

인터넷 인쇄 프로토콜 시스템 구현에 관한 연구

김 수 홍[†]

요 약

본 연구에서는 인터넷 망을 기반으로 하는 인쇄 표준인 인터넷 인쇄 프로토콜(IPP)에 대하여 소개하고 IPP 클라이언트/서버 시스템 구현을 위하여 각 클라이언트와 서버의 기능에 대해 자세히 논의하였다. 특히 효율적인 IPP 클라이언트/서버 시스템 활용을 위하여 사용자그룹, 프린터그룹 등의 관리기법을 채택하여 시스템 운영자와 관리자의 기능을 강화하고 효과적으로 IPP 시스템 사용자나 프린터를 관리할 수 있도록 구현하였다. 또한 IPP 표준을 사용하는 타 회사 제품과의 호환성을 최대한 확보하는데 노력하였다. 그리고 마지막으로 앞으로의 연구 방향을 제시하였다.

A Study of Implementation for Internet Printing Protocol (IPP) System

Soo-Hong Kim[†]

ABSTRACT

This paper describes the IPP (Internet Printing Protocol), a standard that makes network setup for printers potentially much easier and, not so incidentally, also user can print over the internet and specifies an implementation of IPP client/server system. It allows the system administrator and operators to control IPP system users and printer devices. The focus of this effort is optimized capabilities the security features for authentication, authorization, and policies, also improved compatibility with existing IPP devices. Finally this paper presents conclusions and further researches.

키워드 : 프린터(Printer), IPP(Internet Printing Protocol), 인터넷 인쇄 드라이버(Internet Printing Driver), PWG(Printer Working Group), TCP/IP, SSL(Secure Socket Layer), TLS(Transport Layer Security), HTTP(Hyper Text Transfer Protocol), 사용자, 관리자, 운영자

1. 서 론

본 연구에서는 인터넷 망을 기반으로 하는 인쇄 표준인 인터넷 인쇄 프로토콜(IPP)에 대하여 소개하고 IPP 클라이언트/서버 시스템 구현을 위하여 각 클라이언트와 서버의 기능에 대해 자세히 논의하였다. 특히 효율적인 IPP 클라이언트/서버 시스템 활용을 위하여 사용자그룹, 프린터 그룹 등의 관리기법을 채택하여 시스템 운영자와 관리자의 기능을 강화하고 효과적으로 IPP 시스템 사용자나 프린터를 관리할 수 있도록 구현하였다. 또한 IPP 표준을 사용하는 타 회사 제품과의 호환성을 최대한 확보하는데 노력하였다. 그리고 마지막으로 앞으로의 연구 방향을 제시하였다.

컴퓨터 기술의 발달은 많은 분야에서 변화를 일으켰으며 특히 인터넷은 단기간에 비약적인 기술의 발전으로 많은 변화를 가져오고 있다. 인터넷의 비약적인 발전은 여러 분야에서 사용자들에게 편리함을 제공하고 있다. IETF에서 제안하

고 있는 인터넷 인쇄 프로토콜(Internet Printing Protocol : IPP)는 전자우편(e-mail)과 팩스의 장점을 수용하여 사용자들에게 보다 편리하고 저렴하게 문서를 전송할 수 있는 방법의 일종으로 문서를 직접, 원격지의 프린터에 출력할 수 있게 하는 몇 가지 방법과 프로토콜들을 제안하고 있다. IPP와는 별도로 몇몇 기업에서 웹을 기반으로 하는 자신들만의 독자적인 원격지 인쇄 솔루션을 개발하여 공급하고 있다. 이러한 솔루션들은 서로 다른 솔루션간에 상호동작하지 않는 단점이 있다. 이로 인하여 각각의 프린터 공급자들이 각자의 하드웨어에 맞는 다른 전송 프로토콜을 구현하여 공급하여 상호 호환성이 떨어지게 된다. 따라서 인터넷을 통해서 대부분의 일반적인 상황에 적용될 수 있는 프로토콜을 정의할 필요가 있다. 이러한 요구를 표준화 하기 위한 프로토콜이 IPP인 것이다. IPP는 인터넷 및 인트라넷상의 인쇄 작업의 전송, 감시 그리고 관리를 위한 표준을 정의한다[1].

IPP는 전체 인터넷 인쇄 드라이버(Internet Printing Driver)에 대한 중추 역할을 한다. IPP는 인터넷을 통해 널리 사용

* 본 논문은 상명대학교 2003년도 교내연구비 지원에 의하여 연구되었음.

† 종신회원 : 상명대학교 컴퓨터정보통신공학부 교수

논문접수 : 2003년 4월 22일, 심사완료 : 2003년 10월 30일

되는 HTTP(Hyper Text Transfer Protocol)를 이용한다. 제록스(Xerox)사, 아이비엠(IBM)사, 마이크로소프트(Microsoft)사, 노벨(Novell)사 및 휴렛팩커드(HP)사와 같은 30여 개의 프린터 제조업체와, 프린터 어플리케이션 개발업체의 지원을 받는 Printer Working Group(PWG)을 통해 현재 IETF에서 표준개발이 이루어지고 있다.

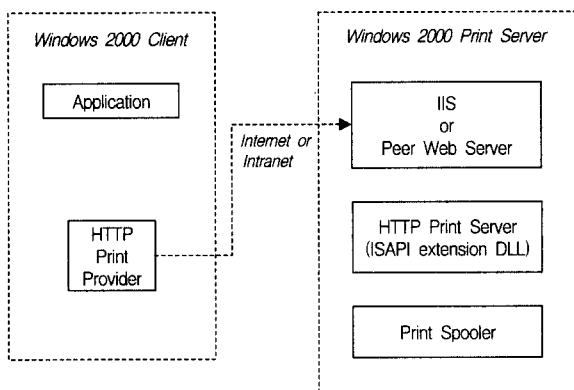
IPP를 기반으로 한 관련 제품들은 현재 주로 하드웨어 개발사를 중심으로 기본 전송 프로토콜을 롬에 저장한 형태의 제품으로 출시되고 있다. 저가형 프린터 서버에서부터 휴렛팩커드사의 고급 프린터 서버까지 다양한 형태의 하드웨어 제품이 개발되고 있다. 마이크로소프트사에서도 윈도우 제품들을 출시하면서 IPP를 지원하겠다고 공약하였고 이러한 공약의 일부로서 기본적인 기능을 제공하고 있다. 그러나 이러한 지원은 하나의 지원정책에 머물고 있으며 이것을 기반으로 한 클라이언트 서버 제품은 전무한 실정이며 실제로 일반 기업에 맞추어 고객화, 통합화 되어야 할 부분에 대한 언급이 미흡한 상태이다. IPP는 다양한 분야에서 사용될 수 있는 가능성을 가지고 있으며 이 때문에 프린터를 만드는 제조업체, 프린터를 이용한 사업을 하려는 출력소, 포털에서도 IPP와 관련된 사항을 주시하고 있다.

본 연구에서는 위와 같은 필요성에 따라 PWG에서 제안하고 있는 IPP 규격에 따르는 IPP 서버/클라이언트 시스템 구현에 대해 논하고자 한다. 본 논문의 구성은 다음과 같다. 2장에서 관련연구에 대해 살펴보고, 3장은 IPP 시스템의 전반적인 개요를 기술하였다. 4장은 시스템을 구현하고, 5장에서 결론 및 향후 연구방향을 제시하였다.

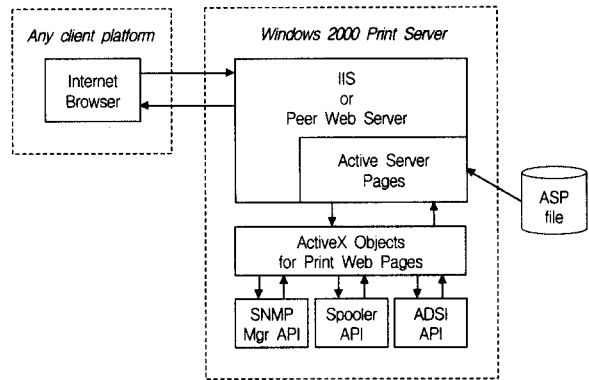
2. 관련 연구

2.1 마이크로소프트사의 IPP 지원 정책

현재 윈도우 95, 98을 위한 IPP 지원 업그레이드를 제공하고 있고 윈도우 2000 제품군과 XP 제품군에서는 다음(그림 1), (그림 2)와 같은 인터넷 인쇄 클라이언트(Internet Printing Client)와 서버(Server)를 제공한다[2].



(그림 1) Printing to an URL-Identified Print Queue



(그림 2) Viewing Print Web Pages

2.2 데스크탑 투 딜리버리 솔루션(Desktop to Delivery Solutions)

한국어도비시스템즈사는 2002년 3월초, 새롭게 부상하고 있는 인터넷 프린팅 시장 공략을 위해 인쇄업자를 위한 B2B 전자상거래 솔루션 제공 업체인 웹 프린트와 공동 마케팅을 펼치기로 했다. 이로써 동사의 PDF 기술 및 다른 인터넷 프린팅 기술에 웹 프린트의 '데스크탑 투 딜리버리 솔루션(Desktop to Delivery Solutions)'이 통합된다.

이 인터넷 기반의 인쇄 서비스는 인터넷 인쇄 드라이버와 표준 웹 브라우저의 기능을 포함하고 있어 보다 저렴한 비용으로 오류 없는 인쇄작업을 지원한다. 또한 이 서비스는 단기적인 주문형 인쇄 서비스들을 목표로 하고 있지만, 궁극적으로는 로컬(local) 데스크탑 프린터를 이용하는 것처럼 간단하고 안전한 인쇄작업을 지원한다.

참고로 웹 프린트의 '데스크탑 투 딜리버리 솔루션'은 고객들이 데스크탑 프린터를 사용하는 것만큼 쉽게 자신의 작업물을 그들이 원하는 지역으로 전송해 인쇄할 수 있게 해준다.

이 솔루션은 어도비사의 그래픽 제품과 마이크로소프트사의 오피스 등 모든 문서작성 프로그램을 이용할 수 있으며, 작업 후 인쇄를 주문하면 웹 프린트 시스템이 그 문서를 PDF 파일로 변환시킨다. 온라인 상으로 고객이 재검토를 끝내면 인쇄 준비가 완료된 파일에 자동으로 가격이 책정되어 인쇄업자에게 전송되며, 이때 인쇄용지의 색깔과 타입, 제본방식, 표지재료, 인쇄물량, 배송 등 구매자의 요구 사항들도 함께 전송된다. 이로써 고객들은 아무런 손상없이 자신들이 원했던 문서를 원하는 방식대로 인쇄, 제본, 마무리할 수 있다.

2.3 Native File Access

노벨사의 NFA(Native File Access)는 가장 중요한 파일 서비스 기능으로, 네트웨어(Netware 6) 서버와 윈도우, 유닉스, 매킨토시 클라이언트와의 직접적인 연결을 가능하게 한다. NFA를 사용하면 노벨사의 거추장스럽고 부담스러운 클라이언트 소프트웨어를 설치하지 않고도 네트워크 드라이버를 네트웨어(Netware6) 서버에 매핑(mapping)할 수 있다.

노벨사는 iForder나 iPrint와 같은 새로운 기능들이 LAN은 물론 인터넷 기반의 작업 영역에까지 확장되기를 바라고 있다.

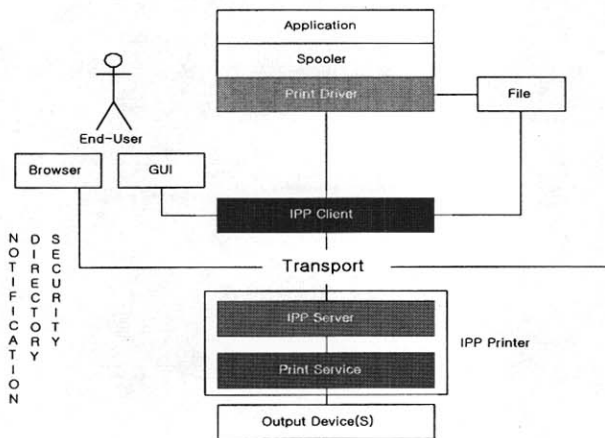
IETF의 IPP를 기반으로 구축된 iPrint는 유용하고 새로운 인쇄 서비스로 클라이언트에서 웹 브라우저를 통해 인쇄 작업을 초기화할 수 있다. 클라이언트는 브라우저를 사용해 프린터를 찾을 수 있으며, iPrint는 작업 완료에 필요한 드라이버를 다운로드, 설치한다. iForder 틀을 사용하면 파일을 업로드/다운로드할 수 있으며 브라우저를 통해 파일 동기화도 실행할 수 있다.

3. IPP 시스템

3.1 시스템 개요

인터넷 인쇄 프로토콜(IPP)는 클라이언트/서버로 구성되어 있으며, 사용자는 개인용 컴퓨터나 워크스테이션 상에 위치하고 서버는 프린터 안에 내장(Embedded) 되거나 프린터 서버로서 분리된 기계에 놓인다. 사용자는 항상 그 서버와 통신을 함으로서 초기화되고, 그 이후부터 프린터를 시작, 종료할 수 있으며 문서의 인쇄 방식이나 페이지 수, 인쇄물의 개수, 인쇄 품질 등을 명시할 수 있다. 또한 사용자는 자신이 보낸 문서에 대해 실시간으로 상태를 확인 할 수 있으며, 서버에서는 여러나 상황 메시지를 통해 사용자에게 내용을 통보한다. 이때 사용자가 보낸 데이터를 보호하기 위해 IPP는 보안을 위해 SSL3 또는 TLS를 사용한다. TLS는 인터넷에 대한 보안을 위한 표준으로 IETF에 의해 개발되고 있으며, 사용자(User), 서버(Printer or Proxy) 양쪽의 요청, 응답과 문서 데이터를 암호화하고 확인한다.

IPP의 계층도는 (그림 3)과 같이 표현할 수 있다[3,4].

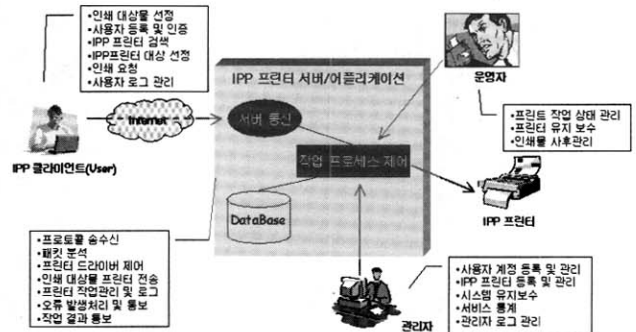


(그림 3) IPP 계층도

3.2 설계 목적

IPP는 (그림 4)와 같이 IPP 클라이언트인 사용자(End-

User)와 운영자(Operator) 그리고 관리자(Administrator)의 세가지 유형의 관계자의 관점에서 설계되었다[5-7].



(그림 4) IPP 구성요소 및 사용사례

각 관계자별 권한을 살펴보면 다음과 같다.

사용자는 프린터에 인쇄할 작업을 보낼 사람을 의미한다. 사용자가 필요로 하는 작업은 다음과 같은 6가지 작업으로 구분된다.

- ① Finding or locating a printer : 인쇄를 할 수 있는 권리를 부여 받은 프린터를 찾고 어디에 위치해 있는지 찾을 수 있어야 한다. 이 기능은 표준의 웹 브라우저나 다른 응용 프로그램을 이용해서 할 수 있다. 프린터를 찾을 때 프린터의 이름, 지리적위치, 프린터의 기능(color, duplex, etc.)등을 포함하는 여러 기준들이 적용될 수 있다.
- ② Create an instance of the printer : 원하는 프린터를 찾은 후에는 사용자의 운영체제 또는 데스크탑에 프린터 로컬 인스턴스를 만들어야 할 필요가 있다. 저장되어있는 어떤 곳으로부터 드라이버를 다운로드하여, 컴퓨터에 설치할 수 있어야 한다. 초기 승인이후에는 아무런 동작 없이 설치될 수 있어야 한다.
- ③ Viewing the status and capabilities of a printer : 프린터의 현재 상태 및 프린터가 지원하는 기능들을 볼 수 있어야 한다.
- ④ Submitting a print job : 프린터가 지정되고 설치되면 사용자는 일반적인 방법을 사용하여 보통 어플리케이션으로 프린트 할 수 있어야 한다. 여기서 보통 어플리케이션은 워드프로세서, 스프레드시트, 웹 브라우저 등을 말한다. 사용자는 레퍼런스에 의해 인쇄물을 보낼 수 있다. 레퍼런스, 즉 URI는 실제적인 프린트가 발생하기 전에 해결되어야 한다.
- ⑤ Viewing the status of a submitted print job : 사용자는 잡(Job)의 상태를 볼 수 있어야 하고, 프린터 큐에서의 인쇄물이 어디에 놓일지 결정할 수 있어야 한다.
- ⑥ Canceling a print job : 인쇄물이 인쇄를 기다리는 동안이나, 시작 후 아직 완성되지 않을 경우, 원래의 생성자는 그 인쇄물을 취소할 수 있어야 한다.

운영자는 프린터의 인쇄물을 관리하고 통제하는 동안에는 프린터의 상태를 모니터링하여 응답해야 한다. 또한 운영자의 작업은 모든 사용자의 작업을 포함한다.

관리자는 프린터의 인스턴스를 생성하여 응답하고, 다른 사용자와 운영자의 권한을 통제한다. 운영자와 관리자는 IPP Version 1.0에서는 지원되지 않는다[8].

3.3 특 징

3.3.1 독창성, 안전성

- 인터넷을 이용하여 원격지의 프린터로 사용자가 직접 인쇄 목적물을 출력할 수 있다.
- 단순 이미지 파일의 전송이 아닌 목적지 프린터로 SSL/TLS 보안 프로토콜을 통한 목적 파일 전송으로 안전성을 보장한다.

3.3.2 혁신성

- 종전 기술과의 비교하면 현재까지 사용되는 네트워크 인쇄 공유 방식은 대부분 이더넷(Ethernet) 혹은 토큰링(Token Ring) 네트워크를 지원하였으나 IPP는 HTTP를 사용함으로써 인터넷 자체를 지원한다고 볼 수 있다[8].

3.3.3 기술적 파급효과

- 기존의 팩스, 우편, 전자우편등 모든 문서 송수신 방법의 대체 방안이 될 수 있다.
- 복사기 및 디지털 통합 장비 시장의 대응품이 될 수 있다.

또한 IPP 표준은 다음과 같은 장점이 있다.

- 어떤 종류의 컴퓨터에서도 네트워크와 연결된 프린터로 인쇄할 수 있다.
- 사용자는 프린터 상태를 결정할 수 있다.
- 리모트 지역에서 인쇄 상태를 모니터링할 수 있다.

3.4 시스템 구현 사례

IPP 시스템의 구현은 구현 방법에 따라 적어도 다음의 4가지 정도로 구분할 수 있다.

3.4.1 Client - One print

사용자와 프린터 둘 다 IPP로 구현된 경우로, 사용자는 직접 그 네트워크에 연결된 프린터에 프로토콜 요청을 보낼 것이다.

3.4.2 One client - Gateway server - One printer

이 구성은 주로 IPP를 지원하지 않고 큰 단위 처리량을 가지고 있는 큰 프린터를 위해 사용될 것이다(예 : 출력소).

3.4.3 One client - Gateway server - Multiple printers

이 구성은 많은 프린터가 같은 네트워크상에서 사용하기 위해 게이트웨이를 제공하는 경우에 사용될 것이다(예 : Workgroup Printer 또는 Printer Pool).

3.4.4 One client-[Gateway server-Repository]-Printer
 사용자가 문서를 참조에 의해 프린트를 요구할 때 사용된다. 서버는 IPP를 이용해서 클라이언트로부터 요청을 받아 저장소에서 그 문서를 회수하고 나서 문서를 인쇄한다[9].

4. IPP 시스템 구현

본 연구에서는 IPP 시스템을 구현하기 위한 프로그래밍 툴(tool)로 Visual C++가 사용되었다. 본 연구로 제작된 IPP 클라이언트/서버 시스템은 본문에서 OKPrinter 시스템으로 명명하였다.

OKPrinter 시스템은 SSL과 TLS를 적용시켜 표준규약에 따른 보안기능을 따르도록 했다.

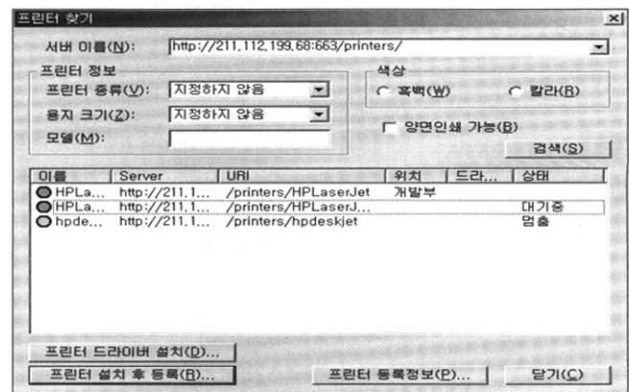
IPP 시스템의 구성요소로는 크게 클라이언트/서버로 나눌 수 있고 다시 세분하여 각 기능별로 다음과 같이 구현하였다[7].

4.1 OKPrinter 클라이언트

클라이언트의 기능으로는 크게 프린터 찾기, 프린터 관리, 문서 전송의 세 가지 기능으로 나뉘 볼 수 있다. 그리고 클라이언트에서는 다음의 순서로 작업을 진행한다.

- ① 프린터 찾기 : OKPrinter 서버에서 지정한 서버 이름 "http://서버IP : 서버포트/URL Prefix/"을 입력
- ② 프린터 설치 : 프린터 찾기가 끝난 후, 사용을 원하는 프린터를 설치 및 등록
- ③ 인쇄보내기 : 인쇄를 원하는 문서를 열고 원하는 프린터를 선택한 후 인쇄

(그림 5)에서 3.2 설계 목적에서 논의한 것처럼 사용자(End-User)와 운영자(Operator) 그리고 관리자(Administrator)의 세 가지 유형의 관점에서 구현하였다.



(그림 5) OKPrinter Client

4.1.1 IPP 클라이언트(사용자 : End-User)

- ① 조건(서버 이름, 색상, 프린터 종류, 용지크기, 해상도, 프린터의 온라인 등)을 지정하여 프린터를 찾을 수 있다.
- ② 검색된 프린터에 대한 자세한 정보는 사용자가 확인

할 수 있다.

- ③ 드라이버의 다운로드 위치를 확인하고 다운/설치 후 프린터에 인쇄하고자 하는 문서를 보낸다.
- ④ 사용자는 그 프린터의 사용자로서 가입이 되어있어야 프린트 작업을 할 수 있다.
- ⑤ 사용자 등록 후 사용자는 그 프린터(상태, 위치, 모델 번호, 드라이버 등)에 대한 자세한 설명을 볼 수 있다.
- ⑥ 전자우편을 통한 통보여부나 마감처리, 보안등에 대해 사용자가 설정을 할 수 있다.
- ⑦ 사용자가 프린트 작업을 원하는 프린터에 보내면 사용자는 그 프린터에 대한 모니터링을 할 수 있다.
- ⑧ 문서등록정보는 다음 세 단계로 구분되며 제공되는 정보는 다음과 같다.
 - 일반정보 : 문서 이름, 문서 URI, 문서 ID, 크기, 페이지 수, 데이터 형식, 프로세서, 소유자, 제출날짜 등
 - 세부정보 : 작업 시작시간과 완료시간, 마감처리, 통보 여부 등
 - 서버정보 : 프린터 이름, 위치, 설명, 모델번호, 프린터 드라이버의 URI, 이름, 부서 IPP의 버전 등[10].

4.1.2 관리자

- ① 사용자 추가시 사용자 정보(이름, 설명), 사용자 ID, 사용자 암호를 설정한다.
- ② 사용자의 암호는 MD5로 암호화되어 있다.
- ③ 사용자가 포함될 소속 그룹을 설정하고 임의의 사용자는 여러 그룹에 포함 될 수 있다.
- ④ 사용자 그룹 추가시 그룹정보(이름, 설명), 구성원 설정, 제한 범위(작업시간, 페이지 수, 파일 크기),
- ⑤ 사용자 추가 권한, 관리자 추가 권한, 사용 가능한 프린터를 추가/삭제 할 수 있다.
이때 사용자의 ID 및 그룹명은 시스템에서 유일해야 한다.
- ⑥ 사용자 및 사용자 그룹을 삭제 및 수정할 수 있다.
- ⑦ 사용자 정보에 표시의 필수 사항은 사용자/그룹별 정렬이 가능하고 임의의 사용자를 찾을 수 있다.
- ⑧ 사용자 정보를 실시간으로 확인 할 수 있다.

4.1.3. 운영자

- ① 프린트 추가 필수 입력 사항은 프린터 이름, 프린터 설명, 설치된 프린터 URI(URL, IP)이다.
- ② 프린터 추가의 경우 선택 입력 사항은 용지크기(A4/A3/B4 ...), 프린터의 종류(레이저/잉크젯/도트/플로터 ...), 해상도, 양면지원, 사용가능 여부, 부서, 지역, 프린터의 최대 인쇄 가능 페이지 수, 페이지 당 인쇄 수 설정.
- ③ 프린터의 사용권한은 임의의 사용자, 특정 IP 및 IP 그룹별로 허가 및 제한 할 수 있다.
- ④ 임의의 프린터에 각 OS별 프린터 드라이버를 추가할 수 있다.
- ⑤ 클라이언트가 접속시 자신의 OS 정보를 서버로 알려주며, 서버는 클라이언트 OS 환경에 맞는 프린터 드

라이버의 URI를 제공한다.

- ⑥ 프린터 및 프린터 그룹을 삭제 및 수정할 수 있다.
- ⑦ 특정 프린터 및 프린터 그룹의 프로세스를 정지/다시 시작 할 수 있다.
- ⑧ 프린터의 상태를 실시간으로 확인 할 수 있다.

4.2 OKPrinter 서버

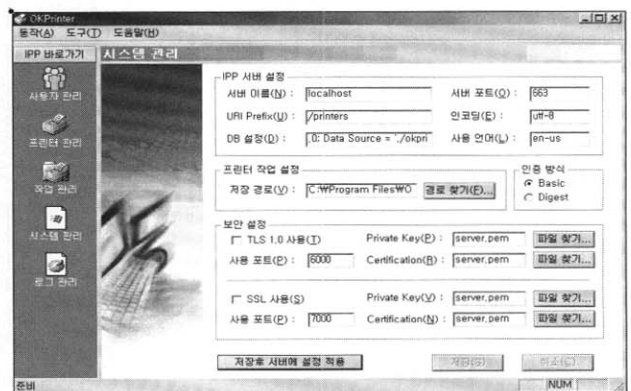
서버의 기능으로는 사용자관리, 프린터 관리, 작업관리, 시스템 관리, 로그관리의 다섯 가지로 분류할 수 있고 이에 준하여 OKPrinter 서버를 구현하였다[11].

전체적인 서버의 기능을 살펴보면 아래와 같다.

- ① 사용자가 요청한 모든 인쇄물의 목록을 나열한다.
 - ② 프린터 및 사용자별 정렬이 가능하다.
 - ③ 임의의 사용자 프린터에 의한 찾기 기능을 지원한다.
 - ④ 로그 및 인쇄물의 유효기간을 설정할 수 있다.
 - ⑤ 프린터 당 최대 대기열 수를 고려한다.
 - ⑥ 시스템 설정을 실시간으로 확인 할 수 있고 수정할 수 있다.
 - ⑦ 인쇄되어진 모든 작업 목록을 시스템 관리에서 설정한 임의의 기간까지 보존한다.
 - ⑧ 보존되어진 임의의 문서는 운영자 및 관리자에 의해 임의 출력될 수 없다.
 - ⑨ 사용자는 클라이언트에서 임의 기간에 보존되어 있는 자신의 목록을 확인하고 반복 출력할 수 있다.
- 각 세부관리 기능을 살펴보면 다음과 같다.

4.2.1 시스템 관리

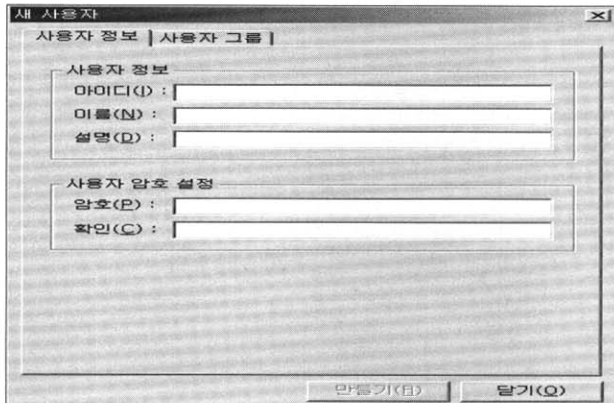
- IPP 서버 설정 : 서버이름, 서버포트, URI Prefix, 인코딩, DB설정 및 사용언어를 설정
- 프린터 작업설정
- 인증방식 : 인증방식은 Basic과 Digest로 나뉘어 있으며, Basic으로 기본 설정
- 보안 설정 : 보안설정을 위하여 TLS, SSL을 선택적으로 이용가능



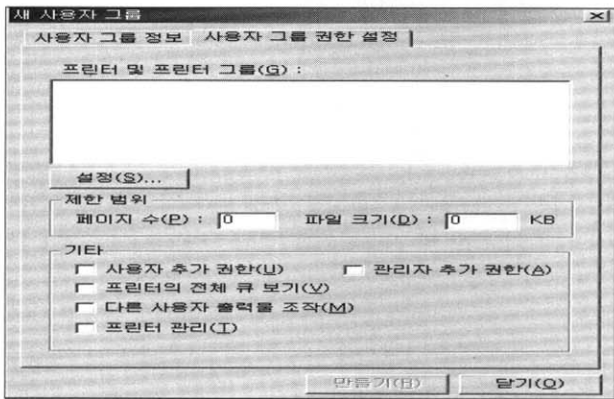
(그림 6) 시스템 관리

4.2.2 사용자 관리

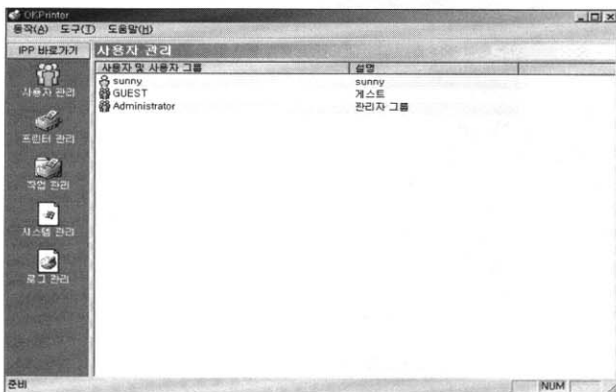
- 모든 사용자의 리스트가 보여지며, 프린터그룹의 사용 권한을 부여할 수 있다.
- 사용자는 관리자 그룹과, 게스트 그룹 등으로 나누어 등록할 수 있으며, 사용자에 관한 등록정보 확인이 가능하다.
- 그룹별 관리는 IPP 표준규약에는 포함되지 않은 OKPrinter의 부가적인 기능으로 사용자와 프린터를 그룹별로 관리함으로써 시스템 관리에 효율성을 더하였다.



(그림 7) 사용자 등록화면



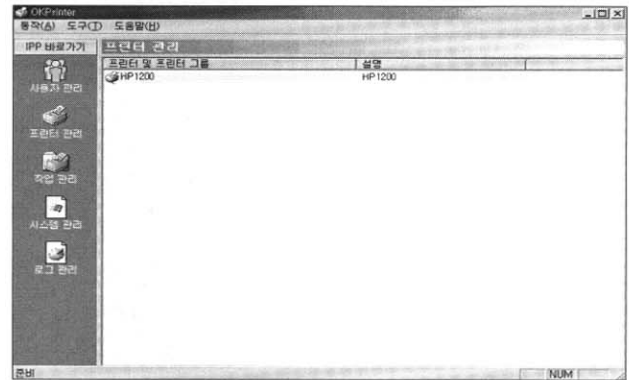
(그림 8) 사용자그룹 설정화면



(그림 9) 사용자 관리

4.2.3 프린터 관리

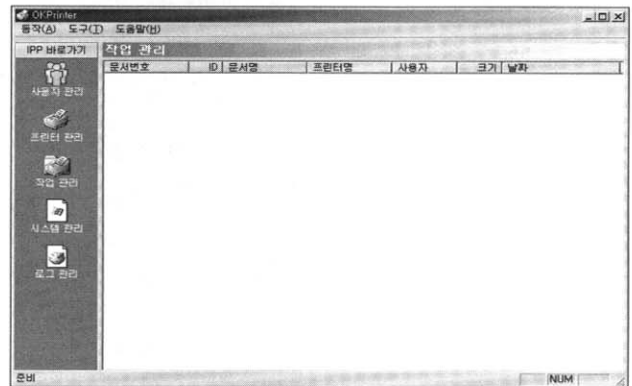
- 사용을 원하는 프린터를 미리 등록한 후 OK Printer 서버에서 프린터 관리를 선택
- 모든 프린터에 대한 리스트가 보여지며, 관리자가 기본 프린터를 설정할 수 있다.



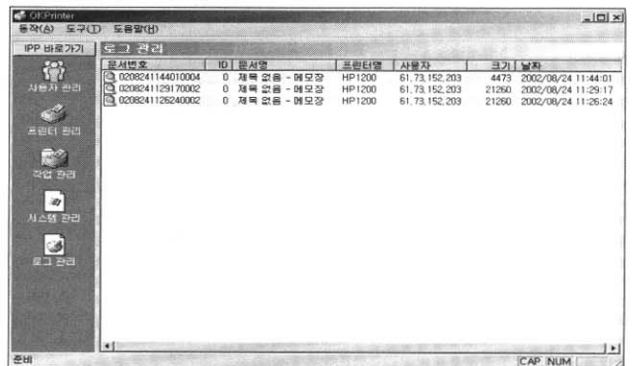
(그림 10) 프린터 관리

4.2.4 작업관리

- 현재 작업중인 문서 ID, 문서 이름, 프린터 이름, 소유자, 페이지 수, 크기, 날짜 등 확인 가능
- 관리자는 현재 진행중인 문서를 일시중지/계속/다시 시작/취소 할 수 있다.

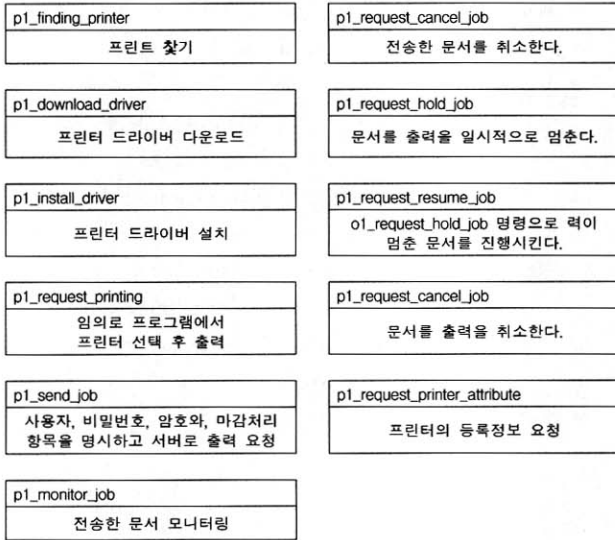


(그림 11) 작업 관리



(그림 12) 로그 관리

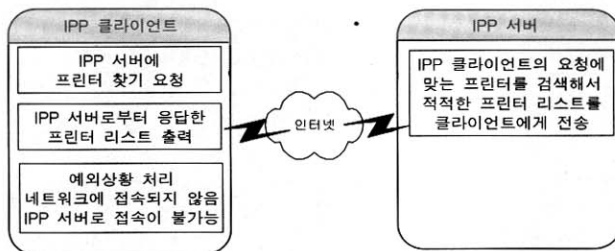
4.3 주요 프로세스 설명 및 다이어그램



(그림 13) 주요 프로세스

4.3.1 프린터 찾기(p1_finding_printer)

- 프로세스명 : p1_finding_printer
- 대응 프로세서 : p3_finding_printer
- 관련 프로세스 : p1_request_print_attributes
- 사용 클래스
 - Dialog : OKPrinterClient.CFindPrinterDlg
 - Socket : OKPrinterClient.CClientSocket, OKPrinterClient.CXHttpRequestObject
 - Packet : OKPrinterClient.CXPrintPacket, OKPrinterClient.CXAttributePacket, OKPrinterClient.CXValuePacket
- 주요 사용 함수 및 객체
 - 제작 API(GetInfo : http://www.pppp.uuuu 부분을 호스트 이름, 포트, URI로 구분해 낸다).
- 프로세스 다이어그램

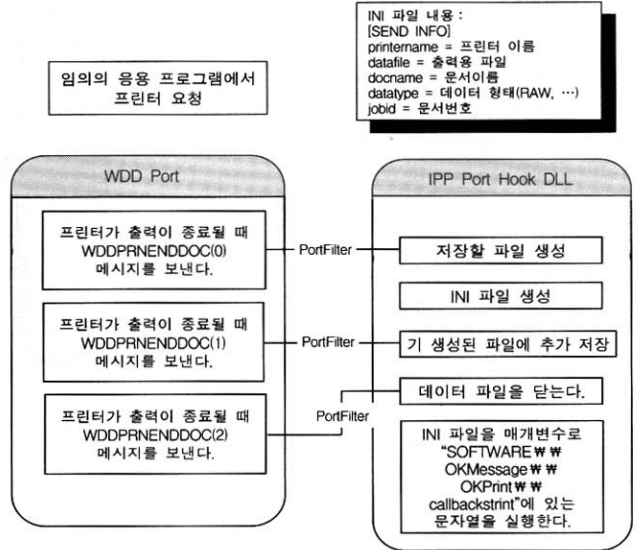


(그림 14) 프린터 찾기

4.3.2 프린터 출력물 포트 후킹

- 프로세스명 : p1_hook_print_job
- 관련 프로세스 : p1_request_printing, p1_send_job
- 사용 클래스

- 레지스트리 제어 : OKPortHook.CXRegKey
- 주요 사용 함수 및 객체 : CreateFile, WritePrivateProfileString
- 프로세스 다이어그램

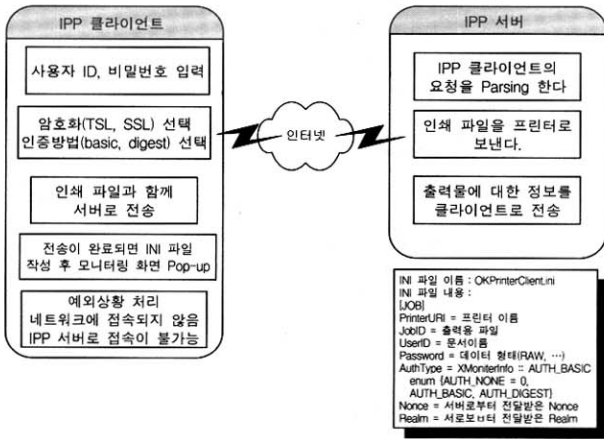


(그림 15) 프린터 출력물 포트 후킹

4.3.3 프린터 출력 요청

- 프로세스명 : p1_send_job
- 관련 프로세서 : p1_hook_print_job, p1_monitor_job
- 사용 클래스
 - Dialog : OKPrinterClient.CPrintingSheet, OKPrinterClient.CPrintingUserPage, OKPrinterClient.CPrintingSetPage, OKPrinterClient.CprintingInfoPage
 - Socket : OKPrinterClient.CClientSocket, OKPrinterClient.CXHttpRequestObject, OKPrinterClient.CXSecureSocket
 - Packet : OKPrinterClient.CXPrintPacket, OKPrinterClient.CXAttributePacket, OKPrinterClient.CXValuePacket
- 레지스트리 제어 : OKPrinterClient.CXRegKey
- 암호화 : OKPrinterClient.CXBase64, OKPrinterClient.GetHttpMD5()
- 주요 사용 함수 및 객체
- Windows API : WritePrivateProfileString, GetPrivateProfileString
- 제작 API
- GetInfo : http://WWW : PPPP/UUUU 부분을 호스트 이름, 포트, URI로 구분해낸다.
- AnsiToUTF8, UTF8ToAnsi : Ansi, UTF8 코드를 서로 변환한다.

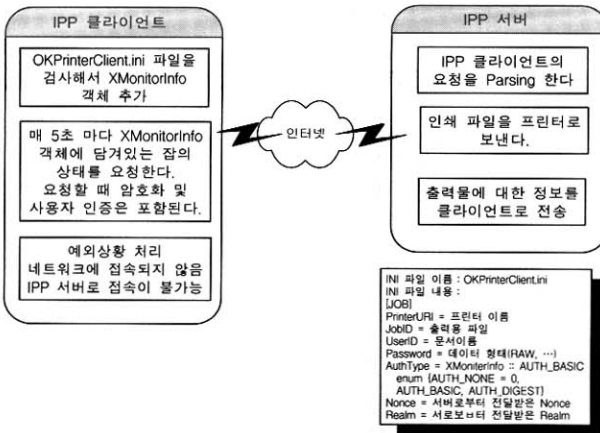
- 관련 INI 파일 : "OKPrinterClient.ini".
- 프로세스 다이어그램



(그림 16) 프린터 출력 요청

4.3.4 출력물 모니터링

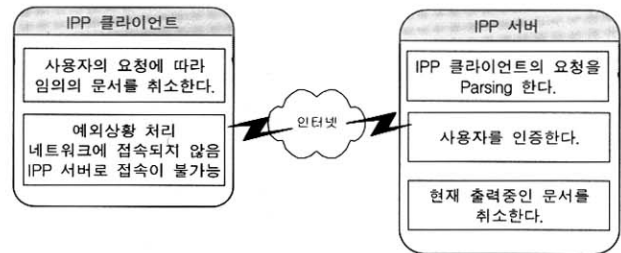
- 프로세스명 : p1_monitor_job
- 관련 프로세서 : p1_send_job
- 사용 클래스
 - Dialog : OKPrinterClient.COKPrinterClientView
 - Socket : OKPrinterClient.CClientSocket, OKPrinterClient.CXHttpRequestObject, OKPrinterClient.CXSecureSocket
 - Packet : OKPrinterClient.CXPrintPacket, OKPrinterClient.CXAttributePacket, OKPrinterClient.CXValuePacket
- 레지스트리 제어 : OKPortHook.CXRegKey
- 주요 사용 함수 및 객체
 - XMonitorInfo : 현재 모니터링할 객체들의 정보를 저장한다.
- 프로세스 다이어그램



(그림 17) 출력물 모니터링

4.3.5 출력물 취소

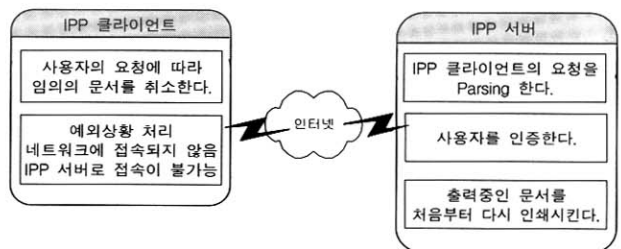
- 프로세스명 : p1_request_cancel_job
- 관련 프로세서 : p1_monitor_job
- 사용 클래스
 - Dialog : OKPrinterClient.COKPrinterClientView
 - Socket : OKPrinterClient.CClientSocket, OKPrinterClient.CXHttpRequestObject, OKPrinterClient.CXSecureSocket
 - Packet : OKPrinterClient.CXPrintPacket, OKPrinterClient.CXAttributePacket, OKPrinterClient.CXValuePacket
- 레지스트리 제어 : OKPortHook.CXRegKey
- 주요 사용 함수 및 객체 : 없음
- 프로세스 다이어그램



(그림 18) 출력물 취소 요청

4.3.6 출력물 재시도

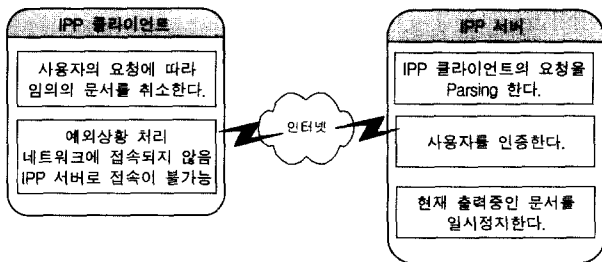
- 프로세스명 : p1_request_restart_job
- 관련 프로세서 : p1_monitor_job
- 사용 클래스
 - Dialog : OKPrinterClient.COKPrinterClientView
 - Socket : OKPrinterClient.CClientSocket, OKPrinterClient.CXHttpRequestObject, OKPrinterClient.CXSecureSocket
 - Packet : OKPrinterClient.CXPrintPacket, OKPrinterClient.CXAttributePacket, OKPrinterClient.CXValuePacket
- 레지스트리 제어 : OKPortHook.CXRegKey
- 주요 사용 함수 및 객체 : 없음
- 프로세스 다이어그램



(그림 19) 출력물 재시도

4.3.7 출력물 일시 정지

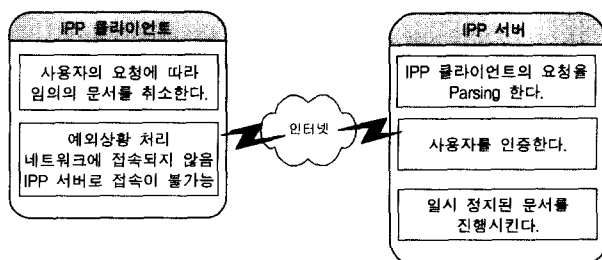
- 프로세스명 : pl_request_hold_job
- 관련 프로세서 : pl_monitor_job
- 사용 클래스
 - Dialog : OKPrinterClient.COKPrinterClientView
 - Socket : OKPrinterClient.CClientSocket, KPrinterClient.CXHttpRequestObject, OKPrinterClient.CXSecureSocket
 - Packet : OKPrinterClient.CXPrintPacket, OKPrinterClient.CXAttributePacket, OKPrinterClient.CXValuePacket
- 레지스트리 제어 : OKPortHook.CXRegKey
- 주요 사용 함수 및 객체 : 없음
- 프로세스 다이어그램



(그림 20) 출력물 일시 정지

4.3.8 출력물 계속

- 프로세스명 : pl_request_resume_job
- 관련 프로세서 : pl_monitor_job
- 사용 클래스
 - Dialog : OKPrinterClient.COKPrinterClientView
 - Socket : OKPrinterClient.CClientSocket, OKPrinterClient.CXHttpRequestObject, OKPrinterClient.CXSecureSocket
 - Packet : OKPrinterClient.CXPrintPacket, OKPrinterClient.CXAttributePacket, OKPrinterClient.CXValuePacket
- 레지스트리 제어 : OKPortHook.CXRegKey
- 주요 사용 함수 및 객체 : 없음
- 프로세스 다이어그램



(그림 21) 출력물 계속 요청

5. 결론 및 향후 연구 방향

최근의 인터넷이 급속한 보급은 10년 전에는 상상조차 할 수 없었다. 이러한 맥락으로 인터넷 인쇄 프로토콜은 인터넷을 기반으로 한 인쇄 기술의 표준으로 급속히 사용범위가 확대되고 있어서, 앞으로 팩스 혹은 전자우편의 대응으로 사용될 수도 있고, 대형 출력소 또는 보다 넓은 범위에서 널리 사용될 것이다. 이에 대비하여 지속적인 연구가 이어지고 향후에 거대한 인터넷 인쇄 관련 시장을 준비해야만 할 것이다. 현재 IPP는 버전 1.1까지 정의되었으며 지금도 끊이지 않고 전세계의 관련 인력들이 지속적으로 새로운 표준 안을 제안하고 있다. 본 연구에서는 IPP 표준 안에서 제시하고 있는 전송 프로토콜을 실제 시스템으로 구현하였고 이론적으로 제시된 표준안과 실제 기업과 사용자들의 요구를 수용할 수 있는 다양한 기능들을 포함하는 서버/클라이언트 시스템을 구현하였다. 이번 연구의 성과로는 세계적으로 몇몇 업체에서만 기초기술을 가지고 있는 하드웨어에 독립적인 IPP 서버/클라이언트 시스템의 개발 성공과 표준 안에 부합하는 프로토콜 소스코드를 확보함으로써 향후 새로 제안되는 표준 안에 빠르게 대응하여 시스템을 수정할 수 있게 된 것이다. 또한 타 회사 하드웨어 제품과의 호환성 테스트에서 만족할 만한 성과를 내었다. 향후에는 실제 출력 업체들에서 사용할 수 있도록 빌링(Billing) 시스템과의 연동에 대한 연구도 수행되어야 할 것이고 또한 현재 PWG에서 중요하게 논의되고 있는 통보(Notifications)기능에 대한 연구가 진행되어야 할 것이다.

참고 문헌

- [1] Carl-Uno M., "The Internet Printing Protocol(IPP)," <http://www.pwg.org/ipp>.
- [2] Microsoft Corp, <http://www.microsoft.com>.
- [3] Herriot, R., Butler, S., Moore, P., Tuner, R., "Internet Printing Protocol/1.1 : Encoding and Transport," RFC 2910, September, 2001.
- [4] R. deBry, T. Hastings, R. Herriot, S. Isaacson, P. Powell, "Internet Printing Protocol/1.1 : Model and Semantics," RFC 2911, September, 2001.
- [5] Paul E. Renaud, "Introduction to Client/Server Systems," Wiley Computer Publishing, 1996.
- [6] Lon Barfield, "The User Interface : Concepts & Design," Addison-Wesley, 1993.
- [7] 김수홍, "클라이언트/서버 시스템 구축", 상조사, 2000.
- [8] Zilles, S., "Rationale for the Structure and Model and Protocol for the Internet Printing Protocol," RFC 2568, April, 1999.
- [9] Wright, D., "Design Goals for an Internet Printing Protocol," RFC 2567, April, 1999.

- [10] Herriot, R., Jacobs, N., Hastings, T., Martin, J. "Mapping between LPD and IPP Protocols," RFC2569, April, 1999
- [11] Hastings, T., Manros, C., Zehler, P., Kuger, C. and H. Holst, "Internet Printing Protocol/1.1 : Implementer's Guide," RFC 3196, November, 2001.
- [12] Herriot, R., Hastings, T., Isaacson, S., Martin, J., deBry, R., Shepherd, M. and Bergman, R., "Internet Printing Protocol/1.1 : IPP Event Notifications and Subscriptions," Work in Progress.
- [13] R. Vonk, "Prototyping," Prentice Hall Inc., 1992.



김 수 흥

e-mail : soohkim@smu.ac.kr

1974년 서울대학교 공과대학 응용수학과
(공학사)

1990년 서울대학교 대학원 전산과학 전공
(이학석사)

1996년 서울대학교 대학원 전산과학 전공
(이학박사)

1992년~현재 상명대학교 공과대학 컴퓨터정보통신공학부 교수
관심분야 : 병렬처리시스템, 소프트웨어 공학, EC/ERP 시스템