

임베디드 시스템을 위한 DBMS 컴포넌트 구성 도구 설계*

문미경^o 서여진 박우창
덕성여자대학교 전산학과

{moon^o, yjseo}@namhae.duksung.ac.kr, ucpark@duksung.ac.kr

Design of a Component Configuration Toolkit for Embedded DBMS

MiKyoung Moon^o, YeoJin Seo, UChang Park
Dept. of Computer Science, Duksung Women's University

요 약

임베디드 데이터베이스 시스템은 사용 목적과 환경에 따라 탑재되는 DBMS 요구 사항이 다르기 때문에 임베디드 시스템 구현시 마다 DBMS를 재개발하여야 하는 문제가 있다. 이러한 DBMS를 단일 통합된 코드 형태로 개발하는 것은 코드 재사용성과 유연성이 낮아 비생산적이다. 본 논문에서는 임베디드 시스템에서 사용되는 DBMS 기능에 따라 분리하여 컴포넌트화된 DBMS를 구성해주는 도구의 설계를 소개한다. 컴포넌트 대상은 SQL 인터페이스, SQL 언어 지원 레벨, 동시성 제어 모듈, 인덱싱 모듈, 회복 관리자 등이다. 도구는 빠르고 유연한 임베디드 데이터베이스 개발시 생산성을 높일 수 있다.

1. 서 론

정보 기술의 급속한 발전과 함께 임베디드 시스템이 많은 수의 가전기와 개인용 휴대기기에서 사용되고 있다. 최근에는 사용자의 편의성을 고려하여 갈수록, 더 많은 데이터의 수용과 관리를 필요로 하는 제품들이 늘어나면서 임베디드 데이터베이스를 많은 곳에서 사용하고 있다. 하지만 각 응용마다 다른 환경의 사양을 지니고 있으며, 필요로 하는 DBMS 기능도 상이한 경우가 많다. 각각의 임베디드 시스템에서 필요한 기능에 따라 매번 사양에 맞는 DBMS를 구현하는 것은 개발시간과 인력소모가 크다, 따라서 임베디드 시스템을 위한 효율적인 DBMS 개발의 필요성이 대두되고 있다.

이에 따라 본 논문에서는 각각의 응용마다 필요한 기능에 따라 구현되어졌던 기존 임베디드 시스템의 DBMS를 컴포넌트화하여 재사용이 가능하도록 해주는 도구를 설계하였다. 이는 임베디드 시스템에 탑재될 DBMS를 구현 할 때 필요한 기능을 가지고 있는 컴포넌트가 이미 구현되어져 있는 경우, 그 컴포넌트를 재사용함으로써 개발에 소요되는 시간과 비용 등을 상당히 줄일 수 있으며, 유지보수에 있어서도 문제가 되는 컴포넌트만 유지 보수해 나가면 되므로 유지보수 비용이 절감된다. 또한, 임의의 시스템에 내장된 형태로 구성되기 때문에 제한되고 전문화된 기능만을 수행하는 임베디드 시스템의 조건인 경량화를 위하여 필요한 기능만 선택하여 메모리를 최소화 할 수 있게 된다. 또 임베디드 DBMS를 구성하는 도구를 따로 두어

임베디드 시스템 개발자들은 컴포넌트 선택을 통하여 유연성과 생산성의 효과를 높여 편리하게 임베디드를 위한 DBMS를 구성 할 수 있도록 하였다.

본 논문의 구성은 다음과 같다. 2절에서는 임베디드 시스템용 DBMS 컴포넌트화와 관련된 연구를 보이고, 3절에서는 컴포넌트 가능 요소 및 도구 설계를 통해 임베디드 시스템에서 DBMS가 효과적으로 구성되도록 생성시켜주는 도구에 대한 설계를 설명 한 후, 마지막 절에서는 본 논문의 결론과 앞으로의 연구 방향에 대해 제시할 것이다.

2. 임베디드 시스템용 DBMS 컴포넌트화 관련 연구

임베디드 시스템은 특정 시스템에서 제한된 작업만을 수행하도록 하는 시스템으로, 1950년대 통신 장비 제어를 위해 등장하여 1970년대 후반부터 표준화된 대량 생산이 가능하였으며, 1990년대 초반까지 군사용 제어, 산업 기기 제어 등의 목적으로 많이 사용되어 왔다. 1990년대 후반부터 컴퓨터 산업과 정보 가전 기기의 발전으로 임베디드 시스템은 첨단 산업으로 각광 받기 시작하며 활발한 연구가 진행되고 있다[1]. 임베디드 시스템은 Windows, Linux 등의 범용 운영체제를 소형화한 것과 VxWorks, PSOS, Lynx등의 실시간 운영체제들이 경쟁을 하고 있다[2,3].

임베디드 시스템에 탑재되는 DBMS 시스템으로는 Pervasive.SQL[4], Polyhedra, Velocis, RDM, Berkeley DB[5], TimesTen[6] 등이 있다. 이들은 하나의 시스템으로 구성되어 있는 단일통합형 DBMS 시스템으로, 기존의 기능을 확장하거나 수정하려면 전체 DBMS 시스템에 연속적으로 영향을 끼치게 되어 확장과 수정이 쉽지 않으며, 다른 기능을 추가시키기

* 본 연구는 한국과학재단 연구비 (과제번호 R06-2002-003-01005-0) 지원으로 수행되었음

나 유지보수 할 때도 복잡하고 비용이 많이 든다. 이처럼 위에 언급된 것과 같은 단일통합형 DBMS를 사용 할 경우 전체적으로 여러 기능들이 하나의 DBMS 시스템을 구성하고 있어서 기능별 분리가 쉽지 않다. 이미 구현되어진 단일통합형 DBMS를 재사용할 때 필요하지 않은 기능을 분리하기 어렵기 때문에 임베디드 시스템에 그대로 탑재되어 최적화된 크기를 갖지 못하게 된다.

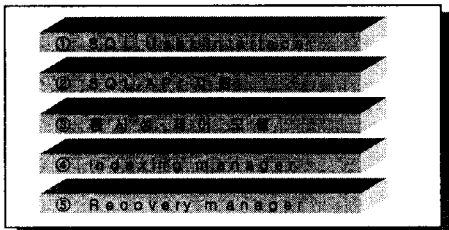
이를 해결하기 위해 DBMS의 기능들을 컴포넌트화 한다. 컴포넌트화로 인해 임베디드에 탑재된 DBMS의 확장, 수정, 교체를 할 때 전체가 아닌 부분 작업이 가능하며, 추가 시에도 전체적으로 컴파일 하지 않고 추가한 부품만 컴파일 함으로써 유지보수가 간단해진다. 또한 필요한 기능만으로 구성되어져서 작은 크기로 임베디드 시스템에 적합한 DBMS가 구성되어 진다.

3. 임베디드 시스템의 DBMS 컴포넌트화 도구 설계

본 절에서는 DBMS 컴포넌트화를 위해 설계되어진 컴포넌트 가능 요소와 도구 설계를 본다.

3.1 컴포넌트 가능 요소

작은 단위의 소프트웨어인 컴포넌트는 완전히 독립적으로 배포할 수 있어야 하며, 제3자에 의해서 조립, 관리, 재사용이 가능해야하며, 제3자가 쉽게 사용할 수 있도록 명확한 인터페이스 명세가 있도록 컴포넌트화 되어져야 한다. 이를 만족시키며 임베디드 시스템에서 적합한 DBMS 컴포넌트 가능 요소 중 본 연구에서는 다음 기능들에 초점을 맞춘다.



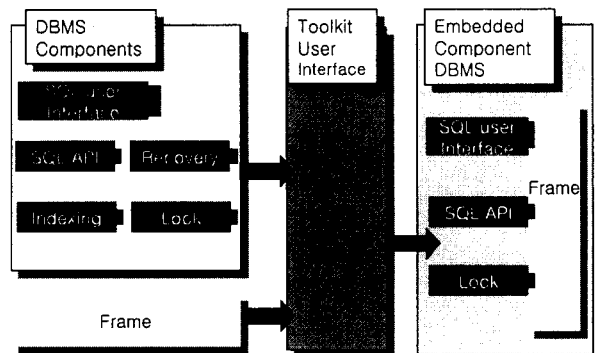
[그림 1] 임베디드시스템 DBMS 컴포넌트 가능 요소

그림 1의 임베디드 시스템용 DBMS의 컴포넌트 가능 요소들에 대해서 상세히 살펴보도록 한다. 첫째, SQL 사용자 인터페이스는 임베디드 시스템에 탑재되는 DBMS를 다루는 사용자가 사용하게 될 SQL 인터페이스 툴로써, 사용자가 SQL 명령어를 쉽게 접근 할 수 있도록 해주는 기능을 가진 컴포넌트이다. 둘째, SQL API 지원은 각 응용에 따라 사용되는 SQL 명령문의 집합이 상이한 것을 감안하여 SQL문 중에서 필요한 부분만 선택하여 쓸 수 있도록 기본 SQL 문 집합, 고급 SQL 문 집합, SQL 함수 집합으로 구분한다. 기본 SQL 문 집합은 검색, 수정, 삭제, 추가와 같이 SQL문에서 필수적인 기능으로 구성된다. 고급 SQL 문 집합은 테이블, 뷰, 사용자의 생성과 제거,

COMMIT, ROLLBACK, GRANT 등의 기능을 말한다. SQL 함수 집합은 수학함수, 문자함수, 집계함수, 보안함수 등을 제공한다. 예를 들어 수학함수를 선택한 경우에 지원되는 함수로는 절대값을 처리해주는 ABS, 나머지를 반환하는 MOD, 제곱근을 구해주는 SQRT 등이 있으며, 문자 함수를 선택한 경우에는 소문자로 변환시켜주는 LOWER, 대문자로 변환시켜 주는 UPPER 등의 함수가 지원되며 집계함수는 SELECT문의 결과 그룹에게 적용되는 함수들을 가지며, 보안함수는 사용자에게 따라 접근권한을 주는 기능을 가지도록 각각 컴포넌트화 한다. 이와 같이 SQL API지원은 기본 SQL 문 집합, 고급 SQL 문 집합, SQL 함수 집합을 지원하며 이중 필요한 부분만 선택하여 사용하면 된다. 세 번째, 동시성 제어 모듈은 여러 트랜잭션의 동시 수행을 제어하기 위해서 데이터의 접근 권한을 잠글 수 있도록 록(lock) 기능을 가지는 컴포넌트이다. 네 번째는 테이블에서 효과적으로 검색 할 수 있도록 하는 인덱싱 기능을 가진 인덱스 관리자(indexing manager)로 트리 데이터 구조를 기반으로 한 B+트리, 해시가 있고, 이들 중에서 기능을 선택하거나 사용하지 않을 수 있다. 마지막으로 여섯 번째는 데이터베이스 시스템에서 작업 도중 문제가 발생하는 경우 처리해주는 데이터 회복 기능의 복구 관리자(recovery manager)이다. 복구 관리자의 undo는 문제가 발생하면 장애시점에서 처리 중인 변경사항들을 취소하는 것이다. redo는 데이터베이스의 사본을 주기적으로 만들어 두고, 보관 이후로 트랜잭션들이 데이터베이스에 대해 수행한 변경사항을 로그로 기록해 두고 장애가 발생하면 문제가 발생한 시점부터 다시 재적용 한다.

이와 같이 DBMS를 구성할 컴포넌트들을 기능별로 크게 분류함으로써 임베디드 시스템에 장착될 DBMS의 컴포넌트 가능 요소를 살펴보았다.

3.2 도구 설계

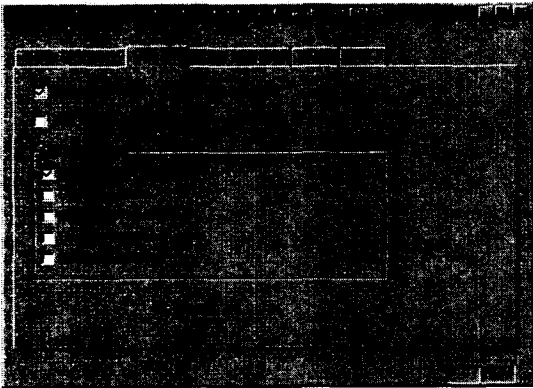


[그림 2] 임베디드 시스템용 DBMS 구성 도구 설계

설계된 임베디드 데이터베이스 시스템 컴포넌트 구성 도구는 그림 2와 같다. SQL 사용자 인터페이스, SQL API 지원, 동시성 제어 모듈, 인덱스 관리자, 회복 관리자 컴포넌트들 중에서 사용자는 임베디드 시스템에 탑재할 DBMS가 필요로 하는 기능을 가진 컴포넌트들을 선택한다. 선택되어진 컴포넌트들과

기본 틀인 프레임(Frame)은 임베디드 시스템에 장착될 DBMS의 핵심 엔진이다.

임베디드 시스템을 위한 DBMS 컴포넌트 구성 도구(그림 2)의 사용자 인터페이스는 그림 3과 같다. 그림 3을 통해 설계한 DBMS 구성 도구 인터페이스를 살펴보면 주메뉴는 파일, 컴포넌트선택, DBMS 생성, 도움말로 구성되어 있다. 파일메뉴는 구성된 DBMS를 저장하거나 불러와서 수정할 수 있도록 열기와 저장, 종료로 되어 있다. DBMS 생성 메뉴는 선택되어진 컴포넌트들을 바탕으로 기본 틀인 Frame과 함께 컴파일하여 임베디드 시스템을 위한 DBMS를 생성하도록 구현되어질 것이다.



[그림 3] DBMS 구성 도구 초기 화면

컴포넌트 선택 메뉴는 필요한 기능에 따라 각 탭 창에서 해당되는 기능을 가지는 컴포넌트를 선택 할 수 있다. 그림 3은 컴포넌트들 중에서 SQL API를 선택하는 화면으로 기본 SQL과 SQL 함수 중 수확 함수 기능을 가지는 임베디드 DBMS로 구성되어 지도록 선택한 것이다.

현재 개발되고 있는 임베디드 시스템을 위한 DBMS의 크기는 일반적으로 표 1과 같으며, 모두 단일통합형 DBMS로 재사용시 필요하지 않은 부분도 가지고 있기 때문에 크기가 최적화 되지 않는 단점이 있다.

DBMS 시스템	메모리 크기
TemesTen	5MB
Velocis	4MB
Polyhedra	1.5 ~ 2 MB
McKoi	1.5MB
RDM	400 ~ 500 KB
Berkeley DB	175KB

[표 1] 상용 임베디드 DBMS의 메모리 크기

본 논문에서 기본으로 사용하고 있는 DBMS는 이미 구현되어 있는 표 1의 여러 DBMS 중 Diehl & Associates에서 구현한 공개 데이터베이스 시스템인 McKoi[7]이다. McKoi의 모든 기능을 구현한 단일통합형으로 크기는 1.5MB이며 표 2를 보면 McKoi를 컴포넌트화 하였다는 가정 하에 전체 기능 중에 SQL Interface를 제외한 크기는 1.43MB, 기본 SQL만 제외한 경우

는 1.46MB으로 추정됨을 볼 수 있다.

구성	크기	감소율
McKoi 전체	1.5MB	-
SQL Interface만 제외	1.43MB	4.6%
기본SQL만 제외	1.46MB	2.6%

[표 2] McKoi의 컴포넌트화 부분선택과 크기

단일통합된 DBMS보다 기능을 선택한 경우 크기가 줄어들음 표 2의 감소율을 통해서 알 수 있다. 전체 기능 중에서 만약 SQL Interface와 기본 SQL의 기능을 제외시키는 경우 DBMS 크기는 7.2%감소한다. 즉, 필요하지 않은 기능을 제외시킴으로써 경량의 재사용가능 DBMS를 생성할 수 있는 것이다. 이를 바탕으로 앞에서 언급된 컴포넌트 가능 요소들로 임베디드 시스템을 위한 DBMS 구성 도구를 구현할 것이다.

4. 결 론

본 논문에서는 임베디드 시스템에서 기본적으로 필요한 DBMS를 주문 생산하도록 하는 도구를 설계하였다. 이는 무수한 데이터를 관리하는 단일통합 DBMS가 제품마다 다른 기능 요구 사항을 가짐으로써 개발시 재사용성이 떨어지는 것을 극복하고 필요한 기능만을 선택 구성하여 임베디드 시스템에 최적화된 컴포넌트화 DBMS를 구현하도록 설계되었다. 컴포넌트화 DBMS 구성 도구를 개발함으로써 임베디드 시스템을 위한 DBMS의 빠른 개발과 코드의 최적화를 하게 된다.

현재 다섯 기능과 그 세부항목들로 컴포넌트화 할 계획을 가지고 있는 설계 단계이지만, 향후 더 필요한 기능들도 컴포넌트화 할 것이며, DBMS 크기를 고려하며 임베디드 시스템에 적합한 경량의 컴포넌트화된 DBMS를 생산해주는 도구로 구현 할 계획이다.

참고문헌

- [1] 배방희, "인퍼노 RTOS 기반의 embedded 웹단말기 구현 방안에 관한 연구", 한양대학교 산업대학원 공학석사논문, 1999.
- [2] 이두원, "실시간 운영체제", <http://www.doall.co.kr>, 1999.
- [3] 김홍섭, 문승진, "내장형 시스템을 위한 실시간 데이터베이스 엔진 설계 및 구현", 한국 인터넷 정보과학회 3권 5호, pp.19-29, 2002.
- [4] Pervasive.SQL, <http://www.pervasive.com/>.
- [5] Berkeley DB, <http://www.sleepycat.com/>.
- [6] Jorgen Hansson Christer Norstrom, "Embedded Databases for Embedded Real-Time Systems: A Component-Based Approach", Technical Report, Linkoping University and Malardalen University, Sweden, 2002.
- [7] McKoi, <http://mckoi.com/database>, 2000.
- [8] 박병형, 불록놀리와 CBD, 태영출판사, 2002.