PATH'} = '/usr/remote/bin';
open(FILE) || die "Cannot read $DATA file";
while (<READ_DATA>) {
        if (/(.+)\s(.+)\s(.+)/) {
            $variable1= $1;    $variable2= $2;    $variable3= $3;
            chop($_);
            print INIT_DATA;
    }
}
close FILE;
exit $?;
```

5.2.2 데이터 관리 도구

데이터 관리 도구는 EIR 의 정보 저장소를 대상으로 하며, 현재 EIR 을 구성하여 서비스를 제공하는 엔트리 정보들을 생성한 유닉스 엔트리 정보 리스트와 원시 데이터로부터 생성한 엔트리 정보 리스트와 상호 비교하여 현재의 EIR 에 새로이 입력 또는 삭제, 변경 등에 관한 내용을 갖는 유닉스 엔트리 정보 리스트를 생성하는 부분과, 새로이 생성된 엔트리 정보 리스트를 현재의 EIR 에 대량의 엔트리 정보를 EIR 의 저장 구조 및 위치에 맞게 로드하는 부분으로 나뉜다. 데이터 관리 도구 중에서 영국의 런던 대학에서 개발한 Bulkload 도구[17]는 이미 요구된 데이터 관리 도구가 갖추어야 할 사항들을 만족하는 기능들을 포함하고 있으며, 다음과 같은 장점을 지니고 있다.

- EIR 서비스 시스템의 운영 중단 없이 엔트리 정보의 조작이 가능하므로 Wide availability 가 높다.
- 대량의 엔트리 정보들을 한 번에 조작할 수 있다.
- EIR 을 직접 조작하는데 따른 오류 발생이 없다. 즉 엔트리 정보 조작 시 입력 오류 점검이 가능하다. 만약 사람의 수작업을 통해 정보 저장소를 직접 조작했을 경우 발생한 오류는 EIR 서비스 시스템의 운영을 중단 시킨다.
- 다양한 데이터 소스로부터 일관성 있는 EIR 엔트리 정보를 생성할 수 있다.
- EIR 의 엔트리 정보 조작에 따른 이력을 남길 수 있으므로 엔트리 정보에 대한 일관성 유지 및 회복이 용이하다.

이 논문에서 구현한 POSTECH EIR 서비스 시스템에서는 위 에서 언급한 데이터 관리 도구[17]를 이용하여 POSTECH EIR 에 대해

서 정보 저장 및 제반 관리에 대한 절차[19]를 마련하여 효율적인 데이터 관리를 수행하고 있다.

VI. 결론 및 향후 과제

중앙 집중식의 Electronic Directory에 접근하여 정보를 이용하고자 할 때는 특별한 절차 및 응용 프로그램이 요구 될 것이다. 더구나 데이터 베이스의 확장성 및 개방성에 있어서도 특별한 관리를 필요로 하며 아예 불가능한 것도 있을 것이며, 많은 양의 데이터를 보관한다 할지라도 원하는 정보를 검색하고 서비스 하는데는 많은 시간과 노력이 소요될 것이며, 비록 동일한 유형의 정보를 제공하는 다른 서비스가 있다 하더라도 해당 서버의 유용한 정보를 이용할 수 없다

이 논문에서 추구하고자 하는 목적은 점차 거대해지는 엔터프라이즈 조직의 정보를 네트워크상에서 단일의 인터페이스를 사용하여 사용자에게 정보를 제공하는 서비스를 위한 시스템을 설계하고 구축하는 것이었다. 이 목적을 이루기 위해, 정보 저장소로는 클라이언트-서버 모델을 근간으로 분산 컴퓨팅 환경을 지원하는 일종의 분산 데이터 베이스인 X.500 디렉토리 표준 안을 기반으로 설정하여 정보 저장 구조가 특정 형식에 얽매이지 않고 모든 정보

객체를 손 쉽게 정의하며, 기존의 정보 제공 서비스 시스템들이 갖는 단점들의 대 부분을 해결하고 있다는 장점을 이용하였으며, 인터넷 상에서 사용자에게 편리한 단일의 인터페이스와 인터넷 통합 서비스를 지원하기 위해 WWW 테크놀러지를 수용하였다. 그리고 구현 단계로는 EIR 서비스 시스템의 개념 모델을 제시하였으며, 이를 확장한 설계 모델에 따라 시범적으로 포항공대 정보 제공 서비스 시스템을 구현하였다.

이 논문에서는 엔터프라이즈 정보 중에서 사람들에 대한 일반적인 정보를 대상으로 하는 화이트 페이지(White Pgae) 서비스를 대상으로 하였다. 이 논문의 향후 계획으로는 X.500 디렉토리가 추구해야 할 명명(name service)서비스와 사용자 편의 위주의 명명(naming) 인터페이스 및 기존의 상용 DBMS 와의 인터페이스에 대한 연구를 포함하여, 주제별로 응용 정보를 제공하는 옐로우 페이지(Yellow Page) 서비스를 포함하는 것이다. 그러나 그보다 먼저 화이트 페이지 서비스 대상이 되는 다양한 엔터프라이즈 정보, 즉 하드웨어 또는 소프트웨어 목록 제공 및 그 밖의 도큐먼트 리스트 등에 관한 서비스를 제공하도록 확장하는 것이다.

참고 문헌

[1] International Telecommunication Union, Telecommunication(ITU-T) X.500 series of Recommendations, “Information Technology – Open System Interconnection - The Directory”, 1994.

[2] Naval Computer and Telecommunications Station, “U.S. Inter-Agency White Pages Directory, Requirements, Architecture and Concept of Operations,” Version1.0, April 26. 1995.

[3] R.Wright, A.Getchell, T.Howes, S. Sataluri, P.Lee, W. Yeong, “Recommendation for an X.500 Production Directory Service,” RFC 1803, June. 1995.

[4] Datapro 8428, “CCITT X.500: A Recommendation for Directory Service”, 1992.

[5] Douglas Steedman, “X.500, The Directory Standard and Its Application”, Great Britain, 1993.

[6] C.J Robbins, S.Kille, “The Developement Environment user’s mannual V.5 : quipu”, 1991.

[7] Ed Krol, “The Whole Internet User’s Guide & Catalog”, p210~290, O’Reilly & Associates INC, 1994.

[8] W.C, Feltstrom,The Whois++ Directory Service, Connexious, vol.8.no.12, 1994.

[9] T. Hows, “The Gopher to X.500 Gateway”, Univ. Of Michigan, 1992.

[10] S.Kille, R.Hobby, S.Kent, “A Strategic for Deployment an Internet X.500 Directory Service”, RFC 1430, Feb. 1993.

[11] C. Kim, K.T.Kim, The 23rd Kiss spring conference, pp527-530, 1996.

[12] T. Hows, M. Smith, “The LADP Application Program Interface”, RFC 1823, Univ. Of Michigan, August. 1995.

[13] Yeong. W, Howes. T, and S.Kille, “Lighweight Directory Access Protocol”, RFC 1777, Univ. Of Michigan, March. 1995.

[14] W.Young, T. Howes, and S. Kille, “X.500 Lightweight Directory AccessProtocol”, RFC 1487, Jul. 1993.

[15] DANTE, “NameFLOW-Paradise Annual Report 1994-1995”, Oct. 1995.

[16] http://www.Tu-chemnitz.de/~fri/web500gw/readme.html

[17] Barker. P, “Preparing data for inclusion in an X.500 Directory”, Research Note RN/92/41/, Univ. Of College London, May. 1992.

[18] K.S.Kim, “Enabling Customizable World-Wide Web Access to X.500 Directory”, ETRI, 1996.

[19] C.H.Kim, “POSTECH Enterprise Information Repository Service Administrtor Guide”, DPE Lab Report, POSTECH, DEC. 1996.