특집 데이터베이스와 사용자 인터페이스

최중민
한양대학교 전자계산학과 교수

1. 서론

컴퓨터의 급속한 보급으로 많은 양의 데이터를 관리하기 위한 데이터베이스의 사용 또한 늘고 있다. 하지만 컴퓨터 사용에는 초보자인 일반 사용자는 데이터베이스 또는 DBMS를 이용하여 데이터를 입력하고 원하는 데이터를 검색하는데 불편과 어려움을 느끼는 것이 사실이다. 이것의 가장 근본적인 이유는 사용자 인터페이스의 문제이다. DBMS의 상위 계층인 사용자 인터페이스는 사용자와 응용 프로그램, 그리고 DBMS간의 정보 교환과 대화를 관리하여, 사용자가 데이터 모델이나 스키마 등의 데이터 정의와 데이터 조작을 쉽고 효율적으로 할 수 있도록 도와주는 역할을 수행해야 한다.

초창기의 DBMS는 데이터의 관리와 수행에 중점을 두었기 때문에 일반 텍스트 위주의 인터페이스만을 제공하였지만 점차 데이터베이스가 널리 보급되고 Microsoft의 Access나 Borland의 Paradox와 같이 PC 전용의 DBMS도 등장하여 일반 사용자가 많이 이용하게 됨으로써 사용자 인터페이스의 유형도 그래픽을 가미한 GUI의 형태로 발전하게 된다. 더 나아가서는 자연어로 입력을 받아 사용자 질의를 처리해 주는 좀더 진보된 형태의 사용자 인터페이스에 대한 연구가 많이 진행되어 왔다. 이 글에서는 데이터베이스를 사용하기 위한 사용자 인터페이스의 종류와 각각의 장단점, 그리고 특히 최근 많이 거론되고 있는 멀티미디어 데이터베이스의 사용자 인터페이스에 대하여 기술하고자 한다.

2. 사용자 인터페이스의 필요성

사용자의 인터페이스를 설계하려면 먼저 어떤 종류의 사용자가 있는지 그 유형부터

살펴볼 필요가 있다. 데이터베이스 시스템과 관계되는 사용자는 데이터베이스 관리자, 데이터베이스 설계자, 시스템 분석가, 응용 프로그래머, 일반 사용자 등 그 범위가 다양하다. 데이터베이스 관리자는 모든 자원을 관리하고 보안이나 재생 등의 기능을 담당한다. 데이터베이스 설계자는 데이터의 구조나 상호관계를 정의하고 실제로 데이터베이스를 구현한다. 시스템 분석가는 사용자의 요구사항을 결정하고 그것을 만족시키기 위한 규격을 정하게 된다. 응용 프로그래머는 이 규격을 프로그램으로 구현하고 테스트와 문서 작성 등의 프로그램 유지 관리를 한다. 일반 사용자는 데이터베이스에 저장된 데이터를 질의를 통해서 검색한다. 이런 다양한 유형의 사용자중 데이터베이스 시스템에서 사용자 인터페이스를 강조하는 것은 대부분 일반 사용자를 대상으로 한다.

일반 사용자는 다시 몇가지 부류로 나눌 수 있다. 첫째, 데이터베이스를 간헐적으로 사용하는 부류 (Casual End Users)로써 사용할 때마다 서로 다른 정보를 요구하게 되며, 주로 복잡한 데이터베이스 질의어를 사용하여 요구를 한다.

둘째, 고급의 데이터베이스 접근 기술은 필요치 않으면서 항상 거의 정형화된 트랜잭션 (Canned Transaction)을 수시로 사용하는 부류 (Parametric End Users)로써 은행직원이나 여행사의 티켓 예약 담당자 등이 이에 속할 수 있다.

셋째, 공학자, 과학자, 경제분석가 등의 전문 분야 종사자로써 주로 복잡한 작업에 대한 요구를 표현하기 위해 DBMS의 기능을 자세히 아는 부류 (Sophisticated End Users)이다.

넷째, PC용 DBMS를 통해 독립적으로 데이터베이스를 이용하는 부류 (Stand-Alone Users)로써 메뉴나 그래픽 인터페이스에 익숙한 계층이다. 이러한 여러 가지 계층의 일반 사용자들이 DBMS를 이용하기 때문에 사용자 인터페이스도 이런 다양한 부류를 골고루 다 만족시키는 방향으로 접근이 되어야 한다.

역사적으로는 1960년대말에 DBMS가 처음 소개되었을때만 해도 일반 사용자는 데이터베이스에 대한 직접 접근이 허용되지 않았으며 구두를 통해 프로그래머에게 그들의 요구를 전달할 수 밖에 없었다. 이 시기에는 거의 모든 요구가 절차언어 (Procedural Language)를 통한 batch program으로 작성되었고 간헐적으로 서식 기반 (Form-Based)의 인터페이스가 상점 직원이나 은행 직원들의 수행을 도와주는 정도였다. 1970년대 들어서는 SQL, QBE와 같은 질의 언어가 보편화되었으며 주로 관계형 DBMS에서 사용되었다. 그 후 강력한 그래픽 기능을 갖춘 워크스테이션과 윈도우 기반 환경의 등장으로 그래픽 사용자 인터페이스가 나타나게 되고 이것이 데이터베이스의 이용을 촉진시키는 계기가 되었다.

현재 대부분의 DBMS가 지향하는 사용자 인터페이스는 서식 기반 질의이다. 이것은 매개 질의 (Parameterized Query)라고 표현할 수도 있으며 사용자가 복잡한 질의 언어의 구문을 몰라도 서식의 빈칸만 채우면 되도록 편의를 도모한다.

이외에도 아이콘 기반 인터페이스와 마우스 기반 인터페이스 등이 등장하게 된다. 자연어를 이용한 인터페이스에 대한 연구도 자연 언어 처리분야와 연계되어 지속적으로 연구되고 있다.

3. 사용자 인터페이스 유형

DBMS에서 제공되는 사용자 인터페이스에는 다음과 같은 종류가 있다.

3.1 메뉴 기반 인터페이스 (menu-based interface)

메뉴기반 인터페이스는 메뉴를 통해 사용자에게 여러 가지 옵션을 제시하고 그 선택을 통해 요구사항에 대한 질의를 형성하게 도와준다. 메뉴의 가장 큰 장점은 사용자로 하여금 질의어의 특정 명령어

나 구문 구조를 일일이 외울 필요가 없도록 도와주는 것이며 메뉴에 나타나는 옵션들을 단계적으로 선택함으로써 질의를 형성하도록 한다. 풀다운 메뉴(Pull-Down Menu)가 윈도우 기반 사용자 인터페이스에서 널리 사용되는 방법이며 사용자가 데이터베이스의 내용을 개략적으로 볼 수 있는 브라우징 인터페이스로서도 이용된다.

3.2 그래픽 인터페이스 (Graphical Interface)

그래픽 인터페이스는 사용자에게 데이터베이스의 스키마를 다이어그램의 형태로 보여주는 역할을 한다. 사용자는 이 다이어그램을 조작함으로써 질의를 할 수 있다. 많은 경우에 그래픽 인터페이스는 메뉴기반 인터페이스와 결합되어 사용되고 있으며 주로 마우스를 통하여 스키마 다이어그램의 한 부분을 선택할 수 있도록 한다.

3.3 서식 기반 인터페이스 (Form-Based Interface)

서식 기반 인터페이스는 사용자에게 서식을 제공하여 각 항목을 채워넣음으로써 질의를 형성하게 도와주는 역할을 한다. 사용자는 서식의 모든 항목을 채워넣을 수도 있고 일부만 기술할 수도 있는데 이때는 DBMS가 디폴트값을 지정하게 된다. 서식은 대부분 상점의 물품판매나 은행의 예금거래 등 정형화된 트랜잭션을 이용하는 전문적인 지식이 요구되지 않는 사용자들을 위해 사용되고 있다. 많은 DBMS에서 서식 기술 언어 (Form Specification Language)를 제공하여 프로그래머가 사용자 서식을 쉽게 설계하도록 도와주고 있으며, 또한 사용자가 화면에 제시된 샘플 서식을 보고 인터액티브하게 새로운 서식을 정의할 수 있도록 한다.

3.4 자연어 인터페이스 (Natural Language Interface)

한글이나 영어 문장과 같이 일반 사람이 사용하는 자연어로 질의를 할 수 있도록 한다. 자연어 인터페이스는 데이터베이스 개념 스키마와 비슷한 스키마를 가지고 있으며, 사용자의 요구를 해석하기 위해 일반적인 사전정보와 더불어 이 자연어 스키마의 단어를 이용한다. 만일 해석이 성공적으로 이루어지면 SQL과 같은 질의어 명령으로 바뀌어 DBMS에게 전달되고, 해석이 잘 되지 않을 경우는 사용자와의 대화를 통해 요구 질의의 정확한 의도를 파악하게 된다. 자연어로 된 질의를 해석하는 기법으로 크게 세가지 접근방법이 있다.

3.4.1 키워드 추출 또는 패턴 매칭 기법

이 방법은 키워드 시스템에 적용되며, 해석 프로그램은 자연어 문장의 각 단어를 데이터베이스의 특정 필드와 연관시켜 질의어로 바꾸어준다. 하지만 문법을 적용하지 않는 이러한 패턴 매칭 기법은 그 사용 효과가 제한적이다.

3.4.2 파싱

자연어 질의어를 자연어 문법에 기준하여 분석한 후 내부 포맷으로 변환시켜준다. 파싱은 크게 두 가지의 기법이 있는데 자연어의 문법 (Grammar 또는 Syntax)을 기반으로 분석하는 방법과 각 단어의 의미 (Semantic)와 여러 production rule을 바탕으로 의미 분석을 하는 방법이다. 예를 들어 LADDER시스템은 의미 구문 (Semantic Grammar)를 이용하여 미국 해군의 데이터베이스를 접근하고, INTELLECT 시스템은 구문 규칙 (Grammar Rules)에 의존한다.

또한 Rendezous 시스템은 단어사전(Lexicon)과 phrase grammar를 이용하여 자연어 질의를 relational calculus로 변화시킨다.

3.4.3 질의 매핑

자연어 질의를 특정 데이터베이스 질의어로 바꾸기 위해 지식 베이스 또는 world model을 이용하였다. 이 방법에서는 특정 응용 영역에 대한 질의 문장의 전형적인 형태 표현과 언어학적 지식, 그리

고 매핑 지식 등을 함께 이용하게 된다. 질의 매핑의 예로 KDA 시스템이 있다.

3.5 데이터베이스 관리자에 대한 인터페이스 (interfaces for the DBA)

대부분의 데이터베이스 시스템은 관리자만이 사용할 수 있는 명령어를 포함하고 있으며 DBA만이 접근이 허용되는 인터페이스를 제공한다. 이러한 명령어로는 사용자 계정의 생성, 시스템 환경 설정, 계정 권한부여, 스키마 생성, 데이터베이스의 저장 구조를 재구성하는 것 등이 있다.

4. 멀티미디어 DBMS 사용자 인터페이스

멀티미디어 데이터베이스는 기존의 텍스트 위주의 데이터 저장에서 탈피하여 화상, 동영상, 음성 등의 멀티미디어 정보를 저장하고 검색, 편집할 수 있다. 멀티미디어 DBMS에서는 각 미디어의 내용과 동작 시나리오를 표현할 수 있는 데이터 모델과 멀티미디어 검색 질의어, 분산 멀티미디어 관리 등의 연구가 병행되고 있다.

이러한 멀티미디어 DBMS를 위한 사용자 인터페이스는 데이터 정의와 데이터 조작시 멀티미디어 정보를 위한 추가적인 기능이 제공되어야 한다. 멀티미디어 데이터는 데이터간의 논리적인 관계뿐 아니라 시공간 관계, 그리고 사용자 상호 작용 등의 동작 정보를 가지고 있는데 이러한 정보를 표현하는 모델은 시간축 기반 모델, 시간간격 기반 모델, 제어흐름 기반 모델, 스크립트 기반 모델로 나누어진다. 이들 모델 대부분은 시각 환경을 제공하는데, 각 객체를 노드로 나타내어 이들간의 시간관계를 선 또는 화살표로 나타내는 그래프 기반 표현 방식이다.

또한 공간 관계 기술을 위한 시각도구와 상호작용을 위한 시각도구도 제공한다. 멀티미디어 데이터의 조작은 크게 멀티미디어 데이터에 대한 검색과 삽입, 삭제, 갱신 등의 데이터 편집 기능으로 나뉜다. 데이터의 검색시 기존의 문자나 숫자위주의 데이터베이스에 대한 검색은 내용 기반으로 하는 것이 효율적이었으나 멀티미디어 데이터의 경우는 브라우징 방법이 효율적이며, 질의와 브라우징을 통합적으로 제공하는 인터페이스의 필요성이 대두된다. 데이터의 편집시에는 각 응용 프로그램에서 요구하는 기능의 수준이 다르므로 DBMS 자체에서 제공하는 내부 모듈로 다양한 조작 기능을 지원하기 어렵고 외부 데이터 편집 도구를 이용해야 한다.

그러나 외부 도구는 화일 시스템에 기초하므로 DBMS와 불일치가 일어나고 이를 해결하기 위해 외부 도구와 DBMS가 데이터와 제어를 공유하는 메커니즘이 필요하다.

5. 결 론

사용자 인터페이스는 비단 데이터베이스 시스템에서만 거론되는 문제는 아니며 사실상 모든 시스템 프로그램 또는 응용 프로그램을 구현하는 과정에서 중요시되어야 하는 개념이다. 특히 데이터베이스 시스템에서는 그 특성상 사용자가 데이터를 정의하고 조작하는 기능을 기반으로 하기 때문에 질의어를 얼마나 편리하게 작성할 수 있도록 도와주는가가 중요하다. 특히 PC와 윈도우 기반 환경의 GUI가 보편화된 현 시점에서는 사용자들이 익숙한 그래픽을 중심으로 한 인터페이스가 많이 요구된다. 하지만 또 중요한 측면은 데이터베이스 사용자의 다양한 계층을 고려하여 응용 프로그램이 어느 계층을 주 대상으로 할 것인지를 결정하고 그 계층에 맞는 인터페이스를 설계해야 한다. 가령 키보드 사용에 아주 익숙한 사용자들은 메뉴 기반이나 서식 기반의 인터페이스에 대해 속도가 느리기 때문에 오히려 불편해할 수도 있는 것이다.

멀티미디어 DBMS에서는 그 데이터의 특성상 브라우징과 질의를 통합한 인터페이스의 개발이 요구된다. DBC