

신기술 해설

전산망 데이터베이스 연계 표준

박 정 선 †

❖ 목	차 ❖
1. 서 론	4. 데이터베이스 접속표준
2. RDA C-ISAM 표준	5. 결 론
3. 관계형 데이터베이스 언어 표준	

1. 서 론

네트워크의 발달로 정보 시스템내의 다양한 하드웨어, 소프트웨어, 데이터베이스가 연결되어 상호 운용될 수 있는 환경이 조성되고 있는 바 사용자들이 여러 원격지에 산재되어 있는 데이터베이스를 지역 사이트에 있는 데이터베이스처럼 사용하기를 원하는 것은 성숙해 가는 정보화 사회의 당연한 귀결이라 할 수 있다.

국가기간전산망에서도 다양한 플랫폼 또는 데이터베이스 시스템간의 데이터 호환에 대한 요구가 끊이지 않고 있다. 국가기간전산망의 환경은 크게 두가지로 분류될 수 있는데, C-ISAM을 사용하는 환경과 데이터베이스 시스템을 사용하는 환경이다. 각 환경하에서 데이터 호환을 위하여 표준을 제작함은 전산망에 있는 시스템들의 효율을 향상 시키는데 필수적이라 할 수 있다. 본 연구는 국가기간전산망에 사용되거나 사용될 표준들을 설명하고 비교하여 국가기간전산망의 데이터베이스 연계를 위한 표준들을 이해하는데 도움을 주는 것이 목적이다.

C-ISAM은 세계적으로 널리 사용되지 않고 있는 화일 관리 시스템이지만 우리나라에서는 주민, 부동

산, 자동차 관리 시스템 등에서 사용되고 있어 우리나라의 중요한 정보들이 C-ISAM에 의해 관리되고 있다. 이러한 C-ISAM의 표준화를 위하여 이미 "주전산기 파일 시스템 표준"[1]을 개발한 바 있다. 현재 행정전산망에서 사용되고 있는 시스템들은 운영에 있어서 별 문제가 없다. 그러나 이들 시스템들이 서로 데이터를 교환해야 할 경우 통신 시스템이 기본적으로 제공해 주는 유틸리티를 사용하여 각 사안별로 처리되고 있다. 그러므로 데이터 교환을 위하여 구현하는 방법이 각 시스템마다 다른 실정이다. 이는 유지보수 측면에서 문제점들이 도출되는 바 이를 해결하기 위해 표준화된 방법인 "원격지 데이터베이스 접근:C-ISAM"(RDA C-ISAM)을 개발하였다[3]. 이 방법은 SQL환경에서 사용될 수 있는 원격지 데이터베이스 접근(Remote Database Access:RDA)[9, 10]을 위한 국제 표준을 C-ISAM 환경에 사용될 수 있도록 수정하였으며 정보통신부의 공식적인 발표만 남은 예정이다.

국가기간전산망내에서 C-ISAM을 쓰지 않는 환경을 데이터베이스 시스템 환경이라고 할 수 있다. 데이터베이스의 역사를 볼 때 1960년대, 1970년대를 제층형 데이터베이스, 네트워크형 데이터베이스 시대라고 한다면 1980년대를 관계형 데이터베이스

† 경희대 원 명기대학고 산업공학과 교수

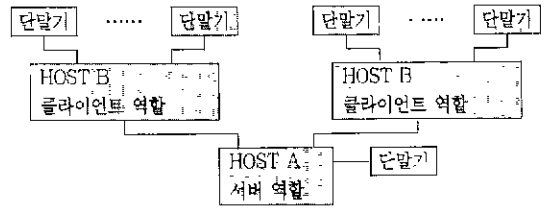
시대라고 할 수 있다. 1980년대가 관계형 데이터베이스가 시작된 시기라고 한다면 1990년대는 관계형 데이터베이스가 성숙된 시기라고 할 수 있다. 상품화된 객체 지향 데이터베이스가 시장에 나오기 시작한 시기이지만 아직 시장 점유율은 아주 미미한 상태에 있다. 관계형 데이터베이스 시대에 있어 그 언어의 통일은 관계형 데이터베이스가 널리 사용되는데 막대한 역할을 하였다. 이러한 추세에 맞추어 “관계형 데이터베이스 언어 표준”[2]을 제정한 바 있다. 국제 표준에 기반을 두고 국내 데이터베이스 사용자들과 제공자들의 의견을 수렴하여 작성하였다.

이러한 언어의 표준은 국가기간전산망에서 권해지는 규격을 정하는 것으로 이기종 데이터베이스간의 연계 방법에도 영향을 미치고 있다. 이기종 데이터베이스를 연재하기 위한 방법으로 데이터베이스 접속 방법이 제안되고 있다. 국가기간전산망에 사용될 데이터베이스 접속 표준을 작성함에 있어서, 기존에 작성된 “관계형 데이터베이스 언어 표준”의 규격을 참조하였고, 데이터베이스 시스템을 공급하는 업체들의 데이터베이스 접속 상품 규격들을 참조하였다. 이기종 데이터베이스간을 연계시키는 방법은 크게 두 가지로 나뉠 수 있는데, 클라이언트에서 두 개 이상의 서버를 접근하여 데이터를 호환하는 방법이 첫 번째 방법이고, 서버 시스템에서 다른 서버를 직접 접근 하는 방법이 두 번째 방법이다. 두 가지 방법 모두 현실적으로 가능하나 후자는 기존 프로그램을 전부 수정해야 하고 비용이 많이 든다는 문제가 제기 되어 첫 번째 방법이 표준으로 채택되었다. 작성된 표준은 “데이터베이스 접속 표준”이며 정보통신부의 정식 공표만 남은 예정이다.

2. RDA C-ISAM 표준

이 표준은 C-ISAM을 사용하는 환경에서 적용될 수 있는 표준으로 (그림 1)과 같은 구성에 적용할 수 있다.

(그림 1)에서 보이고 있는 것은 “원격지 데이터베이스 접근:C-ISAM”을 이용하여 호스트 A를 데이터베이스 서버화 하였을 때의 예이다. 다른 호스트 B, C 등은 각기 자기가 관리하는 데이터베이스



(그림 1) 원격지 데이터베이스 접근: C-ISAM 시스템 구성 예

를 접근하는 이외에, 클라이언트 프로세스를 갖추으로써 호스트 A의 데이터베이스를 접근할 수 있다.

RDA C-ISAM은 “원격지 데이터베이스 접근: SQL”(RDA SQL) 표준을 참조하여 작성하였다. 이들 표준의 공통점과 차이점을 서술하여 이해를 돕고자 한다.

2.1. RDA SQL과 RDA C-ISAM의 공통점

RDA SQL과 RDA C-ISAM은 모두 특수표준으로서, ISO 9579-1 RDA-Generic[9]에 기반을 두고 있다. 즉, RDA 특수표준은 RDA 일반표준에 각 환경을 반영하여 보완적인 내용만을 추가한 것이다. 따라서 SQL 환경에서 RDA SQL 표준을 적용하거나, C-ISAM 환경에서 RDA C-ISAM 표준을 적용하고자 할 경우 RDA 일반표준인 ISO 9579-1 RDA-Generic을 같이 참고하여야 한다.

2.2. RDA SQL과 RDA C-ISAM의 차이점

RDA SQL 표준은 ISO 9075:1992 (SQL Database Language)[8]이 참조 대상이고, RDA C-ISAM 표준에서는 “주전산기 파일 시스템 표준”(KIS 0034:1994)과 TX C-ISAM이 참조 대상이다. RDA SQL 표준의 ConformanceLevel은 그 해당 범위에 따라서 국제표준 전체를 대상으로 하는 Full SQL, Full SQL의 일부분을 대상으로 하는 Intermediate SQL, Intermediate SQL의 일부분을 대상으로 하는 Entry SQL 등으로 구분된다. RDA C-ISAM 표준에서는 전산망 표준인 KIS 0034:1994 [1]와 TX C-ISAM[7]으로 구분하였다.

자원개방 서비스에서 RDA SQL 표준은 Usage-Mode를 update, retrieval으로 읽기 전용 및 수정으로 구분하였고, RDA C-ISAM 표준에서는 Lock-

Mode, AccessMode, TransMode으로 구분하여 각각 C-ISAM의 기능을 반영하였다. 여기에서 LockMode는 ISEXCLLOCK, ISMANULOCK, IS-AUTOLOCK으로, AccessMode는 ISINPUT, ISOUTPUT, ISINOUT으로, TransMode는 ISO-CONTROL, ISTRANS으로 구분한다.

자원개방 서비스에서 RDA C-ISAM 표준은 개방하고자 하는 자원의 최대수가 정해져 있기 때문에 한정된 데이터 파일의 수를 초과하여 개방하고자 할 때 발생하는 오류로서 ISDataFileTooManyOpen이라는 매개변수를 추가로 사용한다.

명령어 직접수행 서비스인 R-Execute에서, RDA SQL 표준은 오류를 전송하기 위하여 STATE, CODE의 조합으로 사용하고, RDA C-ISAM 표준에서는 STATE 매개변수만을 이용하여 오류를 전송한다.

RDA 표준에서 사용하는 데이터의 타입으로서, SQL 표준에서는 character, numeric, decimal, interger, smallInt, float, real, doubleprecision, datetime, interval, varchar, bit, bitvarying 등이 있으며, C-ISAM 표준에서는 character, hcharacter, decimal, interger, long, float, double 등이 있다.

RDA C-ISAM 표준에서는 저장실행 서비스가 생략되었다. 왜냐하면, C-ISAM의 경우 다른 데이터 베이스 언어와는 달리 문장 하나만으로는 의미단위를 형성할 수 없다. 따라서 원하는 일을 수행할 수 있는 의미단위를 구성하기 위해서는, 다수의 C-ISAM 함수를 이용하여 프로시저어를 구성하고, 그 프로시저어를 호출해야 한다. 그러나 RDA 일반표준에는 프로시저어 호출 함수가 없기 때문에, 의미 단위를 형성할 수 없게 되고, 따라서 저장실행은 무의미해진다.

RDA SQL 표준에서는 ISO 9579-1 RDA-Generic에서 정의한 응용-문맥에서 기본 응용-문맥, TP 응용-문맥의 모든 형식을 구성하고 있다. RDA C-ISAM 표준에서는 C-ISAM의 TP 처리를 위한 기능이 부족하기 때문에 TP 응용-문맥의 객체 식별자, ASE 데이터 타입, 순서 규칙 및 적합성 요구사

항 등을 추가로 정의하지 않는다.

3. 관계형 데이터베이스 언어 표준

본 표준(KSQL1)은 ISO(International Standard Organization)의 국제 표준인 ISO IS 9075:1992 [8]의 레벨 1을 기반으로 하고, 필요한 기능을 추가한 정도의 범위를 갖고 있는데, 그 이유는 만약 ISO IS 9075:1989를 기반으로 하게 된다면, ISO IS 9075:1989에 정의된 기능이 현재 상용화되어 있는 관계형 DBMS 제품들이 제공하는 기능에 비해 너무 빈약하기 때문에 제품마다 추가로 제공되는 기능이 다양해질 것이며 이 기능들을 사용하는 응용 프로그램들의 호환성에 문제가 있게 되며, 상용화되어 있는 관계형 DBMS 제품들이 ISO IS 9075:1992의 모든 기능을 만족 시키지 못하기 때문이다. 그리고, SQL2의 레벨 1에 비해 KSQL1에는 동적 SQL 관리자, 스키마 처리어, 오류 관리자, 트랜잭션의 속성을 정의하는 SET TRANSACTION 문이 추가되었으며, 기본 구성 요소에 한글과 한자를 지원할 수 있도록 SQL character language에 <KSC-5601 (완성형) letter>을 추가하였다. 문종류 별로 정리하면 <표 1>과 같다.

<표 1> KSQL1 목록

분 류	함 수 이 름
스키마 처리어	CREATE SCHEMA, CREATE TABLE, CREATE VIEW, GRANT
스키마 처리어	DROP SCHEMA, DROP TABLE, ALTER TABLE, DROP VIEW, REVOKE
자 료 처리어	DECLARE CURSOR, OPEN, FETCH, CLOSE, SELECT:single row, DELETE: positioned, DELETE:searched, INSERT, UPDATE:positioned, UPDATE:searched
트랜잭션 관리어	SET TRANSACTION, COMMIT, ROLLBACK
동적SQL 관리어	ALLOCATE DESCRIPTOR, DEALLO- CATE DESCRIPTOR, GET DESCRIPT- OR, SET DESCRIPTOR, PREPARE, DE- SCRIBE OUTPUT, EXECUTE, dynamic DECLARE CURSOR dynamic OPEN, dy- namic FETCH, dynamic CLOSE, dynamic DELETE:positioned, dynamic UPDATE: positioned
오류관리어	GET DIAGNOSTICS

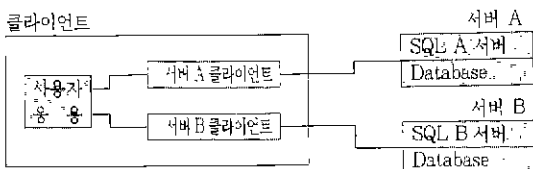
4. 데이터베이스 접속 표준

4.1. 데이터베이스 접속 개요

데이터베이스 접속이란 서로 다른 이기종 데이터베이스를 접근할 수 있도록 개방된 연결성을 보장하는 구조를 말한다. 이는 네트워크로 연결된 클라이언트/서버 데이터베이스에서 각각 다른 종류의 데이터베이스 정보를 일관된 형태로 액세스 할 수 있도록 하는 인터페이스를 제공함으로써, 응용 프로그램의 개발 환경을 표준화하는 것이다.

네트워크 혹은 대형 컴퓨터에서 가능한 모든 업무를 액세스하기 위해 알아야 하는 여러가지 형식에 대한 지식이 없어도 접속의 응용 프로그램 인터페이스(Application Programming Interface, 이하 API)의 표준을 따르기만 하면 다양한 데이터 소스를 접근할 수 있는 응용 프로그램을 만들 수 있다. 따라서 사용자는 원하는 데이터베이스 관리 시스템을 선택하여 해당 드라이버와 링크함으로써 데이터베이스 시스템의 종류에 관계없이 응용 프로그램을 구동할 수 있다.

본 접속 표준 방법에서는 마이크로소프트사에서 제공해주고, 많은 DBMS 제품들이 지원해주는 ODBC(Open Database Connectivity)[11]를 기본으로 한 표준에 대하여 설명한다. (그림 2)의 예는 SQL 데이터베이스를 관리하는 서버 A와 서버 B를 클라이언트 역할을 하게 되는 PC에서 접근하는 예이다. 클라이언트의 사용자 응용은 서버 A와 서버 B를 둘 다 원격 접근하여 결과를 얻은 다음, 두 결과를 취합/비교하는 응용이라고 가정한다. 서버 A는 SQL을 지원하므로 클라이언트는 서버 A를 접근하기 위하여 DB 게이트웨이 클라이언트 프로세스를 갖추어야 한다. 서버 B 역시 SQL을 지원하므로 클라이언트는 서버 B를 접근하기 위하여 DB 게이트웨이 클라이언트 프로세스를 갖추어야 한다.



(그림 2) DB 접속을 사용한 복수 데이터베이스 접근

4.2. 지원 레벨

4.2.1. 지원 함수 목록

데이터베이스 접속에서는 ODBC에서 제공하는 모

(표 2) 지원 함수 목록

분류	함수 이름	레벨
데이터 소스 연결	SQLAllocEnv	1
	SQLAllocConnect	1
	SQLConnect	1
	SQLDriverConnect	2
	SQLBrowseConnect	2
드라이버 및 데이터 소스에 대한 정보 확보	SQLDataSources	2
	SQLGetInfo	2
	SQLGetFunctions	2
	SQLGetTypeInfo	2
드라이버 옵션 설정 및 추출	SQLSetConnectOption	2
	SQLGetConnectOption	2
	SQLSetStmtOption	2
	SQLGetStmtOption	2
SQL 요청 준비	SQLAllocStmt	1
	SQLPrepare	1
	SQLSetParam	1
	SQLParamOption	2
	SQLGetCursorName	1
	SQLSetCursorName	1
	SQLSetScrollOptions	2
요청수락	SQLExecute	1
	SQLExecDirect	1
	SQLNativeSql	2
	SQLDescribeParam	2
	SQLNumParams	2
	SQLParamData	2
	SQLPutData	2
결과 및 결과에 대한 정보 추출	SQLRowCount	1
	SQLNumResultCols	1
	SQLDescribeCol	1
	SQLColAttributes	1
	SQLBindCol	1
	SQLFetch	1
	SQLExtendedFetch	2
	SQLGetData	2
	SQLSetPos	2
	SQLMoreResults	2
	SQLError	1
데이터 소스의 시스템 테이블에 대한 정보 확보	SQLColumnPrivileges	2
	SQLColumns	2
	SQLForeignKeys	2
	SQLPrimaryKeys	2
	SQLProcedureColumns	2
	SQLProcedures	2
	SQLSpecialColumn	2
	SQLStatistics	2
	SQLTablePrivileges	2
	SQLTables	2
문장 종료	SQLFreeStmt	1
	SQLCancel	1
	SQLTransact	1
연결 종료	SQLDisconnect	1
	SQLFreeConnect	1
	SQLFreeEnv	1

든 함수를 제공해 주도록 하였으며, 제공 함수 중 ODBC 코어(core) 부분을 쓴 표준에서는 레벨 1로 정하였고, ODBC의 Level 1 및 Level 2 부분을 쓴 표준에서는 레벨 2로 정하였다. 레벨 1은 표준 함수 중 필수적으로 제공이 되어야 하는 부분이며, 레벨 2는 필수 기능은 아니지만 사용자의 편의 및 시스템의 효율성을 위하여 제공되는 기능이다.

4.2.2. 지원 SQL 문장

데이터베이스 접속에서는 ODBC에서 제공하는 모든 SQL 문장을 제공해 주도록 하였으며, 제공 SQL 문장 중 최소(0) 및 코어(1) 부분을 레벨 1로 정하였고, 확장(2) 부분을 레벨 2로 정하였다.

〈표 3〉 지원 SQL 문장

종류	SQL문장	레벨
스키마 정의어	CREATE TABLE	1
	CREATE VIEW	1
	GRANT	1
스키마 처리어	DROP TABLE	1
	ALTER TABLE	1
	DROP VIEW	1
	REVOKE	1
자료 처리어	SELECT:single row	1
	DELETE:positioned	1
	DELETE:searched	1
	INSERT	1
	UPDATE:positioned	1
	OUTER JOIN	2
카타문	CREATE INDEX	2
	DROP INDEX	2

4.2.3. 지원 자료 형태

데이터베이스 접속에서는 ODBC에서 제공하는 모든 자료 형태를 제공해 주도록 하였으며, 제공 자료 형태 중 ODBC의 지원 Level이 0 또는 1이거나, KSQL1에서 제공되는 자료 형태일 경우 레벨 1로 정하였고, 그 이외의 부분은 레벨 2로 정하였다.

4.3 RDA SQL과 데이터베이스 접속 표준

RDA SQL은 ISO/IEC의 국제표준이고, ODBC는 단체표준이지만 산업체에서의 실제 표준이다. 이 두가지 표준의 유사성과 차이점을 분석하여 전산망에 사용되는 표준들에 대한 이해를 돕고자 한다.

4.3.1. 공통점

여러 벤더에 의해 공급되는 하나 이상의 플랫폼

〈표 4〉 지원 자료 형태

자료 형태	레벨	비고
CHAR	1	KSQL1 제공
DATE	2	
DECIMAL	1	KSQL1 제공
DOUBLE	1	KSQL1 제공
FLOAT	1	KSQL1 제공
BINARY	2	
INTEGER	1	KSQL1 제공
LONGVARCHAR	2	
LONGVARIABLE	2	
NUMERIC	1	KSQL1 제공
REAL	1	KSQL1 제공
SMALLINT	1	KSQL1 제공
TIME	2	
TIMESTAMP	2	
VARCHAR	1	
VARIABLE	2	
BIT	2	

위에 있는 다양한 관계형 데이터베이스 시스템을 접근하기 위해서 만들어 졌다. 또한 표준화된 API (Application Program Interface)에 근거를 두고 있다. ODBC는 X/Open의 표준인 CLI(Call Level Interface)를 근거로 Microsoft가 주도하여 만든 표준이며, RDA SQL은 ISO의 표준이다.

RDA SQL은 서버 시스템의 행태를 정의한다. 데이터베이스 서버는 데이터베이스 저장과 데이터베이스 프로세스 서비스를 RDA 클라이언트에게 제공한다. 더불어 RDA SQL은 RDA 클라이언트가 RDA 서버 시스템의 능력을 사용할 수 있도록 통신 서비스와 규약을 상술하고 있다. ODBC 경우도 데이터베이스 서버는 데이터베이스 저장과 데이터베이스 프로세스 서비스를 RDA 클라이언트에게 제공한다. 또한 대화형 통신 모델을 사용하며 클라이언트와 서버간의 교류가 통신을 위한 논리적 접속의 문맥 안에서 일어난다.

4.3.2. 차이점

본산 데이터베이스를 접근함에 있어 RDA SQL은 두 가지의 일반적인 유형을 지원한다. 한 트랜잭션이 단지 하나의 데이터베이스 서버에 접근하는 RDA 기본 응용 문맥 (Basic Application Context)과 한 트랜잭션이 여러 데이터베이스 서버에 동시에 접근하는 RDA TP 응용 문맥 (Transaction Processing Application Context)을 지원하고 있다.

ODBC의 경우는 한 트랜잭션이 어느 한시점에 단지 하나의 데이터베이스 서버에 접근하는 방법만을 제공한다. 또한 정보 스키마 테이블(Information Scheme Table)의 정보를 접근하는데 있어 ODBC는 이를 위한 서비스를 제공하나, RDA SQL은 제공하지 않는다. 정보 스키마 테이블은 카탈로그 테이블(catalog table)이라 불리우는데 특정한 데이터베이스에 대한 서술적 정보(descriptive information)를 가지고 있다. 그리고 ODBC와 RDA SQL이 지원하는 SQL의 종류에 있어 상당한 차이를 보여주고 있다. RDA SQL의 경우 지원되는 SQL의 수준을 국제표준에 준하여 명시한다. 예를 들면, ISO 9075 1987, ISO 9075 1989, ISO 9075 1992(엔트리, 중간, 고급). RDA 클라이언트와 서버는 클라이언트가 접근하고자 하는 자원을 확인할 때 지원하는 수준을 위해 서버와 타협을 하게 된다. 그러나, ODBC 경우는 다소 제한된 SQL 수준을 제공하나 그 수준 내에서는 네트워크 내에 있는 어떠한 SQL 변이(variants)의 사용도 허용한다. RDA SQL표준은 클라이언트에 통신 기능만을 부가하여 클라이언트는 사용자로부터 제출받은 요청을 그대로 서버에 보내고, 데이터베이스의 실행은 전적으로 서버가 실행한다. 반면에, ODBC에서는 통신 기능과 SQL 기능을 갖춘 SQL 드라이버가 위치한 클라이언트에서 SQL 구문 처리 등을 수행하여 이를 각 서버에 대응하는 동적 함수 호출 형태로 번역한다. 그 다음 번역된 호출을 서버에 보낸다.

5. 결 론

본 논문에서는 국가기간전산망에 데이터베이스 연계 표준 기술을 수용하기 위하여 관련 표준들을 서술하고 비교하였다. 단기적으로는 원격지 데이터베이스 접근 표준이나 데이터베이스 접속 표준 등을 수용하고, 장기적으로는 분산 트랜잭션 처리가 가능한 표준들을 수용하여 다중 데이터베이스 시스템을 구축하는 방향으로 국가기간전산망을 발전시키는 것이 바람직할 것이다.

본 연구에서 설명된 여러 표준들은 여러 데이터베이스를 상호 연계하여 사용하기 위한 기술들이다.

또한 현재의 표준들도 계속해서 진화 발전될 것이므로 이를 예의 주시하면서 역동적으로 대처할 수 있는 것이 바람직하다.

참 고 문 헌

1. 한국전산원, "주전산기 파일 시스템 표준," KIS 0034:1994.
2. 한국전산원, "관계형 데이터베이스 언어 표준," KIS 0044:1994.
3. 한국전산원, "원격지 데이터베이스 접근:C-ISAM (안)," 1994.
4. 한국전산원, "데이터베이스 게이트웨이 표준 (안)," 1994.
5. 한국전산원, "데이터베이스 연계를 위한 연구," 1994.
6. 한국전산원, "국가기간전산망 데이터베이스 전환전략 연구:데이터베이스 연계 표준화 전략 연구," 1993.
7. 행정전산사업단, "행정전산망 도입기종 안정화 위탁과제 보고서 제7권:TX C-ISAM Source 분석서," 1990.
8. ISO/IEC 9075 "Information Technology-Database Language," 1992.
9. ISO/IEC 9579-1 "Information Technology-Remote Database Access," 1993.
10. ISO/IEC 9579-2 "Information Technology-Remote Database Access-Part 2:SQL Specialization," 1993.
11. Microsoft ODBC Interface, "API Reference," 1993.

박 정 선



1983년 서울대학교 산업공학과(학사)
1985년 한국과학기술원 경영과학과(석사)

1985년~88년 금성소프트웨어 근무

1988년~89년 한국유니시스 근무

1993년 The University of Texas At Austin MIS전공(박사)

1993년 한국전산원 근무(데이터관리팀장)

1995년~현재 명지대학교 산업공학과 조교수

관심분야: Expert System Data Base Software Engineer, ng CIM CALS