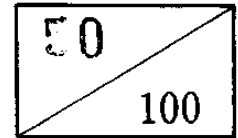


제 1 차년도
년차보고서



다중매체 데이터베이스의 응용에 관한 연구 (I)

A Study on the Implementation of the Multimedia Database(I)

연 구 기 관

한국과학기술연구원
부설시스템공학연구소

과 학 기 술 처

배 포 선

| 사 본 번 호 | 부 수 | 배 포 처 |
|------------------|-----|--------------------|
| 1/100 | 1 | 시스템공학연구소 영구보존용 |
| 2/100 | 1 | 시스템공학연구소 도서과 보관용 |
| 3/100 ~ 6/100 | 4 | 시스템공학연구소 연구개발과 보관용 |
| 7/100 ~ 8/100 | 2 | 시스템공학연구소 해당연구실 |
| 9/100 ~ 11/100 | 3 | 과학기술처 |
| 12/100 ~ 100/100 | 89 | 기타 배포기관 |

제 출 문

과학기술처장관 귀하

본 보고서를 “다중매체 데이터베이스의 응용에 관한 연구” 과제의
1 차년도 년차보고서로 제출합니다.

1991년 7월 9일

주관연구기관명 : 한국 과학기술 연구원
부설시스템공학연구소

| | | |
|-----------|---|-------------------------|
| 연구책임자 : 양 | 유 | 길 (시스템공학연구소) 선임연구원 |
| | 김 | 현 (") |
| 연구원 : 심 | 부 | 성 (") |
| | 양 | 영 종 (") |
| | 강 | 현 용 (") |
| | 홍 | 요 훈 (") |
| | 정 | 효 택 (시스템공학연구소) 연구원 |
| | 정 | 영 은 (") |
| | 조 | 영 표 (") |
| | 김 | 경 일 (") |
| | 김 | 혜 령 (") |
| | 이 | 정 은 (") |
| | 김 | 병 수 (") |

요 약 문

I. 제 목

다중매체 데이터베이스의 응용에 관한 연구(I)

II. 연구 개발의 목적 및 중요성

정보산업의 눈부신 발전으로 컴퓨터의 활용분야와 범위가 확대됨에 따라 실생활과 밀접한 분야에도 컴퓨터의 첨단기술을 적용하게 되었다.

그동안 방송스튜디오에서나 제작, 편집이 가능하던 비디오 및 음성을 포함한 오디오 등의 다중매체 데이터도 컴퓨터에서 제어 및 관리할 수 있게 되었으며, 기존의 문자, 그래픽 및 음향정보와 더불어 한글 윈도우 텍스트, 그래픽 애니메이션, 비디오, 오디오 등 각종의 데이터를 컴퓨터에서 통합 관리할 수 있다는 다중매체 시스템이 선보이게 되었다.

이러한 다중매체 시스템의 기술을 교육, 훈련, 프리젠테이션, 광고및 오락 등의 분야에 광범위하게 이용하기 위해서는 다중매체 데이터의 생성 및 관리와 다중매체 장비의 제어를 위한 관련 하드웨어와 소프트웨어의 기술 개발이 시급히 요구되고 있다. 또한 컴퓨터 시스템 기술과 대용량의 저장매체, 입출력 기기, 디지털 데이터의 압축/복원기술 등 관련 산업기술의 분석과 응용을 통한 다중매체 데이터베이스 개발 및 검색 시스템 기술 개발의 중요성이 부각되고 있다.

따라서, 본 연구 1차년도 목적은 레이저 디스크의 다중매체 정보

이용을 중심으로 한 디바이스 인터페이스, 사용자 인터페이스 및 다중매체 정보검색을 위한 각종 기본 함수를 개발함으로써 다중매체 응용시스템을 누구나 쉽게 구축할 수 있는 기본환경을 제공하고 시범시스템을 개발함에 있다.

Ⅲ. 연구개발의 내용 및 범위

본 과제 1차년도에 세부과제 분야별 연구개발 내용은 다음과 같다.

| 세부과제 | 연구개발내용 | 개발범위 |
|---------------|---|-------------------------|
| 다중매체 관련기술의 분석 | 1. 다중매체 입/출력 기기 인터페이스 기술 2. 저장매체 및 데이터 압축/복원기술 3. 다중매체 개발 지원 응용 소프트웨어 분석 4. 다중매체 데이터베이스 및 통신기술 | 조 사 분 석 " 조 사 |
| 기초함수의 개발 | 1. 다중매체 응용시스템 구축을 위한 개발 환경의 분석 2. 오디오, 비디오 및 사용자 인터페이스 기초 함수의 설계 및 개발 3. 한글 텍스트 입출력 지원 윈도우 함수의 개발 4. 다중매체 정보검색 기능의 설계 및 개발 | 분 석 설계, 개발 " " |
| 시범시스템 개발 | 1. 다중매체 정보검색 시범 시스템 개발을 위한 기초 함수의 통합 및 개발 환경 구축 2. 다중매체 데이터베이스 관리 시범 시스템 설계 및 개발 | 분석, 개발 설계, 개발 |

Ⅳ. 연구개발 결과의 활용방안

1. 다중매체 데이터베이스, 통신, 데이터 압축 / 복원 및 인터페이스기술 등 관련 기술의 분석을 통해 다중매체 기술의 개념정립
2. 오디오, 비디오, 입출력 기기 및 사용자 인터페이스 등 보조기기 제어 함수의 개발을 통해 다중매체 정보의 운용을 위한 기초기술의 습득 가능
3. 다중매체 응용 및 검색 시스템은 각 응용분야별 개발 지원 도구 및 응용시스템 개발을 위한 기술 제공 가능
4. 교육, 훈련, 프리젠테이션, 광고, 데스크 탑 비디오 및 데스크탑 인쇄 등의 응용 분야에 상업화 및 활용이 가능하다.

ABSTRACT

I. Title of the Study

A Study on the Implementation of the Multimedia Database (I)

II. The Goals and Importance of the Study

With the rapid advancement of information industry, the scope and fields of computer utilization are enlarged and the recently developed computer technology became applied to the human's daily life.

Multimedia data, such as video and audio including sounds, which have been produced and manipulated only in the broadcasting stations, now can be created and controlled under the micro computer system, which made Multimedia System come out. In the Multimedia Systems texts, graphics, sounds, Hangul window texts, graphic animations, video, and audio data or information are combined, managed, and delivered all together.

To utilize the Multimedia System technology widely in the fields of education, training, presentation, commercial advertisement and entertainments, efforts for the R & D of the hardware and software integrated technology are very urgent and critical.

And more important technology to be developed now is the multimedia database and information retrieval system technology, through the thorough analysis and application of Multimedia Sys-

tem related techniques such as computer system, multimedia storage, input/output devices, and digital audio/video data compression/decompression.

Therefore, the goals of the first year's study are to implement a prototype multimedia application system and to provide the implementing environments of multimedia database and information retrieval system by developing the basic functions for multimedia device interface, user interface, and information retrieval system, mainly using the multimedia data in the laser discs.

III. Scope and Contents of Study

The followings are the major tasks and detailed contents of the first year study.

| Tasks | Contents | Scope |
|---|--|--|
| Survey & Analysis of Multimedia Technology | <ol style="list-style-type: none"> 1. Multimedia Input/Output Device Interface 2. Compression/Decompression and Storage Device 3. Multimedia Implementation Supporting S/W 4. Multimedia Database & Communication | <p>Survey Analysis</p> <p>Analysis Survey</p> |
| Basic Functions Implementation | <ol style="list-style-type: none"> 1. Implementation Environment for Multimedia Application 2. Basic Functions for Audio, Video and User Interface 3. Window Functions for Hangeul Input/Output 4. Multimedia Information Retrieval System | <p>Analysis</p> <p>Design, Development</p> <p>Design, Development</p> <p>Design, Development</p> |
| Prototype Application for Multimedia Management | <ol style="list-style-type: none"> 1. Integrating the Basic Functions for Application Implementation 2. Multimedia Management and Retrieval System for Demonstration Purpose | <p>Analysis, Development</p> <p>Design, Implementation</p> |

IV. Suggestions on Using the Research Results

1. State of the art will give fundamental ideas and information of multimedia related technology, such as multimedia database, communication, data compression/decompression, and interface devices.
2. Device-control functions for audio/video and user interface can be used as the basic techniques for multimedia information manipulation.
3. Multimedia information management and retrieval system will be used not only as an implementation tool but also as a multimedia application model, especially in the field of digital video manipulation required.
4. This will be used as a gateway for commercializing and utilizing multimedia application systems in education, training, presentation, and desk top video.

CONTENTS

| | |
|---|-----|
| I. INTRODUCTION. | 17 |
| 1. Needs for the Study. | 19 |
| 2. Objectives and Contents of the Study. | 20 |
| II. ANALYSIS OF THE MULTIMEDIA TECHNOLOGY. | 23 |
| 1. Introduction to Multimedia System. | 25 |
| 2. Analysis of Multimedia Related Technology. | 71 |
| 3. Status of Multimedia System. | 110 |
| 4. New Technology for Improvement. | 127 |
| III. ANALYSIS OF THE MULTIMEDIA DATABASE. | 135 |
| 1. Overview. | 137 |
| 2. Relational Database for Multimedia. | 144 |
| 3. Object Oriented Database. | 149 |
| 4. Distributed Database. | 156 |
| 5. Hypermedia Database. | 164 |
| 6. Suggestion for Multimedia Database Technology. | 170 |
| IV. SYSTEM IMPLEMENTATION. | 173 |
| 1. Implementation Environments. | 175 |
| 2. Implementation of Interface Module & Functions. | 185 |
| V. CONCLUSIONS. | 201 |
| REFERENCE. | 207 |
| APPENDIX: Library Functions. | 211 |

목 차

| | |
|-------------------------------------|-----|
| 제 1 장 서 론 | 17 |
| 제 1 절 연구의 필요성 | 19 |
| 제 2 절 연구의 목표 및 내용 | 20 |
| 제 2 장 다중매체 시스템의 기술분석 | 23 |
| 제 1 절 다중매체 시스템의 개요 | 25 |
| 제 2 절 다중매체 관련 기술 분석 | 71 |
| 제 3 절 다중매체 시스템 개발 현황 | 110 |
| 제 4 절 향후 발전 방향 | 127 |
| 제 3 장 다중매체 데이터베이스 기술분석 | 135 |
| 제 1 절 다중매체 데이터베이스의 개요 | 137 |
| 제 2 절 관계형 다중매체 데이터베이스 | 144 |
| 제 3 절 객체지향 데이터베이스 | 149 |
| 제 4 절 분산 데이터베이스 | 156 |
| 제 5 절 하이퍼미디어 데이터베이스 | 164 |
| 제 6 절 다중매체 데이터베이스의 발전 방향 | 170 |
| 제 4 장 시스템 개발 내용 | 173 |
| 제 1 절 시스템 개발환경 | 175 |
| 제 2 절 분야별 개발내용 | 185 |

| | |
|---------------------|-----|
| 제 5 장 결 론 | 201 |
| 참고문헌 | 207 |
| 부록 : 라이브러리 함수 | 211 |

표 목 차

| | |
|---|-----|
| 〈 표 2- 1 〉 VGA 가 지원하는 디스플레이 모드 | 35 |
| 〈 표 2- 2 〉 타가 보드의 종류 및 특성 | 37 |
| 〈 표 2- 3 〉 TARGA+ 보드의 종류 및 특성 | 37 |
| 〈 표 2- 4 〉 비스타 보드의 종류 및 특성 | 38 |
| 〈 표 2- 5 〉 오디오 처리지원 소프트웨어 | 52 |
| 〈 표 2- 6 〉 그래픽 처리지원 소프트웨어 | 54 |
| 〈 표 2- 7 〉 비디오 처리지원 소프트웨어 | 57 |
| 〈 표 2- 8 〉 통합 소프트웨어 | 60 |
| 〈 표 2- 9 〉 다중매체 시스템의 응용분야 | 64 |
| 〈 표 2-10 〉 Laser Disc 와 VTR 의 특성 비교표 | 74 |
| 〈 표 2-11 〉 CD-I 오디오 용량 | 83 |
| 〈 표 2-12 〉 DVI 압축 알고리즘에 의한 이미지 형식 | 87 |
| 〈 표 2-13 〉 CD-ROM에 저장시 매체별 정보량 | 88 |
| 〈 표 2-14 〉 GUI 의 각 요소 비교표 | 93 |
| 〈 표 2-15 〉 압축 / 복원기술 표준화 방안 비교표 | 107 |
| 〈 표 2-16 〉 다중매체 시스템 참여업체 현황 | 112 |
| 〈 표 2-17 〉 Commodore 사의 Amiga 에서의 하드웨어 및 소프트웨어... 114 | 114 |
| 〈 표 2-18 〉 Apple 사의 매킨토시에서의 하드웨어 및 소프트웨어 ... 117 | 117 |
| 〈 표 2-19 〉 IBM 사 PC/AT 와 PS/2 에서의 하드웨어 및 소프트웨어...121 | 121 |
| 〈 표 2-20 〉 다중매체 관련산업의 역사적 배경과 미래 | 129 |

그 립 목 차

| | | |
|------------|--|-----|
| < 그림 2-1 > | 다중매체 시스템의 하드웨어 구성도 | 27 |
| < 그림 2-2 > | CD저장매체 기술의 변천과정 | 79 |
| < 그림 2-3 > | DVI 기술의 발전 추이 | 89 |
| < 그림 2-4 > | 전세계 다중매체 하드웨어 및 소프트웨어 시장 동향 (출처 : IWG) | 134 |
| < 그림 2-5 > | 디지털 비디오를 이용한 PC 시장 | 134 |
| < 그림 3-1 > | 분산 데이터베이스 시스템의 계층적 관계 | 158 |
| < 그림 3-2 > | 하이퍼미디어의 정보 표현 형태 | 165 |
| < 그림 3-3 > | 미래의 데이터베이스 시스템 | 171 |
| < 그림 4-1 > | 다중매체 데이터베이스 시스템 하드웨어 구성도 | 176 |
| < 그림 4-2 > | 다중매체 데이터베이스 시스템 소프트웨어 구성도 | 176 |

제 1 장 서 론

제 1 장 서 론

제 1 절 연구의 필요성

인간은 자기의 의사를 표현하거나 정보를 전달하고자 할 때 가능하면 간결 명료하게 전달하기를 원한다.

따라서 교단에서 학생을 지도하는 교사나 신입사원을 교육하는 기업의 교육담당자나 자신의 회사에서 만든 제품의 판매를 위해 홍보하기를 원하는 영업사원이나 누구든 주어진 환경에서 최선을 다해 자기가 알리고 싶은 내용을 상대방이 쉽게 이해할 수 있도록 하기 위해 시청각도구를 동원하기까지도 한다.

다중매체의 시대라 일컬어지는 90년대를 맞아 다양한 환경변화에 따른 요구에 적절히 대응하고 지식, 정보 등을 전달·수용하기 위해서는 컴퓨터 분야의 새로운 혁신이 요구된다.

따라서 컴퓨터 부문에서도 텍스트나 그래픽, 단순 오디오 데이터를 관리할 수 있던 수준에서 복잡한 그래픽, 애니메이션, 오디오 등의 여러 매체별 데이터를 컴퓨터로 제어·관리할 필요가 있으며, 매우 다양한 자료형태와 이들간의 복잡한 관계를 처리해 줌으로써 사용자가 빠르고 쉽게 자료를 액세스할 수 있도록 관리해 주는 새로운 데이터베이스가 필요하게 되었다. 동시에 데이터를 저장하기 위해서는 대용량의 광 디스크나 동적 화상정보를 보관하기 위한 레이저 디스크 등에 따르는 기술과 데이터를 출력하기 위한 고해상도 디스플레이

기술, 비디오 신호를 디지털 신호로 변환하여 고속으로 디스플레이 하는 기술 등이 필요하다.

다중매체 시스템에서 다루는 데이터는 용량이 매우 크고 가변적이다. 또한 단순한 형태의 레코드로 표현하기 어려운 구조를 가지며, 검색이 어렵고, 데이터의 구성요소가 특정 데이터 형태로 규정하기 어려운 경우가 많다.

따라서 정형화된 데이터와 비정형화된 데이터를 동시에 취급할 수 있는 새로운 데이터 관리개념의 도입이 요구된다.

이러한 다중매체 데이터를 사용자의 입장에서 보다 쉽고 편리하게 액세스할 수 있도록 하기 위해서는 인간과 컴퓨터의 인터페이스 기술 또한 간과할 수 없다.

이러한 모든 요구조건을 충족시키기 위해서는 첨단기술인 다중매체를 이용한 데이터베이스 관리시스템에 대한 연구가 시급히 요구된다.

제2절 연구의 목표 및 내용

다중매체 데이터베이스의 응용을 위해서는 다중매체 데이터관리를 위한 시범시스템의 개발, 다중매체 데이터베이스 관리체계 및 그 응용시스템의 개발을 연구목표로 한다.

따라서 본 연구에서는 다중매체 데이터 관리를 위한 시범시스템의 개발과 이에 따른 입·출력기기와의 인터페이스 방법연구 및 기초합수의 개발을 목표로 하였다.

다중매체를 지원하는 데이터베이스 관리 시스템은 저장해야 할 자료의 성격이나 이들의 복잡한 관계, 그리고 윈도우에 기반을 둔 편리한 사용자 인터페이스 등을 제공해야 하기 때문에 기존의 데이터베이스 관리 시스템의 기능으로는 매우 부족하다.

따라서 본 연구에서는 이러한 요구에 발맞추어 다중매체 시스템의 구성요소별로 해외 및 국내시장 현황과 기술발전 추이를 조사하였으며, 각 매체별 제어를 위한 기본함수와 이를 통합·관리할 수 있는 소프트웨어의 기본함수를 제작함으로써 다중매체 데이터베이스 응용분야의 기초를 이루는데 주력하였다. 특히 사용자 인터페이스 부문에도 초점을 맞추어 터치스크린(Touch Screen)과 윈도우(Windows)의 응용을 위한 기본함수도 제작하였다. 또한 데이터베이스의 시범적 구축을 위해서 기개발된 각 매체별 응용 소프트웨어의 기능과 적용성을 연구하였다.

그리고 정형화된 자료의 관리와 아울러 비정형화된 자료(그래픽, 영상, 음성, 하이퍼 텍스트 등)의 관리를 위하여 기존의 텍스트 위주의 관계형 데이터베이스 기능을 확장시켜 다중매체 데이터베이스의 구현 가능성을 검토하였다.

또한 각 매체별 제어를 위한 기본함수를 개발함으로써 다중매체 시스템의 응용프로그램을 개발하고자 하는 사용자들이 쉽게 이용할 수 있도록 하였으며, 그래픽, 비디오, 오디오 등의 통합 개발환경을 구축하여 컴퓨터를 통해 모든 관련매체를 제어·관리할 수 있게 하였다.

다중매체 데이터베이스의 시작품(Proto-type)을 개발하여 선보임으로써 다방면의 응용분야에 활용 가능함을 보여주었다.

제 2장 다중매체 시스템의 기술 분석

제 2 장 다중매체 시스템의 기술분석

제 1 절 다중매체 시스템의 개요

1. 다중매체 시스템의 개념

다중매체 시스템은 1990년대 인간이 컴퓨터를 이용하여 만들어낸 가장 최신의 주목받는 기술로서 컴퓨터 시스템에 텍스트, 그래픽, 애니메이션, 비디오, 오디오 등의 다양한 정보를 사용하여 모든 시청각 정보를 통합, 조정한 시스템이다.

즉 지금까지 독자적으로 발전해온 다양한 분야의 기술인 영상 기술, 오디오 기술, 사진 기술, 컴퓨터 하드웨어 및 소프트웨어 기술, 저장 기술, 통신 기술 등이 하나로 통합된 시스템으로서, 엄청난 양의 정보를 전달할 수 있을 뿐만 아니라 다양한 문자, 그래픽, 애니메이션, 비디오, 음성 등의 정보를 자유로이 전달할 수 있으며 또한 사용자가 컴퓨터와 서로 의사를 주고 받을 수 있는 상호 작용성을 가진다. 이러한 다양한 매체의 정보를 수용하기 위하여 컴퓨터 화면, 비디오 디스크 플레이어, 광 디스크 드라이버, 비디오 카메라, CD-ROM, 사운드 및 오디오 신디사이저, 정보 통신 등의 다양한 전자 미디어 기능을 한꺼번에 통합, 조정하는데 컴퓨터를 이용한다.

이미 선진 각국에서는 상당한 수준의 다중매체 시스템에 관한 기술이 연구, 개발되고 있는데 미국이나 일본의 경우는 이미 교육 및 훈련 등 사회 각분야에서 응용 시스템이 활용되고 있고 응용 상품들이 판매되고 있다. 그러나 우리나라는 아직까지 다중매체 시스템에

대한 용어도 생소하고 개념도 충분히 알려져 있지 못한 실정이지만 일부 학계와 연구소에서 이에 대한 연구가 활발히 전개되고 있으며 점차적으로 각분야에서 다중매체 시스템을 이용한 응용 시스템이 많이 개발되어 이용될 것으로 전망된다.

2. 다중매체 시스템의 하드웨어

가. 개 요

다중매체 시스템의 출현은 반도체 기술의 발전과 더불어 기존의 텍스트 처리 중심에서 더 나아가 컴퓨터와 각종 매체의 입출력장치를 이용하여 풀모션 비디오, 정지화상, 음성, 사운드, 3차원 그래픽, 애니메이션, 그래픽, 텍스트 등 다양한 매체의 통합된 정보를 생성, 조작, 사용할 수 있도록 하는 통합기술로 발전되고 있다.

이를 위해서는 대용량의 데이터 저장매체와 더불어 방대한 양의 데이터를 읽고, 디스플레이 할 수 있는 각종 입출력장치를 필요로 하며, 이들의 제어를 위한 기술 등이 요구되며, 영상 / 음성정보의 실시간 처리를 위해 대용량의 데이터 전송을 가능케하는 통신기술 등의 발전이 뒤따라야 한다.

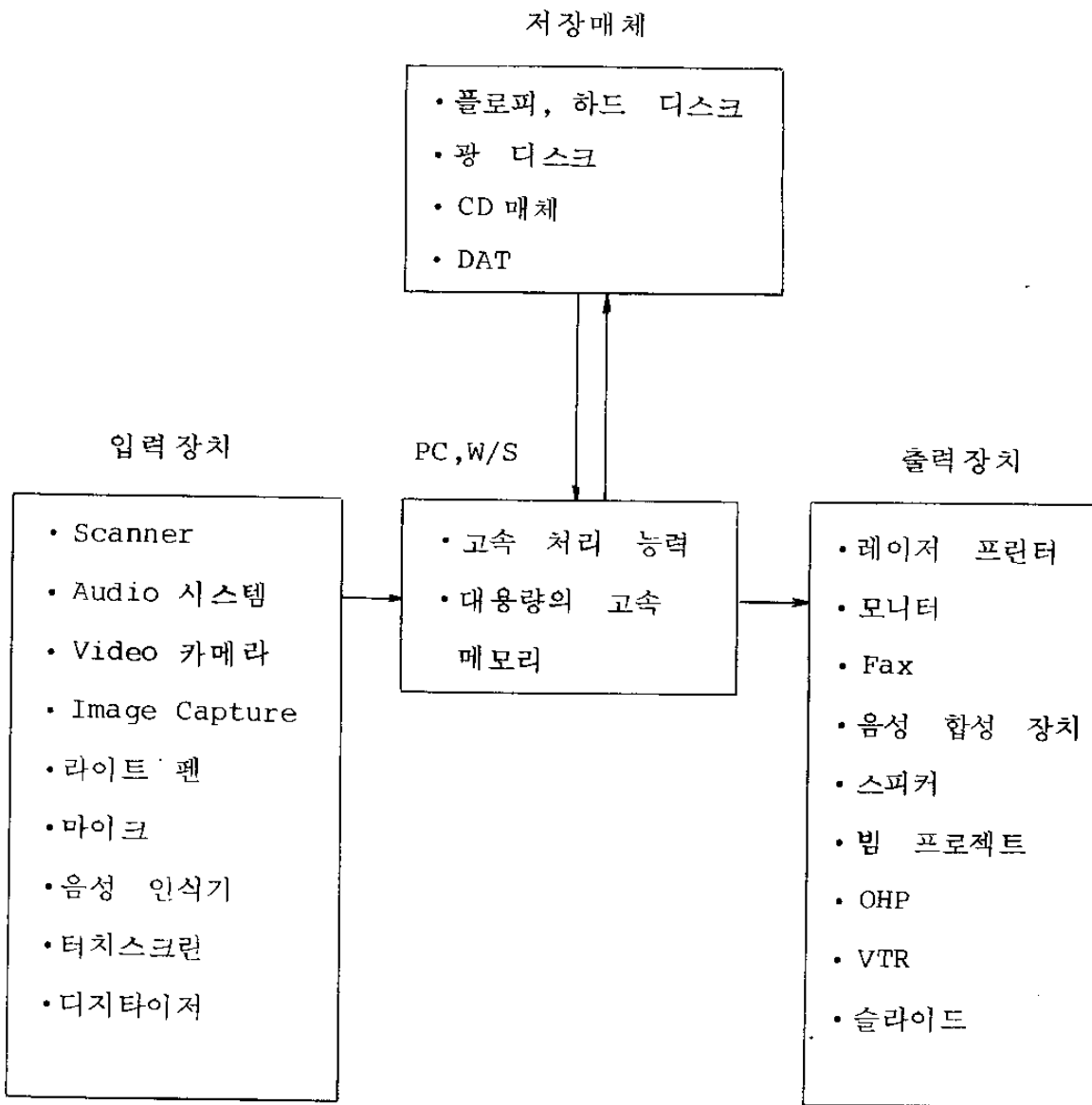
여기서는 이러한 다중매체를 구현하는데 있어서 필요한 하드웨어에 대하여 기술하였다.

나. 하드웨어 시스템 구성

다중매체 시스템을 구현하기 위한 하드웨어는 기본적으로 대용량의 데이터를 처리하고 전달할 수 있는 데이터 전송속도와 주기억 용량, 그래픽 처리를 위한 EGA 이상의 그래픽 보드, VCR이나 TV 영상을 디

지타이징하고 비디오 영상, 음악, 컴퓨터 그래픽 등을 결합시킬 수 있는 비디오 그래픽 보드 등을 갖추어야 한다.

다중매체 시스템을 지원하는 일반적인 하드웨어의 구성은 다음과 같다.



< 그림 2-1 > 다중매체 시스템의 하드웨어 구성도

다. 다중매체 시스템의 기본 하드웨어

(1) 기본사양

(가) 마이크로 프로세서와 주기억장치

일반적으로 컴퓨터의 처리속도는 마이크로 프로세서에 의하여 좌우된다. 특히 한 프레임이 수천만 비트를 차지하는 고화질의 비디오 데이터를 실시간으로 처리하거나 그래픽 이미지, 음성 등의 데이터를 처리하여야 하는 다중매체 환경에서는 곧 마이크로 프로세서의 기술발전이 다중매체의 기술적인 발전의 근간이 된다고 할 수 있다.

수치적으로 대개의 XT는 속도가 8 MHz 정도이고 AT의 경우는 12-16 MHz 인데 다중매체는 한꺼번에 많은 데이터를 읽고, 처리해야 되기 때문에 속도가 30-40 MHz 정도의 처리속도를 갖추어야 한다. 즉, 매초마다 수백 MBytes 이상의 데이터를 읽고 디스플레이할 수 있도록 4 MB 이상의 넉넉한 주기억용량을 갖는 80386 이상의 시스템이 지원되어야 한다. 따라서 이제까지는 CPU의 기능을 다양하게 하기 위해 명령어의 종류를 늘린 방식인 CISC (Complex Instruction Set Computer) 칩에서 CPU의 실행속도를 빠르게 하기 위해서 명령어의 종류를 줄인 방식인 RISC (Reduced Instruction Set Computer) 칩을 채택한 MPU (Micro Processor Unit) 로 변화되어 가고 있다.

RISC 칩은 CISC 칩에 비해 설계 및 제도가 용이하여 저가격에 공급이 가능하며, 고속으로 명령어를 처리하므로 최근에는 RISC 칩을 이용한 고성능 슈퍼 마이크로 컴퓨터도 개발되어 발표되고 있다.

(나) 보조기억장치

비디오 데이터나 그래픽 등의 대용량을 처리하기 위한 기억장치는 비디오 데이터의 경우 1 초에 30 프레임 이상을 스크린에 출력할 수

있는 빠른 전송속도와 대용량의 데이터를 기억할 수 있는 저장장치가 다중매체 시스템 환경에서는 필수적이다. 이로 인해 기억용량의 증대 뿐만 아니라 호출시간의 고속화를 실현할 수 있는 새로운 저장장치가 등장하게 되었다.

일반적으로 사용되는 다중매체의 보조기억장치로는 외장형 하드 디스크, 광 디스크, 레이저 디스크 등이 있다. 이들 기억장치의 공통적인 특징은 대량의 데이터를 빠르고 안전하게 보관할 수 있고 컴퓨터 사용에서의 이동이 용이한 장점을 갖고 있다.

특히 레이저 디스크의 경우 동적 화상정보까지도 저장하여 레이저 디스크 플레이어 (LDP) 를 이용한다면 디지털 비디오 보드를 내장하고 있는 퍼스널 컴퓨터에서 영화를 보는 것처럼 생생한 모션 비디오정보를 디스플레이해서 볼 수 있다.

(2) 비디오 / 오디오 프로세서

다중매체 환경 하에서 비디오나 오디오 처리 프로세서는 여러가지 처리보드가 발표되고 있다. Intel 의 경우 두개의 고밀도 VLSI 칩의 비디오 프로세서가 풀스크린, 풀모션 비디오 스틸 등을 처리한다. C-cube Microsystems 의 CL550 칩은 JPEG에서 채택된 알고리즘에 사용되는 이미지 프로세서 칩으로 AT 혹은 매킨토시에 add-in 보드로서 사용되고 있다.

기존의 오디오 / 비디오 프로세서들은 단지 특정한 디스플레이 해상도에 고정된 스탠다드 기능을 제공하였으나 DVI에서는 VDP 칩에 있는 비디오 프로세서에 의해 새로운 프리미티브를 개발할 수 있는 환경을 융통성 있게 제공하고 있다. 또 다양한 데이터 저장장치로 부터 오디오의 독립된 멀티플 트랙을 만들어 음악이나 음향효과 등의

오디오와 샘플링 레이트를 자유롭게 할 수 있다.

라. 저장매체 및 장비

(1) 외장형 하드 디스크(Hard Disk)

외장형 하드 디스크는 플로피 디스크의 이동성과 하드 디스크의 용량과 속도의 장점을 살려, 아직까지 다중매체 시스템의 기억매체로 널리 사용되고 있지 못하는 광 디스크의 전환기 기억매체로 사용되고 있다.

광 디스크가 아직 가격이 비싸고 속도가 늦은 반면 외장형 하드 디스크는 카트리지가당 기억용량이 44.5MB에 달하며 액세스 시간은 25ms로서 기존의 하드 디스크와 큰 차이가 없으며 카트리지를 교환하면 45MB의 데이터를 저장해 놓을 수가 있고, 다른 컴퓨터에서 작성한 파일은 카트리지만 가져와 읽어낼 수 있다.

외장형 하드 디스크는 많은 데이터를 다루거나 빈번한 데이터의 이동이 요구되는 환경에서 편리하게 사용할 수 있다.

(2) 광 디스크

대용량의 정보를 기록할 수 있는 장치에 대한 요구가 갈수록 높아지면서 출현하게 된 것이 광 디스크이다.

광 디스크는 레이저 기술을 기반으로 높은 기억밀도와 반영구적인 특수성으로 인해 더욱 새로운 기술향상과 더불어 응용의 확대를 불러왔다.

1970년대 Philips 사에 의해 처음으로 광 디스크가 개발된 이래 1980년대 CD-ROM이 상업적인 제품으로 완성되면서 부터 다중매체의 저장장치로 끊임없이 발전하여 왔다.

광 디스크의 장점은 디스크 한장에 600-650MB의 데이터를 저장할 수 있는 대용량, 데이터 검색시 평균 0.7초 정도의 고속 액세스, 단위

바이트당 낮은 비용으로 보급이 가능하며, 경제성이 높고 외상에도 강하다.

최근에는 하나의 박스안에 여러장의 광 디스크를 넣고 오토체인저를 사용하여 정보를 검색할 수 있는 Jukebox 개념이 도입되면서 광 디스크의 용량이 보통 30 GB를 상회하게 되었다. 그래서 용량 부족으로 기록이 어려웠던 고해상도의 칼라 이미지, 화상정보, 그래픽 및 이미지 형태의 대용량 다중매체 데이터베이스의 경우처럼 큰 정보량의 저장 및 검색에 적합한 저장매체로 자리잡아 가고 있다.

광 디스크는 읽기만 가능한 ROM형, 한번 쓰기가 가능한 WORM, 읽기 쓰기가 모두 가능한 Erasable 광 Disk 등으로 구분된다.

(가) ROM 형

DAD(Digital Audio Disc)의 한 방식으로 출발하게 된 CD(Compact Disc)는 스피커와 소프트웨어만 있으면 CD-ROM 드라이버를 통해서도 CD의 음악을 들을 수 있는데 이는 종래의 아날로그 디스크와는 전혀 다른 PCM(Pulse Code Modulation)이라는 디지털 전송방식으로 기록된 재생전용 레코드이기 때문이다.

이런 특성이 컴퓨터의 기억매체로 사용될 수 있음이 1983년 Philips사에 의해서 발표되었는데 이것이 오늘날 광 디스크 기억매체의 주류를 이루고 있는 CD-ROM이다. CD-ROM의 저장용량은 디스크당 640 MB 이상으로 많은 양의 데이터를 저장할 수 있어 대규모의 데이터베이스, 백과사전, 도서관리 업무 등에 주로 이용하고 있다.

CD-ROM은 대용량을 기억시킬 수 있는 반면 액세스 속도면에서는 기존의 하드 디스크와 큰 차이가 있으며 CD-ROM에 저장되는 정보는 단지 재생만이 가능하기 때문에 정보를 추가하거나 변경할 필요

가 없는 정확한 정보를 저장하여야 한다.

(나) WORM (Write Once Read Many)

CD-ROM이 단지 읽기만 가능한 기억매체인데 비해 기록이 불가능하다는 치명적인 약점을 보완하기 위해 1회에 한하여 쓸 수 있도록 한 기억매체가 WORM이다. 처음에는 미니급 컴퓨터에서 사용되다가 1986년 Philips와 Sony 등의 제작회사들이 5.25인치 CD를 표준으로 발표하였으나 8인치, 12인치 등의 여러 제품이 존재하며, 용량도 100MB에서 1GB 이상까지 다양하여 아직 국제적인 표준으로 통용되지는 않고 있다.

한번의 쓰기가 가능하여도 기록할 수 있는 기능을 넣기 위해서는 고도의 레이저 기술을 필요로 하므로 CD-ROM에 비해서는 매우 고가이다.

그래서 WORM 드라이브는 기록 중심의 백업용 장치로 주로 사용한다. 예를들어 보험회사의 경우 조사 자료를 저장할 수 있고 엔지니어링 설계 회사는 모든 설계의 개정판들을 영구 기록하는데 WORM 드라이브가 효과적으로 이용된다.

여기서 발전한 것이 광 파일링 시스템인데 이것이 WORM 드라이브의 특성이 집약적으로 반영된 실용제품이다.

(다) Erasable 광 디스크

Erasable 광 디스크는 사용자가 기록, 소거, 재생을 반복할 수 있는 소거가능형 (Erasable Type)으로서 주로 컴퓨터용 메모리에 많이 쓰이고 있으며 막대한 양의 데이터를 저장, 수정하고 다시 저장하는데 최적이다. 그래서 칼라 그래픽과 의학, 과학 데이터 같은 전문적인 영상처리가 주 응용분야이다. 또 CAD/CAM에서의 데이터의 저장,

수정에 용이하며 컴퓨터 백업용으로 테이프 라이브러리를 대체해 가고 있다.

Erasable 광 디스크는 자기 디스크에 비해 용량이 풍부하고 안정성이 뛰어나긴 하나 액세스 속도가 뒤떨어지고 아직 가격이 비싸 자기 저장 장치를 대체하기까지는 많은 기술개발이 요구된다.

(3) 플로피 디스크 (Floppy Disk)

퍼스널 컴퓨터에서 기억매체로 가장 많이 그리고 널리 사용되고 있는 플로피 디스크는 그 저장용량이 360 KB-1.2MB 정도의 저장용량 밖에 갖지 못하였다. 그러나 90년대 들어 초고밀도 (VHD: Very High-Density) 플로피 디스크의 등장으로 지금의 10배가 넘는 저장용량을 갖게된 플로피 디스크가 발표되고 있다.

VHD 플로피 디스크는 기존의 플로피 디스크와의 호환성을 고려하여 외형의 크기는 같지만 저장용량이 늘어 칼라 이미지, 탁상출판문서, 대용량의 데이터베이스와 다중매체 데이터의 저장매체로 이용될 것이다.

Toshiba 사는 기존의 드라이브 방법을 그대로 사용하면서 비트밀도를 두배로 증가시킨 2.88MB의 포맷을 제안하였고 NEC, Brier, Insite 사에서는 새로운 광 또는 자기방식을 추진하고 있다.

NEC 사의 내장서보 (Embedded-servo) 기술은 10.8MB 까지 저장할 수 있고 Brier 사의 트윈타이어 (Twin Tier) 서보 방식은 21.4 MB 까지 저장할 수 있으며, Insite 사의 광학서보 방식은 한장의 플롭티칼 (Fl-optical) 디스크에 20.8 MB 까지 저장할 수 있다.

마. 다중매체 처리보드

(1) 비디오 처리보드

비디오처리 보드는 VCR, CATV, 비디오 카메라, 레이저 디스크 플레이어 등을 컴퓨터에 연결해서 비디오 신호를 저장, 편집, 재생할 수 있도록 지원하는 보드로서 내부적으로는 여러개의 다른 신호방식(NTSC 방식, PAL 방식, SECAM 방식)이 있다.

NTSC 방식의 VCR이나 TV 신호로 표시된 그래픽, 비디오, 문자 등을 컴퓨터에서 읽고 편집하기 위해서 디코더가 NTSC 신호를 RGB 신호로 디코드하고 수정, 편집된 영상을 테이프 인코딩 하는 과정을 수행한다.

상기 장비의 비디오 신호를 그래픽 이미지로 변환하여 처리 할 시 컴퓨터의 속도때문에 모션의 속도나 칼라의 수준이 떨어진다. 그래서 비디오 데이터의 처리속도를 높이기 위해 그래픽 액셀레이터를 비디오 보드에 첨가시킨 보드가 많이 나오고 있다.

(2) 오디오 처리보드

오디오처리 보드는 각종 음성이나 음향 등의 아날로그 사운드 신호를 디지털 화일로 변환하고 재생하는데 사용된다. IBM, Apple, NEC 계열에서 널리 사용되는 미디(MIDI) 프로세싱 유닛은 전자악기와 컴퓨터를 연결해서 악기를 제어하고 원하는 악기의 음색을 직접 마이크를 통해 샘플링 하는 샘플러 등이 있으며 매킨토시의 경우 음성, 음악 등이 별도의 보드없이 맥레코더를 사용하여 처리한다.

(3) 그래픽 처리보드

(가) VGA (Video Graphic Array)

퍼스널 컴퓨터용 그래픽 보드 시장을 선도하는 IBM이 PS/2 시리즈를 발표하면서 시스템 보드에 내장된 형태로 제공되었던 VGA 보드

는 뛰어난 그래픽 기능과 다양한 칼라로 인해 확장보드의 형태로 발전되어 왔다.

기존에 발표된 그래픽 보드들은 모두 디지털 방식의 한 종류인 TTL (Transistor Transistor Logic) 신호 방식을 이용하고 있는데 VGA 보드는 아날로그 신호 방식을 사용하고 있다. 아날로그 방식은 이론상으로 무한대의 색상표현이 가능한 방식이다. 그러나 램 (RAM)의 용량이 제한이 있어 아직은 256 가지의 색상표현이 최대이며 VGA 보드가 이를 제공한다. VGA 보드의 표준해상도는 640 × 480 이며 램 (RAM)의 용량을 늘이면 1024 × 768 까지의 해상도를 지원할 수 있다. 그러나 이 경우에는 색상표현에 제한을 받아 16 가지 칼라의 색상밖에 나타내지 못한다.

이러한 제약점을 타개하기 위해 VGA 기능을 보다 향상시킨 SVGA (Super VGA)가 1988년 VESA (Video Electronics Standards Association)에 의해서 발표되었는데 현재는 16 칼라의 색상을 지원할 경우 1768 × 1768 까지의 지원이 가능한 SVGA가 개발되었다.

일반적인 VGA 보드의 사양은 아래의 표와 같다.

< 표 2-1 > VGA가 지원하는 디스플레이 모드

| 모 드 | Type | 해 상 도 | Text Format | Char. Box | Colors | Emul- ation |
|--------|------|-----------|----------------|--------------|--------|----------------|
| 0, 1 | text | 320 × 200 | 25 × 40 | 8 × 8 | 16 | CGA |
| 2, 3 | text | 640 × 200 | 25 × 80 | 8 × 8 | 16 | CGA |
| 0*, 1* | text | 320 × 350 | 25 × 40 | 14 × 8 | 16 | EGA |

(다음 페이지에 계속)

(< 표 2-1 > 계속)

| 모 드 | Type | 해 상 도 | Text Format | Char. Box | Colors | Emul- ation |
|-------|---------|-----------|----------------|--------------|--------|----------------|
| 2*,3* | text | 640 × 350 | 25 × 80 | 14 × 8 | 16 | EGA+ |
| 0+,1+ | text | 360 × 400 | 25 × 40 | 16 × 9 | 16 | VGA |
| 2+,3+ | text | 720 × 400 | 25 × 80 | 16 × 9 | 16 | VGA |
| 4,5 | graphic | 320 × 200 | 25 × 40 | 8 × 8 | 4 | CGA |
| 6 | graphic | 640 × 200 | 25 × 80 | 8 × 8 | 2 | CGA |
| 7 | text | 720 × 350 | 25 × 80 | 14 × 9 | mono | MDA |
| 7+ | text | 720 × 400 | 25 × 80 | 16 × 9 | mono | VGA |
| 13 | graphic | 320 × 200 | 25 × 40 | 8 × 8 | 16 | EGA |
| 14 | graphic | 640 × 200 | 25 × 80 | 8 × 8 | 16 | EGA |
| 15 | graphic | 640 × 350 | 25 × 80 | 14 × 8 | mono | EGA |
| 16 | graphic | 640 × 350 | 25 × 80 | 14 × 8 | 16 | EGA |
| 17 | graphic | 640 × 480 | 25 × 80 | 16 × 8 | 2 | VGA |
| 18 | graphic | 640 × 480 | 25 × 80 | 16 × 8 | 16 | VGA |
| 19 | graphic | 320 × 200 | 25 × 40 | 8 × 8 | 256 | VGA |

* : EGA 카드

+ : 고급모드

(나) 타가 보드 (TARGA : Truevision Advanced Raster Graphics Adapter)

80년대 초 Truevision사에서 발표된 타가 보드는 컴퓨터 그래픽 전문 보드로서 애니메이션과 기타 응용 이미지 시스템에 널리 사용

되고 있다.

타가 보드의 종류와 특성은 아래 표와 같다.

< 표 2-2 > 타가 보드의 종류 및 특성

| 종류 특징 | TARGA 8 | TARGA 16 | TARGA 24 | TARGA 32 |
|----------|---------------------|------------------------|------------------------|----------------------|
| 컬러 수 | 256 | 32,768 | 16,777,216 | 16,777,216 |
| 입력신호 | RS 170 | RS 170A, RGB | RGB | RGB |
| 출력신호 | RS 170 | RS 170A, RGB | RGB | RGB |
| 메모리 | 8bit/pixel 256KB | 16 bit/pixel 512 KB | 24 bit/pixel 768 KB | 32 bit/pixel 1 MB |

최근에는 타가 보드에서 더 향상된 기능을 갖는 TARGA+ 보드가 발표되었는데 그 종류와 특성은 아래 표와 같다.

< 표 2-3 > TARGA+ 보드의 종류 및 특성

| NTSC | TARGA+ 16 | TARGA+ 16/32 | TARGA+ 64 |
|--------------|------------|--------------|---------------|
| 메모리 | 0.5 MB | 1 MB | 2 MB |
| 최대 해상도 | | | |
| 8 bit/pixel | 1024 × 512 | 1024 × 512DB | 1024 × 1024DB |
| 16 bit/pixel | 512 × 512 | 1024 × 512 | 1024 × 1024 |
| 24 bit/pixel | | 512 × 512 | 512 × 1024 |
| 32 bit/pixel | | 512 × 512 | 512 × 1024 |

DB = Double Buffered

(대) 비스타 (VISTA) 보드

현재 발표된 컴퓨터 그래픽 전문 보드로는 가장 고가로서 IBM AT 호환기종이나 386 PC 기종 및 매킨토시 기종에서 사용된다. 비스타 보드는 보드상에서 4MB의 비디오 램을 사용하여 NTSC 방식과 PAL 방식을 사용하는 모니터와 카메라를 모두 사용할 수 있는 특징이 있다.

비스타 보드의 종류와 특성은 아래의 표와 같다.

< 표 2-4 > 비스타 보드의 종류 및 특성

| 종류 특성 | ATVista 1 M | ATVista 2 M | ATVista 4 M | NUVista 2 M | NUVista 4 M |
|-------------|----------------|----------------|----------------|----------------|----------------|
| 사용기종 | IBM AT | IBM AT | IBM AT | MAC II | MAC 1024 |
| 데이터 버스 | 8,16 | 8,16 | 8,16 | 8,16,32 | 8,16,32 |
| 비디오 메모리 | 1 M | 2 M | 4 M | 2 M | 4 M |
| 해상도 | | | | | |
| 8bit/pixel | 512×512 | 1024×512 | 1024×1024 | 1024×1024 | 1024×1024 |
| 16bit/pixel | 1024×512 | 1024×1024 | 2048×1024 | 2048×1024 | 2048×1024 |
| 32bit/pixel | 1024×1024 | 2048×1024 | 2048×2048 | 2048×1024 | 2048×2048 |
| VMX 확장 | 2-10 M | 2-10 M | 2-10 M | 2-8 M | 2-8 M |

바. 다중매체 입·출력장비

최근의 입·출력장비들은 인간과 기계와의 인터페이스가 인간에게 친숙하게 접근될 수 있는 방향으로 나아가고 있다. 특히 다중매체 시

시스템에서 갖추어야 할 입·출력 장비들은 기능면에서 뿐만 아니라 그 사용면에서도 다양화되고 고기술이 집약된 형태로 발전되고 있다.

여기서는 널리 사용되고 있는 입·출력 장비들에 대해서 알아보았다.

(1) 입력지원 장비

(가) 디지털화 태블릿 (Digitizing Tablet)

국내에서는 디지털타저로 알려진 디지털화 태블릿은 사용의 편의성과 저렴한 가격, 다양한 어플리케이션 구축 등으로 CAD 사용자의 전유물에서 더 나아가 우리에게 친숙한 입력장치로 바뀌어 가고 있다.

디지털화 태블릿의 최대 강점은 포인팅에 있다. 이 포인팅만으로도면, 청사진 등을 신속하고 정확하게 추적할 수 있다. 디지털화 태블릿이란 용어는 평면상의 위치들을 디지털 값으로 변환시키는 과정을 의미한다.

디지털화 태블릿은 태블릿 상의 위치를 검출할 수 있으며, 그후 태블릿 펜이나 커서를 동일한 위치에 다시 올려 놓음으로써 스크린 상의 동일한 포인터로 돌아갈 수 있게 된다. 결국 포인터는 태블릿 상의 해당 위치에 논리적으로 고정되어 있다고 볼 수 있다.

(나) 스캐너 (Scanner)

스캐너는 종이에 인쇄된 내용물을 RAM이나 디스크 내의 정보로 변환시키는 장비이다. 스캐너는 그래픽 스캐닝과 광학문자 인식기능(OCR)을 할 수 있는데 그래픽 스캐닝은 그림, 차트, 서명 (Sign) 등의 정보를 쉽게 입력할 수 있으며 광학문자 인식은 스캐너가 종이 위의 밝고 어두운 부분을 읽은 다음 컴퓨터 화일로 이미지를 저장하면 OCR 소프트웨어가 문자나 숫자로 그 밝고 어두운 부분의 명암을 분석한다. 스캐너의 해상도는 페이지의 이미지를 보통 인치당 도트 (DPI)

로 나타낸다. DPI가 증가할수록 해상도가 증가하고 그 이미지의 질도 향상된다.

(대) 오디오 시스템

컴퓨터에 내장되어 있는 작은 스피커로는 음의 발생과 증폭에 있어서 한계가 있다. 이러한 한계를 극복하기 위해 전자악기와 컴퓨터를 연결하는 미디카드와 미디프로세싱 유닛 카드를 사용하여 악기들을 제어하거나 외부로 부터 음원을 입력하는 신디사이저 등을 사용한다.

(2) 출력지원 장비

(개) 모니터 (Monitor)

다중매체 시스템 환경을 지원하기 위한 모니터는 기능이나 크기에 있어서 다양성과 높은 해상도가 요구된다. 이는 보다 효과적으로 시각적인 정보를 제공할 수 있기 때문이다. 최근에는 고품질의 모니터 등이 발표되고 있는데 거의 일본이 기술을 독점하고 있는 실정이다.

640 × 480의 해상도를 넘어 1024 × 768, 2048 × 1024의 해상도가 지원되는 모니터도 발표되고 있다.

(나) 프린터 (Printer)

레이저 프린터의 등장으로 인해 간단한 그래픽에서 부터 사진, 고급의 고해상도 그래픽까지 자유자재로 처리할 수 있게 되었다. 그래픽 출력에 있어 일일이 좌표를 지정해 주지 않아도 되는 포스트 스크립트를 지원하는 컬러 프린터도 발표되었다.

앞으로의 출력장치의 특징은 디스플레이와 프린터를 구별하지 않고 다룰 수 있게 된다는 점이다. 화면에서 볼수 있는 영상을 위지윅 (WYSIWYG)이 실현되고 있다. 즉 프린터의 성능향상으로 화면의 영

상과 출력의 일치가 가능해지게 되었다.

사. 사용자 인터페이스 장비

(1) 키보드 (Keyboard)

컴퓨터에 있어서 키보드는 분리해서 생각할 수 없을 만큼 컴퓨터의 일부분으로서 가장 보편화 된 장비이다. 키보드의 키의 작동원리는 어떤 키 스위치 기술을 사용하느냐에 따라 달라지는데 보통 PC 키보드에는 스프링을 이용한 커패시티브 방식과 미케니컬 방식을 사용하며 스프링 대신에 도체로 된 돔 모양의 고무를 이용하는 컨택터브 러버돔과 껌브레인 방식을 사용하는 키보드도 있다.

(2) 마우스 (Mouse)

오늘날 키보드 대체 입력장치로 가장 널리 사용되고 있는 마우스는 현재 다양하고 풍부한 어플리케이션들을 가지고 있다. 마우스가 단순한 장치이면서도 관심의 대상이 되는 것은 손동작이 마치 커서의 이동과 같기 때문에 쉽게 사용할 수 있기 때문이다. 마우스는 신호처리 방식에 따라 3가지로 구분할 수 있다.

먼저 일반적으로 가장 널리 사용되는 방식으로 미케니컬 마우스가 있다. 이것은 다시 볼 타입과 휠 타입으로 나눌 수 있는데, 볼 타입 마우스는 바닥에 돌출된 볼의 움직임에 따라 볼과 맞닿아 있는 두개의 엔코더를 회전시켜 가로축(x), 세로축(y)의 두가지 전기신호로 분해해서 컴퓨터로 전달하면 컴퓨터는 이 신호를 사용해서 커서를 이동시키는 작업을 수행한다. 휠 타입은 엔코더에 연결된 두개의 휠이 바닥에 돌출되어 마우스의 움직임을 휠이 받아 전기적 신호로 변환하여 컴퓨터를 작동시킨다.

옵티컬 마우스는 서로 다른 파장의 빛을 발광 다이오드(LED)로 쏘

아 그 반대편의 감광센서가 이를 읽어들이게 하는 것으로 반드시 격자가 그려진 패드를 필요로 한다. 패드는 검정색과 파란색의 2색이 있는데 발광 다이오드가 발사한 빛이 패드에 그려진 격자를 지날 때 마다 감광센서가 그것을 감지해 전기적 신호로 컴퓨터에 전달한다. 옵토 미케니컬 마우스는 볼을 내장하고 있으며 볼에 연결된 엔코더의 회전에 따라 반대편에 달린 작은 휠이 회전하게 된다. 이 휠에는 작은 구멍이 나 있는데 휠 양쪽에는 각각 발광 다이오드의 감광 센서가 있어 휠의 회전에 따라 발광 다이오드가 발신한 빛을 감광 센서가 읽어들이어 이 신호로 컴퓨터는 커서를 움직여 작업을 수행하게 된다.

(3) 트랙볼 (Track Ball)

트랙볼은 평평한 밑바닥에 볼을 고정시키고 사용자는 손바닥으로 볼을 굴려 원하는 메뉴를 선택하는 장비로서 많은 공간이 필요하지 않으며 다시 위치 선정을 하지 않고 스크린 주위를 신속하게 이동할 수 있다.

터치 스크린보다 불편하지만 미세한 위치를 감지할 수 있어 복잡한 메뉴화면을 구현하기 용이하다.

(4) 조이스틱 (Joystick)

조이스틱은 손가락 길이 정도의 스틱을 좌우상하로 밀고 당겨서 메뉴를 찾아 간 후 버튼을 눌러 선택할 수 있는 장비로 대부분의 전자오락기에서 조이스틱을 사용하고 있으므로 친숙하게 이용할 수 있으나 위치지정의 정확도가 다소 떨어진다.

(5) 라이트 펜 (Light Pen)

라이트 펜은 화면의 메뉴를 펜으로 지적하여 원하는 동작을 실현

시키는 장비로서 터치스크린처럼 사용될 수 있으며 또한 타블렛 펜처럼 사용하여 화면상의 내용을 수정, 변경할 수 있다.

훈련용 장비로 널리 사용되지만 광범위한 소프트웨어의 지원이 부족하고 화면을 터치하기 위해 펜을 화면 높이로 들어 올려야 하는 불편함이 있다.

(6) 터치스크린 (Touch Screen)

터치스크린은 크게 두가지 방식으로 나눈다.

하나는 스크린상에 전해성 물질을 균등히 코팅하여 손가락으로 화면을 누름으로써 명령이 컴퓨터에 전달되거나 스크린에 감지 센서를 설치하여 스크린 표면에 교차하고 있는 적외선을 차단하여 신호를 발생시키는 방식이다. 초보자가 쉽게 작업할 수 있으므로 대중적인 정보 서비스와 교육 훈련용 시스템에 사용되고 있다.

그러나 스크린상에 손가락으로 미세한 부분까지 터치하기가 힘들고 스크린 코팅이 마멸될때 감도를 잃어버릴 수 있으며 지문으로 스크린이 지저분하게 되기가 쉬운 단점이 있다.

3. 다중매체 시스템의 소프트웨어

가. 기본 소프트웨어

(1) 다중매체 지원 운영체제

다중매체 지원 운영체제는 컴퓨터 하드웨어와 분리하여 생각할 수 없다. 다중매체 시스템은 비디오 데이터, 사운드 데이터등, 기존의 간단한 텍스트, 그래픽 데이터와는 비교도 안될 정도로 방대한 데이터를 다루기 때문에 고성능, 대용량의 하드웨어 시스템을 요구하며 다

양한 다중매체 자원을 처리하기 위하여 추가적인 여러 하드웨어가 필요하게 된다. 현재 다중매체 시스템 구축을 위한 운영체제와 보드들이 계속 개발되고 있으며 다중매체 구현을 위한 하드웨어와 운영체제를 가지고 있어서 다중매체 자원을 처리할 수 있는 환경을 지닌 워크스테이션 및 퍼스널 컴퓨터급 들이 늘어나고 있는 실정이다.

다중매체 지원 운영체제는 모든 다중매체 자원들을 일괄하여 관리하는 방식, 객체지향 접근 방식 등을 제공하여 자원을 범용성 있게 관리한다. 따라서 모든 자원을 통합하는 소프트웨어는 자유롭고 용이하게 각종 자원을 제어함으로써 응용 소프트웨어를 개발할 수 있어야 한다.

최근 들어 운영체제는 사용자가 텍스트 뿐만 아니라 오디오, 비디오 등의 다중매체를 다양하고 편리하게 사용하게 하여 생산성을 향상시킬 뿐만 아니라 응용 소프트웨어 개발자의 부담을 덜게 하기 위하여 뛰어난 GUI(Graphical User Interface)를 제공한다. 즉 아이콘, 윈도우, 다이얼로그 박스, 메뉴 구동 방식 등을 사용하여 기술적 지식이 없는 초보자도 편리하게 응용 프로그램을 작성할 수 있게 되었다.

또한 다중매체 지원 운영체제는 멀티태스킹 기능을 갖추어 여러가지 다양한 매체자원을 효율적으로 이용하는 응용 소프트웨어를 동시에 제공하여야 한다. 즉, 사용자는 한 윈도우에서 3차원 그래픽을 그리다가 다른 윈도우로 이동하여 텍스트 작업을 수행하고 또 다른 윈도우에서는 음성합성을 할 수 있는 등 여러가지 작업을 동시에 수행할 수 있어야 한다.

Microsoft사의 윈도우 3.0은 IBM 호환기종 사용자들에게 그래픽 인터페이스를 제공한다. MS-DOS에는 표준 그래픽 화일 포맷이

없는데 윈도우 3.0은 이를 보완해 주며 복잡하고 어려운 DOS 환경을 그림으로 단순화하여 사용자와 연결시켜 주며 하나 이상의 응용 프로그램을 동시에 실행시킬 수 있다.

Apple사의 매킨토시 운영체제는 뛰어난 그래픽 기능을 갖고 있어서 다중매체 운영체제에 가장 가깝게 설계되어 있는데 24비트 칼라(1,670만 색상 지원)와 8비트 알파 채널을 가진 시스템 유틸리티인 Quick Draw는 그래픽 응용 소프트웨어를 위한 표준 프로그래밍 인터페이스를 제공한다. 현재 Apple사는 새로운 다중매체 운영체제를 위하여 이러한 그래픽 기능이외에 하이퍼카드와 매킨토시 시스템을 결합한 새로운 운영체제를 개발 중에 있다.

이외에도 그래픽 인터페이스를 지원하는 대표적인 UNIX로 AT&T의 Open Look과 OSF의 Motif가 있는데 윈도우, 아이콘, 스크롤링 등의 기능을 갖고 있으며 NeXT사의 NeXT Step Interface도 그래픽 인터페이스를 지향하고 있다.

결론적으로 사용자의 관심이 점차로 텍스트 처리에서 여러 매체가 혼합된 다중매체 처리로, 하드웨어 기능에서 소프트웨어 기능으로, 개인단위의 생산성 추구에서 그룹단위의 생산성 추구로 변화하고 있기 때문에 앞으로의 운영체제는 매킨토시, UNIX, PC-DOS 등의 장점을 결합한 운영체제를 추구하고 될 것이며 음성인식 시스템을 채용한 대화체 방식의 운영체제로 까지 발전하게 될 것이다.

(2) 사용자 인터페이스

사용자 인터페이스는 사용자와 전체 시스템, 응용 프로그램, 혹은 시스템 객체들 간의 상호작용을 조절하는 메카니즘을 뜻한다.

먼저 응용 분야와 분리된 사용자 인터페이스에 대해 개략적으로 알

아본 후 다중매체 시스템과 관련하여 GUI 소프트웨어에 대해 간단히 알아보기로 한다.

(가) 응용 분야로부터 분리된 엔티티 (Entity)로서의 사용자 인터페이스

사용자 인터페이스를 사용자와 응용 프로그램으로 부터 분리된 엔티티로 정의하면 다음과 같은 장점이 있다.

- 만일 시스템이 모듈화 계층 형식 내에서 설계된다면 그 시스템에 대한 연구, 개발, 유지 관리 및 미래의 확장이 더욱 쉬워진다. 그러한 시스템에서 한 요소의 변경은, 만일 인터페이스에 대한 사양을 가지고 있다면 이웃하는 모듈들에 대하여 어떠한 영향도 끼치지 않을 것이다. 이러한 사실은 그 시스템의 추가, 유지 및 변경시 시간 및 가격 절감의 효과가 있다.
- 서로 다른 응용 프로그램들을 가진 인터페이스의 이용, 혹은 서로 다른 인터페이스를 가진 같은 응용 프로그램의 이용은 시장성이 넓다.
- 대부분의 사용자 인터페이스 소프트웨어들은 응용 프로그램 소프트웨어들과 혼합되어 있어서 각각의 분리된 구조를 전개하기 어렵다. 분리되어서 정의된 사용자 인터페이스의 이용은 응용 프로그램과 사용자 인터페이스가 상호간에 비교적 독립적으로 개발되도록 허용한다.

인터페이스를 응용 프로그램으로 부터 완전히 분리하여 상호간의 명확한 경계와 잘 정의된 인터페이스를 가지는 것이. 여러가지 이점이 있으나, 현재는 불가능하다. 응용 프로그램 인터페이스 문제는 이용 가

능한 하드웨어와 시스템 소프트웨어에 의하여 제한을 받는다. 따라서 사용자 인터페이스 연구의 결과 및 결론들은 응용 프로그램 설계자들 뿐만 아니라 시스템 설계자들과도 밀접한 관계가 있다. 또한 사용자 인터페이스들의 개발은 응용 프로그램들과 시스템들의 설계방법에 중대한 영향을 끼칠 것이다.

(나) GUI 소프트웨어

사용자 인터페이스는 단순한 키보드 입력뿐만 아니라 각종 입력장비를 이용하여 단순한 텍스트 외에도 음성, 화상 등의 입출력까지 그 범위는 매우 광범위하다. 그 중에서 주요 쟁점이 되고 있는 것이 화면상에 표시하는 그래픽 사용자 인터페이스 (Graphical User Interface)이다. GUI는 여러 그래픽 기능을 구사하여 사용자의 조작환경을 알기 쉬운 형태로 제공한 것이다.

GUI의 개발은 1970년대에 Xerox PARC사에서 개발한 Smalltalk 시스템에서 시작되었으며 (Smalltalk는 이후 Smalltalk-80으로 발전하였고 객체 지향 프로그래밍의 주요 기법들을 제공해 주었다) 현재 GUI의 기본 기념들인 윈도우 (Windows), 아이콘 (Icon), 메뉴 (Menu), 스크롤 바 (Scroll Bar), 다이얼로그 박스 (Dialog Box) 등과 같은 특징들이 도입되었다.

GUI들은 몇가지 예외적인 제품들이 있으나 대부분의 경우 윈도우 시스템, 이미지 모델, 응용 프로그램 인터페이스 (API: Application Program Interface)로 구성되어 있다.

오늘날의 GUI들은 많은 다양성을 가지고 있으나 GUI 제품이라고 하여 반드시 이러한 모든 특성을 갖춘 것은 아니다. 예를 들어 몇몇 GUI들은 아이콘을 사용하지 않으며, 아이콘을 사용하는 제품에서

도 아이콘은 선택적이거나 가끔 쓰일 뿐이다. 또한 마우스를 사용하는 제품이 있는 반면, 몇몇 GUI 들은 사용자에게 키보드로 마우스의 역할을 대신할 수 있도록 되어 있다.

(3) 다중매체 데이터베이스 시스템

다중매체 데이터베이스 관리 시스템은 다중매체 데이터베이스의 구축을 목적으로 사용하는 소프트웨어로서 기존의 정형화된 데이터는 물론 비정형화된 데이터의 처리도 가능하여야 한다.

즉 다중매체 데이터는 텍스트와 정형화된 데이터 뿐만 아니라 그래픽, 오디오, 비디오 등과 같이 특정 데이터 타입으로 일률적으로 규정할 수 없는 방대한 양의 데이터로 구성되어 있으므로 다중매체 데이터베이스 관리 시스템은 이러한 데이터를 효과적으로 관리할 수 있어야 한다.

다중매체 데이터의 화상 정보는 텍스트 및 수치 데이터에는 없는 특징을 갖는 유용한 정보이므로 사용자는 다음과 같은 요구를 만족시키는 데이터베이스 관리 시스템을 필요로 한다.

- 텍스트 데이터 뿐만 아니라 시각적으로 전체 정보를 한눈에 쉽게 파악할 수 있도록 그래프, 도면, 사진 등의 데이터를 처리할 수 있어야 한다.
- 사용자가 화면을 보면서 필요한 정보를 간단하고 빠르게 검색하여 정확하고 해상도 높은 정보를 얻을 수 있어야 한다.
- 검색한 결과를 단순히 보는 것 뿐만 아니라 편집, 가공하여 여러 분야에 이용할 수 있어야 한다.
- 사용자가 원하는 정보가 데이터베이스에 구축되면 자동적으로 그 정보를 획득할 수 있어야 한다.

기존의 데이터베이스 관리 시스템은 제한적인 형태의 데이터만 관리할 수 있으며 사용자가 정의할 수 있는 데이터 형식이 제한적이며 확장성이 부족하고 짧은 트랜잭션만 지원이 가능하다는 등의 단점이 있다.

이러한 단점들을 개선하고 새로운 다중매체 데이터들을 다양한 사용자의 요구에 맞게 처리할 수 있는 데이터베이스를 구축하기 위해서 기존의 데이터베이스 관리 시스템을 기초로 하여 더욱 발전된 데이터베이스 관리 시스템 개발에 대한 연구가 진행 중이다.

다중매체 데이터베이스를 구축하기 위한 미래의 데이터베이스 관리 시스템을 예상하여 정리하면 다음과 같다.

- 다중매체 데이터베이스 관리 시스템 : 기존의 데이터베이스 관리 시스템에서 처리하지 못하는 그래픽, 오디오, 비디오 등과 같은 새로운 데이터 형식들과 매체를 지원한다.
 - 다양한 매체의 통합제어에 의한 편리한 사용자 인터페이스를 제공
 - BLOB(Binary Large Objects) 데이터 형식 사용
 - 매체 포맷에 대한 표준화 작업의 진행
- 분산 데이터베이스 관리 시스템 : 지역적으로 분산된 데이터베이스들에 대한 통합적인 사용을 가능하게 한다.
 - 신뢰성 향상 (Improved Reliability)
 - 응답시간 개선 (Improved Response Time)
 - 일의 분배 (Load Distribution)
 - 책임의 지역화 (Locality of Responsibility)
 - 모듈별 확장 (Modular Expansion)

- 객체 지향형 데이터베이스 관리 시스템 : 객체 지향형 기술과 데이터베이스 관리 시스템의 기술 결합에 의해 생산성을 향상시키고 데이터 모델을 확장한다.

(4) 다중매체 통신 시스템

다중매체 통신 시스템은 기존의 텍스트 통신에 그래픽, 오디오, 비디오 등의 다양한 매체를 포함하여 정보를 교환할 수 있는 시스템으로서 현재 활발히 연구가 진행되고 있다.

다중매체 통신 시스템은 처리하여야 할 오디오 데이터가 서로 다른 통신 대역폭을 필요로 하기 때문에 다양한 대역폭을 제공하여야 하며 방대한 양의 비디오 데이터를 전송 지연 없이 신속하게 전달하기 위해서는 고속 통신망이 필요하다.

기존의 통신망은 공중 통신망과 사설 통신망으로 구분할 수 있는데 공중 통신망 중 패킷 교환망 (PSDN)은 Fast Packet을 사용하면 음성 위주의 다중매체 통신 서비스가 가능하다. 사설 교환 통신망의 LAN(Local Area Network), MAN(Metropolitan Area Network)은 지역적 한계를 극복하고 고속통신을 위해 광케이블을 이용한 FDD-I, FDD-II, DQDB 등의 고속통신 기술을 이용하면 다중매체 통신이 가능하다.

현재 연구가 진행중인 다중매체 통신망으로는 AHENA, ANDREW, IMES, HECTOR, ISI, AGORA 등이 있으며 CCITT, IEEE 등에서는 다중매체 통신망에 대한 표준화가 진행중이다.

나. 시스템 개발을 위한 기능별 소프트웨어

다중매체를 구성하는 여러 자원들은 컴퓨터 내부적으로 소프트웨어를 이용하여 작성되기도 하지만 외부로부터 입력되는 것이 보통이다.

즉, 비디오 카메라, VCR로 부터는 움직이는 영상 데이터가 입력되고, 오디오 시스템으로 부터는 음악이 입력되며 마이크를 통하여 사람의 음성어, 스캐너를 통하여 사진이나 그림 등이 입력된다. 이렇게 다양한 외부의 자원들은 고유한 특성으로 인하여 각기 다른 방식으로 입력되어야 하며 이들 자원이 컴퓨터 내부 데이터로 변형이 되어 응용 소프트웨어의 재료가 된다. 다중매체 응용 소프트웨어의 구성요소가 되는 이들 자원을 입력, 편집, 재생하기 위해서는 하드웨어와 그 하드웨어를 제어하는 드라이버 소프트웨어 및 이들 데이터를 1차적으로 가공하는 소프트웨어가 필요하다.

여기서는 오디오, 그래픽, 비디오 처리 지원 소프트웨어로 구분하고 대표적인 소프트웨어를 소개한다.

(1) 오디오 처리 지원 소프트웨어

오디오를 합성하는 소프트웨어는 문자를 인식하여 오디오로 재생할 수 있는데 이 소프트웨어를 이용하면 음성처리를 위한 별도의 음성을 입력시킬 필요가 없기 때문에 대용량의 음성 데이터가 차지하는 용량이 줄어들며 또한 일일이 입력시킬 필요없이 자유롭게 음성을 합성할 수 있다. 이 소프트웨어는 보통 다른 소프트웨어와 결합할 수 있는데, 예를 들면 사람의 입의 움직임을 표현하는 애니메이션 소프트웨어와 결합하면 컴퓨터에서 말하는 사람을 만들 수 있다. 또한 단순히 텍스트로만 표현된 각종 메시지가 이 소프트웨어를 이용하면 훨씬 생동감 있게 메시지를 전달할 수 있다.

또한 오디오를 외부로부터 디지털 데이터로 바꾸어 입력, 저장, 편집이 가능한 소프트웨어가 있는데 별도의 하드웨어가 필요하다. 이 하드웨어를 통하여 입력되는 음성어나 외부 CD 플레이어 등의 오디오

시스템으로 부터 입력되는 사운드는 1차적으로 가공될 수 있다. 즉 속도를 빠르게 하거나 별도로 입력된 오디오를 합성, 편집하는 것이 가능하다. 이들 소프트웨어로 만들어진 오디오 자원은 단순한 시각적 효과를 능가하며 생동감을 더해준다.

컴퓨터와 외부 악기를 연결하는 MIDI(Musical Instrument Digital Interface) 기기 및 그 기기를 제어하는 소프트웨어를 사용하면 수십가지 악기를 동시에 연주할 수 있으며 악보로 표현 불가능한 음색 및 악기로는 불가능한 것도 연주할 수 있다.

오디오 데이터를 처리하는 소프트웨어 기능은 다음과 같다.

- 모든 오디오 사운드의 입력과 출력이 가능하다.

모든 소리를 디지털 데이터로 변형하여 저장매체에 입력할 수 있으며 입력된 데이터는 스피커를 통해 바로 들을 수 있다.

- 사용자는 샘플하는 음향의 속도 및 스테레오 샘플링 레이트(Sampling Rate)를 임의로 지정할 수 있다.

- 사운드에 각종 효과를 첨가할 수 있다.

- 편집할 사운드의 음파를 그림으로 나타내어 눈으로 직접 보고 필요없는 소리의 파형을 절단하여 원하는 부분에 추가하는 등의 오디오에 대한 조작이 가능하다.

<표 2-5> 오디오 처리 지원 소프트웨어

| 소프트웨어 | 업 체 | Platform | 특 징 |
|----------|-----|----------|---------------------|
| Cakewake | | PC | • MIDI 소프트웨어 |
| Ease | | PC | • 여러 종류의 악기로 동시에 연주 |
| Personal | | PC | 가능 |

(다음 페이지에 계속)

(< 표 2-5 > 계속)

| 소프트웨어 | 업 체 | Platform | 특 징 |
|--------------------------|----------|-----------|--|
| - Composer Texture II | | PC | <ul style="list-style-type: none"> • 자유롭게 한 부분을 다른 부분으로 이동시켜 작곡 및 연주 가능 • 악보로 표현하기 어려운 음색의 연주 가능 |
| MediaTracks | Farallon | Macintosh | <ul style="list-style-type: none"> • MacRecorder, Sound Edit 등에서 만들어진 음성, 음악 화일 이용 가능 • 화상, 이미지에 부가적 설명, 소리 첨가 가능 • Interactive User Interface 제공 |
| MacinTalk | Apple | Macintosh | <ul style="list-style-type: none"> • interFACE 등과 같이 Apple에서 사용되는 프로그램에서 임의로 Macintosh 내장 마이크를 통해 음성합성 가능 |
| Audio Master II | Oxxi | PC | <ul style="list-style-type: none"> • 디지털 사운드를 샘플링 하고 편집 가능 |

(2) 그래픽 처리 지원 소프트웨어

그래픽 소프트웨어는 2차원 그래픽 소프트웨어와 3차원 그래픽 소프트웨어로 구분할 수 있다.

좁은 의미의 컴퓨터 그래픽으로 호칭되는 2차원 그래픽 소프트웨어는 평면으로 출력된 이미지 및 2차원 상의 작업 프로세스를 총칭

하며 텍스트타일 디자인, 타일 디자인, 회화, 만화영화 제작, PC 슬라이드 프리젠테이션, 광고분야 등에 활용된다. 그래픽 소프트웨어는 흑백의 간단한 페인팅 소프트웨어로부터 수백만 종류의 색상을 표현하고 각종 특수 효과를 낼 수 있는 고기능의 소프트웨어까지 다양하다. 이 소프트웨어를 이용하면 새로운 그래픽 이미지를 창조하거나 컴퓨터로 입력된 비디오 데이터, 스캐너로 입력된 이미지, 흑백 페인팅 소프트웨어로 작성된 그래픽 등의 자유로운 편집이 가능하다.

3 차원 그래픽 소프트웨어는 입체적 영상에 실물과 같은 효과를 주어 실세계에서는 불가능한 물체를 만들어 내거나 실물과 같은 입체적 영상을 만들어 큰 효과를 낼 수 있다. 이렇게 모델링한 3 차원 입체영상은 위치, 조명에 따라 다양한 표현이 가능하며 이를 이용하여 애니메이션을 위한 그래픽 자원을 생성할 수 있다. 즉 움직이는 비디오 영상을 정지화면으로 몇개의 프레임을 잡아내듯이 시각과 위치에 따라 만들어지는 형상을 시각의 변화, 연속적인 위치의 변화 등에 따른 형상을 생성하므로써 현실세계에서는 불가능한 물체를 만들 수 있다. 2 차원 그래픽 소프트웨어 보다 더욱 생동감을 필요로 하는 상업광고, 영화제작, 디자인, 영상 프리젠테이션, 모의 비행실험, 시뮬레이션 등 광범위한 분야에 사용된다.

< 표 2-6 > 그래픽 처리 지원 소프트웨어

| 소프트웨어 | 업 체 | Platform | 특 징 |
|--------|-----------|----------|--|
| Lumena | Time Arts | PC | <ul style="list-style-type: none"> • Professional Art, Design 용의 2D Painting 소프트웨어 • Draw, Tools, XFORM, Block, Input, Output, Vector, Layout 그룹으로 |

(다음 페이지에 계속)

(< 표 2-6 > 계속)

| 소프트웨어 | 업 체 | Platform | 특 징 |
|---------------|-------------------|----------|---|
| | | | 메뉴 구성 |
| Crystal 3D | Time Arts | PC | <ul style="list-style-type: none"> • 3D Solid Modeling, 애니메이션 - 용이하고 완벽한 Modeling 과 애니메이션 가능 • Material Mapping 가능 |
| Tips | AT&T | PC | <ul style="list-style-type: none"> • 2D Painting 소프트웨어 • 아이콘 메뉴, Tint 등의 특수효과, 편집 가능 |
| Topas | AT&T GSL | PC | <ul style="list-style-type: none"> • Topas Supershade, Topas Modeler, Topas ProModeler, Topas Anima- tor 등으로 구성 • Rendering, Modeling, 애니메이션 등이 강력한 통합 소프트웨어 • 방송등에 사용 |
| Softimage | Softimage Inc. | W/S | <ul style="list-style-type: none"> • Model, Motion, Matter, Tools 등으로 구성 • Targa, Alias, Wavefront 등에서 이미지 변환 가능 • 광고, 애니메이션, 방송, 자동차, 건축 디자인, 비디오 제작에 사용 |
| Links | Links | W/S | <ul style="list-style-type: none"> • Animation Tool, Surface Editor, L/Manage, Material Library 등 으로 구성 사용 |

(다음 페이지에 계속)

(< 표 2-6 > 계속)

| 소프트웨어 | 업 체 | Platform | 특 징 |
|---------------------|-------------------|-----------|--|
| | | | <ul style="list-style-type: none"> • META Editor - 인체의 유연한 표현이 가능 • Material Library - 다양한 재질감 표현 가능 |
| Alias/2 | Alias Research | W/S | <ul style="list-style-type: none"> • Flow Data 형식으로 작업 • 5개의 통합된 모듈로 구성 • Film Animation, 산업 디자인, 과학용 시뮬레이션 등에 사용 |
| Advanced Visualizer | Wavefront | W/S | <ul style="list-style-type: none"> • CG 애니메이션, 제품 디자인, Engineering 시뮬레이션, 자동차 디자인, 건축설계 등에 사용 • Model Design, Animation Preview, Material Edit, Rendering 과정 |
| Pixel Paint | Apple | Macintosh | <ul style="list-style-type: none"> • 그래픽, 텍스트, 이미지, 사진 등을 다양한 색상으로 가공, 처리 가능 |
| Delux Paint III | Electronic | PC | <ul style="list-style-type: none"> • 텍스트, 이미지 등을 자유롭게 편집 가능 • 거의 모든 그래픽 카드 지원 • 그래픽적인 기능과 다양한 폰트 제공 |
| Rio | AT&T | PC | <ul style="list-style-type: none"> • 편집용 2D 그래픽 소프트웨어 • 상업용 프리젠테이션 • 풍부한 색상, 이미지 편집 용이 |

(3) 비디오 처리 지원 소프트웨어

비디오 처리 지원 소프트웨어는 움직이는 비디오 데이터를 읽어들이 비디오 영상위에 텍스트를 써넣거나 비디오 영상과 그래픽 등을 결합할 수 있으며 또한 비디오 영상의 밝기를 조절하거나 이미지 크기를 축소, 확대하고 윈도우를 만들어 위치를 변경하거나 색상을 변형시키는 등의 여러가지 변화를 줄 수 있으며 원하는 비디오 영상의 속도를 조절할 수도 있다.

이러한 소프트웨어는 만화 제작, 광고, 각종 시뮬레이션, 교육 분야등 여러 분야에 널리 활용할 수 있다.

< 표 2- 7 > 비디오 처리 지원 소프트웨어

| 소프트웨어 | 업 체 | Platform | 특 징 |
|---------------------------------|------------------------|----------|--|
| PhotoMac Edit Video Quill | | PC | • Color Capture 2.0으로 읽어들이 영상을 편집 |
| Diaquest | | PC | • 비디오 이미지를 자유로이 입력 • 비디오 이미지의 편집, 디지털이징 가능 |
| Quicksilver Video/Disk | Sundance Technology | PC | • 비디오 이미지를 자유로이 입력 • 비디오 디스크 플레이어 가능 - VTR, VCR과 접속 가능 |
| AVC | IBM | PC | • 음성과 영상 포착 가능 • 비디오 이미지 편집 • 특수효과 기능 |

다. 응용 시스템 개발을 위한 통합 소프트웨어

통합 소프트웨어는 크게 저작 도구, 프리젠테이션 도구, 애니메이션 도구 등으로 크게 구분할 수 있는데 각 도구들의 특징 및 소프트웨어들을 <표 2-8>로 정리하였다.

<표 2-8>에서 예를 든 소프트웨어들은 모두가 다양한 매체를 이용하여 사용자들에게 정보를 전달할 수 있는 기능을 갖추고 있기 때문에 명확하게 구분하기가 힘든데 여기에서는 VideoLogic사가 구분한 것을 인용하였다.

(1) 저작 도구

저작 도구는 처음에는 컴퓨터 지원교육 (CAI : Computer Aided Instruction) 시스템 혹은 컴퓨터 지원훈련 (CBT : Computer Based Training) 시스템 개발을 위해 사용되었는데 이 저작 도구를 사용하면 프로그래밍에 대한 전문적인 지식이 없는 일반 교육자도 쉽게 코스웨어 (Courseware) 를 개발할 수 있어서 이제까지 비교육 전문가가 코스웨어를 개발할 때 발생하는 단점을 극복할 수 있다.

저작 도구를 사용하여 코스웨어를 개발할 경우 코스웨어는 유기적으로 결합된 그래픽, 오디오, 비디오, 텍스트 등 다양한 다중매체 자원을 포함하여 이제까지의 텍스트와 간단한 그래픽 위주의 코스웨어에서는 불가능 하였던 교육효과를 얻을 수 있다.

또한 인간의 사고방식대로 자유롭게 정보의 검색이 가능하도록 서로 데이터를 연결하여 학습자의 필요나 관심에 의해 정보를 추적, 조화할 수 있는 비순차적 텍스트 원리인 하이퍼텍스트 개념을 도입하여 기존의 학습방식인 교과서나 교과과정을 중심으로 한 순차적 전개방식이 아닌 복합적인 학습을 유도하여 학습자 나름대로 학습방식

이나 정보의 필요에 의해 스스로 학습할 수 있도록 유도한다.

현재는 코스웨어 제작에 국한하지 않고 텍스트, 그래픽, 오디오, 비디오 등의 다양한 매체가 결합된 각종 소프트웨어를 개발하는 도구로서 그 개념이 확장되어 널리 사용되고 있다.

대부분의 저작 도구는 디자인과 개발 환경을 동시에 지원하며 다양한 형태의 데이터를 결합, 통합할 수 있으며 비디오 디스크, CD-ROM 및 각종 외부장비와의 입·출력도 가능하다.

(2) 프리젠테이션 도구

프리젠테이션 시스템은 표현되는 내용이 인간의 주의를 최대한 집중시킬 수 있도록 구성되어야 하는데 이를 위해서는 다중매체를 이용하면 가장 효과적으로 구성할 수 있다

따라서 프리젠테이션 시스템 개발을 위한 도구는 다중매체 자원을 효과적으로 표현할 수 있는 다양한 기능을 가져야 하는데 자체적으로 그래픽 기능을 갖기도 하고 이미 입력되었거나 음성합성 소프트웨어를 통해 만들어진 사운드등을 쉽게 이용할 수 있어야 하며 애니메이션을 처리할 수 있어야 한다 특히 CD-ROM이나 레이저 디스크 등의 외부장비를 자유로이 조정하여 보다 생동감 있게 표현하며 그 밖의 화면을 다채롭게 표현할 수 있는 특수효과도 필요하다.

프리젠테이션 도구는 프로그래밍이 가능한 언어를 제공하여 사용자와 컴퓨터와의 상호작용성을 최대한 보장함으로써 사용자가 자유롭게 프리젠테이션을 진행시킬 수 있도록 진행 속도를 조절할 수 있어야 한다. 또한 각 자원들을 일관되게 관리할 수 있는 관리방식이 필요한데 각 자원이 별개의 관리방식을 가진다면 도구를 사용하여 응용 소프트웨어를 개발하기에는 곤란하기 때문이다.

< 표 2-8 > 통합 소프트웨어

| 소프트웨어 | 업 체 | Platform | 특 징 | 용 도 |
|---------------------------|-----------------------------------|-----------------|--|---------------------|
| Aldus Persuasion 2.0 | Aldus Corp | Macintosh | <ul style="list-style-type: none"> · DeskTop 프리젠테이션 소프트웨어 · PICT, PICT2, EPS 그래픽 화일 이용 가능 · Word Processing, Charting Tool, Mini-Spreadsheet, Drawing Tool 등을 제공 | 프리젠테이션 |
| Ask*Me 2000 | Innovative Communications Systems | PC | <ul style="list-style-type: none"> · Authoring Package Run time Display Package 로 구성 · 그림, 그래픽, 애니메이션, 비디오 등을 처리 · Stratos : 매체를 통합하고 응용 프로그램을 생성하는데 사용 되는 언어 | 저작도구 |
| Authorware Professional | Authorware | PC Macintosh | <ul style="list-style-type: none"> · 텍스트, 그래픽, 애니메이션, 사운드, 비디오 등을 이용한 정보와 지식 표현 · Data Source 연결 조정 · 시뮬레이션 가능 · 아이콘을 기초로 한 개발 환경 | 애니메이션, 프리젠테이션, 저작도구 |
| Autodesk Animator | Autodesk | PC | <ul style="list-style-type: none"> · 그래픽, 텍스트, 사진, 고품질의 비디오 애니메이션 가능 · Business, Storyboard, Training, Education, Home Video 제작 등에 사용 | 애니메이션, 프리젠테이션 |
| BigTime TV | HyperPro | Macintosh | <ul style="list-style-type: none"> · Video Digitizer, Overlay Board를 통해 비디오 입력 처리 · XCMD : Video Window 와 Macintosh Window 통합 · Video Maker : 일련의 비디오를 Select, Edit, Save, Retrieve · I/F Constructor : Multi-Window 생성, 전환면 인터페이스 제공 | 저작도구 |
| CAL Toolbox & Autographic | PenguinSoft | PC | <ul style="list-style-type: none"> · CAL Toolbox : Video Logio 사의 IVA 카드로 레이저 디스크를 제어 · Autographic : Drawing Package | 저작도구 |
| Course Builder 3 | TeleRobotics International | Macintosh | <ul style="list-style-type: none"> · Macintosh를 위한 객체 지향형 코스웨어 저작도구 환경 제공 · 코스웨어 개발 도구 · Macintosh 소프트웨어와 호환성 | 저작도구 |
| Crystal | Intelligent Environments | PC | <ul style="list-style-type: none"> · Rule-based Programming 을 하기 위한 전문가 시스템 · 프로그램을 제어하고 다른 시스템 과 소프트웨어를 통합 | 저작도구 |
| Grasp 3.5 | Paul Mace Software | PC | <ul style="list-style-type: none"> · 그래픽 프로그래밍 환경 제공 - Palette, Movement, Timing, Sound, User Interaction을 포함한 프리젠테이션 요소를 제어 · 명령 인터프리터 내장 - 프로그램 개발, 디버깅속도 향상 · Grasp : 애니메이션 프로그래밍 언어 · Pictor : Painting 프로그램 · Arttools : 애니메이션을 위한 특수 효과 Generator | 애니메이션, 프리젠테이션 |

(다음 페이지에 계속)

(< 표 2-8 > 계속)

| 소프트웨어 | 업 체 | Platform | 특 징 | 용 도 |
|-----------------------------|-----------------------------|-----------------|---|---------------|
| Guide 2.0 | Office Workstations Ltd | PC Macintosh | <ul style="list-style-type: none"> · 비디오 디스크, CD-ROM 지원 · Hypertext Product - Document 개발 및 링크 · MacDraw 타입 그래픽 생성 기능 | 프리젠테이션, 저작도구 |
| Guide 3.0 | Office Workstations Ltd | PC | <ul style="list-style-type: none"> · Interactive Hypermedia-based 소프트웨어 시스템 · Hypermedia Linking Tool · 향상된 Document Formatting · 향상된 Link Maintenance | 프리젠테이션 저작도구 |
| HyperCard | Apple Computer | Macintosh | <ul style="list-style-type: none"> · 다양한 디자인 환경 제공 - 직관적, 시각적인 사용자 인터페이스 · CD-ROM, 비디오 디스크, 사운드 라이브러리 애니메이션에의 액세스 · HyperCard Stack - 텍스트, 그래픽, 그림, 사운드, 음성 통합 가능 · Macromind Director Animation 통합 가능 · 비디오 디스크 플레이어 액세스 - Realtime 비디오 제공 | 프리젠테이션, 저작도구 |
| Hyperdoc | Hyperdoc | PC | <ul style="list-style-type: none"> · 응용 프로그램을 Assemble하는 소프트웨어 제공 · 텍스트, 이미지, 숫자, 비디오, 사운드 등의 처리 가능 · 외부 소프트웨어와 시스템의 액세스 가능 | 프리젠테이션, 저작도구 |
| IconAuthor | AimTech | PC | <ul style="list-style-type: none"> · MS-Window, Mouse, Pull Down Menu 제공 · 다양한 Platform 제공 - Dynamic Interactive Application 개발 가능 | 저작도구 |
| Instant Replay Professional | Nostradamus | PC | <ul style="list-style-type: none"> · 그래픽, 텍스트, 애니메이션, 비디오, 오디오, DOS 프로그램 통합 · Dynamic Self Running · User Controlled 다중매체 소프트웨어 | 애니메이션, 프리젠테이션 |
| MacroMind Director 2.0 | MacroMind | Macintosh | <ul style="list-style-type: none"> · 다중매체, 애니메이션 생성 도구 · 사진, 그래픽, 텍스트, PICT Image Scrapbook Image, Color Palette 등의 다중매체 데이터베이스 제공 · 애니메이션에서 Play Back, Speed-Up 가능 | 애니메이션, 프리젠테이션 |
| MacroMind MediaMaker | MacroMind | Macintosh | <ul style="list-style-type: none"> · Visual-based 편집 시스템 - 정지화상, 동적화상과 사운드와 동기 유지 · 비디오 디스크, 비디오 테이프 플레이어, CD, 오디오 등을 제어 · Macintosh 그래픽 통합 가능 | 프리젠테이션 |
| MentorII | Mentor Interactive Training | PC | <ul style="list-style-type: none"> · CBT에 사용 (특히 Financial 부분) · 비디오 테이프, 디스크 플레이어, 오디오, 슬라이드 시스템 지원 · Version 4.0 - Menu Editor, Speech Digitizing On-line Editing, Window Emulation, C/W Distribution 기능 | 저작도구 |

(다음 페이지에 계속)

(< 표 2-8 > 계속)

| 소프트웨어 | 업 체 | Platform | 특 징 | 용 도 |
|--------------------------|------------------------------------|-----------------|--|---------------------|
| Microsoft PowerPoint 2.0 | Microsoft | PC Macintosh | · 자유로운 스크린 디자인이 가능한 프리젠테이션 도구 - 텍스트, 그래프, Diagram, Chart, Table 등이 혼합 가능 · Overhead, 35mm 슬라이드 사용 | 프리젠테이션 |
| ProCal | Epic Interactive Media Company | PC | · CAL(Computer-Assisted Learning), Interactive Video 를 위한 환경 · 80개의 명령으로 구성된 언어 - Control, Branching, Input, Matching, Calculation, Display Graphic, Video 등의 기능 | 저작도구 |
| Quest 3.0 | Allen Communication | PC | · 프레임 단위로 일련의 Lesson 학습 가능 · 정답판정, 그래픽, 워드 프로세서, 색상 배합, 애니메이션, Flow Diagram 등의 다양한 기능 · 워드 프로세싱 강화 | 애니메이션, 저작도구 |
| SAM | The technologies Application Group | PC | · Prompt-Driven 저작도구 시스템 · 다양한 기능 제공 - 비디오 윈도우, 비디오 디스크, 테이프 플레이어 제어, 텍스트와 그래픽의 Overlay 및 동기 | 저작도구 |
| Show Partner F/X | Brightbill-Roberts and Co | PC | · IBM PC, PS/2 에서 Desktop 프리젠테이션을 위한 강력한 도구 · MS-DOS, MS-Window 소프트웨어의 텍스트, 그래픽 화면을 Capturing 한 후 스크린상에 출력 가능 | 프리젠테이션 |
| Spinnaker Plus 2.0 | Spinnaker Software | PC Macintosh | · 객체 지향형 프로그래밍 언어제공 · Macintosh와 IBM PC 응용 프로그램 간의 인터페이스 제공 | 프리젠테이션, 저작도구 |
| Storyboard Plus | IBM | PC | · 프리젠테이션을 위한 도구 · Picture Maker, Picture Taker, Story Editor, Story Teller, Text Maker로 구성 | 프리젠테이션 |
| SuperCard | Silicon Beach Software | Macintosh | · Macintosh HyperCard 와 호환성 - 색상, Pull Down Menu, Dialog Box, Full Screen 지원, Window Draw, Paint 그래픽 환경 제공 | 저작도구 |
| TenCore | Computer Teaching Cor. | PC | · 고품질의 IV, CBT 개발환경을 제공 · 그래픽, Character Editor, Source Code Editor, Disk Manager, File Manager 기능 포함 | 저작도구 |
| ToolBook 1.0 | Asymetrix Corp. | PC | · 응용 프로그램 개발 도구 - Color Drawing, Hypernavigation, 애니메이션, 텍스트 포매팅, 프로그래밍 언어 제공 · 객체 지향형 프로그래밍 가능 · Hypertext 사용 가능 | 애니메이션, 프리젠테이션, 저작도구 |

(다음 페이지에 계속)

(< 표 2-8 > 계속)

| 소프트웨어 | 업 체 | Platform | 특 징 | 용 도 |
|-----------------------------|------------------------------|-----------------|---|-------------|
| V.C.N. Concorde Version 3.0 | V.C.N. International | PC | · 통합 그래픽 Package · 그래픽, 텍스트, 이미지, 음악, 애니메이션, 특수 효과 등의 결합 가능 | 애니메이션, 저작도구 |
| Video Builder | TeleRobotics International | Macintosh | · 비디오 디스크를 제어하는 다중 매체 생성 도구 - Interactive Video 제공 | 저작도구 |
| Videodisk Writer | Whitney Educational Services | PC Macintosh | · HyperCard를 기초로 한 비디오 디스크 저작도구 · On-line Help 기능 · Visual Frame 선택, Text Overlay 등이 가능 · Pioneer LD-V 42000 지원 가능 | 저작도구 |

(3) 애니메이션 도구

컴퓨터 애니메이션은 3 차원 공간에 시간축을 더한 것으로 4 차원 시간예술로 지칭된다.

애니메이션 소프트웨어와 그래픽 소프트웨어를 명확하게 구분하기가 힘든 경우가 있는데 이는 보통 애니메이션 소프트웨어가 그래픽 기능을 포함하고 있기 때문이다. 애니메이션 소프트웨어의 주된 기능은 얼마나 쉽게 애니메이션을 작성할 수 있는가에 달려 있는데, 예를 들면 어떤 물체의 움직이는 위치를 지정하면 그 움직임에 따라 지정된 수의 프레임을 자동적으로 생성되기 때문에 일일이 그래픽 소프트웨어로 프레임을 만들 필요가 없어 시간과 노력을 절약할 수 있다. 애니메이션 소프트웨어를 이용하여 만들어지는 그래픽은 비디오 영상으로 부터 받아들인 애니메이션과는 다른 독특한 효과를 연출한다. 애니메이션 소프트웨어는 만화영화 제작, 비디오 게임, 시뮬레이션 등에 사용된다.

4. 다중매체 시스템의 응용

다중매체 시스템의 응용분야는 무척 광범위하기 때문에 분야별로 구분하기가 힘들지만 크게 교육·훈련 분야, 광고 및 판매 분야, 홍보 및 프리젠테이션 분야, 시뮬레이션 분야, 오락 분야 등으로 크게 나눌 수 있다.

문자, 그래픽, 애니메이션, 오디오(음향, 음악, 음성), 비디오 등의 다양한 데이터를 사용하여 모든 시청각 정보를 통합하여 제공함으로써 엄청난 양의 정보를 전달할 수 있을 뿐만 아니라, 이들을 결합한 다중매체 시스템을 활용하면 기존의 매체를 활용할 때보다 주의력, 이해력, 기억력, 집중력 등을 증가시켜 보다 효과적으로 정보를 전달할 수 있다.

외국에서는 많은 다중매체 시스템 응용 소프트웨어가 개발되어 있으며, 이들 응용 소프트웨어는 독자적인 시스템으로 구축되어 있기도 하고, 각종 틀을 사용하여 개발되기도 한다.

< 표 2-9 >는 다중매체 시스템의 응용분야를 사용자별로 구분하여 기술한 것이다.

< 표 2-9 > 다중매체 시스템의 응용 분야

| 응용분야 이용범위 | 교육 훈련 | 광고 및 판매 | 홍보 및 프리젠테이션 | 시뮬레이션 |
|------------------------------------|----------------------------------|---------------------------|-----------------------|-------|
| 일반기업체, 공사, 자동차생산업체, 정보산업체, 섬유·의류업체 | 기업연수교육, 기계정비교육, H/W 정비교육, S/W 학습 | 대형·고급상품 판매, 자동차영업소, 디자인선택 | 홍보물제작, 브리핑, 회의, 신제품발표 | 충격실험 |

(다음 페이지에 계속)

(< 표 2-9 > 계속)

| 응용분야 이용범위 | 교육 훈련 | 광고 및 판매 | 홍 보 및 프리젠테이션 | 시뮬레이션 |
|--|--|--|-----------------------------|-------------------------------|
| 건설·설계업체, 유통업체, 광고 기획사, 의료기 관, 기타서비스 업체 (여행사, 변호사) | 지점·대리점요 원교육, 기술교 육 | 샘플설계, 조감 도, 아파트·콘 도분양 안내, 상품정보 안내, 부동산판매, 인 테리어선정, 여 행사 관광안내 | 시설안내 | 인체 반응 |
| 금융기관, 종교 단체, 연구기관, 스포츠기관, 교 육기관, 언론기 관 | 성경공부, 선수 교육훈련, 기술 교육, 음악 교 육, 과학실험교 육, 외국어교육, 입시학원, 미술 사교육, 기자연 수교육 | 금융 상품 안 내 및 이용 안내 | 행원교육, 회 의, 시청각 교육용 | |
| 정부 / 행정기관, 군 / 경찰, 정당, 협회, 공공장소, 예술관 | 군장교 교육 | 매 표관리 | 회의, 브리핑, 작전회의, 시 설물안내 | 도로교통상황, 전쟁발발, 전 투 시뮬레이션 |

가. 교육 훈련 분야

이제까지 개발된 CAI 및 CBT 목적의 많은 교육용 소프트웨어는 단순히 교과서나 문제집의 내용을 컴퓨터를 이용하여 보조교육 수단으로 활용하여 왔으나 다중매체 시스템을 통한 교육이나 훈련, 학습은 책이나 강의를 통해서 얻을 수 없는 청각적, 시각적 효과를 직접 컴퓨터와 대화형식으로 자세한 설명과 함께 제공해 줌으로써 학습자의 흥미를 유발시켜 훨씬 더 깊은 이해를 할 수 있다. 즉, 필요한 화면의 일부분을 확대하거나 축소할 수도 있어 복잡한 회로의

자세한 부분을 실제 크기보다 더 자세히 볼 수 있거나 큰 구조물의 레이아웃을 전체적으로 한 화면으로 축소하여 한번에 볼 수도 있다.

결국 다중매체 시스템을 이용한 학습용 소프트웨어는 문자, 숫자, 사진, 음성, 소리, 비디오 등의 합성을 통해 실험이나 실습 등의 직접 경험학습까지도 대체할 수 있으며 또한 학습자 개인의 능력에 따라 학습진도의 진행이 가능하므로 충실한 개인교사 역할을 할 수 있어서 학습의 개별화를 실현할 수 있다.

- 응용 분야

- 개인학습을 위한 교육용 프로그램
- 기업의 기계 정비 교육, 연수원 훈련 시스템
- 외국어 교육기관의 외국어 교육
- 컴퓨터 하드웨어 및 소프트웨어 이용 교육

- 적용 사례

- CTE North사 (미국): 사원들에게 전화케이블 설치하는 방법을 가르치는 프로그램을 개발하여 교육을 실시한 결과 교육효과가 3 - 4 배 높은 것으로 나타났다.
- 듀폰사 : 트럭 운전수 교육을 위한 비디오 시스템
- 미국심장협회 : 혈액에서 콜레스테롤이 어떻게 조절되는지를 보여주는 디지털 교육시스템을 의과대학에 설치
- MIT : 어학능력을 향상시키기 위한 목적으로 Direction Paris 라는 툴 개발

나. 광고 및 판매 분야

광고는 기업이 생산한 상품, 서비스, 아이디어 등을 신문, 잡지, 라

디오, 텔레비전 등의 여러 대중매체를 통하여 수요를 창출하여 기업의 이윤을 증가시키기 위한 수단으로서 대량판매, 대량소비를 유도하는 방향으로 발전하였다. 컴퓨터의 2차원, 3차원 그래픽을 기업 이미지 광고나 상품광고에 많이 사용하고 있는데 컴퓨터 그래픽은 현실 세계에서는 불가능한 장면의 창출이나 특수한 영상효과를 이용해서 소비자들의 관심을 끌고 있다.

다중매체 시스템을 이용한 광고는 단순히 그래픽으로만 이루어진 광고의 차원을 넘어 비디오 영상과 그래픽, 애니메이션의 결합이 가능하다. 이제까지는 비디오 영상과 그래픽을 결합하는데 고가의 장비가 필요하며 많은 경비가 소요되었지만 다중매체 시스템을 이용하면 훨씬 저렴한 비용으로 단시간에 제작이 가능해졌다.

다중매체 시스템을 이용하면 고객의 편의를 위해 구매상담 시스템, 판매자를 위한 판매상담 시스템도 가능하다.

상품구매의 경우 잠재적인 고객이 상품이나 서비스에 대해 세일즈맨의 중재나 간섭없이 상품을 구매할 수 있는 정보를 제공하는데 백화점이나 전시장을 직접 찾아가서 보는 것 뿐만 아니라 우편으로 보내온 디스크나 비디오텍스를 통해서 상품의 구매 결정을 내릴 수 있다. 또한 통신망을 이용하여 가정에서도 상품에 관한 정보를 얻고 또 구매결정을 내리고 대금도 은행에 자동이체가 가능하게 될 것이다.

상품판매의 경우 음성, 이미지, 애니메이션 등을 이용한 다중매체 시스템 프로그램을 통하여 고객들에게 빠르고 명확한 상품에 대한 정보를 전달함으로써 상품이나 서비스 판매에 소요되는 시간을 줄이고 또한 상품정보를 암기할 필요가 없게 될 것이다.

결국 다중매체 시스템을 이용하면 고객은 취향에 맞는 상품을 빠

른 시간내에 구입 가능하며 판매자는 고객관리는 물론 상품판매 면에서도 효율적이고 용이하게 할 수 있다.

- 응용 분야

- 자동차 회사의 구매상담 및 고객관리
- 백화점의 판매 상담
- 인테리어 디자인업체의 주문형 설계 지원
- 부동산 안내
- 여행사의 패키지 상품 안내

- 적용 분야

• 스틸케이스사 : 사무가구 설계 및 판매에 전자 카탈로그 이용
다. 홍보 및 프리젠테이션 분야

과거의 기업 홍보물은 주로 책자, 팜플렛, 슬라이드, 비디오 등으로 제작되었으나 일방적으로 정보를 전달하거나 일회용으로 낭비가 심한 단점이 있는 등 넓은 장소에서 많은 사람들에게 정보를 전달하는 프리젠테이션용으로는 부적절 하였다.

다중매체 시스템을 이용한 프리젠테이션은 모든 시청각 효과를 포함하는 프로그램으로 제작이 가능한 컴퓨터를 활용하기 때문에 환경 변화에 따른 요구에 적절하게 대응할 수 있으며 과거의 기업 홍보의 단점을 극복할 수 있게 되었다. 장소나 인원등의 환경에 따라 적절한 장비 (일반 모니터 혹은 빔 프로젝터 등)를 선택하여 사용할 수 있으며 프리젠테이션 과정이 고정되어 있는 것이 아니라 필요에 따라 변경이 가능하여 일방적인 정보전달이 아닌 대화식으로 정보전달이 가능하며 홍보용 비디오보다 단기간에 저렴한 비용으로 개발이 가능하다.

따라서 다중매체 시스템을 활용한 프리젠테이션은 일반 대기업이나 관공서의 세미나, 광고, 홍보물 제작, 방송국의 컴퓨터 그래픽스 제작, 문화시설의 안내용으로 많이 이용되어질 것으로 전망된다.

- 응용 분야

- 기업 및 관공서의 홍보실
- 미술관 또는 박물관 등의 문화기관의 시설물 안내
- 기업이나 단체의 시행사업이나 기업이념 홍보

라. 시뮬레이션 분야

시뮬레이션이라 함은 모형을 통하여 반복 실험하여 얻은 결과를 통해 실제현상을 예측하는 모의실험을 말하는데, 그 동안 불가능 했던 과학분야나 의학분야 등의 어떠한 시뮬레이션도 응용이 가능하다.

DNA 모형으로 부터 배치모형 등 어떠한 시각적인 모형도 3차원 그래픽을 통해 만들어낼 수 있다.

핵실험과 같이 실제로 실험할 수 없는 경우, 실제와 똑같은 환경을 설정해 준 뒤 복잡한 계산을 해서 실제 실험과 같은 결과를 예측할 수 있는 시뮬레이션도 가능하며, 또한 애니메이션을 이용하여 신체내부의 혈액의 흐름과 같이 눈으로 볼 수 없었던 것도 현실감 있게 볼 수 있는 등 흥미롭고 능동적으로 학습을 할 수 있게 해 준다.

- 응용 분야

- 혈액의 흐름이나 관절, 근육의 움직임 관찰
- 위험한 무기나 핵실험
- 자동차 충돌시 안전성 검사

마. 오락 분야

다중매체를 이용한 응용시스템이 현재까지 가장 많이 개발된 분야 중의 하나는 오락분야이다. 최근에는 정보(Information)와 오락(Entertainment)의 합성어인 인포테인먼트(Infortainment)와, 교육(Education)과 오락(Entertainment)이 합성된 에듀테인먼트(Edutainment)라는 용어가 사용되고 있다.

현재 많이 사용되고 있는 컴퓨터 게임은 주로 1차원이나 2차원의 그래픽과 간단한 효과음으로 이루어진 것이 대부분이지만 다중매체 시스템을 이용한 오락, 게임은 일반적으로 단순한 게임뿐만 아니라 게임을 즐기면서 새로운 정보를 얻을 수 있다. 예를 들면 포드사에서 개발한 포드 시뮬레이션Ⅱ는 흥미있게 운전하는 게임을 즐길 수 있을 뿐만 아니라 포드 자동차사의 전제품에 대한 자세한 정보도 얻을 수 있다.

제2절 다중매체 관련기술 분석

1. 관련 기술분야

다중매체 시스템과 관련된 기술은 그 분야가 매우 광범위하여 컴퓨터와 관련된 시스템 기술과 다중매체 응용 시스템을 개발하기 위한 응용 기술로 크게 나눌 수 있다. 응용 기술이라 함은 다중매체 제작을 위한 시나리오 저작, 편집과 비디오, 오디오와 관련된 방송제작 기술 및 이미지 처리와 관련된 컴퓨터 그래픽 등 종합적인 기술을 말하나 여기서는 컴퓨터와 관련된 시스템 기술에 한하여 분석하였다. (이하 다중매체 관련 기술이라 함)

다중매체 관련 기술은 컴퓨터의 시스템 환경 차원에서 볼 때 하드웨어 관련 기술, 소프트웨어 관련 기술(및 주변 기술)로 나눌 수 있다.

다시 하드웨어 기술은 기억장치, 저장매체, 마이크로 프로세서, 다중매체 처리 전용 프로세서 및 압축 / 복원장치, 통신 처리장치 등과 관련된 기술로 나눈다. 소프트웨어 기술은 오퍼레이팅 시스템, 데이터베이스를 포함한 다중매체 관리시스템, 사용자 인터페이스 지원 시스템 관련 기술과 같은 시스템 소프트웨어 기술과 다중매체 데이터 저작 및 처리를 위한 Tool Base 의 기본적인 응용 소프트웨어 기술로 구분된다. 다중매체 관련기술의 대표적인 기술 분야로는,

- 첫째, 대용량의 다중매체 데이터의 신속한 입출력 및 처리를 관장하는 마이크로 프로세서 기술
- 둘째, 대용량의 다중매체 데이터를 광 디스크와 같은 매체에 수록,

저장하는 저장매체 기술

셋째, 이미지를 포함한 풀모션 비디오와 음성을 포함한 오디오 데이터와 같은 대용량의 데이터를 디지털화하여 처리하는 압축/복원 기술

넷째, 비정형의 다양한 다중매체 데이터를 처리, 관리하는 다중매체 데이터베이스 관련 기술

다섯째, 대용량의 다양한 다중매체 데이터를 안정적으로 전송하는 다중매체 통신기술 등이다.

본 연구에서는 다중매체 데이터베이스 구축을 위한 관련 기술의 사전 분석 차원에서 다중매체 시스템의 핵심적 기술인 저장매체 기술, 압축/복원 기술, 다중매체 데이터베이스 기술, 사용자 인터페이스 기술, 다중매체 통신기술 및 다중매체 기술과 관련한 국제 표준화 동향등에 한하여 현존 기술을 중심으로 조사, 분석하였다.

2. 저장매체 기술

대용량의 다중매체 데이터를 적절히 보관 저장하는 저장매체로 외장형 하드디스크와 CD-ROM, WORM, Erasable 형태의 광 디스크등이 개발되어 사용되고 있으며 이들의 일반적인 특성 및 용도는 이미 밝힌 바와 같다.

여기서는 현재까지 개발 발표된 다중매체의 저장기술 중에서 아나로그 형태의 Interactive Video Disc (일명 Laser Disc)의 저장기술과 디지털 형태의 CD 매체와 관련된 저장기술을 중심으로 분석하

였다.

가. 비디오 디스크

1980년 부터 소개된 비디오 디스크 기술은 1983년 Panasonic사가 고품질의 아날로그 비디오와 디지털 오디오 데이터를 동시에 1회 기록할 수 있는 “Optical Memory Disc Recorder”를 발표함으로써 현실화 되었으나 Pioneer사가 그 기술을 더욱 향상시켜 Laser Disc 또는 Laservision이라 명칭하고 상품화하였다. 최근에는 Pioneer사와 Sony사가 기술의 판권을 갖고 가정용 및 산업용의 다양한 옵션의 레이저디스크 플레이어(Laser Disc Player)를 보급하고 있다.

레이저디스크의 정보를 랜덤 액세스하는 방법 및 컴퓨터와 인터페이스하여 Interactive하게 사용하는 정도에 따라 레벨 I, II, III, IV등으로 구분된다. 레벨 I, II는 컴퓨터와 인터페이스 없이 TV와 단독 연결하여 사용되는데 레벨 I은 주로 가정오락용으로 생동감 있는 오디오 및 비디오를 제공하며, 레벨 II에서는 LDP에 마이크로 프로세서가 내장되어 있어서 자체 메모리에 약간의 프로그래밍을 하여 기초적인 POS(Point Of Sale)에 사용하는 등 능동적으로 사용이 가능하다. 다중매체 시스템에서는 레벨 III 이상으로 컴퓨터와 인터페이스 할 수 있게 제어모듈이 내장된 산업용 LDP를 사용해야 한다.

비디오 디스크는 데이터를 인코딩(Encoding)하는 방법에 따라 CLV(Constant Linear Velocity)와 CAV(Constant Angular Velocity)의 2가지 기록형태를 갖는데 다중매체 시스템에서는 내장된 비디오를 프레임 단위로 랜덤 액세스 할 수 있는 CAV형태의 디스크를 사용해

해야 한다. < 표 2 - 10 > 은 레이저 디스크 와 VTR 의 특성 및 기능적, 기술적 특징을 비교하였다.

< 표 2 - 10 > Laser Disc 와 VTR 의 특성 비교표

| 구 분 | | Laser Disc (12"의 경우) | | 1/2" VTR |
|---------------------|-----------|----------------------|----------------|---------------|
| | | 표준형 (CAV) | 확장형 (CLV) | |
| 실행시간 / 면 (Side) | | 30분 | 60분 | 2,4,6 시간등 다양 |
| 정지화면수 / 면 | | 54,000 | 87,000 | - |
| RPM at Inner Track | | 1,800 | 1,800 | - |
| RPM at Outer Track | | 1,800 | 600 | - |
| Frames at Inner TRK | | 1 | 1 | - |
| Frames at Outer TRK | | 1 | 3 | - |
| 특수 실행 기능 | 저속 플레이 | 가 능 | 불 가 | 가능 (수명단축요인) |
| | 프레임별 정지 | 가 능 | 불 가 | 가능 (") |
| | 프레임별 이동 | 가 능 | 불 가 | 가능 (") |
| | 고속 전진, 후진 | 가 능 | 가 능 | 가능 (") |
| | 정지 기능 | 가 능 | 가 능 | 가능 (") |
| Pick Up 방식 | | 비접촉 (Optical) | 비접촉 (Optical) | 접촉 (테이프, 헤드) |
| 랜덤 액세스 | | 가 능 | 가 능 | 불가 |
| 프로그램 출력 | | 가 능 | 가 능 | 불가 |
| Pick Up 수명 | | 5,000 시간 이상 | 5,000 시간 이상 | - |
| 데이터 수명 | | 무 한 | 무 한 | 제한적 (테이프 질) |
| 오 디 오 | | 디지털 HiFi, 2 채널 | 디지털 HiFi, 2 채널 | 일반 HiFi, 2 채널 |

1. 상기 자료의 LD 및 VTR의 비디오는 NTSC 방식을 기준으로 표기했으며 PAL의 경우는 다소 다르나 본고에서는 생략하였다.

2. LD는 12"와 8" 크기의 2종류가 있으며, 8"(20cm) LD인 경우 실행시간은 CAV에서 14분, CLV는 20분이고 프레임수는 CAV에서 25,000 프레임이다.

LD와 LDP를 다중매체 시스템의 부분으로 구성하려면 데이터 형식 외에도 컴퓨터 시스템과의 인터페이스(RS 232C) 제공 여부, 지원 가능한 디지털 비디오 보드, 액세스 타임(2.5초 이내), 저작 도구(Authoring Tool 또는 Authoring System) 및 저장된 데이터의 프레임당 디렉토리를 담은 보조디스켓 소프트웨어의 제공 여부 등에 대해서도 고려할 필요가 있다.

나. CD를 이용한 저장기술

CD(Compact Disc)는 1982년 Philips사와 Sony사가 가정용의 고품위 오디오 매체로 개발한 디지털 오디오 형태의 디스크로 최대 70분, 2채널의 오디오를 제공한다. CD를 이용한 다른 저장매체와의 구분을 위해 CD-DA(CD-Digital Audio)라고 부른다. 최근에 저장기술이 발전함에 따라 CD에 오디오 데이터와 더불어 비디오 데이터를 저장하려는 많은 기술들이 발표되어 상업화되고 있다. 여기서는 CDV, CD-VI 및 CD-ROM XA에 대하여 기술하고, CD를 이용한 저장기술인 동시에 대표적인 압축/복원 기술인 CD-I(Compact Disc-Interactive) 및 DVI(Digital Video Interactive)는 차후에 분석 제시기로 한다.

(1) CDV (Compact Disc Video)

CDV 는 1986 년 Philips 사가 제안하여 1987 년에 상품화된 금색의 디스크로 비디오는 아날로그 신호로 기억되어 있다. 한장의 디스크속에 음성신호 기록방식이 혼합되어 있는데 비디오 부분의 오디오는 디지털 사운드 LD 와 같은 방식으로, 그 이외의 오디오 부분은 CD-DA (Compact Disc-Digital Audio : 오디오 CD)와 같은 방식으로 기록되어 있다. 따라서 비디오 부분과 오디오 부분에서는 디스크의 회전수가 달라서 랜덤 액세스를 행하면 약간의 시간이 걸린다. CD 와 CDV, LD 를 동시에 재생할 수 있는 호환적인 플레이어도 생산되고 있어 장래에 많이 사용될 것으로 전망되지만 아직은 소프트웨어가 많지 않다.

(2) CD-VI (Compact Disc - Video Interactive)

DVI 에 대응하여 Philips 사가 제안한 CD-VI 는, CDV 가 5 분간의 아날로그 영상과 20 분간의 CD-DA 가 가능하지만 이 CD-DA 부분을 CD-I 포맷으로 데이터를 기록하여 보자는 것이다. 비록 5 분간의 짧은 시간이지만 풀사이즈의 아날로그 영상을 재생할 수 있다는 장점을 가진 CD-VI 는 정식으로 발표되지 않고 계속 연구 개발중이다.

(3) CD-ROM XA (CD-ROM Extended Architecture)

CD-ROM XA 는 CD-ROM 의 확장 규격으로 1989 년 Sony, Philips, Microsoft 3 사가 기본 사양을 발표하였고 1990 년 2 월 표준 화상 형식을 발표하였다. 이것은 특정 하드웨어에 한정되지 않고 여러 시스템에서 채용 가능하다.

CD-ROM XA 는 기본적으로 CD-ROM 을 응용하기 위해 확장한 것으로

로 CD-ROM 사양에 CD-I 사양을 합치고 정보 교환을 위한 볼륨 및 화일 형식으로 ISO 9660 을 채택하였으며 오디오를 위해 ADPCM을 채택하여 양질의 오디오를 제공하고 있다. 최근에는 Sony 와 Fujitsu사 가 자사의 퍼스널 컴퓨터에 사용할 수 있는 CD-ROM XA 디스크 작성이 가능한 개발 지원용 소프트웨어인 CD-ROM XA Common Library 를 개발하여 Sony Quarterl 과 Fujitsu 의 FM-TOWNS 에 공통으로 사용할 수 있는 소프트웨어를 제작할 수 있게 되었다.

다. CD 매체의 표준화

최초 디지털 오디오 디스크의 한 방식으로 출발한 CD는 Philips, Sony 의 두 회사에서 제창되었기 때문에 CD의 사실상의 표준은 이들 두 회사에 의해 진행되었다. 디지털 오디오 CD(CD-Digital Audio)에 대한 규격은 1980년 붉은색 표지의 규격서에(레드북), 컴퓨터 데이터 CD 즉 CD-ROM에 관한 규격은 1983년 노란색 표지의 규격서에(옐로우북), CD-I에 대한 규격은 1989년 녹색 표지의 규격서에(그린북) 각각 규정하였다. 이들은 사실상의 세계 표준으로 점차 인식되었고 이것을 만든 Sony사와 Philips사가 라이선스를 갖고 있으며 레드북을 전제로 하여 옐로우북이, 그 연장선상에서 그린북이 규정되었다.

옐로우북은 레드북을 기준으로 하여 데이터의 구조를 규정하고 있는데 특히 에러 검출 정정 코드를 포함한 모드1과 포함하고 있지 않는 모드2로 나누어진다. 이렇게 두개의 모드로 나눈 이유는 데이터의 높은 신뢰성이 요구되는 경우와 신뢰성보다는 더 큰 디스크 용

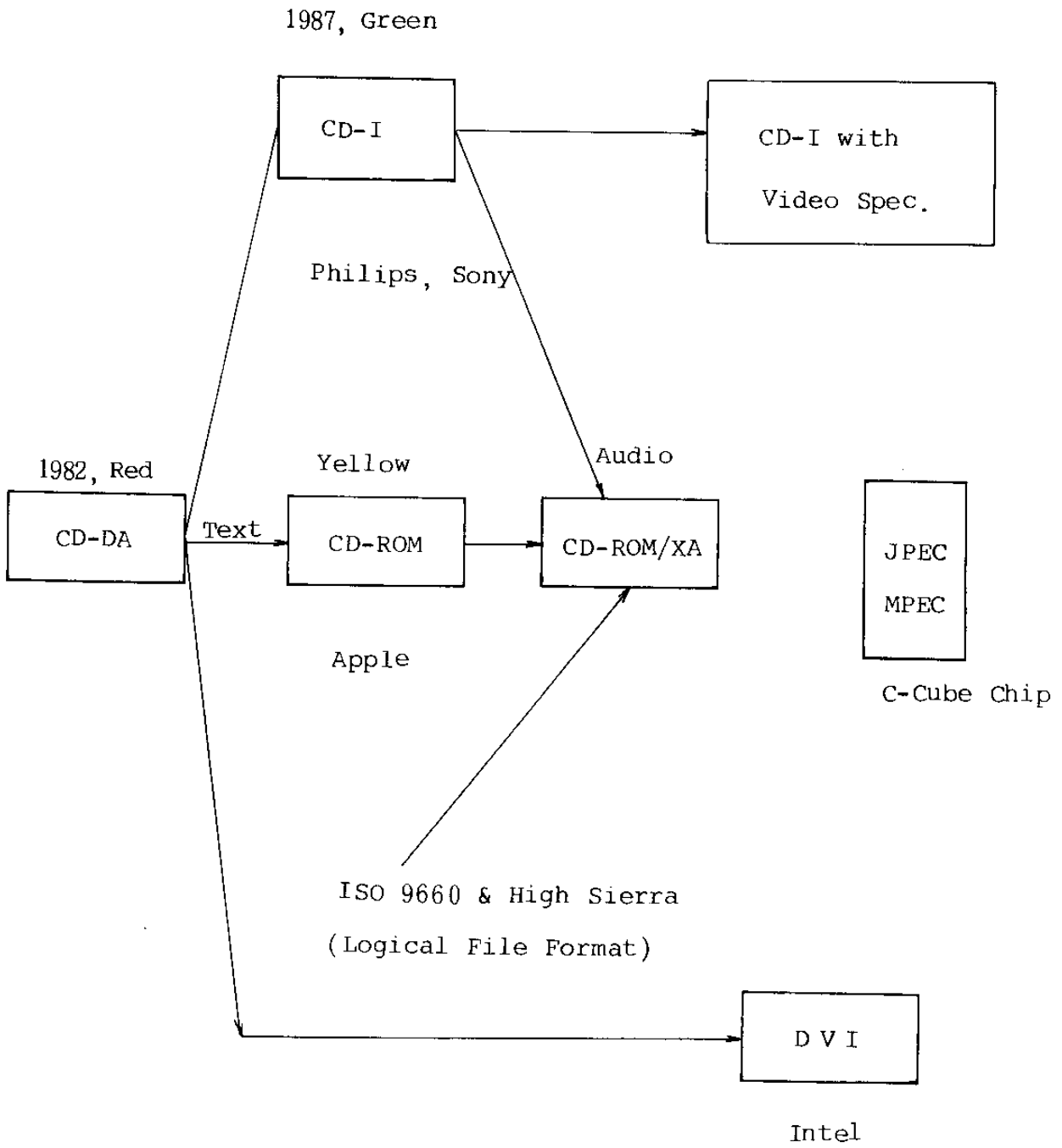
량이 필요할 때를 구분하여 효율적으로 이용하기 위해서이다. 모드 2에서는 디스크에 기록되는 화상과 음성 데이터의 구조에서부터 재생용 하드웨어 디스크의 기능, OS와 CPU의 규격까지 상세하게 규정한 것이 CD-I 규격인데 이것을 보면 CD 매체가 넓은 호환성을 갖고 있음을 알 수 있다.

1985년 11월 국제 표준화 기관에 논리 포맷을 제안하기 위한 모임이 미국의 네바다주의 하이 시에라 (High Sierra) 호텔에서 열렸다. 이때 Sony, Philips 이외에 Microsoft, Digital Equipment, Hitachi, 3M, Apple 사 등의 컴퓨터 제조회사 및 CD-ROM 드라이버 생산업체, 소프트웨어 하우스 등이 참가하였다.

토의 내용에는 CD-ROM의 성능을 최적화하고, WORM과 Erasable 광 디스크 등을 확산하기 위한 내용, CD-ROM에 대한 문자 사용의 표준화등이 포함되었다. 또한 MS-DOS, VAX, UNIX, Apple DOS 등의 대중화된 운영체제와 함께 사용하기 위한 내용과 CD-I와의 호환성 문제 등도 토의되었다.

이때 토의된 내용은 매우 긍정적으로 평가되었으며 기본내용은 Sony사와 Philips사가 제안한 옐로우북과 유사하였는데 그 결과를 하이 시에라 규격이라고 명하여 1987년 부분적 수정을 하여 보완한 후 국제 표준규격으로 ISO에 제안되었고, ISO는 이 규격을 ISO 9660이라고 명명하여 발표함으로써 CD-ROM에 대한 국제적 표준 규격으로 통용되었다.

<그림 2-2>는 CD 매체를 이용한 저장매체기술의 변천과정을 간략히 기술하였다.



< 그림 2 - 2 > CD 저장매체 기술의 변천과정

3. 다중매체 데이터의 압축 / 복원 기술

가. 데이터 압축의 필요성

다중매체 시스템의 구현에 있어서 가장 필수적인 기술요소는 다중매체 데이터 특히 풀모션 비디오 (Full Motion Video) 의 실시간 표현을 위해 개발자 환경에서 데이터의 캡처, 디지털화, 압축 (Compression), 필요시 전송과 사용자 환경에서 압축된 데이터의 실시간 복원 (Decompression) 과 관련된 제반 처리 기술이라고 볼 수 있다. (이후 다중매체 데이터의 압축 / 복원 기술이라고 함)

이미 오래전 부터 아날로그 형태의 레이저 디스크 (또는 Interactive Video Disc) 에 보관된 다중매체 데이터나 TV, VTR 이 제공하는 실시간 비디오 (Real Time Video) 를 퍼스널 컴퓨터에서 디지털 비디오 보드 또는 어댑터를 통해 디지털화하여 풀모션 비디오를 실시간으로 출력하는 방법이 개발 사용되어 왔다. 그러나 이것은, 사용자의 입장에서는 매체를 제어하는 보조기기들이 필수로 소요되어 비용면에서 부담이 되고 있고, 다중매체 시스템 개발자의 입장에서는 개발환경 즉 컴퓨터 시스템 상에서 원시 데이터에 대한 자유로운 처리 및 가공과 추가적인 변환 및 저장이 불가능하며 또한 압축되지 않은 대용량의 아날로그 데이터이므로 고속 통신에 어려움을 주고 있다.

TV, 비디오의 정지된 한 프레임 (Still Frame 또는 정지화상) 을 캡처하여 컴퓨터 데이터 파일로 변환하면 해상도 및 지원되는 칼라에 따라 약 500 KB에서 2 MB가 소요된다. 1초의 완전한 풀모션 비디오의 경우는 30 프레임 (1초 동안에 정지화상 30 프레임을 연속적으로 출력하면 1초의 모

션 비디오와 동일한 시각적 효과를 얻음.)을 기준으로 할 때 약 15MB에서 60MB가 필요하며 이러한 비디오 데이터를 650MB의 CD-ROM 디스크에 압축하지 않은 채로 저장할 경우 약 30 초 (10 초에서 40 초) 의 풀모션 비디오를 저장하게 된다. 그러나 CD-ROM 드라이브의 데이터 처리 속도는 고속의 하드 디스크 처리 속도와 비교할 때 50분의 1 정도인 초당 150 KB 이므로 디스크 한 장 즉 30 초 정도의 풀모션 비디오를 보려면 무려 한 시간 이상이 소요된다.

결국 현재의 저장매체인 CD-ROM 에 풀모션 비디오를 저장할 경우 비디오 데이터를 일정한 비율 (약 100 배 : 15 MB/0.15 MB) 이상으로 컴퓨터에서 압축하여 한 프레임당 약 5 KB 로 CD-ROM 에 저장해야 하고 일단 저장된 데이터는 다시 컴퓨터 상에서 같은 비율로 복원하여야 풀모션의 실시간 비디오로 출력이 가능해 지는 것이다.

그러나 그래픽 이미지 및 정지화상에 대한 압축/복원은 풀모션 비디오와는 달리, 높은 압축배율 보다는 정확한 비디오 데이터의 압축/복원이 요구되므로 데이터의 특성에 따라 각기 다른 압축배율 및 관련 기술이 복합적으로 요구된다.

본 고에서는 오디오 및 비디오 (그래픽 이미지 포함) 데이터의 압축/복원을 위한 기술적인 방법론보다는 이미 발표되고 그 기술이 상품화된 CD-I 와 압축/복원기술에 대하여 기술의 특성, 성능 및 발전방향등을 중심으로 비교 분석하였다.

나. CD-I 기술

(1) 개 요

CD-I (Compact Disc Interactive) 는 Philips사와 Sony사에 의해

공동 연구개발된 기술로 '86년 3월에 표준안을 확정하여 '88년 11월 Green Book에 표준안으로 수록됐다. 컴퓨터와 연결하여 사용하며 산업 및 기술분야 시장을 목표로 한 DVI 기술과는 달리 CD-I는 가정의 TV에 직접 연결하고 리모콘을 사용하여 다중매체 정보를 제공받을 수 있도록 한, 가전시장을 겨냥한 기술이다.

현재 CD-I 기술 개발에 참여하는 업체로는, CD-I 플레이어 생산에 Philips, Sony, Fujitsu, Pioneer, Sanyo 등 일본을 중심으로 한 십여개 업체와, CD-I 응용 소프트웨어 개발에 협력하는 Philips사와 PolyGram사가 합작한 AIM(American Interactive Media)사가 있으며, 모션 비디오 칩세트 개발에 Motorola, Sony, Matsushita의 3개 업체가 참여하고 있다. 고속의 CMOS 형식의 이 칩이 개발 생산되면 CD-I 플레이어의 공급가는 현재의 2,500불에서 1,000불로 하락이 예상되며 소비자가 구입 가능한 가전제품 차원의 다중매체 시스템으로 보급될 것으로 본다.

(2) 오디오 형식 및 특성

- 4가지의 오디오 제공 : CD-DA(PCM 방식), 3가지 레벨의 오디오(ADPCM 방식)
- 오디오 용량은 레벨 및 Mono, Stereo 방식에 따라 2배수로 저장 가능
- 한장의 650MB 용량의 CD-ROM에 저장되는 오디오의 용량은 <표 2-11>과 같다.

〈 표 2 - 11 〉 CD-I 오디오 용량

| 구 분 | Encoding 방 식 | 주 파수 영 역 | Quantizat- ion | S/N Ratio | 재 생 시 간 | 음 질 비 교 |
|-------------------|-----------------|-------------|-------------------|--------------|---------------------|----------|
| CD (디지털 HiFi) | PCM | 20 KHz | 16 Bit 선형 | 98 dB | 1Hr ST | CD-DA |
| HiFi Music | ADPCM | 17 KHz | 8 Bit 압축 | 90 dB | 2Hrs ST 4Hrs MO | LP-Album |
| Midi-Fi Music | ADPCM | 17 KHz | 4 Bit 압축 | 60 dB | 4Hrs ST 8Hrs MO | FM Radio |
| Quality Speech | ADPCM | 8.5 KHz | 4 Bit 압축 | 60 dB | 8Hrs ST 16Hrs MO | AM Radio |

- * PCM : Pulse Coded Modulation
- ADPCM : Adaptive Differential PCM
- ST : Stereo
- MO : Mono

(3) 비디오 형식

(가) 비디오 해상도

| 구 분 | NTSC 방식 | PAL 방식 |
|--------|-----------|-----------|
| 보통 해상도 | 360 X 240 | 384 X 280 |
| 2배 해상도 | 720 X 240 | 768 X 280 |
| 고 해상도 | 720 X 480 | 768 X 560 |

(나) 비디오 Encoding 형식

- DYUV : Delta Y (Luminance) UV(Chroma)
최상의 사진 이미지를 위한 Encoding
- CLUT(Color Lookup Table) : 텍스트와 그래픽 질의 향상을 위한 Encoding
- RGB 555 : 이미지 또는 복잡한 그래픽을 위한 Encoding

(4) 모션 비디오의 특징

- 부분화면의 모션 비디오가 72 분간의 디지털 비디오 형태로 제공됨
- 초당 25 에서 30 프레임의 비디오에 대한 압축 / 복원 기능 제공
- 비디오 알고리즘은 DCT (Discrete Cosine Transforamtion) 방식 채택
- CD-I 플레이어의 최대 액세스 속도는 초당 170 KB

(5) 응용 시스템 개발 도구

Philips 사가 개발한 CD-I 저작시스템은 3 단계로 구분 된다.

- Level 1 : Starter Pack
(Sequence Editor, Paint Package, Clip Art Disc, Rapid CD-I 포함 : 샘플 수준의 소프트웨어 개발용)
- Level 2 : CD-I 950/PC Personal Publishing System
CD-I 800/MAC MacAuthor Series
(완전한 저작 시스템으로, 비디오 / 오디오 처리 및 디스크 제작용 소프트웨어, 하드웨어 모두 포함)
- Level 3 : CD-I 700/SUN SPARC Emulator
(워크스테이션에서 통합적인 개발 환경 제공)

다. DVI 기술

(1) 개요

DVI (Digital Video Interactive)는 CD 및 하드디스크를 저장 매체로 한 디지털 비디오의 압축/복원 및 저장에 관한 기술로 '82년 General Electric사와 RCA사의 공동연구로 시작됐다. '86년 GE사가 RCA사를 인수할 때 이 기술을 같이 인수하여 GE의 David Sarnoff Research Center에서 개발하고 '87년 3월에 발표하였다.

Intel사는 '88년 10월에 GE로부터 이 기술을 인수하고 본격적인 기술 개발 및 상품화를 주도했다. 최근에 IBM이 공동 참여하여 IBM PC AT 및 PS/2 등의 하드웨어를 기본 시스템으로 하고 ASIC 기술이 이용된 2장의 비디오 처리보드를 내장한 상품(Action Media 750)을 발표했으며, AT&T사는 UNIX용 버전을, Olivetti사는 유럽 표준용 DVI 버전을 개발중이다. 소프트웨어 협력업체로는 Microsoft사와 Lotus사 등이 참여하고 있다.

가전시장을 목표로 하며, 가정의 TV에 플레이어를 연결하고 리모콘으로 사용하는 가전제품화된 CD-I 기술과는 달리, DVI는 기술 및 일반산업 분야의 시장을 일차 목표로 하고 '93년까지 점진적으로 학교 및 일반 가정에 보급할 계획으로 추진되고 있다. 특히, DVI 기술의 핵심적 하드웨어인 Capture Board (오디오/비디오 디지털이저) 및 Delivery Board (오디오/비디오 Play Back)는 각각 2000불 내외 ('90년)이나 기술개발을 통해 '92년에는 500불로 가격 하락을 추진하고 있으므로 이같은 보급 계획의 실현 가능성을 뒷받침하고 있다.

(2) 비디오의 압축/복원

(가) 풀모션 비디오의 구현

— 전체 사항

- 해상도 (예) : 512*480
- 칼라 : 24bits/pixel (16 M가지의 색상 : 3Bytes)
- 소요 프레임 : 30frames/sec
- CD-ROM 최대 처리 속도 : 0.15 MB/sec (9MB/min)
- CD-ROM 저장 용량 : 650MB

— 압축 전 데이터량

- 프레임 당 : $512*480*3 = 720KB/frame$ (737,280Bytes)
- 초 당 : $720KB*30 = 21.6MB/sec$ (22,118,400Bytes)

— 풀모션 비디오의 압축 배율 및 압축 정보량

- 소요 압축 배율 : $21.6MB/0.15MB = 144$
- 장당 저장 용량 : $650MB/9MB = 72.2$ 분 / CD-ROM

- 실제 모션 비디오에 있어 압축된 비디오는 첫 프레임 (일명 reference frame) 과 연속 프레임 (일명 delta frames) 으로 구성되는데, 연속 프레임은 앞 프레임 (reference frames 또는 delta frame) 에서 변경된 정보만을 보관하기 때문에 전체적으로 높은 배율 (약 160 배) 의 압축/복원이 가능한 것이다.

(나) 정지화상의 압축

정지화상의 경우는 이미지의 종류 및 특성에 따라서 압축되는 이미지 형식 및 압축배율이 달라진다. 자연의 풍경 등 일반적인 이미지에 대해서는 압축배율이 최대이고, 최대 범위의 색도 및 명도를

지원하는 '9bit 압축' 형식이 적절하고(복원시 분실되는 색정보가 있으나 육안으로 감지되지 않음), 압축 후 복원해도 색정보가 분실되지 않는 '16bit 압축' 형식과 '8bit CLUT' 형식은 지도와 같은 그래픽 이미지의 압축/복원에 사용함이 효과적이다.

< 표 2-12 > DVI 압축 알고리즘에 의한 이미지 형식 및 크기

| 이미지 형식 (bits/pixel) | 세 부 내 용 | 이미지 크기 (KB) |
|------------------------|--------------------------|----------------|
| 24 | 자연에 가까운 화상 (160여만 색상) | 720(#1) |
| 16 비압축 | Captured Image(#2) | 480 |
| 9(YUV) 비압축 | Captured Image(#3) | 272 |
| 16 압축 | #2 의 압축 (자료에 따라 배율 다름) | 136 |
| 8CLUT | #2 의 압축 (2 : 1) | 240 |
| 9(YUV) 압축 | #3 의 압축 (14:1 까지 가능) | 108 |

1 . 해상도는 512*480 의 경우임 (512*480*3bytes)

(3) CD-ROM에 저장시 매체별 정보량

DVI의 기술로 다중매체를 압축하여(144:1의 비율) 각 매체별로 CD-ROM(650MB 용량 기준)에 저장할 때 저장 가능한 최대의 정보량과, 1장의 CD-ROM에 매체를 혼합하여 저장할 때 저장 가능한 정보량의 예는 <표 2-13>와 같다. 후자의 경우 전체 저장량은 650MB의 92%(595MB)에 해당된다.

<표 2-13> CD-ROM에 저장시 매체별 정보량(압축 배율: 144배인 경우)

| 구 분 | 매체별 최대정보량 | 매체혼합시 정보량(예) |
|---------|---|-----------------------------------|
| 텍스트문서 | 약 650,000 페이지 | 약 15,000 페이지(15 MB) |
| 오 디 오 | FM 수준의 스테레오 : 약 5시간 중간 수준의 모노 : 약 22시간 AM 수준의 모노 : 약 44시간 | 정지화상 출력상태에서 6시간의 AM 모노 오디오(90 MB) |
| 정 지 화 상 | 1024X512고해상도 : 5,000 프레임 512X480 해상도 : 10,000 프레임 256X240 해상도 : 40,000 프레임 (12:1 압축) | 5,000 프레임의 고해상도 정지화상(310 MB) |
| 모션비디오 | 풀모션 비디오 : 72분 1/4 화면 모션비디오 : 4시간 1/8 화면 모션비디오 : 14시간 (15 프레임 / 초) | 20분 풀모션 비디오 (180 MB) |

(4) DVI 기술의 발전 전망 및 문제점

(가) DVI 기술의 발전 전망

주목받고 있는 DVI 기술은 현재 Intel 사가 주도하고 IBM 이 공동 참여중으로 비디오 프로세서 / 보드, 압축 알고리즘, 응용 시스템 개발용 하드웨어 시스템 및 지원 소프트웨어 (도구 수준) 등이 향상된 기술로 제품화되면 '93 년까지 업무용, 상업용, 교육용을 포함한 다중매체 시스템 기술 및 시장의 해결방안으로 제시될 전망이다.

| | | | | |
|-------------|--|--|---|---|
| 비디오 프로세서 | Prototype Video Processors | i750PA/DA Video Processors | i750PB/DB Video Processors | Third Generation i750 Video Processors |
| 알고리즘 | PLV 1.0 Video | PLV, RTV, PIC 1.0 | PLV, RTV, PIC 1.5 & 2.0 | PLV, RTV, PIC 3.0, MPEG, JPEG, CCITT, etc |
| 하드웨어 제품 | Prototype System | Pro750 Products for Commerial Applications | ActionMedia Products for Desktop Applications | Products For Consumer and School Applications |
| 기술동향 및 보급계획 | <ul style="list-style-type: none"> • DVI 기술발표 | <ul style="list-style-type: none"> • Intel사가 판권획득 • IBM 공동참여 • Olivetti 참여 (유럽지역) | <ul style="list-style-type: none"> • 협력업체 대거 참여 • 보드를 지원하는 개발도구 발표 • 응용소프트웨어 및 시스템개발 | <ul style="list-style-type: none"> • 산업체및 상업용 H/W 제품화 • 영상회의용 다중매체 시장점유 |
| 시 기 | 1986-87 | 1988-89 | 1990-91 | 1992-93 |

< 그림 2 - 3 > DVI 기술의 발전 추이

(나) DVI 기술의 문제점

DVI는 발전적인 기술이지만 해결해야 할 몇 가지 문제점이 있다. 첫째, 최종 제품의 해상도는 VTR의 해상도 수준을 상회하고 있으나 실시간 비디오를 얻기 위해 압축한 현재의 해상도는 VTR의 해상도보다 낮다. 둘째, 압축배율이 높으므로 압축시 많은 정보가 생략되므로 압축 이전의 아날로그 영상보다 화질이 떨어진다. 셋째, 판매를 목적으로한 VTR 비디오 수준의 응용 소프트웨어를 만들려면 오디오/비디오 마스터테이프를 Intel사에 보내 메인프레임에서 제공하는 PLV (Production Level Video)를 채택하여 압축해야 하는 문제점들이 노출된다. 또한 DVI 개발을 지원하는 소프트웨어로 ActionMedia 750 Production 팩키지 및 CEIT System사의 저작도구 Authology와 Time Arts사의 2차원 그래픽팩키지 Lumena 등이 있으나 종류가 부족한 실정이다.

라. 주요 다중매체 기술의 특성 비교

| 구 분 | CD-I | DVI | IVD | CD-ROM XA |
|--------|-------------------------------------|--|--|--|
| 데이터형식 | 디지털 | 디지털 | 아날로그 | 디지털 |
| 저장매체 | CD-ROM | CD-ROM, HDD | Video Disc | CD-ROM |
| 저장매체종류 | 5.25" CD-ROM | 5.25" CD-ROM, HDD | 12"디스크 | 5.25" CD-ROM |
| 하드웨어형태 | 소비자전용 가전품 | PC/AT, 가전품 추진 | LDP, PC에 I/F | PC에 SCSI I/F |
| 모션비디오 | • 72분, 부분 화면 (해상도 : 120*80) | • 72분, 전화면 (해상도 : 256*240, VTR 수준) | • 면당 30분, 전화면 (해상도 : 640*240, TV수준) | • CD-I와 호환 (해상도 : 640*480, 320*200) |
| 정지화상 | • 최대 8,000 프레임 (해상도 : 720*480) | • 최대 40,000 프레임 (해상도 : 720*512, 또는 768*480) | • 면당 54,000 프레임 (해상도 : 640*480) | 3 형식 (CLUT 1, 4, 8) 소요 |

(다음 페이지에 계속)

(전 페이지에서 계속)

| 구 분 | CD-I | DVI | IVD | CD-ROM XA |
|-----------|------------------------|---------------------------|---------------------|-------------------|
| 그래픽 오버레이 | 가능 | 가능 | 비디오보드 필요 | 가능 |
| 비디오 처리 방식 | 저속, H/W 지원요 | 고속, S/W 구동 방식 | 불가 | H/W적 지원 |
| 압축 / 복원여부 | 사용 | 사용 | 비사용 (Standard) | 사용 |
| 오디오 | • 16 시간 (AM 수준의 모노) | • 최대 44 시간 (AM 수준의 모노) | • 면당 30 분 • 2 채널 | • 19 시간 • 2 채널 |
| 소스편집기능 | 제공 | 제공 | 비제공 | 제공 |
| 텍스트 문서 | 300,000 페이지 | 650,000 페이지 | - | - |

4. 그래픽 사용자 인터페이스 (Graphical User Interface)

가. GUI의 역사적 배경과 정의

흔히 그래픽 운영체제로 불리워지는 GUI는 1970년대에 Zerox사의 Palo Alto 연구소에서 시작되었다. 그 당시는 단순하고 기본적인 것 이어서 실용화 되지는 못했지만 Apple사에서 매킨토시에 구현함으로써 상용화 되었다. 그 이후 컴퓨터의 보급 확산과 더불어 전문적인 사용자 이외에도 많은 사람들이 컴퓨터를 사용함에 따라 사용이 용이한 시스템에 관한 요구가 계속 증가하였고, 하드웨어의 놀라운 발달로 이러한 요구를 충분히 수용할 수 있게 되었기에 GUI는 많은 시스템에서 연구가 진행되어 채택되어 왔다. 매킨토시의 Desk Top IBM PC의 Windows, 워크스테이션의 (UNIX) X-Window를 근간으로 한

OSF의 모티프나, SUN의 오픈룩등이 그 대표적인 것이다. GUI의 궁극적인 목적은 사용자에게 이용의 편리성을 극대화하여 제공함에 있기에 다양한 툴의 개발과 공통된 인터페이스를 제공하여 새로운 환경으로의 변화에 의한 재교육의 시간, 노력을 줄일 수 있으며, 컴퓨터를 보다 쉽게, 보다 생산적으로 이용하도록 그림뿐만이 아니라 음성, 영상등의 다중매체를 지원할 수 있는 방향으로 발전되고 있다.

나. GUI와 CUI(Command User Interface)

초창기의 컴퓨터 시스템은 명령어 방식의 인터페이스만을 허용하여 DOS나 UNIX의 많은 명령어를 일일이 배우고 능숙히 사용하는 데에는 상당한 시간과 노력이 필요했다. 그러나 GUI의 등장은 이런 문제를 해결하여 일반 사용자에게도 쉽게 컴퓨터를 사용할 수 있는 방향을 제시하였다.

TBS사 (Temple, Barker & Solane Inc)가 GUI와 CUI를 비교분석한 자료에 의하면 다음과 같이 GUI의 이점을 표현하고 있다.

- 평균적으로 능숙한 GUI 사용자는 37 퍼센트 정도의 작업을 더 수행할 수 있다.
- 작업의 정확도에 있어서도 91 대 74 퍼센트로 우수하다.
- 두 결과에 의해 생산성은 약 58 퍼센트 증가하고 있다고 보여진다.
- 초보자의 교육결과 GUI의 교육을 받은 사람이 정확하고 빠른 업무처리 성과를 보여 GUI의 이용의 편리성과 이해도를 반영하였다.

결론적으로 GUI의 사용자들은 CUI의 사용자에 비해 높은 생산성, 정확성, 숙련도를 보임으로써 GUI의 우월성을 보여주었다.

GUI의 가장 큰 매력은 그 사용과 교육의 편리성이라 할 수 있다. 적은 노력과 시간으로 정확하고 능률적으로 작업할 수 있으며, 그림과 음성의 지원을 통하여 좀더 친근하고 익히기 쉬운 모습으로 바뀌어가고 있다.

다. GUI의 현황비교

(1) GUI 요소 비교

그래픽 사용자 인터페이스는 세가지의 요소로 이루어 지는데

첫째, API(Application Program Interface)는 프로그래밍 언어 함수들의 집합으로 이것은 프로그래머가 윈도우, 메뉴, 스크롤 바, 그리고 아이콘등을 정의하는 방법을 제공한다.

둘째, Windowing System은 윈도우, 메뉴, 그리고 다이얼로그 박스를 화면에 나타나도록 하는 명령어와 프로그래밍 툴의 집합이다.

셋째, Imaging Model은 폰트와 그래픽을 실제적으로 화면에 나타나도록 하는 방식을 정의한다.

각각의 GUI의 요소는 아래의 <표 2-14>와 같이 비교하였다.

<표 2-14> GUI의 각 요소 비교표

| | WINDOWS | Macintosh | Motif | Open Look | NextStep |
|------------------|---------------------------|----------------|-----------------|-----------|--------------------|
| API | | MAC I/F | XUI | X View | Kits |
| Windowing System | Graphics Device Interface | Window Manager | X Window | X11/NEWS | Window Server |
| Imaging Model | CDI Output Functions | Quick Draw | Not yet Decided | | Display PostScript |
| Operating System | MS-DOS | MAC OS | UNIX | | |
| CPU | 80x86 | 680x0 | 80x86 | | 680x0 |

- * OSF(Open SoftWare Foundation) - Motif
- * SUN - OpenLook
- * NeXT - NextStep
- * 80x86 - Intel 8086, 80286, 80386
- * 680x0 - Motorola 68000, 68020, 68030

(2) 매킨토시

매킨토시에서의 인터페이스 환경을 보면

- 마우스로 대표되는 Pointing Device
- 마우스에 의해 나타나거나 사라지는 스크린상의 메뉴
- 컴퓨터가 작업중인 상황을 보거나 사용할 수 있는 윈도우
- 여러기능을 표시하는 아이콘 (Icon)
- 컴퓨터에게 작업지시를 위한 툴 (Dialog Box, Button, Sliders, Check Box)

등이 있으며 매킨토시의 GUI는 인터페이스의 요소를 만들어 주는 “툴박스”와 그래픽 명령을 수행하는 “윈드로우”로 구성된다.

이러한 형식이 모든 GUI에 공통적인 것은 아니지만 대부분 기본적인 개념은 동일하며 매킨토시에서의 GUI는 다른 시스템에서의 그것보다 사용의 용이성이나 응용프로그램의 이용에 있어서도 편리하다. 그것은 기존의 운영체제위에 인터페이스를 추가한 다른 시스템과는 달리 시스템의 룰상에 인터페이스를 구현하여 모든 프로그램이 동일한 방식을 지원하고 뛰어난 그래픽처리와 음성기능을 통해 사용자의 편의를 도모하며 다양한 사용자 인터페이스 툴박스를 제공하여 쉽고 동일한 인터페이스 환경을 지원하기 때문이다.

(3) 퍼스널컴퓨터 레벨의 GUI

퍼스널컴퓨터 레벨의 GUI는 MS-DOS 상의 Windows나 OS/2의 PM (Presentation Manager)로 크게 나눌 수 있다. 이들의 모습은 상당히 비슷하며 특히 Windows는 MS-DOS의 이용자들의 불만이었던 메모리, 다중처리등의 문제를 상당부분 해결해 주었지만 그래픽 인터페이스로서의 기능은 초보적인 수준이며, 음성이나 영상장치의 접속등 해결해야 할 문제가 남아있으며 더구나 한글의 처리문제등은 대다수의 퍼스널컴퓨터 사용자의 불만이 그대로 남아 국내에서의 독자적인 GUI의 개발필요성이 증가하고 있다.

참고로 최근에 발표된 Microsoft사의 WINDOWS 3.0의 개선부분을 보면 아래와 같다.

- 메모리 관리 기능의 개선
- 좀 더 쉬어진 사용자 인터페이스
- 다중작업 능력의 보강
- 응용 프로그램의 보강 (File Manager, Task List, Program Manager 등)
- 네트워크 기능 보강

(4) UNIX에서의 GUI

업계의 표준으로 자리잡아가고 있는 UNIX에서도 GUI의 접목은 꾸준히 발달되어 오고 있다. 대표적으로 OSF의 Motif와 SUN의 Open-Look을 들 수 있다. 이 시스템은 X/Window를 근간으로 발전되었으며, UNIX의 난해한 사용자 인터페이스를 쉽게 지원함으로써 전문 인력이 주로 사용하던 시스템을 일반 업무에도 이용하는 계기를 만들었다. MS-DOS에서와는 다르게 UNIX자체의 막강한 네트워크 기능과 다중처리

능력을 이용함으로써 전체적으로 민활하게 동작하여 사용의 편의성도도모하고 있으며, 표준화되어 가고있다.

그리고 X Window의 소스가 공개되어 있기에 표준 그래픽 사용자 인터페이스를 구현하는 기반이 되고 있으며, 각각의 GUI들은 스타일 가이드와 툴키트를 제공하여 프로그래머가 사용자 인터페이스를 쉽게 하도록 해주며 동일한 사용자 인터페이스 환경을 가진 응용 프로그램을 개발할 수 있도록 하고 있다.

(5) NextStep

NextStep은 현재까지의 GUI들중 가장 첨단의 인터페이스를 제공한다. 많은 유틸리티를 자체로 보유하고 있으며 NextStep 자체가 운영체제의 대부분을 차지한다. 그러나, 다른 UNIX의 GUI와는 다르게 X Window를 사용하지 않아서 NextStep을 네트워크환경에서는 사용할 수가 없다. 단일 사용자만의 환경으로는 강력한 툴키트의 제공과 고품질의 출력을 위한 Display Postscript, 음성지원기능 등으로 차세대 GUI의 방향을 제시하고 있다고 평가된다.

5. 다중매체 통신 기술

다중매체 통신은 텍스트만 가능했던 종래의 전자메일에 비디오(이미지, 그래픽 포함), 오디오(음성, 음향, 음악 포함) 등의 다중매체 데이터를 포함시켜서 정보를 교환하는 것을 말한다.

현재 활발히 진행되고 있는 다중매체 통신을 위한 기술 및 시스템은 아직 실험단계에 있어 다중매체 데이터를 완벽히 처리, 관리, 전달하기에는 한계가 있으나 이러한 기술과 시스템 개발의 경험을

바탕으로 점차 개선됨에 따라 기술적인 문제들이 해결되고 있다.

가. 다중매체 통신의 특성

다중매체 통신의 특성은 기존의 텍스트 데이터 뿐만 아니라 다양한 종류의 데이터를 수용하여야 한다는 점과 이러한 데이터를 신속 정확하게 전송할 수 있는 고속 대용량의 통신망을 필요로 한다는 점이다.

(1) 데이터의 종류 및 특성

- 텍스트 데이터 : 일반적인 텍스트 데이터의 형태로서, 특정 코드 값을 2진 형태로 전송하며 비교적 저속의 통신으로도 가능하다. 전송시 에러 발생은 치명적이므로 대체로 에러 발생율이 낮아야 하며 기존의 통신망으로도 수용이 가능한 데이터이다.
- 비디오 데이터 : 이미지, 그래픽 등을 포함한 비디오 데이터는 사용하는 고해상도의 모니터에 따라 전송되는 데이터의 양이 달라지며, 동작화면을 구성하기 위해서는 대용량의 고속통신을 필요로 하므로 전송 지연에 민감하다.
- 오디오 데이터 : 음성, 음향, 음악 등을 포함한 오디오 데이터는 종류에 따라 대역폭이 달라지며 전송 지연에 민감하다.

(2) 다중매체 통신을 위한 고속 통신망의 필요성

고속통신을 위한 필수적인 사항으로는,

- 대용량의 정보 공급과 수요의 원활한 소통
- 전송지연을 낮추고, 실시간 통신 실시
- 채널간의 동기화 확립
- Remote Procedure Call 기능

- Light Weight Thread 의 사용 등이며 다중매체의 고속 통신에 있어서는 전송 매체의 특성에 따라 다음과 같은 사항들이 요구된다.

(가) 풀모션 비디오 통신

- 표준 TV 비디오 영상을 초당 30 프레임으로 가정할 때 한 프레임당 600-750 KB가 필요하다. 즉 초당 약 18 MB - 22.5 MB의 전송이 이루어 진다. (압축기술을 통해 1/100으로 압축이 가능함)
- HDTV와 같은 고선명 비디오 서비스는 1.2 Gbps의 전송이 필요하다.
- 네트워크 지연이 매우 적은 실시간 통신을 요구하기 때문에 회선 교환이 가장 적합한 방식이다.
- 채널의 전체 대역폭은 풀모션 비디오 트래픽과 나머지 낮은 데이터 속도 서비스를 통합할 수 있도록 충분히 커야 한다.

(나) 음성 데이터, 정지화상 통신

- CD 수준의 음성 데이터를 가정하면 1 샘플당 16 비트의 데이터로 초당 44,100 샘플이 필요하며 이를 계산하면 초당 약 705 KB의 전송이 이루어진다.
- 네트워크 지연은 심각한 고려사항이 아니다.
- 고속 패킷교환 기술이 사용될 수 있다.

(다) 텍스트 데이터 통신

- 기존의 통신망으로 수용이 가능하다.
- 패킷 교환기술이 사용된다.

비디오 및 오디오 데이터는 압축기법을 사용하여 절대량을 줄일 수 있으나 압축된 데이터도 일반 텍스트 데이터보다 크기 때문에 위와 같은 대용량의 데이터를 전송하기 위해서는 고속 대용량의 통신망이 필수적이다.

나. 기존 통신 기술의 문제점 및 통신망 현황

(1) 기존 통신 기술의 문제점

(가) 고속 통신 기술의 부족 : 기존의 통신망은 주로 9.6 Kbps 이하의 저속 통신에 적합하며 ISDN의 기본 서비스인 2B+D가 허용하는 Band Width 안에 다중매체 통신을 하기에는 기술 수준이 미흡하다.

(나) 통신망의 단순성 : 기존의 통신망은 텍스트 데이터 위주로 설계되어 다양한 미디어의 데이터를 수용하기에 부적합하다.

(다) OSI 7 계층과 다중매체 통신 : OSI 통신 구조는 계층간에 엄격한 독립성이 확보되어 있어 다중매체 통신에 장애가 되며 이러한 구조는 한 종류의 미디어 통신에만 적합하다.

(2) 기존 통신망 현황

| 통신매체 | | | 현황 | 비고 |
|-------|----|-----|---|---|
| 공중통신망 | 회선 | 음성급 | <ul style="list-style-type: none"> 가장 널리 사용되고 있는 통신회선 저속의 (9.6 Kbps 이하) Analog Modem을 이용 높은 에러율 (BER 10⁻³ : KTA 기준치 4.8 Kbps) | <ul style="list-style-type: none"> 높은 에러율과 속도상의 제한 - 다중매체 |

(다음 페이지에 계속)

(전 페이지에서 계속)

| 통신매체 | | | 현황 | 비고 |
|-------|----|--------------|--|---|
| 공중통신망 | 회선 | 부호급 | <ul style="list-style-type: none"> • 통신회선의 Digital화로 사용이 늘어가는 통신회선 • 음성급 회선에 비해 고속 (56 Kbps) • DS1급 (1.544 Mbps) 이상 서비스는 제한적으로 제공 (국내) • 통신회선의 고속화, 고품질화에 따라 점차 광통신으로 발전 | <ul style="list-style-type: none"> • DS1급 이상속도에서는 일부 다중매체 서비스 가능 • DS3급 이상의 회선은 다중매체 장거리 연결에 사용가능 |
| | | 전화회선 (PSTN) | <ul style="list-style-type: none"> • 기존의 전화 회선 사용 • 저속의 Dial-Up Modem 사용 | 높은 에러율 및 속도상의 제한으로 다중매체 통신에 부적합 |
| | | 패킷교환망 (PSDN) | <ul style="list-style-type: none"> • 데이터를 패킷 단위로 교환 • 회선 사용상 효율이 높은 텍스트 데이터 전용의 교환망 • 현재까지는 비교적 저속의 서비스 제공 (9.6 Kbps : 국내) | 접속, 전송 지연 및 속도상의 제한으로 다중매체 통신에 부적합 →Fast Packet은 음성위주의 다중매체 서비스 가능 |

(다음 페이지에 계속)

(전 페이지에서 계속)

| 통신매체 | | 현황 | 비고 |
|-------|---|--|--|
| 사설통신망 | Local Area Network (LAN) Metropolitan Area Network (MAN) | <ul style="list-style-type: none"> • Mbps 단위의 고속 통신을 기초로 지역을 연결한 통신망 • 구리선 (동축케이블) 이용한 LAN <ul style="list-style-type: none"> - 지역 한계 극복에 어려움 (다수의 중계가 필요) - 초고속 통신에 어려움 • 지역적 한계를 극복하고 고속통신을 이루기 위해서는 광통신이 필수 | <p>현재 사용하고 있는 통신망중 비교적 다중매체의 통신특성에 부합</p> <p>FDDI-I FDDI-II DQDB 등</p> |
| | Wide Area Network (WAN) | <ul style="list-style-type: none"> • 지역적으로 멀리 떨어진 LAN 등을 연결하여 구축되는 통신망 • 장거리 전송에 적합한 체제 | |

이제까지의 LAN은 10 Mbps의 속도이나 최근에는 100 Mbps 이상의 전송 속도를 갖는 고속 LAN이 등장하고 있는데, 현행 컬러 TV 데이터를 단순히 디지털화 하는데에도 약 100 Mbps가 필요한 정도임을 고려할때 이 수준도 만족할만한 정도는 되지 못한다. 다중매체의 완벽한 전송을 위해서는 1,000 Mbps 이상의 전송속도가 바람직한데 FDDI와 같은 고속 LAN (또는 MAN) 은 광섬유 케이블을 사용하여

1,000 Mbps 의 많은 데이터를 한꺼번에 전송할 수 있는 매체로 개발될 수 있을 것이다.

다. 고속 통신 기술

최근에 실험 개발되거나 발표된 고속통신의 기술들을 요약 정리하면 다음과 같다.

(1) FDDI-I

- 하드웨어의 개발을 전제로한 기술
- ASC X3T9 에서 개발하여 ISO 의 IS 로 채택
- 광섬유를 이용한 광통신에 기초한, 초고속 MAN(100 Mbps 이상)
- Clock 을 분산하여 Jitter 를 최소화
- 최대 500 Node 의 물리적 접속 가능
- 100 km 정도의 지역 대상
- Dual Ring Bus 구조
- IEEE 802.2 의 LLC(Local Link Control) 사용
- Timed Token Rotation Protocol 채용
- Wiring Concentrator 이용
- 중요도에 따라 Station 을 Class 로 분리하여 Primary Ring 과 Secondary Ring 에 구분 연결 가능

(2) FDDI-II

- 기본적으로 FDDI-I 의 기능 보유
- 동시성 데이터의 전송을 위한 Hybrid Ring Control 기능 추가
- 표준안 작성 중

- Broadband ISDN 과 Frame Format 상의 호환성이 용이치 않아서 표준안으로서 문제 여지 있음

(3) DQDB

- 호주의 Telecom Australia 에서 제안한 QPSX(Queued Packet and Synchronous Exchange) 방식
- IEEE 802.6 위원회에서 추진
- 현재까지 가장 무난한 MAN망으로 평가
- DUAL BUS 구조
- 광섬유를 전송매체로 사용
- CCITT 에서 장래에 권고할 Broadband ISDN 의 Asynchronous Transfer Mode 와 동일한 Cell 구조 사용하여 상호연동 용이함
- 공평성과 우선순위 기능에 아직 해결하지 못한 문제가 남아 있으나 기능적으로 음성이나 화상 같은 서비스를 제공할 수 있고 향후 통신의 기초가 되리라 생각되는 Broadband ISDN 과 호환성이 우수하여 공중망과의 연계도 용이하므로 MAN 의 주류가 되리라 예상

(4) METROCORE

- Bellcore 에서 개발 중
- Fasnet Protocol 을 기본으로 사용
- 8 개의 MAC 을 동시에 병렬처리
- 최고 1.7 Gbps 의 전송 속도
- 실험망 단계

(5) Broadband ISDN

- ISDN 의 고속화
- CCITT 에서 표준화 작업 진행
- ATM 에 기초 : 각 채널의 속도가 다양하며 실시간 데이터 처리에 유리하고 전송 지연의 최소화가 가능
- 150 Mbps 이상의 데이터 처리능력
- 데이터의 압축 기법과 함께 통신을 실현함으로써 다중매체 통신의 성패를 좌우할 요소

(6) CRMA(Cyclic Reservation Multiple Access)

- IBM Research(Zurich)에서 제안
- Slotted Bus Shared-Medium Access Protocol
- SONET Bit Rates(0.622, 1.244, 2.488 Gbps)
HIPPI Rates(0.8, 1.6 Gbps)
- Reservation-Based

(7) SMDS(Switched Multi-Megabit Data Service)

- Bellcore 의 서비스 개념
- Metropolitan/Wide Area 에 LAN의 성능과 Feature 제공
- Public, Packet Switched Data Service
- SMDS Interface Protocol(SIP)는 DQDB 의 Connectionless 부분에 기초함

(8) ATM (Asynchronous Transfer Mode)

- CCITT SG 11, 18
- Broadband ISDN 의 정보 전달 방식
- 일정크기의 Cell

- 모든 종류의 트래픽을 구분없이 비동기식으로 전달
- 시분할 다중화 방식에 의한 패킷형 전달모드

라. 다중매체 통신시스템의 사례

(1) AHENA

- CAI, 전자 OHP, Workbench, 전자메일 / 게시판, 전자회의
- 83년부터 MIT, DEC, IBM 공동 Project

(2) ANDREW

- CAI, OA, 전자메일 / 게시판, 전자회의
- '82년 카네기멜론 대학, IBM

(3) IMES

- 일본 Kyoto 대학

(4) HECTOR

- 독일 칼스웰 대학
- Distributed Academic Computing
- OSI 프로토콜 이용 User 통신 서비스
- CAM 지원 DB 구축

(5) ISI

- 미국 DARPA의 컴퓨터 네트워크 통신 프로젝트의 일부분
- MPM:Message의 Routing, Transmission, Delivery 담당
- UIP:Message의 작성, 읽기, 송달을 담당

(6) AGORA

- 프랑스에서 Kayak이라 불리는 Project 부분으로 설계
- 기능적 모델은 IFIP 모델과 같이 정의되었다.

6. 다중매체 기술의 국제표준화 동향

가. 관련 기술 표준화의 필요성

다중매체 데이터는 자료의 특성 및 시스템에 따라 처리방식이 다르므로 관련 기술의 표준화에는 많은 어려움이 따른다. 특히 관련 기술을 개발하는 기업들이 독자적인 표준화 방안을 제시하고 있으므로 기술적인 문제와 더불어 시장 석권과 관련된 경제적인 이해관계도 다중매체 기술의 국제 표준화에 어려움을 야기시키고 있다. 표준화 작업은 ISO 및 CCITT 등에서 주도적으로 이루어지고 있다.

다중매체 기술의 표준화는 다중매체 데이터의 종류에 따라 각기 분리하여 독자적으로 표준화하는 방법과 모든 종류의 다중매체 데이터를 통합하여 표준화하는 방법 등 두 가지가 있다. 여기서는 다중매체 기술의 핵심적 과제인 모션비디오와 이미지 데이터의 저장, 전송 등의 처리를 위한 압축/복원 기술에 대한 국제적인 표준화 방안과 사용자 인터페이스 관점에서의 다중매체/하이퍼미디어의 국제표준화 방안을 중심으로 고찰한다. 매체저장 기술과 관련한 CD의 표준화에 대해서는 앞서 기술한 바 있다.

나. 압축/복원 기술에 관한 표준화 동향

〈표 2-15〉는 압축/복원 기술을 중심으로한 국제표준화 활동에 대한 비교 요약표로 모션비디오의 표준화 추진 단체인 MPEG와 기술 개발업체인 Intel사, 그래픽 이미지 데이터 표준화 단체인 JPEG, 영상회의를 위한 통신기술의 표준화 단체인 Px64(공식 명칭은 H.261)의 표준화 활동을 비교하였다.

〈 표 2 -15 〉 압축 / 복원기술 표준화 방안 비교표

| 구 분 | MPEG | JPEG | Px64 (H.261) | DVI |
|-----------|-----------|-----------|-----------------|------------|
| 확 정 시 기 | '92 년 초 | '91 년 말 | '91년 중반 | '90 년 말 |
| 표 준 화 기 구 | ISO/CCITT | ISO/CCITT | CCITT | (Intel 사) |
| 표 준 화 대 상 | 모션비디오 | 이미지 | 영상회의 | 모션비디오 |
| 압 축 방 식 | DCT(#1) | DCT | DCT | 자체기술 |
| 압 축 비 율 | 80-275:1 | 8-100:1 | 100-300:1 | 80-120:1 |
| 상 대 적 비 용 | 고 가 | 저 렵 | 저 렵 | - |
| 영 상 의 질 | 고 급 | 고 급 | 저 급 | - |

(#1 DCT : Discrete Cosine Transformation)

(1) MPEG(Moving Pictures Expert Group) 표준화 방안

- ISO/CCITT 표준안
- 대상 데이터 및 목적 : Full Screen 모션비디오의 압축 / 복원 기술
- 압축 알고리즘
 - DCT 방식이 기준 : 8*8 2-Dimension
 - DCT 와 Movement Compensation : 16*16 Block Size
- 압축비율 목표 : 수백대 1 (80-275:1, 1.5 MB/sec)
- 개발된 예 : C-Cube Microsystem 사의 Chip/Board
(768KB → 4KB, 30 Frames/sec, 가격 : \$ 1,000)
- 표준안 확정시기 : '92 년 초 목표

(2) JPEG(Joint Photographic Expert Group) 표준화 방안

- ISO/CCITT 표준안
- 대상 데이터 및 목적 : 정지화상 (이미지) 의 전송을 위한 압축 / 복원 기술
- 압축 알고리즘 : ADCT(Adoptive Discrete Cosine Transform) 방식
- 압축비율 목표 : 8-100:1
 - RGB 는 10:1 - 20:1
 - YUB 는 25:1 - 80:1
- 개발된 예 : C-Cube Microsystem사의 Chip/Board (768KB → 4KB, 30 Frames/sec, 가격 : \$ 1,000)
- 표준안 확정 시기 : '90년 하반기 목표 (1차)

(3) Px64 표준화 활동

- 공식적인 명칭은 H261
- CCITT 표준안
- 대상 데이터 및 목적 : ISDN을 통한 영상회의 및 비디오 전화에 사용, 저가격, 저급의 비디오 이미지 전송
- 압축 알고리즘 : DCT 방식
- 압축비율 목표 : 100-300:1
- 표준안 확정 시기 : '91년 중반 목표

다. MHAC(Multimedia and Hypermedia Information Coding Ad-Hoc Group) 표준화동향

(1) 표준화 목적

- 다중매체 / 하이퍼미디어 용어의 정의 및 코딩 원칙

- 다중매체 오브젝트와 상호작용을 정의하는 규칙 / 메타언어의 정의 등

(2) 표준화 대상

- 다중매체 / 하이퍼미디어 데이터의 저장, 관리
- 하이퍼미디어 시스템을 포함한 사용자 인터페이스 기능의 표준화 방안
- 다중매체 데이터를 포함한 문서, 자료의 통신

(3) 표준안 확정 시기 : 다중매체는 '92년 말, 하이퍼미디어는 '93년 말 목표

(4) 참여 회원 : CCITT, CSELT, NTT, IBM, JVC, Philips, Fujitsu, Star Signal

라. 다중매체 통신관련 표준화 추진 단체

(1) CCITT : FDDI, BISDN 의 표준화

(2) IEEE

802.2 : Local Link Control

802.3 : CSMA/CD

802.4 : 광대역 서비스

802.6 : DQDB MAN 표준안 작업

제3절 다중매체 시스템 개발 현황

80년대 중반 다중매체 시스템의 개념을 처음으로 도입한 미국의 뒤를 이어 최첨단의 전자기술을 가진 일본에서 현재 활발한 연구가 진행중이다. 특히 일본의 기업들은 소니를 중심으로 다중매체 시스템 콘소시움을 결성하여, CD-ROM 뿐만 아니라 ISDN을 통하여 오디오와 비디오가 전달될 수 있도록 압축 알고리즘의 표준을 개발하고 있고 다중매체 시스템 소프트웨어 보다는 장비에 더욱 신경을 쓰고 있으며, 이러한 부문에서 미국보다 우세를 보일 전망이다.

또한, 다중매체 시스템으로서는 매킨토시에 뒤지던 IBM PC 계열은 Microsoft사의 윈도우 3.0에 매킨토시에서 사용하고 있는 다중매체 시스템용 소프트웨어들이 이식되고 있어서 수년내에 매킨토시에 뒤지지 않는 다중매체 시스템 환경이 구축될 것이다.

다중매체 시스템 개발에 박차를 가하고 있는 업체들은 생산성 향상 및 편리한 사용자 인터페이스에 대한 요구에 부응하여 컴퓨터 처리 속도의 향상, 광디스크를 비롯한 대용량 저장매체의 개발, 멀티윈도우, 이미지 스캐너, 디지털라이저 등의 입출력 매체의 개발, 광통신과 ISDN 등의 통신기술의 개발 등을 끊임없이 경주하고 있다.

여기서는 다중매체 시스템의 국내 동향과, 연구개발이 진행중인 주요 업체들의 기술개발 현황 및 기업의 전략을 살펴본다.

1. 국내 현황

다중매체 시스템의 관련 기술은 그야말로 첨단기술로 기술 개발에 막대한 비용을 감수하여야 한다. 따라서 선진국은 계속해서 그들의 자

본력을 바탕으로 기술을 발휘할 수 있지만, 후진국들은 기술의 낙후성으로 이미 선진국에서는 보잘것 없는 기술이나 노하우(Knowhow)에 의존하고 있는 것이 사실이다.

현재 국내의 다중매체 시스템에 대한 연구는 그 역사가 짧고 기술수준이 미약하다. 이제 정부, 학계, 기업체에서 연구를 시작하는 단계이기 때문에 우리나라 다중매체 시스템에 있어서 문제점은 역시 기술력의 부족을 들 수 있다. 우리나라 하드웨어 업체들을 보면 그런대로 세계적인 흐름에 따르고 있으나, 자체적으로 제품 생산을 하기보다는 외국의 제품을 그대로 수입해서 판매한다든지 또는 조립공정만을 거치는 경우가 대부분이다. 이와같은 하드웨어의 대외 종속은 하드웨어 시장의 침체뿐만 아니라 소프트웨어 종속으로 직결된다. 우리나라에서는 CD-ROM을 자체 생산하지 못해 외국 CD-ROM에 의존하고 있는 형편이지만 일부 대기업에서는 자체 제작하려 하고 있는 등 하드웨어 개발을 서두르고 있으며, 미래형 컴퓨터를 순수하게 우리 기술로 구현해 보려는 시도로 음성과 영상, 그래픽 등을 동시에 처리하는 다중매체 시스템의 개념을 추구하고 있다.

정부차원의 정책의 일환으로 상공부는 공업기반 기술조성책으로 산학연 공동으로 다중매체 시스템 구축을 위한 다중매체 데이터베이스, 사용자 인터페이스, 고성능 워크스테이션 등의 중요 과제들을 연구하게끔 육성하고 있고, 대기업 뿐만 아니라 중소규모의 다중매체 시스템 전문업체들이 속속 출현하고 있다.

국내의 업체들은 교육용 저작도구를 자체개발하여 발표하고, 컴퓨터에 음향을 첨가, 전자출판에 이미지 음성 등을 삽입, 영국의 비디오로직사, 미국의 매스마이크로 시스템사 등의 외국업체와 제휴등을 통

하여 다중매체 시스템의 응용 프로그램을 개발하고 있으며 다중매체 시스템 전용 퍼스널 컴퓨터의 독자적인 개발에 나서고 있는 등 다중매체 시스템에 많은 관심을 표명하고 있다.

2. 해외 현황

한 보고서에 따르면 다중매체 시스템은 연간 80%가 넘는 성장률을 기록하리라는 예상을 하고 있다. 이에 따라 전자업체들은 연구 개발에 심혈을 기울이고 있다.

〈표 2-16〉은 다중매체 시스템 참여업체 현황을 약술한 것이다.

〈표 2-16〉 다중매체 시스템 참여업체 현황

| 회 사 | 핵심 기술 목표 |
|-----------|--------------------------------------|
| Apple | 매킨토시 기종의 그래픽 기능에 비디오와 하이파이 사운드 기능 추가 |
| IBM | 인텔의 DVI 기술을 응용, 교육 및 판매시장 공략 |
| Intel | PC에 비디오 기능을 부여하는 DVI 기술 자체 개발 |
| Microsoft | 다중매체 시스템의 표준을 설정하고 제반 기술 확보 |
| Commodor | PC로 비디오 정보와 하이파이 오디오 처리 |
| NeXT | 하이파이사운드 기능을 갖추고 광디스크가 내장된 워크스테이션 설계 |

(다음 페이지에 계속)

(< 표 2-16 > 계속)

| 회 사 | 핵 심 기 술 목 표 |
|---------|--|
| Philips | 소니와 공동으로 CD-I 표준을 이용한 인터랙티브 비디오 시스템 개발 |
| Fujitsu | CD 플레이어가 내장된 FM-Towns PC 판매 |
| NEC | 비디오 게임기, TV, 컴퓨터를 일괄 생산 |

가. Commodore 사

다중매체 시스템의 선구자격인 Commodore 사는 모토로라 칩을 이용하여 값싼 가정용 다중매체 시스템 Amiga 를 개발하였다. Amiga 는 그래픽 처리를 용이하게 하는 그래픽 및 비디오 코프로세서, 표준비디오 신호들의 속도에 쉽게 동조할 수 있는 디스플레이 회로를 장착시켜 사용자의 취향에 따른 주문식 생산을 하고 있다.

비교적 저렴한 Amiga 하드웨어는 그래픽, 애니메이션, 사운드 까지 능숙하게 처리하며 다양한 매체들을 혼합, 사용할 수 있는 멀티태스킹 운영체제를 사용하고 있으며 dBASE 와 호환되는 관계형 데이터베이스로 구성되어 있다.

Amiga 는 그래픽 코프로세서와 비디오를 동조시키고 오버스캔할 수 있는 디스플레이 하드웨어, 스테레오 사운드 및 음성합성 회로 등의 다중매체 시스템을 위한 하드웨어를 갖추고 있다. 다만 그래픽 디스플레이의 빈약한 해상도와 칼라수, 멀티태스킹 메모리를 위한 하드웨어 제품의 부족, 동작 비디오를 위한 압축기능의 결여 등의 단점이 있다.

< 표 2 - 17 > 은 Amiga 에서 사용되는 하드웨어 및 소프트웨어를 기능별로 나타낸 것이다.

〈 표 2 - 17) Commodore 사의 Amiga 에서의 하드웨어 및 소프트웨어

| 하 드 웨 어 | 기 능 |
|--|--|
| 디스플레이 회로 <ul style="list-style-type: none"> — Noninterlaced — Interlaced — Overscan | <ul style="list-style-type: none"> • 비디오 신호 속도에 동조 • 한줄씩 건너 디스플레이한 다음 나머지 줄 디스플레이 • 해상도 768*480 • 화면의 가장자리를 벗어나 사진을 확장 |
| 68000 프로세서 512 KB RAM | 아미가 500 |
| 68020 프로세서 IBM RAM 확장 슬롯 | 아미가 2500 |
| 68030 애드인 모드 (25MHz) (수치 코프로세서, 2MB 32Bit RAM) | |
| Agnus (그래픽 코프로세서) Paula (다중 채널 스테레오 사운드에 관여) - 영어 음소, 사운드, 음성합성 하드웨어 포함 Denise (애니메이션) | <ul style="list-style-type: none"> • 주문형 칩 • 비디오, 오디오 입출력 (스테레오 사운드 4개 채널 핸들링) |
| 모니터의 수직 Blanking간격 제어 칩 | 주파수 호환성 제공 |
| 젠록 박스 | 비디오 신호를 직접 편집 |
| 애드인 고해상도 디스플레이 어댑터 | 256 칼라 디스플레이 |
| 디지털타이저, 샘플러 | 음악 및 사운드 주변장치 |

(다음 페이지에 계속)

(< 표 2-17 > 계속)

| 소 프 트 웨 어 | 기 능 |
|--|---|
| 멀티태스킹 운영체제 | |
| Digi-Paint (New Tek 사) Deluxe Paint III (Electronic Arts 사) | Painting Package |
| Music-X (Microillusions 사) Audio Master II (Oxxi 사) | 오디오 작업 |
| Photon Cel Animator (Microillusions 사) | 애니메이션 프로그램 |
| Digi-View (New Tek 사) | 실물영상을 그래픽으로 처리 |
| VIVA (Michtron 사) | 다중매체 시스템 창출 기능 |
| Pro Video CCI (Shereff Systems 사) | 테이프에 타이틀이나 로고 첨가 |
| 단일 교환 화일 포맷 표준 | 화일 편집 |
| 아이콘을 사용한 프로그래밍 언어 | 매체들과 응용 프로그램을 통합, dBASE와 호환되는 관계형 데이터베이스로 구성 |

나. Apple 사

Apple 사는 1984년 매킨토시 플러스가 발표되면서 기본으로 마우스를 제공하며 그래픽을 자유자재로 사용할 수 있도록 하였고, GUI (Graphic User Interface) 기능을 제공하여 80년대 후반에 활발하게 사용된 GUI의 시초가 되었으며, 매킨토시를 기반으로 다중매체 시스템을 구현하고 있는데 하이퍼카드 소프트웨어와 뛰어난 사용자 인터페이스를 바탕으로 타업체들보다 앞서서 개발하고 있다. MAC-II

fx에서는 DMA(Direct Memory Access) 기능을 첨가하였고, MIT의 Media Lab과 공동으로 비디오와 움직이는 화상의 압축기술을 연구하고 있다.

Apple사는 다중매체 시스템을 논리적 2 단계로 구분하고 있는데 현재의 데스크탑 퍼블리싱과 데스크탑 기능에 고품위의 사운드, 생동감 있는 비디오, 애니메이션을 추가하는 것으로 간주하고 있다. 이러한 2 단계 요소를 하나로 통합하는데 하이퍼카드를 사용하고 있는데 하이퍼카드는 단일 데이터베이스, 하이퍼텍스트 프로그램, 프로그래밍 언어, 자유롭게 만들어진 패키지등의 요소를 하나로 묶는 교량 역할을 한다. 하이퍼카드는 자체 데이터베이스가 정교하지 못하지만 모든 매킨토시 데이터 포맷과 호환성을 지니고 있으며, 광학 스캐너 및 CD-ROM 드라이버 등에 직접 링크될 수 있다.

또한 현재 비디오 디스크, CD 오디오 디스크, 비디오 테이프 등의 정보를 액세스하기 위한 AMCA (시스템 레벨의 아키텍처 혹은 프레임 워크)를 규정하는 작업을 진행중인데 만약 AMCA를 사용할 경우 개발자는 각 다중매체 시스템 주변장치마다 주문형 드라이버를 제작할 필요가 없다.

Apple사는 다중매체 시스템 연구소를 활성화 시키고 개발툴에 관한 카타로그(Wings For The Mind)와 코스웨어 제작에 관한 안내서를 출판하고 있으며 장래에 실시간 비디오를 지원할 수 있는 운영체제 기반과 드라이버 표준을 확립하는데 많은 노력을 기울이고 있다.

Apple사는 그동안 신제품에 대한 전략으로 고가격, 고 성능 위주의 신제품만을 고수하였으나 신제품 전략을 180도 회전하

여 고부가 가치에서 소형화, 다기능화하여 시장점유율 위주로 급선회함에 따라 90년대 시장을 겨냥한 Mac II si 와 Mac II Classic 등의 저가격 모델들을 선보였다.

Mac II si 는 시스템 기능을 확장시키고 시스템을 안정시키기 위하여 모든 것을 커스텀칩을 사용하여 디자인 하였고, 다중매체 시스템 시장을 주도하기 위하여 별도의 장치가 필요없이 각종 형태의 소리를 입력, 저장, 재생할 수 있는 기능을 기본적으로 제공하며, MACE (Machintosh Audio Compression Expansion) 유틸리티를 제공하여 사운드 데이터를 3:1, 6:1 Mode 로 압축하여 저장이 가능 (40MB의 하드 디스크에 3시간의 사운드를 저장) 하며, "Apple Talk" 프로토콜을 지원하는 LAN 카드가 내장되어 있어 별도의 장치 없이 LAN 구성이 가능하도록 하여 1995년까지 전세계 퍼스널 컴퓨터 시장의 30% 점유를 목표로 하고 있다.

〈표 2-18〉은 매킨토시에서 사용되는 하드웨어 및 소프트웨어를 기능별로 나타낸 것이다.

〈표 2-18〉 Apple사의 매킨토시에서의 하드웨어 및 소프트웨어

| 하 드 웨 어 | 기 능 |
|--------------------------------------|--------------------|
| 68030 프로세서, 8MB RAM, 256 칼라 디스플레이 | |
| Color Space FX (Mass Micro System 사) | 칼라 비디오 영상을 포착해서 수정 |
| Micro TV (Apple 사) | 흑백 비디오를 윈도우상에 표현 |

(다음 페이지에 계속)

(< 표 2-18 > 계속)

| 하 드 웨 어 | 기 능 |
|--|-----------------------------------|
| Video-Overlay Card(Apple IIe, IICS 용) | Interlaced 안됨 |
| MacRecorder | 사운드를 포착 디지털화시켜 응용 프로그램에 활용 |
| CD-SC | CD-ROM 드라이버 |
| Audiomedia 애드윈 본드, Hyper-card 호환 소프트웨어 패키지 | 오디오 활용 가능케 함 |
| Voice Navigator (아티클레이트 시스템사) | 음성인식 시스템 |
| MidiDancer | 위치 센서, 라디오 송수신기, 움직임을 감지해 음악으로 변환 |

| 소 프 트 웨 어 | 기 능 |
|--|------------------------|
| MacPaint, MacDraw, Studio/8 (Electronic Arts 사) | Drawing, Modeling Tool |
| Swivel 3D (Paracomp 사) | 애니메이션을 위한 3차원 물체 생성 |
| MacRecorder, ScreenRecorder (Farallon Computing 사) | 소리, 영상 포착 |
| 비디오 편집시스템 (Avid Technology 사) | 질이 떨어짐 |

(다음 페이지에 계속)

(< 표 2-18 > 계속)

| 소프트웨어 | 기능 |
|--|--|
| Authoring Tool -- Storyboarder (American Intelliware사) - MediaMaker (Apple 사) | Macpaint 영상을 이용 흑백 애니메이션 그래픽, 오디오, CD 오디오 등을 10 분간 배열 가능 |
| Course of Action, Authorware professional(Authorware 사) | |

다. IBM사

IBM사는 1980년대 초부터 다중매체 시스템 제품을 내놓기 시작하여, 오늘날 복잡한 프리젠테이션 그래픽 시스템, 정지화상과 동화상 비디오, MIDI(Music Instrument Digital Interface), 음악 제어, 각종 사용자 인터페이스 장비등을 지원한다.

IBM사는 1981년 퍼스널 컴퓨터용 StoryBD(Four-color CCA)의 발표를 시작으로 1984년 부터 1986년 사이에 EGA에서 오디오, 비디오 등을 포함한 다중매체 시스템 제품과 NTSC 방식 비디오로 부터 동화상 비디오와 스테레오 오디오를 결합하여 Play 하는 비디오 디스크 플레이어인 InfoWindow 를 발표하였으며, 1987년에는 오디오 제어와 생성을 가능하게 하는 Music Feature Card, VGA, 8514A 그래픽 어댑터, VGA 용 StoryBD plus 를, 1988년에는 하이퍼미디어 응용 프로그램 제작을 위한 편집, 프리젠테이션 툴인 LinkWay 를, 1989년에는 VCA(Video Capture Adapter/A), ACPA(Audio Capture and Playback Adapter), StoryBD plus 2.0 을 발표하였으며, 또한 여러 매체를 어느 데이터베이스에 있던 간에 효율적인 객체지향 방식으로 연결할 수

수 있는 개발도구인 AVC(Audio Visual Connection)을 제공하였다. AVC는 주기억공간이 2.5MB이며 IBM의 비디오 포착 및 오디오 어댑터 카드를 장착한 PS/2의 OS/2나 MS-DOS 4.0에서 실행되는데 음성과 영상을 포착하여 편집할 수 있고 작품을 만들때 특수효과를 첨가시킬 수 있고 이미지위에 텍스트를 중첩시킬 수 있으며 전문가 시스템인 Knowledge Tool을 사용하면 AVC는 다른 응용 프로그램도 불러 이용할 수 있다. 1990년에는 M-Motion Video Adapter/A, M-Control PGM, Action Media 750 Series, LinkWay 2.0, XGA 그래픽 어댑터, 그리고 M-MVA/A와 MIDI Music File Play 기능을 추가한 AVC 1.03을 발표하였다.

또한 IBM사는 다른 업체들로 부터 특수한 수십종의 비디오 어댑터 카드, 고속 고해상도 디스플레이용 그래픽 코프로세서들을 공급받고 있으며 소프트웨어에서는 기본 DOS가 그래픽 위주가 아니고 표준 그래픽 화일 포맷도 갖추지 않았지만 마이크로 소프트 윈도우로 이를 보완하고 있다.

미국내의 프로그램이 가능한 비디오 디스크 플레이어들은 대부분 IBM 퍼스널 컴퓨터나 호환 기종을 제어하도록 구성되어 있다.

IBM사의 다중매체 시스템에 대한 전략은 다음과 같다.

첫째, 시스템은 반드시 상위기종과 호환성이 있어야 한다. 실제로 3090 메인 프레임이 다중매체 시스템 화일 서버가 될 수 있다는 것이다.

둘째, 기술은 쉽게 통합할 수 있도록 모듈화 되어야 한다.

셋째, 최대 기능은 마더보드로 합쳐져야 한다.

넷째, 시장은 전세계적이어야 하므로 비디오의 경우 전세계의 표준을 모두 지원한다.

<표 2-19>는 IBM PC/AT와 PS/2에서 사용되는 하드웨어 및 소프트웨어를 기능별로 나타낸 것이다.

< 표 2-19 > IBM사의 PC AT 와 PS/2 에서의 하드웨어 및 소프트웨어

| 하 드 웨 어 | 기 능 |
|---|---|
| MIDI Card, Digitizer (Microsoft 사) | DOS 상에서 그래픽 지원 |
| PS/2 : VGA, 8514/A Display Adapter | 고해상도 (1024 * 768 Resolution) |
| Video Charley (Progressive Image Technology 사) | 젠록, 인코더, 문자발생기 |
| Targa 보드 (Truevision 사) | 비디오 편집 |
| Willow Peripherals 사 VGA 계열 (VGA Card 의 일종) | VGA 출력을 NTSC 비디오로 변환 하여 TV나 비디오 테이프에 출력 |
| DVA 계열 Digital Video Adapter (Video Logic 사) | 그래픽과 비디오 영상을 Real Time 지원 |
| Double Take AV Audio/Video Digitizer (Add-in Board) (Logos Systems International 사) | 비디오, 오디오 입력을 디지털화 하여 압축 |
| M-Motion Video Adapter/A | Still-and Full-Motion Video, Two line Audio Source 지원 NTSC, PAL, S-VHS Source Support 다른 응용 프로그램과 통합 가능 윈도우 3.0 제어 |
| Video Capture Adapter/A (VCA/A) | NTSC, RGB Source에서의 비디오를 정지 화상으로 저장 |
| Action Media 750 Series | DVI 용 어댑터 |

(다음 페이지에 계속)

(< 표 2-19 > 계속)

| 하 드 웨 어 | 기 능 |
|------------------------|---|
| Info Window | NTSC 비디오로 부터 동화상 비디 오와 스테레오 오디오를 결합하여 실행 가능한 비디오 디스크 플레 이어 |
| Music Feature Card | 오디오 제어와 생성 |
| XGA Graphics Adapter/A | 8514A 보다 10 배의 속도 |

| 소 프 트 웨 어 | 기 능 |
|---------------------------------------|---|
| InfoWindow(IBM) | 비디오 디스크 플레이어 |
| Guide(Owl) | 윈도우, 그래픽, 한정된 애니메이션 기능 |
| Authology(Intel사) | DVI 용 다중매체 시스템 창작 패키지 |
| AVC(Audio Visual Connection : IBM) | 음성, 영상 편집, 특수효과 첨가시 키는 기능, Knowledge 툴(IBM 전문가 시스템 패키지)를 이용 다른 응용 프로그램 사용 가능, 저작도구 언어(AVA) 포함 |
| AVC 1.03 | M-MVA/A, MIDI 화일 실행 기능 |
| Plus(Spinnaker 사) | Mac, OS/2, 윈도우 3.0 하에서 실행 가능 |

(다음 페이지에 계속)

(< 표 2-19 > 계속)

| 소 프 트 웨 어 | 기 능 |
|--|--|
| Autodesk Animator(Autodesk 사 & Auto CAD사) | 애니메이션 프로그램, 자동적으로 대상물의 모양 변형 |
| LinkWay | Hypermedia 제작을 위한 편집, 프리젠테이션 툴 |
| StoryBD plus 2.0 | Video Capture Adaptor/A, Audio Capture and Playback Adaptor 지원 |
| InfoWindow Presentation System(IWPS) | 저작도구, 프리젠테이션 시스템 |
| Learning System/1(LS/1) | 교육과 훈련 지원 저작도구, 프리젠테이션 시스템 |

라. Intel 사

Intel 사가 GE 와 함께 개발한 DVI(Digital Video Interactive) 는 오리지널 비디오 화일의 크기를 1% 정도로 압축시킬 수 있는 압축 / 복원 장치로서 1시간짜리 디지털 풀스크린, 풀모션 비디오 (매 초당 30 프레임) 를 한개의 표준 CD-ROM 에 담을 수 있다. 따라서 움직이는 활동 비디오를 퍼스널 컴퓨터 화면상에 디스플레이 시켜 이를 텍스트 및 그래픽과 합성시킬 수 있다. 이렇게 생성된 비디오는 오리지널 비디오와 똑같은 질을 유지하지는 못하지만 퍼스널 컴퓨터 상의 풀스크린, 풀모션 비디오를 접속시킬 수 있다.

이러한 DVI 용으로 독특하게 설계된 다중매체 시스템 패키지인 Authology 를 발표하였으며, 89년 10월말에는 RTV(Real-Time Video) 라는 새로운 DVI 소프트웨어를 발표하였다. 이는 풀스크린, 매초 30

프레임으로 VCR의 화면이 제공하는 수준의 압축/복원이 가능하다.

마. SUN사

SUN Microsystems사는 기존의 워크스테이션 기술을 기반으로 하여 다중매체 데이터를 다룰 수 있는 보다 사용하기 좋은 인터페이스에 중점을 두고 있다. SUN은 프린스턴의 David Sarnoff Research Center, Texas Instrument사와 공동으로 DARPA (Defense Advanced Research Projects Agency)의 지원을 받아 고도의 다중매체 시스템 워크스테이션의 개발계획에 착수하였다. 아직은 다중매체 시스템 워크스테이션을 위한 특정한 계획은 없지만 워크스테이션의 기능을 향상시켜 다중매체를 현재 통신망을 기반으로 하는 환경으로 통합하고자 계획을 수립하고 있다.

또한 고품위 TV 수준의 고해상도 모니터를 이용하여 이를 바탕으로 문자, 그래픽스, 비디오 등의 개발에 주력하고 있으며 이는 소프트웨어나 오디오 보다는 HDTV 등의 기술이 워크스테이션 구조에 맞도록 새로운 버스 구조의 개발등에 초점을 맞춘 것이다.

바. HP사

HP사는 89년에 객체지향적이며 모든 응용 소프트웨어간의 데이터 전달이 가능한 NewWave라는 프로그래밍 환경을 발표하였는데 이는 비디오, 오디오, 정지화상 등의 새로운 데이터 유형들을 통합할 수 있다. 실제로 여러 하드웨어 업체들이 NewWave 환경에서 비디오 윈도우를 띄울 수 있는 하드웨어를 개발 공급하고 있는데 미국의 New Media Graphic사와 영국의 Videologic사가 그 대표적 업체이다.

AIMS(Advanced Image Management System)은 손으로 쓴 문서를 포함한 종이를 바탕으로 둔 정보를 디지털화하고 저장을 위해 영상

을 압축할 수 있는 제품이다.

사. Sony/Philips 사

다중매체 시스템의 오디오, 비디오 파일들을 저장하려면 광학 저장 장치가 필요한데 이 분야는 Sony, Philips 두 업체가 높은 시장 점유율을 보이고 있다.

Sony와 Philips의 다중매체 시스템에 대한 전략은 다음과 같다.

첫째, 이들은 다중매체 시스템에 필요한 광 디스크 드라이브를 공급하기 위해서 IBM, Apple, Commodore 등의 여러 업체들과 제휴하고 있다.

둘째, 컴퓨터 없이도 다중매체 시스템을 제공할 수 있는 기술인 CD-I(Compact Disc Interactive)를 추구하고 있다. 이것은 CD-I가 소비자에게 빨리 접근할 수 있다고 판단하였기 때문이다.

셋째, 비디오 회의와 같은 새로운 시장을 창출하기 위해 새로운 오디오, 비디오 및 컴퓨터 기술들을 가전제품과 조화시키는 방법을 연구하고 있다.

특히 Sony사는 자사의 워크스테이션인 NEWS 시스템을 다중매체 시스템을 위한 CD-ROM 개발 시스템으로 발전시키기 위해 일회용 CD-ROM 제작기를 NEWS 시스템에 발표하였다. 또한 자사의 Discman CD 플레이어, Watchman 비디오 카세트 레코더를 결합한 다중매체 시스템도 개발중이다.

아. Fujitsu 사

Fujitsu사는 대표제품인 FMTOWNS의 가격을 '90년 10월 대폭 인하하고, 미국의 다중매체 시스템 소프트웨어 하우스인 Macromind사와 제휴하여 윈도우 3.0 위에서의 구현에 착수했다. 이 시스템은 CD-

ROM을 내장하고 있으며, 스테레오 CD 사운드, PCM, FM 사운드를 지원하며 MIDI 인터페이스가 가능하다. Towns GEAR 라는 하이퍼미디어 저작도구가 제공되며, 일본의 여러 소프트웨어 회사들이 100여개 이상의 소프트웨어를 제작하고 있다.

자. NEC 사

NEC 사는 자사의 PC-8801 MC 가 FMTOWNS 보다 가격 경쟁력이 강하다고 보고 있으며, 20여개의 소프트웨어 하우스들이 8801 MC 를 바탕으로 하여 개발한 소프트웨어들이 있다.

차. Pioneer 사

Pioneer 사는 레이저 디스크를 CD-ROM 형식으로 변환한 LD-ROM 을 선보였다. LD-ROM 은 LD 사양을 맞추면서 동시에 CD-ROM 형식에 맞추고 있어서, LD-ROM 사용자는 CD-ROM 을 사용하는 것처럼 사용할 수 있다.

제4절 향후 발전 방향

컴퓨터의 신기술 분야로 각광받기 시작하고 있는 다중매체 시스템분야는 데스크탑 비디오, 교육, 프리젠테이션, 맵핑, 자료관리, 시청각 통신, 데스크탑 인쇄(출판), 시뮬레이션 등 그 응용범위가 광범위하고 독립된 매체의 응용에서 발전하여 각 매체의 통합제어를 통하여 아직까지는 제약된 기술수준 때문에 정착되기에 좀 더 시간과 과제들이 필요하지만 벌써부터 많은 사람들의 호기심을 불러 일으키고 있다.

다중매체 시스템은 기존의 정보시스템에 비해 자료의 관리 및 통신 프로토콜 등의 측면에서 새로운 변혁을 요구하고 있다.

그 응용대상도 개인용 컴퓨터로부터 워크스테이션, 중·대형 컴퓨터에 이르기까지 확장되어 나갈 것이며, 기존 정보시스템과의 융합을 위해서는 다중매체 시스템의 표준화와 관련기술 개발에 박차를 가해야 한다. 인간의 다양한 정보욕구와 지적활동을 지원하기 위해서 보다 자연스러운 조작환경을 제공하려면 인공지능 기술과도 연계할 필요가 있을 것이며, 응용시스템 개발을 용이하게 하기 위한 저작환경의 제공도 중요한 역할을 담당할 것이다.

다중매체 시스템의 표준화는 크게 자료의 표준화와 통신의 표준화로 분류할 수 있으며, 자료의 표준화는 각 다중매체 자료의 종류에 따라 분리하여 독립적으로 표준화하거나 가능한 모든 종류의 다중매체 자료를 하나의 틀안에 넣어 표준화하는 방법 등 크게 두 종류 흐름으로 볼 수 있다.

그 접근방법중 하나는 객체지향적 방식으로써 사용자 인터페이스의

관점에서 객체와 상호작용을 정의하는 메타언어의 개발이라든지 각 매체별 자료간의 동기화를 피한다든지 텍스트 데이터와 바이트(Byte) 데이터를 동시에 취급할 수 있는 BLOB(Binary Large Objects)의 탄생 등이 그 노력이라고 볼 수 있다.

또한 궁극적으로 ISDN에 접목시키기 위해서는 통신의 표준화도 필요한데 OSI 모델에 있어 각 Layer 간의 엄격한 독립성 때문에 다중매체 통신에는 장애가 되고 있으며 현재의 통신 프로토콜이나 전송속도로 보아서는 음성정보의 실시간 전송이나 동화상정보의 전송은 거의 불가능한 상태이므로 통신시설의 개량을 통해 고속, 고품질의 통신망을 구축하여야 한다.

다중매체 시스템의 실현을 위해서는 표준화를 위한 노력뿐만 아니라 관련산업분야의 기술개발 또한 중요하다.

이는 다양한 데이터 형식을 취급할 수 있는 기술이나 자료 저장 기술, 오디오와 비디오 데이터의 저장, 전송 및 조작을 위한 데이터의 압축과 복원기술(실시간 처리 알고리즘 H/W화, 강력한 실시간 처리 프로세서 도입 포함), 다중매체 데이터 베이스 관리체계의 개발 등 주요 기술개발과 다중매체용 개인용 컴퓨터의 개발, 각종 S/W(저작용, 프리젠테이션용, 애니메이션용 등)의 개발 등이 라고 할 수 있다.

1. 관련산업 기술개발

지난 1990년 2월 샌프란시스코에서 열렸던 CD-ROM 회의 결과를 통해 관련산업 기술개발의 역사적 배경과 다중매체 관련산업 부문의 미래를 예측해 보면 <표 2-20>과 같다.

<표 2-20> 다중매체 관련산업의 역사적 배경과 미래

| 연 도 | 관련산업 기술개발 내용 |
|------|---|
| 1986 | <ul style="list-style-type: none"> • 정보의 과부하를 해결하기 위해 CD-ROM이 제안됨 • CD-I가 발표됨 |
| 1987 | <ul style="list-style-type: none"> • DVI가 발표됨 (David Sarnoff Research Center에서 개발, General Electric사와 계속 연구) |
| 1988 | <ul style="list-style-type: none"> • Microsoft사가 CD-ROM 제품인 Bookshelf를 도입 • Open Standards 토의 • Intel/Lotus/Microsoft사들이 DVI 표준에 상호 협조하기로 함. (Intel사는 마이크로 프로세서를 중점적으로 하고, Lotus사와 Microsoft사는 CD-ROM을 기초로 액세스하게하기로 함) • Intel사는 GE사로부터 DVI를 취득 비디오 원격회의와 다중매체간의 연결을 거의 인식 못함 |
| 1989 | <ul style="list-style-type: none"> • IBM사에서 Intel사의 DVI를 지지 • Intel은 Pro750개발 Platform을 발표 • 응용제품들이 개발되기 시작 • 낮은 비용의 복원환경이 요구됨 |
| 1990 | <ul style="list-style-type: none"> • 산업분야에 박차 (연합이 일어남). 2월에 Intel사에서 'Action Media 750' 이라는 Board를 내놓음. • CD-ROM 주제가 2배 이상이 됨 • IBM을 위한 낮은 비용의 Delivery Board (DVI Boards 의 Action Media Series) |

(다음 페이지에 계속)

(< 표 2-20 > 계속)

| 연 도 | 관련산업 기술개발 내용 |
|------|--|
| 1990 | <p>: SCSI Interface, Audio + Video DSP, 512 × 240 해상도, Motion MCA 혹은 AT 버스</p> <ul style="list-style-type: none"> • Olivetti사가 Intel사와 유럽용 Board를 발표 • 제작수준의 (Compressed) 비디오와 실시간 (Compressed) 비디오를 비교해 볼때 Offline Compression이 장비나 알고리즘이 더 좋기 때문에 압축을 더 잘함. • Offline DVI는 이제 TV 품질과 거의 유사 - VCR보다 더 좋음 • 제작수준의 비디오에 대한 '90년대 수요는 증가할 것으로 보임 • 실시간 비디오가 '90년 5월 i750 프로세서 상에서 작동 가능 <p>실시간 비디오는 VCR 품질만큼 좋지 않다.</p> <ul style="list-style-type: none"> • 새로운 Tool들이 유용하게 됨 <ul style="list-style-type: none"> - CEIT Systems, Inc.사의 저작용 소프트웨어인 "Autology : Multimedia" - Lumina - 프리젠테이션 저작을 위한 MEDIAscript • 비디오 압축표준 <ul style="list-style-type: none"> - ISO + CCITT - Visual Telephony Group. - ISO = JPEG - 정지화상 |

(다음 페이지에 계속)

(< 표 2-20 > 계속)

| 연 도 | 관련산업 기술개발 내용 |
|------|---|
| 1990 | <p>MPEG - 동작화상+오디오 : 다중매체 시스템에 대한 주요 표준화그룹</p> <ul style="list-style-type: none"> • Intel사와 표준 <ul style="list-style-type: none"> - 다중매체와 비디오 전화가 1990년대에 통합될 것으로 보임. • 알고리즘 개발 <ul style="list-style-type: none"> - Bellcore와 MPEG에 제안 - 프로그램이 가능한 진보된 비디오 프로세서 |
| 1991 | <ul style="list-style-type: none"> • i750b 프로세서 발표될 예정 <ul style="list-style-type: none"> - i750a보다 성능이 2배, 전적으로 프로그램 가능, i750a 크기의 절반, 호환성 증대, 가격이 싸고 제작수준 및 실시간 수준의 비디오가 모두 가능하며 새로운 표준을 다룰 수 있음. • Series 2 Action Media Board : 30% 낮은 비용 (1,000\$ 이하)으로 발표될 예정 현재의 Series 1 Action Media Board는 약 5,000 \$, 1989년 Pro 750 Board는 거의 10,000 \$ 이었음. |
| 1992 | <ul style="list-style-type: none"> • PC Motherboard상에 DVI설치 가능 (500\$정도로 가격 인하) • i750c는 MPEG의 최종 표준안과 호환성있게 될 것이며, 보다 더 광범위한 압축 알고리즘을 다룰 수 있을 것임. (즉, DCT, Subband Encoding, CCITT-JPEG) : 100-1,000 MIPS |

(다음 페이지에 계속)

(< 표 2-20 > 계속)

| 연 도 | 관련산업 기술개발 내용 |
|------|---|
| 2000 | <ul style="list-style-type: none">· 응용분야 : 가정 비디오게임, PC, 학교, 공학용 워크스테이션· 단일 칩에 통합 비디오 기능을 갖는 'Super Multi CPU PC' 가 개발될 계획임. |

DVI의 기본기술은 VDP Chip Set, 시스템 소프트웨어, 압축 알고리즘, 오디오 서브 시스템 등이라고 볼 수 있으므로 DVI Chip Set의 저가격화, DVI 응용 소프트웨어의 지원, 압축 알고리즘의 고성능화 등의 문제를 해결해야 한다.

특히 시스템 소프트웨어 측면에서 집적된 서브루틴들을 내장하게 하여 비디오-그래픽 그림 그리기, 선·다각형·원·타원·텍스트 등의 고속 도시, 음성·동화상의 플레이 백 지원, 2차원의 비디오·텍스트·이미지를 3차원의 그래픽 모델로 매핑하는 기능, 마이크로 코드 기능으로 확장된 라이브러리, 오디오 비디오 지원 시스템이라는 라이브러리를 통해 플레이백·엑세스 가능하게 동기화하는 기능 등 다양한 시스템 소프트웨어를 지원하도록 해야 한다.

2. 시장규모 및 전망

다중매체 시스템의 시장규모 및 전망에 대해서는 조사기관마다 차이를 보이고 있으나, 다중매체에 대한 기술조언 전문 컨설팅 업체인 Information Workstation Group은 오는 '94년 전세계 다중매체 시장은 가전제품으로 약 60억불, 업무용 프리젠테이션으로 44억불, 정보검색으로 34억불, 교육과 훈련 등으로 25억불 등 170억불의 규

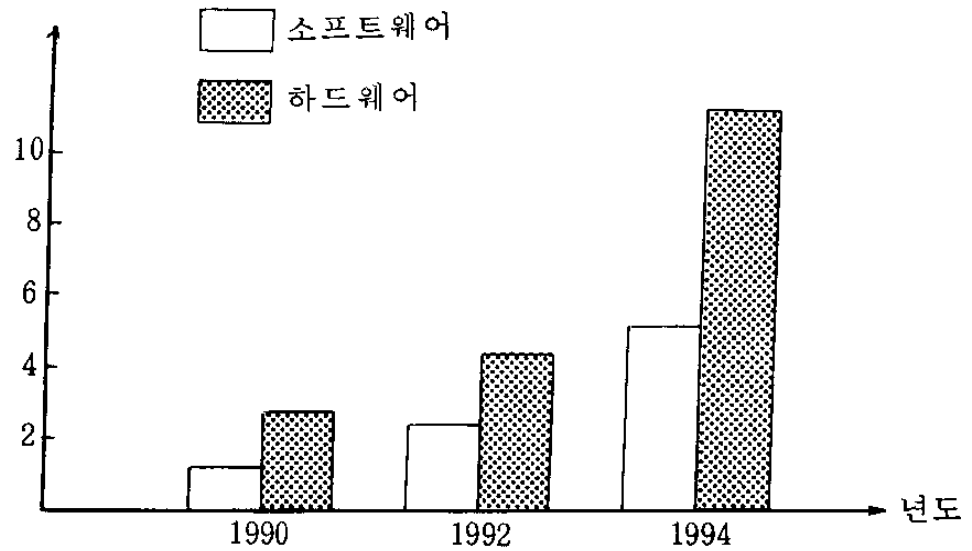
모일 것이라고 예측 (〈그림 2 - 4〉 참조) 하고 있으며, 시장조사기관인 DPI 사에 따르면 PC 다중매체 시장은 1993년에는 1백억달러 이상이 될 것으로 보고 있으며, Diversified Data Resources 사는 1994년에 100만대에서 200만대 가량의 다중매체용 PC가 팔릴 것으로 보고 있다. 이는 약 40억불에 해당하는 금액이다. 〈그림 2 - 5〉는 이 회사가 예측한 디지털 비디오와 관련한 개인용 컴퓨터의 판매 예측 자료이다. 따라서 컴퓨터 회사들은 80년대를 PC가 변화시킨 것처럼 다중매체 시스템이 90년대를 변화시킬 것이라고 예상하고 있다.

그러나 예전처럼 미국의 컴퓨터 업체들이 다중매체 시장을 장악하기는 어려울 것이다. PC역사의 초기에는 IBM사나 Apple사 등의 미국업체들이 가전제품을 장악하고 있던 일본 업체들과 조화롭게 발전할 수 있었다. 반면 지금은 Sony사나 NEC사 등의 일본업체들이 비디오 모니터나 광 디스크 등의 다중매체 기술을 가지고 있어 기존의 TV 기술과 컴퓨터 기술을 결합, 다중매체 시장에서 미국업체들보다 오히려 유리한 위치를 점할 수도 있을 것이다.

그동안 새로운 기술의 부족으로 PC 시장은 10-15% 성장에 그쳤지만, 이제는 소리와 비디오의 제어를 PC에 추가함으로써 업체들은 보다 많은 매출을 기대하고 있다. 또한 다중매체 관련 하드웨어 가격이 급속히 하락하고 있어 조만간 누구든지 다중매체 관련장비를 쉽게 구입할 수 있을 것이다.

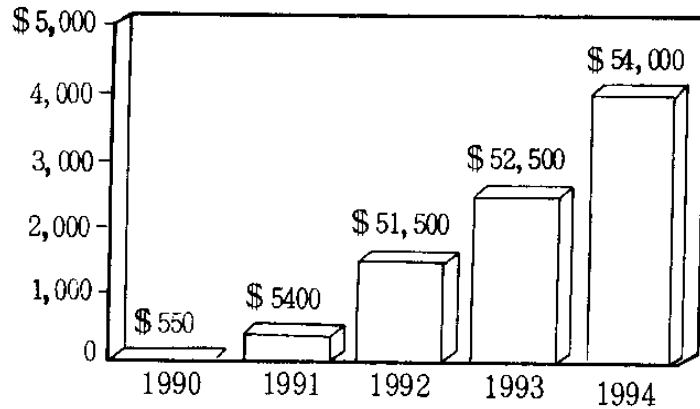
앞으로 CD-I를 대체할 것으로 여겨지는 DVI(Digital Video Interactive) 기술에 극복해야 할 기술적인 문제가 많다. 완전한 다중매체 시스템이 언제쯤 적정가격으로 대중화될 수 있을지는 미지수

지만 현재의 기술발전 속도로 미루어볼 때 머지않아 실현될 전망이다.



< 그림 2 - 4 > 전세계 다중매체 하드웨어 및 소프트웨어 시장 동향 (출처 : IWG)

(\$ millions)



Source: Diversified Data Resources Inc.

< 그림 2 - 5 > 디지털 비디오를 이용한 PC 시장

제 3 장 다중매체 데이터베이스 기술분석

제 3 장 다중매체 데이터베이스 기술분석

제 1 절 다중매체 데이터베이스의 개요

다중매체 데이터베이스란 기존의 데이터 관리, 데이터 모델 및 설계, 조작언어, 데이터 제어 등의 정형화된 기법 외에, 텍스트, 그래픽 비디오, 음성 등의 비정형화되고 복잡한 데이터를 효율적으로 처리할 수 있는 데이터 베이스라 할 수 있다. 다중매체를 지원하는 DBMS (Data Base Management System)는 저장해야할 자료의 성격이나 이들의 복잡한 관계설정, 그리고 윈도우에 기반을 둔 편리한 사용자 인터페이스 등을 제공해야 하기 때문에 정형화된 자료만을 다루는 기존 DBMS의 기능으로는 매우 부족하다.

우선 다중매체에서 다루는 데이터는 용량이 매우 크고 가변적이다. 또한 단순한 형태의 레코드로 표현하기 어려운 구조를 가지며, 검색이 어렵고, 데이터의 구성요소가 특정 데이터 형식으로 규정하기가 어려운 경우도 많다. 그리고 초기의 데이터베이스 프로그램은 절차적 언어를 바로 이용하여 작성되었으므로 데이터베이스 환경과 프로그램 환경이 별도로 존재하였고, 그래서 프로그램 언어 자체가 데이터베이스의 고유한 특성과는 무관하게 개발되어 결합 불일치 문제를 야기하였다. 또한 사용자가 데이터베이스의 내부구조와 질의어의 문법을 알아야 하는 불편이 있었다. 따라서 개발될 시스템의 성능을 높이기 위해서는 다양한 저장 구조와 그래픽, 메뉴등을 이용한 편리한 사용자 인터페이스, 손쉬운 프로그래밍 등의 특성을 갖

는 새로운 다중매체 데이터베이스 시스템을 개발하는 것이 합당하다. 이를 위해 국내, 국외의 다중매체 데이터베이스 시스템의 환경을 분석하여 요구사항을 분석, 종합한 후 이를 만족하는 시스템을 개발하여야 한다. 다중매체 데이터베이스의 연구 방향은 대체로 다음과 같은 변화요인에 기인한다.

— 데이터의 다양화 : 레코드, 텍스트, 이미지, 음성, 지식, 복합객체

— 네트워크의 발전 : 퍼스널컴퓨터 또는 워크스테이션, 분산처리

— 하드웨어의 발전 : 광 디스크, 비디오 디스크, 스캐너, 고해상도 출력 장치

— 기능강화 요구 : 추론, 계승기능, 능동적 처리기능, 사용자 인터페이스

— 응용영역의 다변화 : CAD/CAM, CASE, OA, FA, POS, DSS, 데이터뱅크
위와 같은 변화 요인을 고려하여 다중매체 데이터를 개발, 관리하기 위해서 다중매체 데이터베이스 시스템은 다음과 같은 요건과 기능을 만족하여야 한다.

1. 다중매체 데이터베이스 시스템의 요건

— 현실세계의 자원들을 표현하고 관리할 수 있는 모델이 필요하다.

— 매체의 차이에 관계없이 다중매체 데이터베이스의 구조를 통일적으로 정의할 수 있어야 한다.

— 다중매체 데이터들은 특성상 특별한 지원을 통해 효과적으로 처리할 수 있어야 하고 그 양이 방대하기 때문에 데이터를 최대한 공유하여야 한다.

- 다중매체 데이터에 포함되는 정보를 서로 관련시킬 수 있어야 한다.
- 다중매체 데이터의 내용에 대한 검색이 가능하여야 한다.
- 다중매체 데이터를 관리하는데 관련된 시스템들을 잘 사용할 수 있어야 한다.
- 비정형화된 데이터에 대한 질의에서 필요한 조건을 명확하게 작성하기 어렵기 때문에 점차적으로 세부조건을 제시하고 데이터의 범위를 줄여 나갈 수 있는 방법을 연구하여야 한다.
- 내용결합을 위한 내용정보의 세분화와 세분화된 정보를 결합시킬 수 있어야 한다.
- 시스템의 확장이 용이하여야 한다.
- 동시 처리, 버전관리, 에러로부터의 복구, 자료에 대한 보안 등의 기능이 있어야 한다.
- 질의를 잘 표현하기 위해서 강력한 표현 기능과 브라우징(Browsing) 기능이 요구된다.

2. 다중매체 시스템 자원 관리 기능

시스템 자원은 크게 하드웨어 자원, 각종 프로세서, 다중매체 데이터, 각종 Event 등으로 분류될 수 있다. 위의 시스템 자원중 하드웨어 및 Event 에 대한 시스템 관리 기능은 기존의 시스템에서와 유사한 형태로 처리될 것으로 생각되나 데이터에 대한 처리가 많은 시스템에서는 그 관리 기능이 달라져야 한다. 즉 하나의 다중매체 데이터에 대하여 각 종류별로 데이터 및 프로세서를 적절히 할당해주

고, 이들 각각의 데이터 및 프로세스들 사이에 동기를 맞추어 주어야 한다.

3. 다중매체 데이터의 표현 특성

다중매체 데이터베이스 시스템이 관리하여야 하는 데이터는 크게 다 음과 같이 분류할 수 있다.

- 텍스트와 정형화된 데이터

- INGRES, SYSTEM R 과 같은 관계형 및 계층형 모델에 기초한 데이터베이스 시스템은 정형화된 데이터에 적합하다.
- 효율적인 액세스를 위하여 B-Tree, Hashing 등의 인덱스 기법을 사용한다.

- 오디오 데이터

- 고음질 엔코딩(Encoding) 기법은 많은 기억공간이 필요하다.
- 음악신호는 음성신호보다 더 규칙적이며 구조적인데 이것은 High-Level 이 가능함을 의미한다.

- 이미지 데이터

- 이미지 데이터 저장을 위해 Pixel-oriented, Quardtrees, R-tree, Vector-based 등의 데이터 구조가 제안됨.
- Bellcore 에서 다양한 응용 프로그램간의 비호환성 문제를 해결하기 위해 이미지를 수집하고 처리하는 패키지인 FIVE(Format Independent Visual Exchange) 가 개발됨.
- 그외 Tiff, Gif, G3, G4 Fax Format 등이 있다.
- 기억공간을 절약하고 통신성능을 향상시키기 위해 이미지 데이터의 압축, 복원기술이 개발되었다.

－ 풀모션 (Full Motion) 비디오 데이터

- 비디오 프레임과 관련된 오디오 신호를 포함하고 있는 아날로그 정보는 비디오 카세트와 같은 방법으로 저장될 수 있어야 하며 통합되기 위해서는 디지털화 하여야 한다.
- 대용량의 기억장치 공간과 빠른 액세스 속도를 요구한다.

4. 화일 관리 기능

음성 데이터, 이미지 데이터 그리고 텍스트 데이터를 각각 혹은 하나의 표현단위를 이용하여 저장하고 관리하게 된다. 이때 하나의 객체가 음성, 이미지, 텍스트의 복합적인 데이터로 구성되었을 때 이들의 데이터를 상호관련 유지시키면서 관리하는 방법에는 대체로 두가지 경우가 가능하다. 우선 각 매체에 해당되는 데이터마다 별도의 화일에 저장하고 이 화일들을 일종의 링크를 이용하여 서로 연관성을 유지하도록 관리하는 방법과 또 하나의 입력되는 데이터를 Time Slice에 해당하는 모든 데이터를 하나의 화일에 저장하고, 이렇게 만들어진 화일들을 시간의 축에서 링크를 연결하여 관리하는 방법이 있다. 전자는 검색할때에 보다 편리한 특성을 갖고 후자의 경우는 사용자 인터페이스에 출력하는 경우에 저장된 기본 단위별로 처리하면 되는 점이 유리하다.

5. 동기화 기능

저장된 다중매체 데이터를 다중매체 사용자 인터페이스에 출력하기 위해서는 서로 관련이 있는 각 매체의 데이터를 재조립하여 입력

당시의 형태를 그대로 유지하여 출력될 수 있도록 동기화시키는 것이 중요하다. 이 동기화를 구현하기 위해 데이터 화일의 내용을 동기화 링크에 따라 읽어내고 각 매체의 출력 장치마다 별도의 버퍼를 두어 동기화 신호에 따라 각 장치가 해당 데이터를 정상적으로 출력할 수 있도록 조정하는 기능이 요구된다.

6. 분산화일 시스템 관리

각 화일의 분산 상황을 유지하는 기능으로 다음 절에서 기술할 분산 데이터베이스 시스템에서 자세히 다루기로 한다.

7. 현존 데이터베이스의 기술적인 문제점

다중매체 데이터베이스 시스템은 메타 데이터 관리, 질문처리, 트랜잭션 관리의 기능을 수행하는 것은 종래의 데이터베이스 관리 시스템과 거의 동일하지만 객체지향적이기 때문에 사용자 인터페이스 및 기억구조를 포함하는 관리체계가 기존의 데이터베이스 시스템과는 크게 다르다.

현재의 데이터베이스 기술을 다중매체 시스템의 관점에서 고찰할 때 다음과 같은 문제점이 있다.

- 다중매체의 데이터 구조에 대한 정의방법이 명확하지 않다. 즉 다중매체 데이터베이스를 실현하기 위해서는 데이터의 구조인식이 필요한데 다중매체의 경우 종래의 문자, 수치 데이터에 비해 추상적이기 때문에 표현 데이터 모델의 확립이 어렵다.
- 다중매체 데이터를 조작할 수 있는 통일적인 액세스 방법이 없다.

- 다중매체 데이터베이스의 사용자 언어가 없고 사용자 인터페이스가 응용 시스템마다 제각기 개발되고 있다.
- 효율적인 다중매체 데이터의 기록방법이 명확하지 않으며 데이터의 압축기법 및 여러 매체를 결합시키기 위한 정보의 기록방법이 필요하다.
- 매체의 변환이 어렵다. 즉 임의의 도형을 가지는 화상으로부터 도형추출, 도형과 텍스트가 혼합된 화상으로부터의 텍스트 추출 등 패턴인식과 관련된 작업이 과제가 된다.

이상의 요건들을 만족시키고 문제점들을 보완하는 방향으로 새로운 다중매체 데이터베이스 개발에 그 연구 흐름이 조성되고 있는데 그중 몇가지를 살펴보면 다음과 같다.

- 관계형 데이터베이스의 확장
- 객체중심 데이터베이스
- 데이터베이스 네트워크와 분산 데이터베이스
- 하이퍼미디어 데이터베이스

제2절 관계형 다중매체 데이터베이스

관계형 다중매체 데이터베이스는 일반적인 관계형 데이터베이스를 확장한 것으로 관계 데이터베이스 레코드내에 있는 기존의 필드외에 전혀 또다른 필드를 구성하여 구조화된 커다란 다중매체 데이터 객체들을 저장할 수 있도록 발전시킨 것이다. 관계형 다중매체 데이터베이스는 이 객체들을 저장, 보호할 수 있을 뿐만 아니라 내재되어 있는 인덱싱 (Indexing) 능력으로 즉시 검색할 수 있다. 그리고 표준 SQL 의 확장을 통하여 다중매체 데이터베이스용으로 객체들을 생성, 처리할 수 있게 한다. 이 객체는 데이터베이스 레코드 내에서 또다른 데이터 필드로서 표현되므로 기존의 SQL 의 검색기능을 통해서도 질의할 수 있다. 다른 데이터유형과 마찬가지로 이 객체는 작업의 원자단위의 일부분으로 간주되고 있기 때문에 기존의 데이터베이스가 제공하는 정상적인 메카니즘을 통해서도 기록, 재저장 및 복구될 수 있다. 따라서 관계형 다중매체 데이터베이스는 기존의 관계형 데이터베이스위에 새로운 다중매체 데이터 유형과 그외 앞절에서 기술했던 여러 기능들과 요건들을 첨가 보완시킨 관계형 확장 데이터베이스라 할 수 있겠다. 먼저 새로운 데이터 유형의 구성을 위한 방법과 기술을 살펴본 후 관계형 데이터베이스의 그밖의 특징과 기술들을 검토해 본다.

1. BLOB(Binary Large Objects)의 개념과 특성

다중매체 데이터베이스 응용 프로그램들은 데이터 필드와 영상 및 텍스트 객체등이 결합된 형태로 나타날 수 있다. 이러한 관계형 다중

매체 데이터베이스시스템은 이들 객체들을 포함할 레코드내의 필드들을 정의하기 위해 BLOB으로 불리우는 새로운 범주의 데이터 유형들을 사용한다.

BLOB 데이터 유형에는 텍스트와 바이트가 있다. 텍스트 BLOB는 메모, 매뉴얼의 장, 계약서 및 소스 코드와 같이 유효한 텍스트 문자들을 포함한다. 바이트 BLOB는 스프레드 시트, 그래프, 팩스, 목적코드 모듈, 위성데이터, 음성패턴 및 디지털이화된 데이터와 같은 객체를 포함할 수 있는 2진 데이터 스트림이다.

BLOB는 매우 클 수 있기 때문에 (최대 2 Giga Byte) 다중매체 데이터베이스로 디스크의 독립된 분할 혹은 전용의 자기 디스크나 광 디스크상에서 BLOB 열 (Column) 전체를 위치시킬 수 있어야 한다. (관계형에서는 행과 열로 정의함) BLOBspace란 BLOB의 열을 포함하는 데이터베이스의 논리적 영역을 지칭한다. BLOB의 열이 BLOBspace에 위치하게 될 경우 그 레코드 내의 대응필드가 그 위치를 가리키게 될 것이다. BLOB의 위치는 관계형 데이터베이스에서 다른 모든 데이터 필드들과 마찬가지로 응용프로그램에 대해 투명 (transparent)하다.

기존의 레코드와 분리되어 있는 위치에 BLOB을 저장할 수 있다는 것은 두가지 이점을 제공한다.

첫째, 커다란 객체가 없는 대형 응용 프로그램들을 최적으로 수행할 수 있게 해준다.

둘째, 값싼 저장 매체로 저장할 수 있다.

2. 관계형 다중매체 데이터베이스 기술

다른 데이터 유형이 그렇듯이 이러한 BLOB의 열이 정규적인 필드

와 동일하게 보이도록 다중매체 데이터베이스를 구현해야 한다. 이는 모든 트랜잭션들이 다음과 같은 속성을 나타내어야 한다.

가. 원자성

시스템이 다운(Down) 될때 트랜잭션이 진행중이었다면 다운전에 이루어진 어떤 트랜잭션 변경도 시스템 복구후에 데이터베이스 내에 남아있지 않아야 한다.

나. 일관성

처리되는 트랜잭션이 다른 사용자들과 관련될 경우에 한해서 트랜잭션이 완성될때까지는 진행중인 트랜잭션에 의해 가해진 변경을 볼 수가 없어야 한다.

다. 격리성

멀티유저(Multi User) 환경에서의 트랜잭션 효과들이 싱글유저 환경에서 실행되었을 때와 동일할 것을 요구한다.

라. 내구성

일단 트랜잭션이 완료되면 후속시스템과 매체에 이상이 있을지라도 그 변경사항들이 영구적으로 그 데이터베이스에 반영될것을 요구한다.

대부분의 데이터베이스는 위의 특성을 보장하기 위하여 서로 비슷한 잠금(Lock)과 로깅 및 복구 프로세스를 사용하는데 관계형 다중매체 데이터베이스에서도 커다란 객체들을 능률적으로 처리하기 위해 그에 알맞는 수정이 가해져야 한다. 게다가 기록 보관하는 작용 및 메모리와 디스크 공간의 이용 역시 특별한 처리를 요구한다.

3. 질의 처리 (Query Procedure)

기존의 질의 기능으로도 관계형 다중매체 데이터베이스를 액세스할 수 있지만 특히 BLOB 과 같은 필드를 액세스할 경우에는 그 값이 정수처럼 단일값이 아니기 때문에 그 사용에 있어 보다 제한적이다. 따라서 질의 기능의 확장과 보완이 요구되는데 그 방법으로 4 세 대언어 즉 객체지향 프로그래밍언어를 사용하여 다른 변수에서 판독하는 것과 동일한 방식으로 프로그램에 정의되어 있는 필드로 BLOB 을 읽어들이 수 있다. 이는 기존의 데이터베이스 시스템이 갖는 결합 불일치성의 단점을 보완해주는 것이다.

4. 물리적인 측면

복잡한 데이터모델을 관리하기 위한 저장방법을 3 가지 유형으로 구분할 수 있다.

가. DSM(Direct Storage Model)

- 전체를 한꺼번에 저장
- Object Id 에 의해 Object 를 검색
- Key 값에 의한 검색은 지원하지 못하고, Shared Object 표현에 는 부적합
- 음성이나 화상과 같은 Object 저장

나. FDSM(Fully Decompressed Storage Model)

- Object Id 와 Value 의 조합으로 표현됨

다. NSM(Normalized Storage Model)

- 복잡한 Object 를 Object Id, Value1, Value2 ... , Value k 의 조합으로 구성
- 일반적인 Index 기법이 이들에 응용될 수 있다.

5. 발전방향

관계형 다중매체 데이터베이스 시스템들이 발전됨에 따라 객체의 위치를 찾고 처리하는 것은 점점 더 세련되어지고 보다 쉬워질 것이다. 서로 다른 객체들간의 관련성은 객체지향 데이터베이스 시스템에서와 마찬가지로 지원될 수 있을 것이다. 확장된 SQL 을 통하여 액세스할 수 있는 완전한 텍스트 검색 기능어외에도 바이트 객체들을 통한 검색이 가능할 것이다. 한편, 상업용 응용 프로그램에서 사용하기 위해 다각적으로 다중매체 데이터베이스 기술을 확장시킬 가능성이 많다. 이러한 기능들이 제공하는 파워를 사용하기 시작함에 따라 그 잠재력은 보다 확실해지고 제품들의 성능은 더욱 향상될 것이다.

제3절 객체지향 데이터베이스

1980 년대에 들어서면서 “객체지향 기술” 이라고 총칭되는 새로운 분야에 대한 관심이 고조되면서 이에 따른 연구가 활발히 진행중에 있으며, 특히 CAD 와 CASE 와 같은 복잡한 설계 응용 프로그램 등에서 그 효용과 가치를 인정받고 있다. 데이터베이스 분야에서도 이 새로운 기술을 응용하여 기존의 데이터베이스의 문제점을 보완 개선하려는 시도가 이루어지고 있으며 이미 실험용 및 상업용으로 개발, 운영되고 있는 것도 상당수에 이른다. 따라서 먼저 객체지향 기술의 표본이라 할 수 있는 객체지향 프로그래밍의 개념과 특성을 살펴본 후 객체지향 데이터베이스를 연구해보는 것이 타당할 것이다.

1. 객체지향 프로그래밍

객체, 메세지, 계승의 세가지 요소가 그 근간을 이루고 있다. 하나의 객체는 여러개의 자체구조와 이들 자료구조에 적용 가능한 연산들로 구성되며 자료구조는 변수 형태로서 연산은 메소드 (method) 형태로 구현된다. 메세지는 어떤 객체에게 어떤 연산을 수행하기를 요구하는 요청으로써 개념적으로는 어떤 연산의 수행 주체가 그 객체 자체가 되어야 한다는 시각에서 나온 개념이다. 대개의 경우 Procedure Call 의 형태로 실현되며 메세지를 받았을 때 수행되는 객체내부의 연산들을 메소드라 한다. 따라서 메세지는 정의된 객체간의 인터페이스 프로토콜 (Interface Protocol) 이라고 할 수 있으며 메세지 발생 및 발생된 메세지의 수신도 객체이다. 계승은 한 객체가 다른 객체의 특성을 자신의 특성의 일부로 사용하는 관계를 말하는 것으로

로써, 이것은 자료의 수정이나 갱신을 단순하게 하고 복잡한 자료를 단순화시킬 수 있도록 한다.

2. 객체지향 데이터베이스

사실 객체지향 데이터베이스는 네트워크나 계층구조의 데이터베이스와 매우 유사하다. 하지만 계층적 데이터베이스가 오퍼레이터에 의하여 일일이 조종되어야 하는 레코드 구조로 구성되어 있는데 비하여, 객체지향 데이터베이스는 중첩된 객체 구조로 되어 있는 점이 다르며, 둘다 포인터 개념을 사용한다는 점에서는 같다. 객체지향 데이터베이스에서 객체 지정자는 나중에 다시 사용될 수 없는 논리적 포인터인데, 이들은 객체의 존재 유무뿐만 아니라 시스템이 메시지를 보내려고 할때 그 객체가 어떤 클래스에 속하게 되는지를 알려 준다. 이를 통하여 객체가 임의의 보전 법칙을 지킬 수 있도록 해주는 메소드를 객체지향 데이터베이스에 포함할 수 있게 된다. 객체지향 데이터베이스가 기존의 계층적 또는 관계형 모델에 비하여 가장 크게 부각되는 점은 계승(Inheritance), 클래스, 메소드, 메시지 등의 완전히 새로운 개념을 사용한다는 것이다. 객체지향 데이터베이스를 사용하는 사용자는 보다 많은 코드(메소드)를 데이터베이스 자체에 포함시킬 수 있다. 이와 같이 응용에 따라서 새롭게 지식을 첨가할 수 있도록 함으로써 질의 처리를 최적화하고 트랜잭션을 동시에 처리할 수 있게 되는 것이다. 따라서 기존의 데이터베이스가 수동적이라면, 객체지향 데이터베이스는 능동적이라 할 수 있다.

3. 기능과 특성

객체지향 데이터베이스들이 갖는 기능들로서 우선 기존의 데이터베이스 시스템으로부터 기대할 수 있는 모든 기능과 특성이 제공되어야함은 물론 지속성, 동시성 조절, 탄력성, 일관성, 질의 처리 기능들이 아울러 포함되어야 한다.

- 복합객체 (Complex Objects): Set, List, Tuple 등의 표현이 쉬워야 한다.
- 객체개성 (Object Identity): 내장된 값이나 그들값의 내용과는 독립적으로 둘 이상의 객체들을 관계지을 수 있으며 객체들은 고유한 Identity에 의해 서로 구분된다.
- 캡슐화 (Encapsulation) : 데이터 내부구조와 외부 인터페이스를 구분하여, 내부구조를 보이지 않게 함으로써 프로그래머는 외부 인터페이스 부분만 고려하게 하고 불필요한 Side Effect를 방지한다.
- 계승 (Inheritance) : 보다 일반적인 자료와 세부적인 자료간에 관계를 설정하고, 보다 세부적인 구조는 일반적인 구조의 속성을 계승받도록 하여, 보다 많은 의미를 표현하고 코드의 재사용이 가능하게 한다.
- 형과 클래스 (Types and Classes) : 모든 객체는 객체가 이해하고 반응할 수 있는 메시지를 정의하는 클래스에 속해있다.
- 완전성 (Computational Completeness) : 완전 프로그래밍 언어의 제공
- 확장성 (Extensibility) : 새로운 응용 프로그램들은 시간이 지남에 따라 진화하는 복잡한 데이터 형태들을 불시에 포함하기 때문에 확장성이 필요하다. 사용자 정의형

- 지속성 (Persistence) : 객체를 생성한 프로세서들이 계속 존재할 수 있게 하는 기능이다.
- 탄력성 : 시스템이 실패할 경우 불일치를 막기 위해 탄력적이거나 실패를 극복할 수 있어야 함.
- 질의 기능 : 질의 최적화를 확장적으로 처리
기억장치 사양들을 인터페이스에 대해 캡슐화시키거나 숨길 수 있도록 처리할 수 있어야 한다.
- 병렬처리
- 복구
- 버전 관리 기능

4. 관계형 데이터베이스와의 비교

관계형 데이터베이스

- 구조화된 질의
- 데이터 의존성 최소화
- 비방향적 질의
- 짧은 트랜잭션, 최적의 동시성
- 내연적 관계
- 절대적 식별자들이 없음

객체지향 데이터베이스

- 방향적 질의
- 프로시저어 의존성 최소화
- 긴 트랜잭션, 최적의 동시성

- 외연적 관계

- 절대적인 객체 식별자

기존의 관계형 데이터베이스를 사용하여서는, CAD와 같은 시스템에서 요구하는 집합적 접근의 결합이나 복합적 내부나 외부구조의 효율적 구현이 어렵다. 그러나 지리정보 시스템(GIS)과 같은 분야에서는 기존의 관계형 데이터베이스에 필요로 하는 데이터 형식을 확충 구현함으로써 상당한 효과를 본 사례가 있다. 이 경우에 관계형 데이터베이스 기술이 유용했던 이유는 구조자체의 처리방식에 있었다기 보다는 질의어가 적절하였다는 점에 있다. 관계형 데이터베이스가 특별한 질의어를 잘 처리하고, 내부와 외부의 데이터베이스를 연결시키는데 장점이 있다는 것이 이러한 경우에 중요한 가치를 더할 수 있다는 것이다. 결국 사용자가 작업을 하면서 중요한 것은 어떤 것이 더 좋은가를 결정하는 것이 아니라 현재 해결하여야 하는 문제의 요구사항에 어떤 방법이 더 적합한지를 결정하는 것이다.

5. 객체지향 데이터베이스의 소개

가. ORION

- MCC에서 개발
- 이론적으로 선구적인 시스템
- 고유의 객체지향 모델 사용
- 복합객체, 버전 제어기능 제공
- LISP언어로 구현(속도 문제됨)

나. Iris

- HP사에서 개발
- 기존 관계형 데이터베이스를 확장하는 방법으로 개발
- SQL을 확장한 객체 SQL 제공

다. O2

- Altair사에서 개발
- 잘 구비된 프로그래밍 환경 제공
- C O2, BASIC O2, LISP O2
- 수행 모드와 개발 모드 분리 제공
- 고유의 객체지향 모델사용

라. Versant

- Versant Object Technology사에서 개발
- 빠른 속도
- 분산형 모델 구현
- 고유의 객체지향 모델 사용

마. VADAK

- GMD-IPSI에서 개발
- 분산형 모델 구현
- 고유의 객체지향 모델 사용

6. 발전방향

객체지향 데이터베이스는 이미 이전에 개발되었던 의미 데이터 모

델과 어느정도 상통하는 면이 있고 관계형 모델보다 훨씬 풍부한 환경을 제공하는 장점이 있다. 객체지향 데이터베이스는 관계형 모델의 한계, 특히 복잡한 데이터형식을 처리할 수 없다는 점을 극복하기 위한 결과로서 발전하고 있다. 많은 학자들과 대학교수 및 발전론자들에 의해 제안된 관계형 모델의 확장은 관계형 데이터베이스계에 불가피한 새로운 개념으로의 방향전환을 의미하고 있다.

실제로 그들의 제품의 전위에 객체지향의 확장을 추가하고 있는 실정이며, 시간이 흐르면 이들 모두는 완전한 객체지향 데이터베이스로 바뀌어질 것이다.

제 4 절 분산 데이터베이스

지난 몇년에 걸쳐 네트워크 기술에 있어서의 급속한 발달이 다른 형태의 컴퓨터들을 함께 묶는 것을 손쉽게 해주고 있다. 과거에는 이러한 연결의 주요 목적은 프린터와 디스크 드라이브들을 공유하는 것이었다. 그런데 지금은 분산 데이터베이스 기술에 의하여 본질적으로 서로 다른 컴퓨터들과 데이터베이스 시스템들이 중요한 데이터를 서로 공유할 수 있게 된 것이다.

특히 방대하고 비정형화된 데이터를 저장, 검색해야 하는 다중매체 데이터베이스 시스템에서는 분산 데이터베이스 기술이 필연적이라 하겠다. 데이터는 비록 압축과정을 거쳐서 저장된다 하더라도 그 양이 매우 방대하므로 저장 및 검색의 효율성을 위하여 하나의 화일이 여러개의 디스크 시스템에 분산되어 저장되어야 하는 가능성을 배제할 수 없다. 이러한 경우에 하나의 화일을 분산 저장하는 정책과 특정 데이터를 검색하는 경우에 각 화일의 분산상황을 유지하는 기능이 지원되어야 하는데 그에 따른 분산 데이터베이스 기술현황과 각각의 장단점 그리고 앞으로의 발전방향 등을 다중매체 데이터베이스와 연계하여 살펴본다.

1. 분산 데이터베이스 시스템의 기술

다중매체 데이터베이스 시스템을 효과적이고도 능률적으로 운영하기 위하여 특별히 요구되는 분산 데이터베이스 기본 기술중 중요한 몇 가지 개념들을 살펴보면 다음과 같다.

- 단락화 (Fragmentation)

단 하나의 테이블이 네트워크 사이트(Site) 들에 어떻게 분산되는지를 설명한다. 수평단락, 수직단락, 혼합단락 등의 방법이 있다.

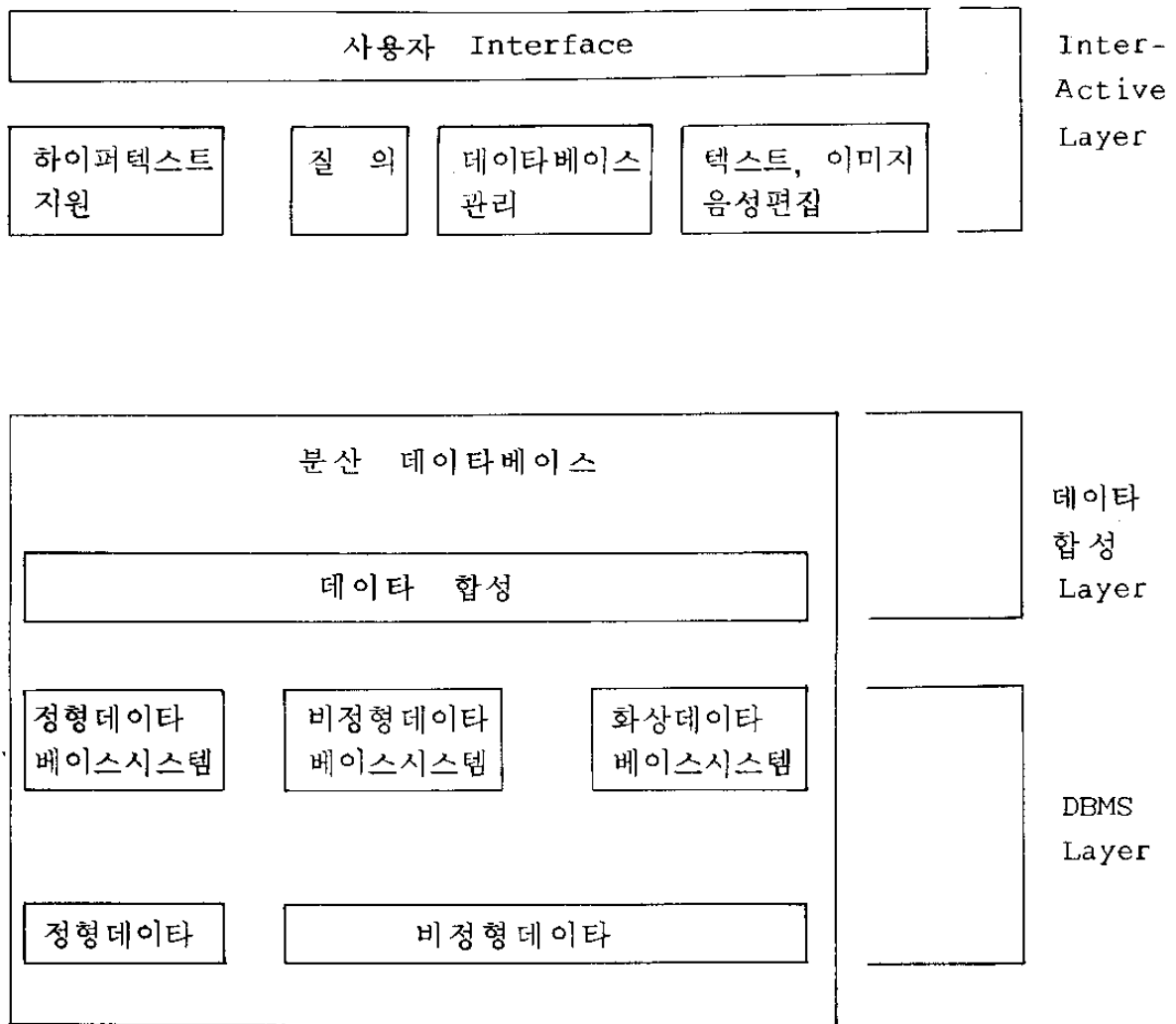
- 테이블 복제 (Table Replication)

복제는 네트워크 주위에 테이블을 복사, 분배하는 것이다. 이는 데이터의 지역이용성을 최대화하고 특정 네트워크 사이트가 실패할 경우 테이블들의 백업 복사본들을 제공하기 위함이다. 그러나 테이블이 갱신될 경우 테이블의 모든 복사본들이 분산 데이터베이스의 완전한 상태를 위해 갱신되어야 하기 때문에 데이터베이스의 성능을 저하시킬 수 있다.

- 할당 (Allocation)

단락화와 복제의 결합된 형태로 나타나며, 어떤 사이트들이 어떤 단락들을 저장할지 결정하는 요소로서 응용 프로그램들이 지역적으로 저장된 데이터만을 액세스할 필요가 있도록 데이터를 저장하는 역할을 담당한다.

또한 실질적 사용지들에게는 분산 시스템을 하나의 중앙 집중식 시스템으로 보이게 하기 위하여 상기 기술들 각각의 캡슐화가 아울러 이루어져야 함도 유의해야 한다. 단락화와 복제 및 할당은 분산 데이터베이스 시스템의 아키텍처를 결정한다. 이러한 아키텍처는 몇개의 레이어 (Layer) 나 스키마타 (Schemata) 로 분리된다. 다음 그림은 분산 아키텍처의 각 구성들간의 계층적 관계를 나타낸 것이다.



< 그림 3-1 > 분산 데이터베이스 시스템의 계층적 관계

가. DBMS Layer

위 그림에서 보면 데이터 합성 모듈과 데이터베이스 관리 모듈이 서로 나뉘어져 있음을 볼 수 있는데 이는 곧 다중매체 시스템 환경에서의 다중 서버 시스템 개념으로 각각 분산되어 있음을 뜻한다. DBMS Layer에서는 실질적인 다중매체 데이터 관리를 담당한다.

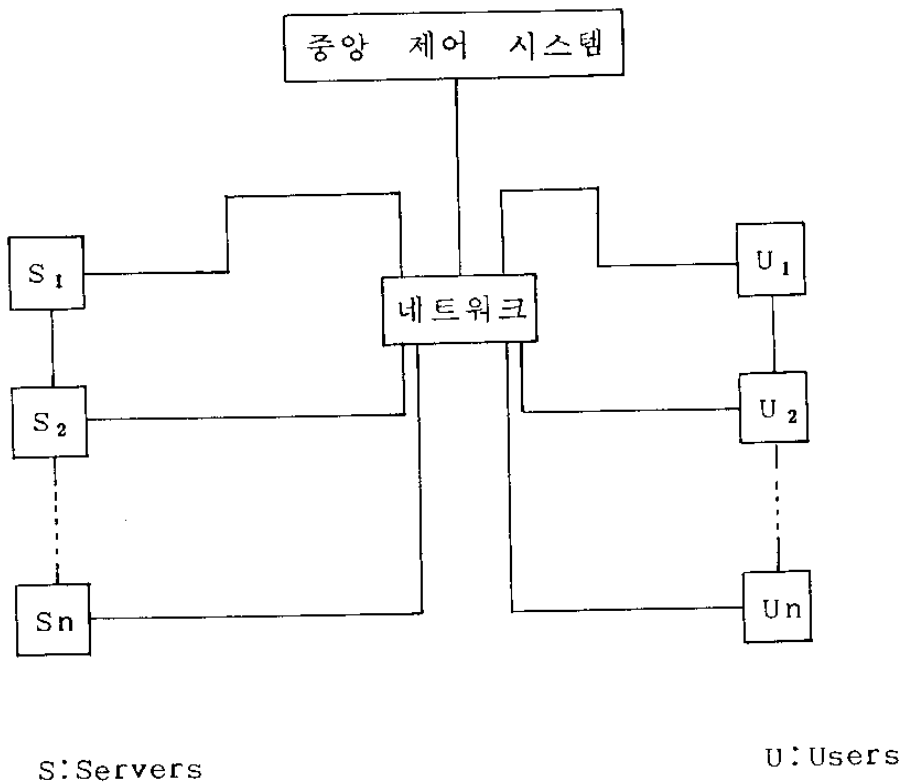
나. 데이터 합성 Layer

DBMS Layer 와 Interactive Layer 를 상호연결 시키면서 데이터의 동기화와 통합화를 전담한다.

다. Interactive Layer

사용자 인터페이스로 구성되며 데이터베이스의 질의 연산을 포함한다.

2. 분산 데이터베이스의 시스템구성



가. 중앙 제어 시스템

- 객체관리, 서버 할당 등을 담당하는 부분으로 다음과 같은 역할을 담당한다.

- 제공된 서비스에 필요한 파라미터를 사용자로부터 수신
- 질의를 해석하여 필요한 서버를 선정
- 선정된 서버와 사용자간의 연결
- 질의간의 우선순위 결정
- 통신 프로토콜 지원

나. 분산 서버

— 서버의 역할

- 사용자와의 상호 작용
- 여러 부속 서버와의 통신
- 지역 데이터와 다른 서버로부터 받은 외부 데이터를 효율적으로 통합하기 위해 대용량의 메모리 필요

다. 사용자 워크스테이션

— 데이터를 출력한다.

- 강력한 연산 능력 필요
- 저장 능력 요구
- 통신 네트워크와의 인터페이스
- 중앙 제어 시스템과의 상호작용

라. 네트워크

— 서버와 사용자를 상호 연결

— 고속 및 고품질의 통신망 필요

— 사용 가능한 통신망

- 광 통신을 근간으로 하는 Metropolitan Area Network(DQDB)
→ QSPX

- 현 기술 수준에서 2-5 Gbps 정도의 전송속도 가능
- 전송지연 최소화, 고속전송이 가능한 B-ISDN

3. 분산 시스템의 장점과 문제점

가. 장점

- 지역처리가 지역 사이트에서 처리되게 하는 것은 모든 처리를 중앙 사이트에 집중시키는 것보다 훨씬 효과적이다.
- 호환성이 없는 하드웨어와 운영체제, 데이터베이스들 및 통신프로토콜들을 전부 하나의 통일된 전역 시스템으로 묶어준다.
- 단순히 새로운 기종들을 네트워크에 추가시키므로써 분산 시스템이 급변하는 상황등을 손쉽게 수용할 수 있다는 사실과 여유분의 데이터 저장장치로 인한 시스템의 신뢰성과 효율성을높여준다.

나. 문제점

- 상대적으로 느린 데이터 전송율을 지닌 Wide Area Network 과 같은 전송회선에서는 심각한 통신 오버헤드를 수반한다.
- 트랜잭션 관리와 동시성 조절 및 실패로부터의 복구가 어렵고 복잡하다.

4. 표본 시스템

아래 시스템들은 느린 전송속도로 인하여 다중매체 데이터베이스 시스템에 당장 응용되기는 어렵지만 앞으로 네트워크 기술이 발전함에 따라 그 효용이 발휘될 수 있을 것이다.

가. SDD-1

- DEC PDP-11 기종 위에서 작동한다.
- ARPANET 로 연결, 전송속도가 느리기 때문에 다중매체 시스템 용
으로 부적합
- 완전한 단락화, 복제, 할당, 캡슐화 제공

나. R *

- CICS/ISC 를 통하여 통신
- IBM 메인프레임 기종들 위에 작동하는 다수의 시스템 R 사본
이 협력하여 구성
- 할당, 캡슐화를 제공하지 못한다.

다. 분산 INGRES

- DEC PDP-11 기종 위에서 작동
- 단락화, 복제, 할당, 캡슐화 기능 제공
- 광역망과 지역망을 같이 취급하도록 지원
- 매우 정교한 질의 최적화 장치
- 많은 다른 데이터베이스와 화일 시스템들에 대한 게이트웨이
제공
- 비교적 빠른 전송속도로 인해 다중매체 시스템으로 사용가능

5. 발전방향

앞으로 다중매체 데이터베이스 시스템에서는 분산 처리 기능이 필
수적이 될 것이다. 또한 이제까지의 100 Mbps 의 LAN 에 비해 1000
Mbps 이상의 고속 LAN 이 등장하게 되면 훨씬 더 효과적으로 분산

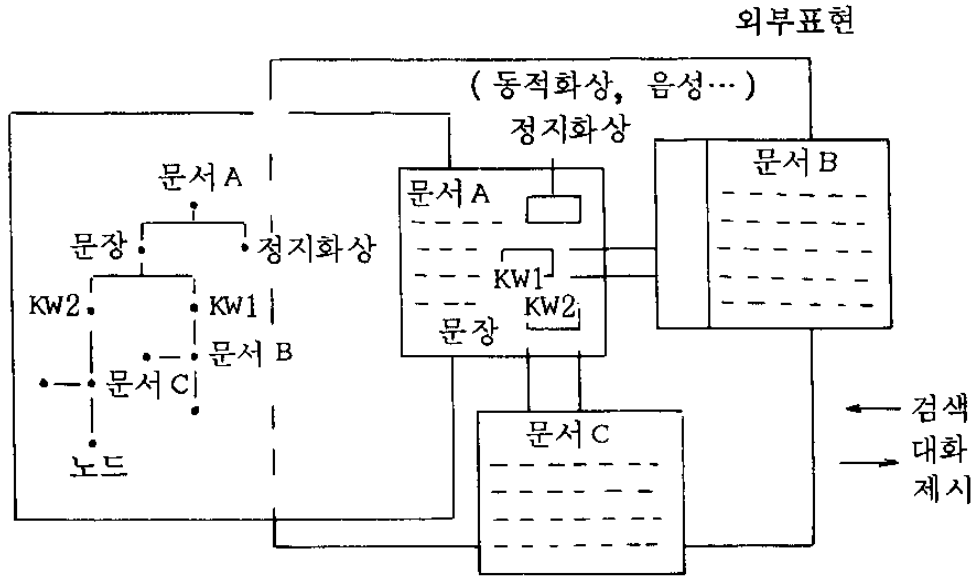
처리 기능을 이용할 수 있을 것이다. 또 현행 LAN 형태의 네트워크 체계가 바뀌어 NLAN(Nonlocal Area Networking) 형태로 바뀌게 되면 전체 네트워크가 하나의 시스템이 되는 분산 오퍼레이팅 시스템이 될 것이다. 이들 분산 오퍼레이팅 시스템과 통신 프로토콜은 세계 도처에 흩어져 있는 사용자들을 서로 음성 및 화상을 통해 접촉하게 하고 서로 데이터를 주고 받을 수 있게 해 줄 것이다.

제5 절 하이퍼미디어 데이터베이스

하이퍼 텍스트는 분리된 데이터들간에 링크를 생성, 표현하는 일로 정의될 수 있으며 이들 데이터에 텍스트 수치와 더불어 그래픽, 음성이 포함된 구조를 하이퍼미디어라 한다. 하이퍼미디어는 이미 정해진 구조대로 정보를 제공받는 것이 아니고 다음번에 어떠한 정보를 필요로 하는가에 따라 사용자가 즉석에서 선택할 수 있도록 해준다. 이러한 하이퍼 텍스트의 비구조적 성질이 정보처리를 곤란하게 한다는 회의론도 일었지만 문서공간의 자유로운 탐색을 가능하게 해 데이터베이스의 사용자에게는 이상적인 시스템으로 간주되고 있다. 더우기 직접적으로 노드간에 링크를 시키는 것과 직감적으로 데이터를 검색해 갈 수 있다는 것은 하이퍼미디어 시스템에서 다중매체를 다루는 이점이기도 하다.

1. 구조

하이퍼미디어는 정보의 단위를 표현하는 노드와 정보간의 관계를 나타내는 링크로 이루어진다. 컴퓨터에 축적된 내부 모델로서는 네트워크 모양, 또는 계층형의 정보관계가 있으며, 이용자가 볼 수 있는 외부 표현으로서는 윈도우 또는 카드 양식의 형태를 취하고 있다. 이러한 구조를 그림으로 설명하면 다음과 같다.



< 그림 3 - 2 > 하이퍼미디어의 정보 표현 형태

2. 종래의 정보관리 시스템과의 비교

가. 링크를 의식한 정보 검색

사용자의 검색의식이 희박하거나 검색 키워드(Keyword)가 애매해도 링크를 둘러보면 관련정보를 얻을 수 있다. 일반적 정보 검색 시스템이나 데이터베이스가 선언적 질의어나 논리적 연산을 검색 수단으로서 이용함에 비해 하이퍼미디어는 이런 개념을 의식하지 않는다.

나. 사용자 인터페이스

하이퍼미디어에서는 사용자 인터페이스가 중요시 되는데 Tiling Window, Pop-up Window, 커서(Cursor) 모양 변경에 의한 의미 전달 등 다양한 표현을 고려한다.

다. 외부 표현과 내부 모델과의 적절한 대응

표시방법과 정보 내부 표현의 대응이 간결한 것도 하이퍼미디어의

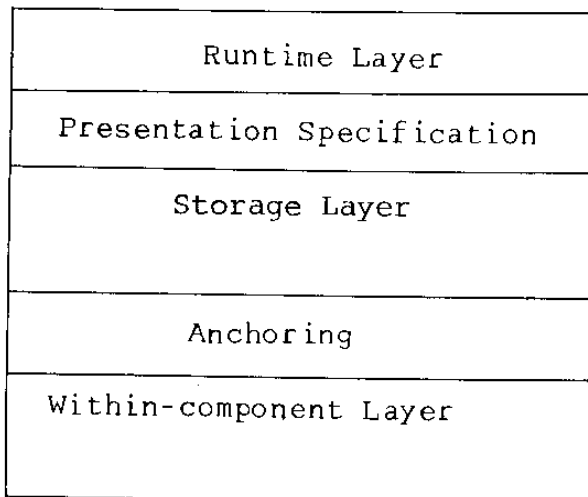
특징이다. 문서, 문장, 대단락, 키워드 등은 표시상으로 포함관계에 있지만 내부 데이터 구조로서는 계층관계로 표현되어 이해가 쉽다.

라. 정보처리의 종합환경의 제공

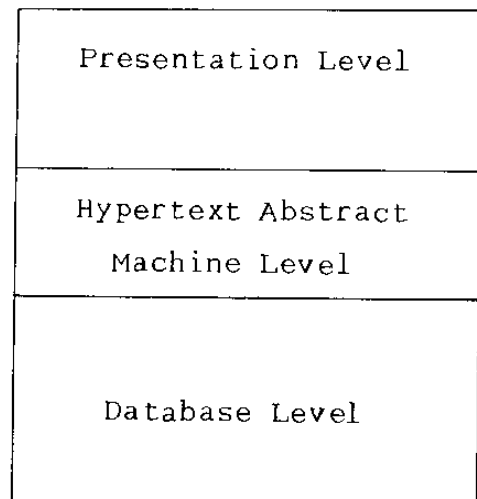
여러 종류의 에디터(Editor)를 종합하고 통일적인 정보이용 환경을 제공한다.

3. 시스템 아키텍처

정립된 이론은 아직 없지만 다음 두 모델이 대표적이다.



Dexter Reference Model



Campbel and Goodman's
Model

4. 하이퍼미디어의 연구 및 상용시스템

| 구 분 | Guide | HyperCard | Intermedia |
|----------|--|---|--|
| 개발시기 | 1983 | 1987 | 1986 |
| 개발기관 | Kent 대학 | 애플사 | Brown 대학 |
| 구현된 머신 | PERQ 워크스테이션 | Mac | Mac (Au/x) |
| 사용 가능 머신 | Mac, IBM PC | Mac | Mac |
| 노드 타입 | - Stretch Text - Replace Button - Reference Button - Command Button | - 카드메타프 - 스택 - Non-sticky 버튼 - SuperCard | - Document - 매우 길다 |
| 링크방법 | - 저작시스템이 담당 - 윈도우를 Scroll 해서 이동 | - 3 가지 | - 관련 문구만 - 두 앵커를 연결 |
| 오버뷰다이어그램 | 없음 | 제공 | - Web View - 오버뷰 Document |
| 특징 | - Scrolling 윈도우 - Genesis 언어 | - HyperTalk - 고정 프레임 - 가장 보편화 - 배우기 쉬움 | - Unix 사용 (제약요인) - Scrolling 윈도우 - 다양한 퍼실리티 제공 |

(다음 페이지에 계속)

(전 페이지에서 계속)

| 구 분 | NoteCard | KMS | Hyperties |
|--------------|--|--|----------------------------|
| 개발시기 | 1987 | 1975 | 1983 |
| 개발기관 | Xerox PARC | CMU(ZOG) | Maryland 대학 |
| 구현된 머신 | D- 머신 | PERQ 워크스테이션 | IBM PC |
| 사용가능 머신 | Sun | Sun, Apollo | Sun, PS/2 |
| 노드타입 | -Single Notecard - 데이터에 따라 다른 형태의 노드 타입 (최소50가지) | - Frame - 단일타입 - 신속히 중착노드 를 화면에 보여줌 | - Article - 여러 Page 로구성 |
| 링크방법 | -Label | - 한 페이지만 | - 전체 기사 |
| 오버뷰 다이어그램 | -Browser Card | 없음 - Home Frame | 없음 |
| 특징 | -File Box -Scrolling 윈도우 | -역추적 기능 -고정 프레임 | - 매우 간단 - 고정 프레임 |

5. 응용분야

- 컴퓨터 분야

- On-line Documentation : Symbolic Document Examiner
- CASE
- 운영체제

- 비즈니스 분야

- 유지 보수 지침서 : MIT Movie Manual
- 사전 및 참고문헌 : DRUID System

- 회계감사 : Procedure Summary Form

- Brochure/Pamphlet : SoftAd

- 교육분야

- 아이디어 창출 : NTT Team Workstation

- 신문 및 방송 : The '88 Vote

- 전문 분야 : Staney's Mathematica Notebook

- 외국어교육 : IRIS InterLex Server

- 고서 연구 및 고고학 : Perseus

- 박물관 : London Design Museum

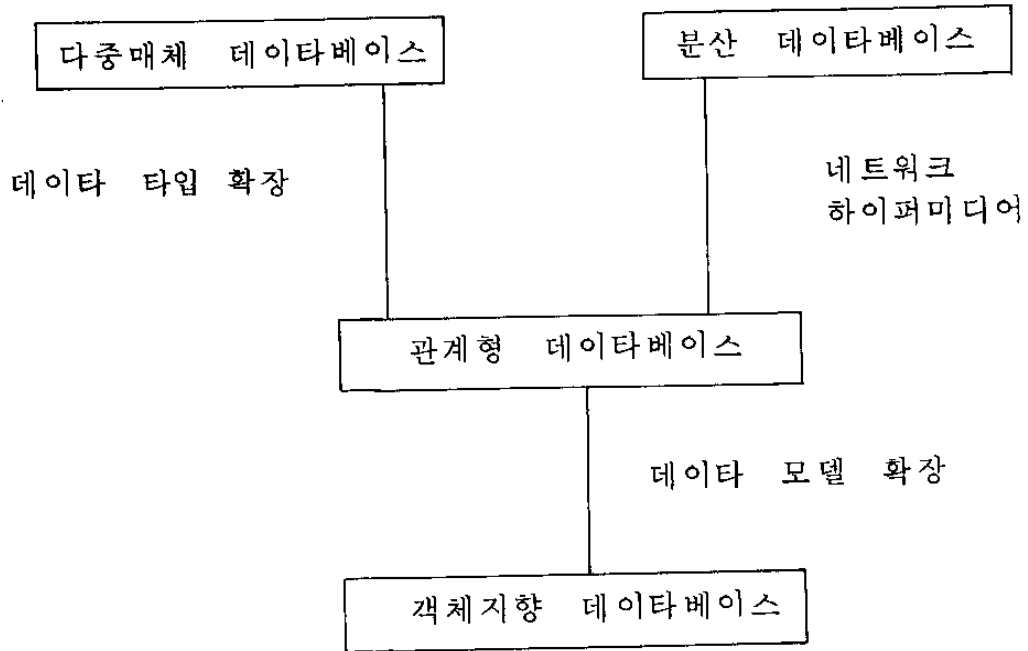
- 오락 및 레저

- 관광안내 : Glas Gow Online

- 도서관 : Book House

제 6 절 다중매체 데이터베이스의 발전 방향

텍스트, 이미지 그래픽, 음성 등과 같은 복합 데이터를 저장 관리해야 하는 다중매체 데이터베이스 시스템은 단순히 정형적, 비정형적 데이터 관리 차원을 넘어서 데이터를 저장, 획득, 출력하는 장비, 광 디스크, 음성 인식자, 운영체제 등에 대한 다중매체 시스템 환경 제반문제의 해결이 함께 이루어질때 진정 다중매체 데이터베이스의 목적을 달성할 수 있을 것이다. 지금까지의 다중매체 데이터를 다루는 연구는 관계형 데이터모델을 기초로 하여 정형화된 데이터와 비정형화된 데이터를 분리 저장하는 방법을 취해왔는데 이는 다중매체 데이터의 특성상 많은 문제점을 야기할 수 밖에 없다. 이에 기존의 관계형 데이터베이스 시스템을 보완, 확장하는 방법과 한편으로는 데이터의 추상화를 통해 복합 객체의 구성이 용이한 새로운 개념의 객체지향 데이터베이스 개발의 방법등이 논의되고 있는 실정이다. 여기에 네트워크의 기술발전, 분산 시스템의 적용, 하이퍼텍스트 개념 첨가 등을 통해 복잡하고 다양한 형태의 데이터베이스 시스템이 연구 개발되고 있다. 하지만 각각의 데이터베이스 시스템들은 각자의 장점과 단점을 갖고 있기 때문에 서로 그 장점과 단점들을 보완하는 선에서 다양한 사용자의 요구에 적합한 데이터베이스 시스템이 개발되어야 할 것이다. (아래 그림 참조)



< 그림 3 - 3 > 미래의 데이터베이스 시스템

제 4 장 시스템 개발 내용

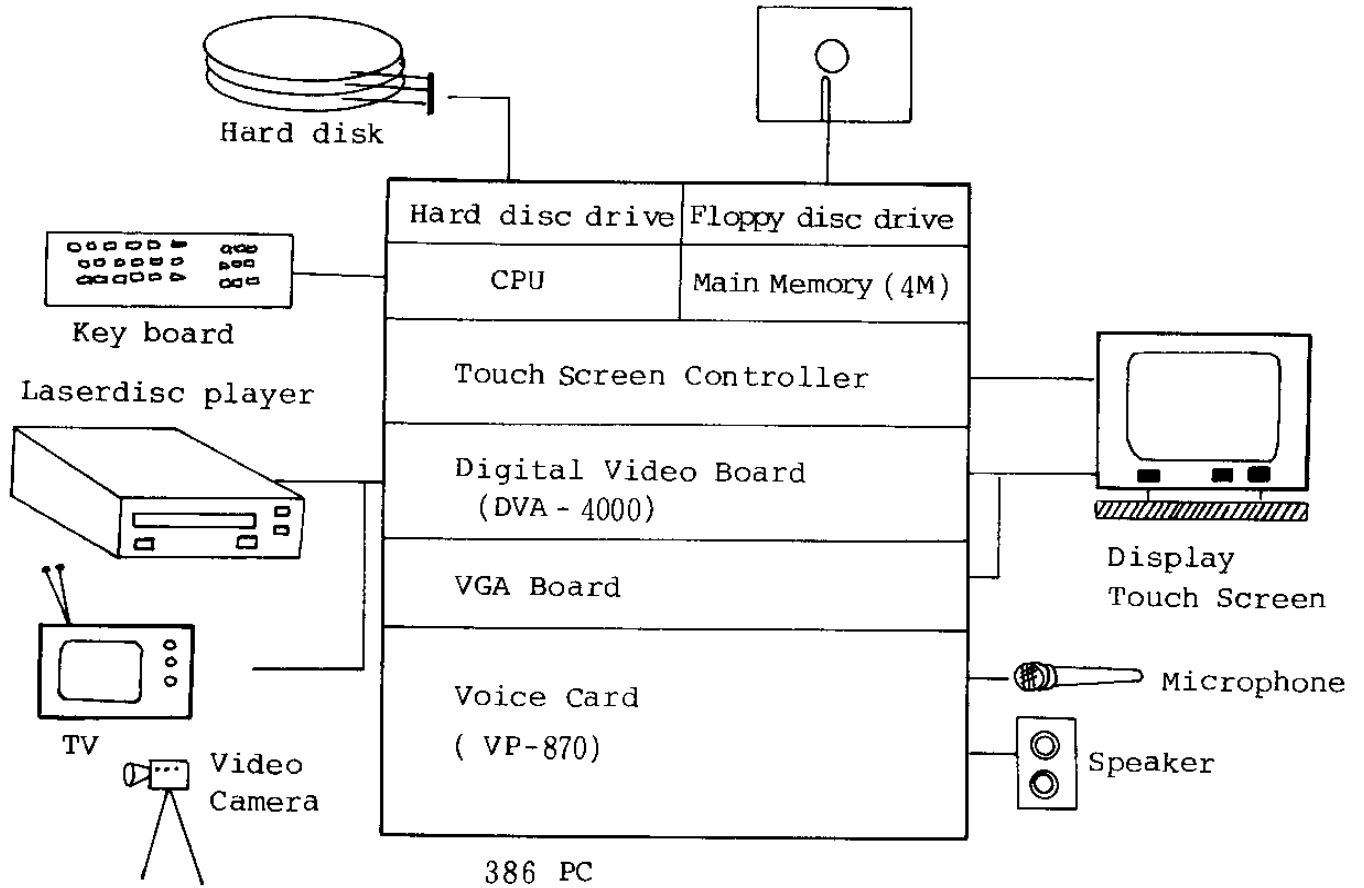
제 4 장 시스템개발 내용

제 1 절 시스템 개발환경

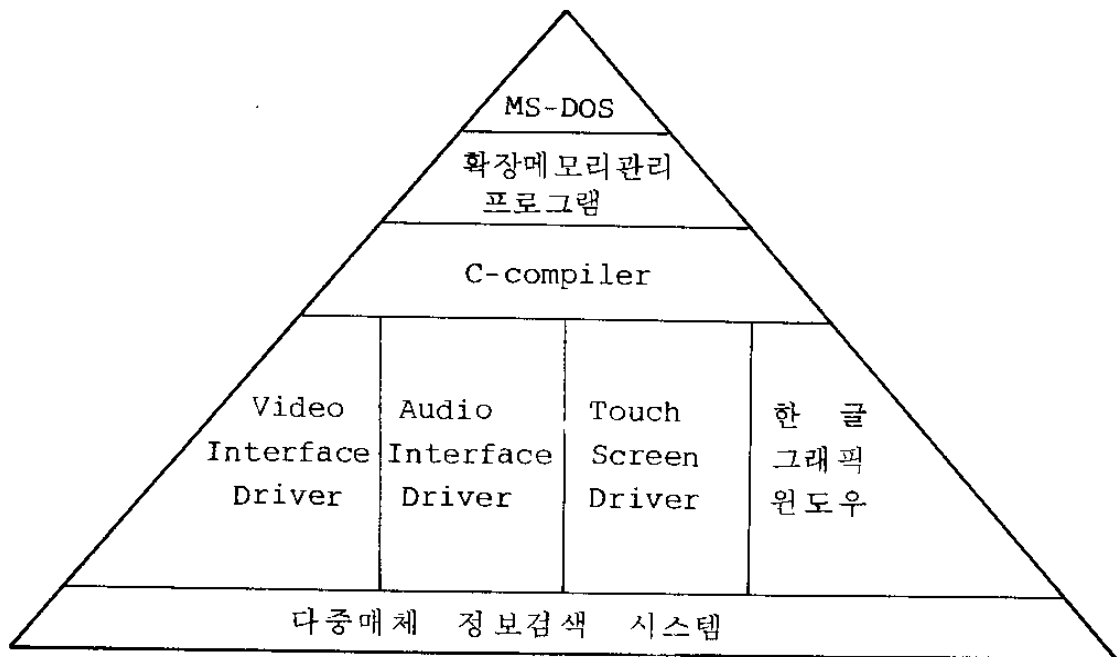
다중매체 응용시스템을 위한 개발환경으로는 그래픽, 오디오, 비디오 등 각 매체별로 데이터를 편집할 수 있는 에디터 (Editor)를 갖추어야 하며, 방대한 양의 데이터를 빠른 속도로 검색할 수 있게 하려면 DVI, CD-I 등의 데이터 저장매체 관련 기술을 적용하여 데이터의 압축 및 복원기술에 대한 하드웨어 사양과 소프트웨어 알고리즘에 대한 심층연구가 요구된다. 그러나 아직 DVI, CD-I 등의 기술로는 동화상 정보를 선명한 화질로 보여줄 수 있는 정도에 이르지 못하고 있는 실정이다. 데이터베이스에 요구되는 매체별 데이터를 만드는 작업은 기존에 개발되어 있는 응용프로그램들 (Paintbrush, IRPRO, Toolkit 등)을 이용하거나 부족한 부분은 개발팀이 직접 프로그램을 개발하여 데이터 화일을 작성하였다.

그리고 동화상 (Motion Picture) 정보와 같이 1 초에 30 Frame 이상의 방대한 데이터가 고속처리되어야 하는 경우 데이터의 압축 및 복원기술을 적용해야 하지만 본 연구에서는 이를 적용하지 못하고 IVD (Interactive Video Disc) 중의 하나인 레이저디스크에 저장된 아나로그의 동화상 정보를 레이저 디스크 플레이어로 액세스하여 디지털 비디오 보드를 통해 디지털 데이터로 변환하여 디스플레이 하는 것으로 연구범위를 제한하였다.

본 연구에서는 기존의 정형화된 데이터의 관리 및 검색기능에 덧



<그림 4-1> 다중매체 데이터 베이스 시스템 하드웨어 구성도



<그림 4-2> 다중매체 데이터베이스 시스템 소프트웨어 구성도

불여 비정형화된 데이터의 관리·검색기능까지 갖는 다중매체 데이터 베이스의 응용을 위하여 관계형 데이터베이스의 개념을 확장하여 다중매체 데이터를 검색할 수 있는 기본함수를 개발하는데 초점을 맞추었다.

또한 다중매체 시스템 및 데이터베이스 응용을 위한 기본함수를 만들되 하드웨어(〈그림 4-1〉 참조) 보다는 소프트웨어(〈그림 4-2〉 참조)의 개발과 통합에 주력하였다.

다중매체 응용시스템을 개발하기 위한 기본개발 환경을 구축함으로써 다중매체 시스템 기술에 익숙하지 못한 각 응용분야의 개발자들도 손쉽게 자기가 원하는 다중매체 응용시스템을 개발할 수 있도록 비디오 인터페이스, 오디오 인터페이스, 사용자 인터페이스 등의 기본함수를 개발하여 제공하였고 사용자가 한글 입·출력을 용이하게 할 수 있는 윈도우 조작함수도 개발하였다.

1. 비디오 인터페이스

이 분야의 라이브러리 함수들은 디지털 비디오 보드(DVA-4000/ISA)상에서 구현되도록 개발하였는데, 그 특성 및 기능은 다음과 같다.

- Background에 비디오를 디스플레이하고 Foreground에 텍스트나 그래픽을 디스플레이 함으로써, 비디오를 디스플레이 하는 모니터와 일반 모니터를 구분하여 사용할 필요 없이 VGA 보드를 통해 화면에 텍스트, 그래픽, 비디오를 함께 디스플레이 하는 것이 가능하다.

- DOS 의 Command Level 에서 혹은 응용 프로그램 내에서 레이저 디스크의 제어가 가능하다.
- NTSC, PAL, RCB, S-VHS, CDI, DVI 등의 입력중 두가지를 동시에 입력할 수 있으며, VTR, 비디오 카메라, 레이저 디스크 등의 화상데이터를 하드 디스크와 같은 저장매체에 저장이 가능하고, 저장된 이미지를 화면에 디스플레이할 수도 있다.
- 화면에 디스플레이 할 화상데이터의 디스플레이 기준점 위치나 디스플레이되는 영역의 크기를 조정할 수 있으며, 동시에 여러 개의 정지화상을 디스플레이할 수 있다.
- Foreground 상의 그래픽을 원하는 색상별로 Fade On 또는 Off 하는 것이 가능하다.

또한 DVA4000 비디오 보드는 프로그래머의 생산성을 향상시키고 프로그램 개발에 소요되는 시간과 비용을 절감하기 위하여 다음과 같은 프로그램 개발 툴들을 제공한다.

가. TOOLBOX Plus

프로그램 개발시 MIC (Multimedia Interactive Control) 시스템 소프트웨어를 쉽게 사용할 수 있도록 개발된 시스템 관리 프로그램으로서 다중매체 시스템을 제어하고 관리한다. 텍스트 모드 (80 Column) 와 그래픽 모드 (640*200) 를 지원하며 TSR (Terminate and Stay Resident program) 로서 MS-DOS Command level 에서 한 번의 키입력으로 실행할 수 있으며 현재 작업 중인 화면의 내용 (모드, 색상, 커서 위치 등) 을 보관한다.

나. MONITOR (Command Trace Program)

응용 프로그램과 MIC 시스템 간에 주고받는 명령어 및 실행결과 등을 디스플레이 하고 제어하며, 응용 프로그램과 함께 Background 에서 작업을 진행하며 동시에 명령어와 그 실행결과를 확인할 수 있는 트레이스 모드 (Trace mode) 와 MONITOR 의 Configuration 을 변경할 수 있는 COMMAND 모드를 제공한다.

다. MASK

화면의 특정 부분 및 키보드의 입력을 논리적인 아이콘 값으로 자동적으로 변환시켜 줌으로써 키보드는 물론 각종 포인팅 입력장비 (마우스, 터치 스크린 등) 를 사용한 응용 프로그램 개발에서 직접 좌표값을 처리하는 불편을 없애준다. 정의된 아이콘들은 포인팅 장비와는 독립적이므로 마우스로 정의된 마스크는 터치스크린, 키보드, 라이트 펜 등으로도 액세스할 수 있다.

2. 오디오 인터페이스

현재 발표되고 있는 오디오 인터페이스 장비는 그 기능과 특성면에서 몇가지로 구분지워 질 수 있다. 단지 음성을 입력받아 디지털화하여 파일로 저장하고 저장된 파일을 재생시키는 기능에서부터 음악을 연주할 수 있는 기능이나 음성을 합성하고 인식할 수 있는 기능을 갖는 보드들이 발표되고 있다.

여기서는 프로그램내에서 자유롭게 음성 데이터를 입·출력할 수 있도록 음질이 우수한 VP-870 보이스 카드 (Voice card) 를 사용하였다. 이 카드를 이용하여 레코더와 동일하게 Record, Play, Stop,

Forward Winding, Backward Winding 과 Winding Indicator 등의 기능을 갖도록 기본함수를 개발하였으며, 새로운 프로그램을 개발할 수 있도록 다음과 같은 인터럽트 서비스 루틴을 제공한다.

- START Recording Processing
- LOAD (보이스 데이터를 메모리로)
- GET
- SAVE
- START

VP-870 보이스 카드의 기본 사양은 다음과 같다.

- 음성입력장치 : 마이크와 테이프 레코더
- 음성출력장치 : 4 또는 8 Ω (오옴)의 스피커
- Input Filter : 300 Hz - 3.4 MHz
- 출 력 : 최고 1.5 Watts
- I/O Address : 2B8H-2BFH

3. 사용자 인터페이스

사용자 인터페이스를 보다 편리하게 지원하기 위해서 극히 제한적이기는 하지만 터치스크린을 구동하는 기본함수 개발에 힘썼다.

터치 스크린은 적외선을 이용한 광센서 방식과 스크린 표면에 전도물질을 입히는 Voltage field 방식이 있다. 본 연구과제에서 사용한 방식은 후자의 방식으로 아날로그 센서와 양방향 컨트롤러로 구성되어 있으며 스크린 센서는 유리판의 표면에 전도물질을 코팅 처리하여 만든다. 이때 스크린 표면에는 Linear Voltage Field가 생

성되는데 손가락 또는 도체로 스크린을 터치하면 이 Voltage Field 센서를 자극하여 컨트롤러가 터치점을 계산해 낸다.

터치스크린의 응용프로그램을 쉽게 개발할 수 있는 툴에는 터치스크린 드라이버, MICROCAL, PAD Manager 등이 있다.

가. 터치 스크린 드라이버 (Touch Screen Driver)

터치 스크린 드라이버는 일종의 메모리 상주 프로그램이다. 이를 통해 프로그램을 손쉽게 작성할 수 있도록 표준 인터페이스를 제공한다. 이 터치스크린 드라이버는 DOS의 디바이스 드라이브인 것처럼 인식된다.

나. MICROCAL

MICROCAL은 터치 스크린의 좌표보정과 아울러 터치 스크린 및 모니터를 테스트하는 프로그램이며 그 기능은 다음과 같다.

- AUTOTEST : 간이 터치 스크린 진단 및 테스트
- MONITOR TEST : 터치 스크린이 장착된 모니터 테스트 및 SETUP
- 터치 스크린 및 하드웨어 컨트롤러의 SETUP 및 좌표보정
- 터치 스크린의 정확도 테스트

다. PAD Manager

PAD Manager는 터치 스크린의 응용 소프트웨어 개발을 용이하게 해주는 유틸리티 프로그램이다. 모든 기능은 대화식으로 진행되며 사용자는 이를 이용하여 화면상에 원하는 터치존 (Touch zone)을 지정할 수 있다.

PAD Manager 는 RAM 상주 프로그램으로 “HOT KEY”를 이용하면 언제든지 POP UP 시킬수 있으며 그래픽이나 텍스트 등의 실제 사용중인 프로그램 화면위에 PAD를 손쉽게 정의할 수 있다.

4. 한글그래픽 윈도우

키보드와 디스플레이를 통해 텍스트를 입·출력하는 기능은 모든 마이크로 컴퓨터가 하드웨어 또는 시스템 소프트웨어 수준에서 구현하고 있는 기본 기능이기도 하나, 본 연구과제의 Target Machine 인 IBM PC의 경우에는 하드웨어적인 문자 처리기능이 텍스트 모드만을 지원하는 한계가 있으므로, 부득이 다양한 영상정보와 문자정보를 동일 화면에서 병행 처리할 수 있기 위해서는 그래픽 모드에서 문자의 입·출력을 가능케 하는 기능을 구현할 필요가 있다.

이 경우에도, 영문 문자의 경우에는 이른바 “Graphic Library”라고 하는 각종 프로그램 개발 툴에서 다양한 서체와 다양한 크기의 문자 처리 기능을 지원하므로 일반적인 어플리케이션의 개발을 위해 새로운 문자 처리 시스템을 개발할 필요가 없으나, 한글의 경우에는 이러한 요건을 갖춘 Function Library가 상용화되어 있지 않으므로, 이러한 기능을 지원하는 기초 함수를 개발해야 한다.

따라서 본 연구에서는 그래픽모드의 설정과 취소, 한글입·출력 환경의 설정과 취소, 한글문자열의 입·출력에 관한 함수, 한글 텍스트의 윈도우 처리 등에 관한 기본함수를 개발하였다.

5. 다중매체 정보검색시스템

이 시스템에서 제공하는 검색 기능의 기본 개념은, 화상, 음성 등의 다중매체 데이터들이라고 할지라도 그 검색 접근점은 어디까지나 유관한 화상, 음성 데이터의 내용에 관해 기술한 문헌정보(텍스트)여야 한다는 점이다. 화상과 음성 데이터는 그것이 재생 장치를 통해 일정한 형태의 그림과 소리로 구현되었을 경우, 그것을 접하는 사람들에게 일정한 의미를 줄 수 있지만, 재생 과정을 거치지 않은 시점에서는 그 의미를 확인할 수 없는 Binary Object에 불과할 뿐이다. 그와 같이 맹목적인 Binary Object들 가운데에서 사용자의 목적에 부합하는 대상들을 추출할 수 있기 위해서는 개개의 Binary Object들이 데이터의 상태에서도 의미 식별이 가능한 텍스트 정보와 일정한 형태로 연계되어 있어야만 한다.

이러한 관점에서 보면, 다양한 다중매체 데이터 사이에 의미를 가질 수 있도록 연관관계를 부여하는 작업을 얼마나 합리적이고 효율적으로 할 수 있느냐 하는 점이 성공적인 다중매체 정보 검색 시스템 구현의 관건이 된다고 할 수 있다. 결국, 그와 같은 다중매체 데이터 간의 상호 연계성 부여는 정교한 다중매체 데이터 통합 편집기의 개발에 의해 구현될 수 있을 것이다.

그러나, 본 과제의 당해년도 연구는 주로 다중매체 시스템의 국내외 현황에 대한 분석과, 하드웨어 인터페이스 부분에 초점을 두어 행해졌기 때문에, 그와 같은 통합 편집기의 설계 및 구현은 차년도 과제로 미루어지게 되었다. 다만, 당해년도에는 이상과 같은 개념의 다중매체 정보 시스템의 효용성에 대해 검증하고 보다 발전된 시스

템 구현의 방향을 모색하고자 하는 목적에서 텍스트 화일형태의 Configuration File에 다중매체 시스템 데이터간의 연계성을 정의하고 이를 토대로 정보검색을 수행하는 형태의 검색 기능 함수들을 개발하고, 이를 여러가지 형태의 응용 프로그램에서 사용할 수 있도록 라이브러리화 하였다.

제 2절 분야별 개발 내용

1. 비디오 인터페이스

가. 개발부문개요

비디오 인터페이스 부문은 모션비디오(동화상)와 정지화상으로 구분된다. 모션비디오의 경우, 부드러운 동작을 연출하기 위해서는 초당 30프레임(한 프레임당 크기는 약 150KB정도)을 디스플레이 하여야 하는데, 현재의 기술로는 하드 디스크등의 저장매체에 저장되어 있는 정지화상을 초당 30프레임이상 디스플레이 하는 것이 용이하지 못하다. 따라서 비디오 영상을 처리하기 위해서 레이저 디스크를 사용하였고, 정지화상을 위해서는 레이저 디스크와 하드 디스크등의 저장매체를 사용하였다.

따라서, 레이저 디스크에 들어 있는 비디오 영상과 카메라, TV, VTR, 레이저 디스크 등으로 부터의 모션 비디오를 캡처링(Capturing)하여 하드 디스크에 저장한 정지화상을 제반 응용 프로그램에서 손쉽게 활용할 수 있도록 레이저 디스크 플레이어를 제어하는 기능, 모션비디오를 캡처링하여 프레임별 정지화상으로 하드 디스크에 저장하는 기능, 저장되어 있는 정지화상을 모니터상에 다양하게 표현시키는 기능, ForeGround 상의 그래픽을 색상별로 제어하는 기능 등을 디지털 비디오 보드상에서 구현할 수 있도록 함수를 제공하였다.

단, 현재의 시스템에서 구현한 비디오 인터페이스 함수는 비디오보드에서 제공하는 모든 기능을 라이브러리화 하지 않았기 때문에 다양한 응용분야에서 이 함수들만으로 적용시키기에는 미흡한 점이 없

지 않다. 이러한 문제는 차년도 과제수행시 보완될 예정이다.

나. 운영환경

비디오 인터페이스는 DVA-4000/ISA 비디오 보드가 지원하는 여러 입·출력 장비중 다음의 장비를 PC 386 시스템에 부착하여 구성하였다.

- Muitisync 모니터
- 레이저 디스크 플레이어 (Laser Disc Player)
- VGA 그래픽 카드

비디오 보드가 지원하는 시스템 화일 및 실생화일들은 MICFILES 디렉토리내에 존재하며 config.sys 화일에 다음과 같이 작성을 해주어야 한다.

```
device=\MICFILES\DVA-4000.SYS/IRQ15
```

```
device=\MICFILES\MIC-DD.SYS /NTSC
```

또한 비디오 보드를 제어하기 위해서 DOS 프롬프트 상에서 다음을 수행하여야 한다.

```
\MICFILES\MIC-LOAD \micfiles\m40dyn.cnf
```

다. 주요 함수의 기능

비디오 인터페이스에 관한 함수 라이브러리에 등록된 함수중 중요한 것은 다음과 같다.

(1) 비디오 보드의 초기화 및 취소에 관련된 함수

— InitVideo()……

비디오 보드를 초기화 시킨다.

- EndVideo()

전화면을 클리어하고 레이저 디스크를 드라이버에서 제거한다.

(2) 환경 설정에 관련된 함수

- SetCoords()

출력화면의 좌표계를 설정한다.

- VideoControl()

비디오, 오디오, 인택스의 출력 여부를 결정한다.

(3) 윈도우 조작에 관련된 함수

- MoveWindow()

디스플레이 할 윈도우의 가로, 세로 시작점의 위치를 지정해 준다.

- SizeWindow()

디스플레이 할 윈도우의 가로, 세로 크기를 지정해 준다.

(4) 화상정보조작에 관련된 함수

- PlayVideo()

레이저 디스크의 화상정보를 다양한 형태로 출력한다.

- SaveImage()

레이저 디스크로부터 입력되어 화면에 출력된 화상정보를 캡처링하여 하드 디스크에 파일로 저장한다.

- DisplayImage()

하드 디스크에 파일로 저장된 화상정보를 화면에 출력한다.

(5) 그래픽에 관련된 함수

— FadeWindow() FadeAll()

Foreground 상의 특정 색상 혹은 모든 색상을 Transparent 하게 하여 Background 의 화상정보를 볼 수 있게 한다.

2. 오디오 인터페이스

가. 개발부문 개요

음성정보를 처리하는 과정이 복잡하고 기능 또한 쓰이는 대상에 따라 음성 인식기나 보이스 에디터 등 매우 다양하다. 그러나 본 연구과제에서는 다중매체 시스템을 구현하는데 있어 도움말 기능이나 텍스트 등에 해당되는 음성데이터를 저장하고 저장된 내용을 스피커를 통해 전달하는데 필요한 기본적인 함수들만을 개발하였다.

나. 운영환경

이 시스템에서 제공되는 함수들은 IBM PC 호환기종에 VP-870 보이스 카드, 마이크, 스피커를 갖춘 시스템에서 사용이 가능하며 TURBOC, MSC 등의 C 컴파일러를 사용하는 응용 프로그램에서 호출할 수 있다.

다. 주요함수의 기능

현재 라이브러리에 등록되어 있는 주요 함수는 다음과 같다.

(1) 음성 데이터의 저장 및 출력

— RecordVoice()

지정된 메모리에 음성 신호를 저장한다.

- PlayVoice ()

지정된 메모리에 로드되어 있는 음성신호를 스피커를 통해 출력한다.

(2) 음성 데이터 화일의 입/출력

- LoadVoice()

음성신호가 저장된 화일로 부터 버퍼로 로드한다.

- SaveVoice ()

음성신호를 화일에 저장한다.

3. 사용자 인터페이스

가. 개발부문 개요

다중매체 시스템에서 사용자 인터페이스로 사용되는 장비는 여러가지가 있다. 그중 터치 스크린은 사용자가 키보드를 사용하지 않고서도 손가락으로 스크린 표면을 가볍게 눌러 원하는 정보를 쉽게 얻을수 있는 점에서 매우 편리한 장비이다.

본 연구과제에서는 터치 스크린 드라이브를 이용하여 터치 스크린으로 들어오는 명령을 받아서 처리할 기본적인 함수들을 개발하였다.

나. 운영환경

이 시스템에서 제공하는 터치 스크린은 IBM PC 호환 기종에 터치 스크린과 콘트롤러카드를 갖추어야 하고 RS-232C 컨넥터를 사용하여 SERIAL Controller 를 본체 COM1 이나 COM2 에 연결한다.

DEVICE DRIVER 는 CONFIG.SYS 에 다음과 같이 작성해주어야 한다.

- DEVICE =KTCAPDI.SYS /* 터치 스크린과의 통신 */
- DEVIKA =KTCTOUCH.SYS/* 터치 스크린 드라이버의 정보를 활용 */

드라이버와의 통신은 일반 DOS 드라이버와 같이 실행되며 “TCH - SCRN” 이란 Device 를 Binary 화일 형식으로 Open 하고 화일을 입·출력하는 것과 마찬가지로 통신하며 TURBOC, MSC 등의 C 컴파일러를 사용하는 응용 프로그램에서 호출할 수 있다.

다. 주요함수의 기능

현재 라이브러리에 등록되어 있는 주요 함수는 다음과 같다.

(1) 좌표 보정의 초기화

- InitCalibration() ...

터치 Pannel 은 X 축, Y 축이 각각 256 등분되어 있어 640 × 480 모드로 좌표를 보정한다.

(2) 좌표 출력

- ReadXY() ...

터치 스크린에서 좌표값을 읽는다.

4. 한글 그래픽 윈도우

가. 개발부문 개요

한글 그래픽 윈도우 부문에서는 다중매체 데이터를 조작하는 제반 응용 프로그램에서 손쉽게 활용할 수 있는 한글 입출력 함수와 이를 토대로 한 한글 윈도우 조작 함수를 개발하여, 이를 다중 매체 시스템의 데이터 조작을 위한 기초 라이브러리에 포함시켰다.

단, 현재의 시스템에서 구현한 한글 입출력 함수는 단일 서체와 단일 크기의 Bitmap 폰트만을 사용하므로 다방면의 응용 시스템에 적용시키기에는 미흡한 점이 적지 않다. 이러한 문제는 차년도 과제 수행시 한글 벡터 폰트의 도입으로 보완할 예정이다.

나. 운영 환경

이 시스템에서 제공하는 한글 그래픽 윈도우 조작 함수들은 IBM PC의 VGA 또는 EGA 그래픽 어댑터가 장착된 시스템에서 사용할 수 있으며, TURBO C, MSC 등의 C Compiler를 사용하는 응용 프로그램에서 호출할 수 있다.

다. 주요함수의 기능

현재의 다중매체 시스템의 데이터 조작 함수 라이브러리에 등록된 한글 윈도우 함수 중 중요한 것은 다음과 같은 것들이다.

(1) 그래픽 환경 설정 및 취소에 관련된 함수

— OpenScreen() ...

스크린 모드를 640 × 480(VGA) 또는 640 × 350 (EGA)의 그

래픽 모드로 설정한다.

- CloseScreen () ...

스크린 모드를 텍스트 모드로 되돌린다.

(2) 한글 입출력 환경의 설정 및 취소에 관련된 함수

- OpenHan () ...

디스크 상에 존재하는 한글 폰트 파일을 열어 한글 폰트 이미지를 메모리로 읽어들이므로써 화면에 한글 문자가 표시될 수 있는 환경을 마련한다.

이때, 사용하고자 하는 한글 코드 시스템이 KS5601 완성형인지 상용조합형인지를 지정해야 한다.

KS5601 완성형을 선택하였을 경우에는 외부 폰트 파일인 ASCII.FNT 와 HAN1.FNT 를 읽어들이고, 상용조합형을 선택하였을 경우에는 ASCII.FNT 와 HAN2.FNT 를 읽어들인다.

- CloseHan()...

메모리 상의 한글 폰트 버퍼를 제거한다.

(3) 한글 문자열의 입출력에 관련된 함수

- DispLine()...

지정된 행, 열에 주어진 한글 문자열을 출력한다.

- EditLine() ...

지정된 행, 열에서 주어진 길이만큼 한글 문자열을 입력받는다. 좌우커서 이동 키와 BS, DEL, INS 키를 편집 키로 사용할 수 있다.

(4) 한글 텍스트 윈도우의 처리에 관련된 함수

— ShowWindow() ...

2 차원 문자 테이블에 담긴 여러 행의 한글 문자열을 화면상에 만들어진 창 (윈도우) 속에서 조회할 수 있는 기능을 제공한다.

— EditWindow() ...

화면상에 일정 규격의 창을 열고 그 안에서 여러 행의 한글 문자열을 편집할 수 있는 기능을 제공한다.

— SelectWindow()...

2 차원 문자 테이블에 담긴 여러 행의 한글 문자열을 화면상에 만들어진 창 (윈도우) 속에서 조회할 수 있는 기능과 함께, 그 중 한행의 정보를 선택할 수 있는 기능을 제공한다. 이 함수는 Pull Down Menu의 구현을 위해 쓰일 수 있다.

5. 다중매체 정보검색 시스템

가. 개발부문 개요

텍스트, 영상, 음성 등 다양한 유형의 데이터를 통합적으로 조작하는 다중매체시스템의 응용분야는 매우 광범위하지만, 그러한 응용시스템의 제반 기능중 가장 일반적인 것은 다량의 데이터 중에서 특정 요건에 부합하는 데이터만을 제한적으로 추출하여 그 내용을 확인하는 “정보검색 기능”이 될 것으로 생각된다.

따라서, 본 연구 과제의 궁극적인 목표 역시 다양한 다중매체 시

스텝 데이터를 자유롭게 구축하고 그 내용을 검색할 수 있는 통합 정보 관리 시스템을 구현하는 데 있다고 할 수 있는데, 이를 위해 1차년도에서는 가장 기본적인 형태의 다중매체 정보검색 시스템의 골격을 구상하고, 이를 응용 시스템 상에서 구현할 수 있는 몇 개의 기초함수를 개발하였다.

물론, 현재 개발된 기초 함수의 기능들은 본 연구팀에서 궁극적으로 구현하고자 하는, 다중매체 데이터를 하이퍼 텍스트적 개념으로 조작하는 “하이퍼미디어”시스템의 개념에는 아직 접근하지 못한 단일 차원의 검색 기능만을 제공하지만, 데이터베이스 구축 및 검색기능을 매우 간편하게 구현할 수 있다는 점에서 나름대로의 효용성을 가질 수 있다고 생각된다.

나. 운영환경

여기서 제공되는 다중매체 정보검색 기능 함수들은 텍스트 및 영상, 음성 데이터의 표현을 위해, 내부적으로 앞에서 소개된 다중매체 데이터 조작 함수들을 호출하므로, 앞 절에서 제시된 운영 환경하에서만 정상적인 기능을 수행할 수 있다.

다. 주요 함수의 기능

(1) 다중매체 데이터 인덱스 생성에 관련된 함수

- CreateNDX() ...

다중매체 데이터 상호간의 연계성을 정의한 Configuration File을 읽어들이어 Index Table을 생성한다. 이 함수의 호출시 Configuration File의 이름을 파라미터로 지정해 주

어야 한다. 이 함수의 기능이 정상적으로 수행되면 Configuration File 과 동일한 파일 이름에 확장자가 “NDX”, “KWD”, “PST” 로 붙은 세 종류의 인덱스 파일이 생성된다. 이 인덱스 파일들은 정보 검색시 참조된다.

(Configuration File 의 확장자는 “SRC” 로 준다)

만일, Configuration File 내에 불합리한 내용이 포함되어 있을 경우, 해당 레코드는 인덱스 생성에서 제외되며, 대신 그 내용은 확장자가 “RJT” 인 Reject 파일에 쓰여지게 된다.

Configuration File 은 다음의 <<예>>에서 보이는 바와 같은 약속에 따라 기술한다.(“/*” 와 “*/” 사이 부분은 주석문)

<<Configuration File 의 작성 예>>

@ 123/*Start of Record */

\$t /*Start of Text */

거북선

조선시대 / 에 왜구 / 의 격퇴 / 를 / 위하여 돌격선 / 으로 / 특수하게
제작 / 된 장갑선 임진왜란 / 당시 이순신 / 에 / 의해 창작 / 된 / 것
이 / 대표적이다.

[그림 1] “이충무공전서”의 통제영귀선

[그림 2] 해군사관학교 복원 거북선 (1980년 제작, 전장 : 113 척)

[그림 3] 청백철화귀선문항아리 / 의 귀선도. 해군사관학교 박물관
소장

\$x /* End of Text */

\$i imagel.img \$v voicel.voc /* Page 1: Still Image and Voice */

\$l 22333 : 34567 /N /* Page 2: Motion Picture */

\$l 40000 /C \$v voice3.voc /* Page 3: 1 Cut and Voice */

@124

\$t

거석기념물

큰 돌 / 로 만든 고대 / 의 기념물. 우리나라 / 의 거석기념물 / 로
는 선돌, 고인돌, 돌널무덤, 돌무지무덤, 환상석열 / 등이 / 있다.

[그림 1] 은울 고인돌. 길이 약 8 m. 황해 은울군 운산리

[그림 2] 월촌 입석. 높이 2.1 m. 전라북도 민속자료 제 7 호. 전
북 김제군

[그림 3] 나주 운흥사 석장승. 높이 2 m. 중요민속자료 제 12 호.
전남 나주군

\$x

\$i \picture \ image 4. img \$v \ voice \ voice 4. voc

\$i \ picture \ image 5. img

\$l 41234: 45987 /M \$v \ voice \ voice 6. voc

<<Configuration File 의 기술 규칙 >>

(가) 각 레코드 (유관한 다중매체 데이터 세트에 관한 기술) 의 시
작은 “@” 문자 (ascii 64) 와 숫자로 된 레코드 고유번호로
표시한다.

(나) 검색 대상이 되는 텍스트 정보의 기술은 식별자 “\$t”(텍

스트 정보의 시작 지시)로부터 "\$x" (텍스트 정보의 종료 지시) 사이에 사용자에게 보여질 형태대로 기술한다.

(다) 텍스트 데이터 중 인덱스 생성에서 제외시킬 부분은 해당 단어 또는 어절 앞에 "/" 표시를 하거나, (이 표시는 정보 출력시에는 감추어지게 된다.) 또는, 문자열 전체를 "["와 "]" 로 묶는다. (이 표시는 정보 출력시 그대로 보여진다.)

(라) 이 텍스트 정보와 관련한 영상 (동화상 및 정지화상) 정보 및 음성정보의 소재를 아래와 같은 방법으로 기술한다.

— 디스크에 저장된 정지 화상 데이터

식별자 "\$i" 와 이미지 화일 이름을 차례로 기술한다. 이 이미지 화일은 비디오 인터페이스 부문에서 소개된 Save Image() 함수에 의해 생성된 이미지 화일이다.

— 디스크에 저장된 음성 데이터

식별자 "\$v" 와 음성 화일 이름을 차례로 기술한다. 이 음성 화일은 오디오 인터페이스 부문에서 소개된 Save Voice() 함수에 의해 생성된 음성 화일이다.

— 레이저 디스크에 저장된 동화상 데이터

식별자 "\$l" 과 레이저 디스크에 수록된 동화상의 시작 프레임 번호 및 최종 프레임 번호 그리고 옵션 기호를 차례로 기술한다.

동화상 출력 옵션은 다음의 여섯가지 중 하나를 선택하여 줄 수 있다.

- /N NORMAL ⇒ 영상과 음성을 정상속도로 출력한다.
- /M MUTE ⇒ 영상만을 출력한다.
- /S SLOW ⇒ 영상만을 느린 속도로 출력한다.
- /F FAST ⇒ 영상만을 빠른 속도로 출력한다.
- /FF VERY_FAST ⇒ 영상만을 매우 빠른 속도로 출력한다.
- /C CUT_BY_CUT ⇒ 영상을 한 프레임씩 출력한다.

— 영상, 음성 데이터간의 동기 부여

다중매체 정보 기술부의 각 행에는 동시에 출력되어야할 영상 데이터와 음성 데이터를 기술한다. 한 행에 기술된 영상 데이터와 음성 데이터는 1 페이지분의 데이터로 묶여 지게 된다.(여기서 페이지란 유관한 화상과 음성을 동시에 조작하기 위한 동기 번호를 의미한다.)

(2) 다중매체 데이터 검색 기능에 관련된 함수

— InitQuery()

정보 검색을 위해 다중매체 데이터의 상호 연관 관계를 지정한 Configuration File 과 각종 인덱스 파일을 Open 한다.

파일 이름을 파라미터로 지정하면, 해당 파일 이름에 확장자가 "SRC", "NDX", "KWD", "PST" 로 붙은 파일들이 Open 된다.

각 인덱스 파일의 기능은 다음과 같다.

*.src: 사용자가 편집한 다중매체 정보 Configuration File

- *.ndx:Configuration File 의 내용에 대한 인덱스 테이블
- *.kwd:Configuration File 의 Text 부문에서 추출된 키워드
리스트
- *.pst:kwd 파일과 ndx 파일의 정보를 연계시켜 주는 인덱스

— ExecQuery() ...

파라미터로 전달된 검색 명령문에 따라 정보 검색 작업을 시행하고, 그 결과 (선택된 레코드 번호)를 돌려 준다.

검색은 개별 단어 검색과 논리 검색이 가능하다. 논리 검색에 사용되는 연산자는 &(and) 와 |(or) 두가지이며, 괄호를 사용하여 우선되는 조건을 묶어 줄 수 있다.

<<예 >>

- 거북선 & 이순신
- 거북선 | 돌무덤
- (민속자료 | 중요민속자료) & 고대기념물

— EndQuery() ...

검색작업 종료시, 또는 다른 분야의 정보를 검색하고자 할 때, 현재 사용중이던 각종 인덱스 파일을 Close 한다.

(3) 다중매체 정보 출력에 관계된 함수

— BrowseAll(), BrowseResult() ...

전체 레코드 또는 검색 결과 얻어진 다수의 레코드의 내용을 열람하고 그 가운데 특정 레코드를 선택할 수 있는 기능을 제공한다.

— ShowText() ...

선택된 특정 레코드의 텍스트 부분을 화면에 출력한다.

— ShowImage() ...

선택된 특정 레코드의 데이터 중 정지화상 (Still Image) 을 페이지 단위로 화면에 출력한다.

(여기서 페이지란 유관한 화상과 음성을 동시에 조작하기 위한 동기 번호를 의미한다.)

— PlayVoice() ...

선택된 특정 레코드의 데이터 중 음성 데이터를 페이지 단위로 음향장치에 출력한다.

— PlayMovie () ...

선택된 특정 레코드의 데이터 중 동화상 데이터를 페이지 단위로 화면 및 음향장치에 출력한다.

제 5 장 결 론

제 5 장 결 론

1. 연구추진

본 과제는 3개년 연구개발 프로젝트중 1차년도의 과제로서 다중매체 데이터베이스를 구축하기 위한 Tool 과 그 응용시스템 및 한국형 하이퍼미디어 시스템의 개발을 위한 기초연구 단계로써 다중매체 시스템의 각 표현매체별 인터페이스 방법에 대하여 연구하였고, 다중매체의 정보관리 방법론에 대하여 연구함으로써 관계형 데이터베이스의 확장개념을 도입하여 그 기초함수 및 시범시스템을 개발하는데 초점을 맞추었다.

기존의 데이터베이스 시스템위에 비디오, 오디오, 한글텍스트 등의 처리기능을 추가하여 시스템의 상부구조에서 응용프로그램이나 유틸리티들을 고안할 때 이들을 효과적으로 사용할 수 있게 하였다.

또한 다양한 표현 매체별 관련 산업분야의 국·내외 기술현황, 하드웨어·소프트웨어의 개발현황 등에 대한 조사분석도 소홀히 하지 않았다.

2. 활용방안 및 기대효과

가. 활용방안

- (1) 학생들에게 교육을 시키기 위한 시청각 교육용으로 생생한 동화상 정보까지 지원해 줌으로 거의 직접 경험에 가까운

산교육을 실시할 수 있다. 보다 상세하고 구체적인 정보의 전달을 통해 교육효과를 증진시킬 수 있다.

- (2) 기업이나 회사의 경우에도 신입사원 교육이나 특수훈련, 전문가 양성, 제품소개 및 홍보, 각종 문서관리용 등으로 다중매체 시스템을 활용할 수 있다.
- (3) 박물관의 자료 수록, 민속자료 수록, 고대역사에 관한 기록 수록 등 인문과학 분야 뿐만 아니라 CAD/CAM 과 같은 응용시스템 개발도구로 이용할 수 있다.

나. 기대효과

- (1) 다중매체 시스템 소프트웨어 기술을 향상시킨다.
- (2) 기업체와 연계하여 연구결과를 교육훈련용, 프리젠테이션용 등의 상품화로 유도할 수 있다.
- (3) 다중매체 시스템의 고급기술인력을 확보할 수 있다.
- (4) 각 매체별 관련기술을 통합하여 데이터베이스에 연계시키는 고급기술과 사용자 인터페이스에 대한 체계적 고찰능력을 배양할 수 있다.

3. 문 제 점

개발과정을 통하여 볼 때 국내의 하드웨어와 소프트웨어 기술은 모두 다중매체 데이터베이스 관련기술에 있어 후진성을 면하지 못하고 있다. 보다 다양한 비디오 및 오디오 데이터의 편집 소프트웨어가 개발되어야 하며, 그래픽·애니메이션 부문의 편집 소프트웨어 역시 국내실정에 맞게 한글 벡터폰트(Vector Font)를 처리할 수 있

는 기능을 추가하여야 한다.

또한 각 표현매체별 오디오, 비디오, 사진, 폰트 등의 데이터 변환 및 압축, 패턴인식 및 처리 등에 대한 연구가 체계적으로 진행되어야 한다.

4. 제 언

다중매체 관련 연구와 산업기술분야의 발전이 균형과 조화를 이루어야겠지만 각 분야별 기술을 총체적으로 통합·관리할 수 있는 기술과 데이터베이스에의 접목기술은 매우 비중있는 분야로 자리잡게 될 것이며 단순한 컴퓨터 부분의 기술로써 그치지 않고 예술적 감각을 함께 실현시키는 시스템 기술로 발전시켜 나가야 할 것이다.

관계형 데이터베이스의 확장 개념뿐 아니라 객체 지향형 혹은 분산형 데이터베이스 개념까지 도입하여 데이터베이스 관리자나 사용자의 입장에서 효율적으로 관리·사용할 수 있는 시스템으로 발전시켜야 한다.

참 고 문 헌

1. Kuan-Tsae Huang, "Multidatabase for Multimedia Computing", pp. 21-84, 1991. 3, Proceedings of Int'l Workshop on New Database Technology.
2. Scott Swix, "IBM Multimedia Product Chronology", pp. 26-32, 1990, IBM Personal Systems Developer.
3. "The Benefits of Graphical User Interface", 1990. Temple, Barkers, & Sloane, Inc., A Report on New Primary Research.
4. Sandra Morris, "Multimedia Application Development", pp. 20-23, 1990. 9.10, Microcomputer Solution.
5. "The Multimedia Roadmap Unfolds", 1990. 2, Intel.
6. "Videodisc Technology", 1988, The Learning Center of Lister Hill National Center for Biomedical Communications.
7. "Interactive Media Systems: Authoring Systems", 1991, Philips.
8. G. David Ripley, "DVI-A Digital Multimedia Technology", pp. 811-823, VOL. 32, NO. 7, 1989, Communications of the ACM.
9. Karen A. Frenkel, "The Next Generation of Interactive Technology", pp. 872-832, VOL. 32, NO. 7, 1989, Communications of the ACM.
10. Phillip Robinson, "The Four Multimedia Gospels", pp. 203-

- 214, VOL. 15, NO. 2, 1990, BYTE.
11. Tim Shetler, "Birth of the BLOB", pp. 221-228, VOL. 15, NO. 2, 1990, BYTE.
 12. Greg Loveria & Don Kinstler, "Multimedia? DVI Arrives", pp. 105-108, VOL. 15, NO. 11, 1990, BYTE.
 13. Frank Hayes & Nick Baran, "A Guide to GUIs", pp. 250-256, VOL. 14, NO. 7, 1989, BYTE.
 14. Mark D. Veljkov, "Managing Multimedia", pp. 227-232, VOL. 15, NO. 8, 1990, BYTE.
 15. Stephen Satchell, "Mexa Floppies", pp. 301-309, VOL. 15, NO. 10, 1990, BYTE.
 16. David A. harvey, "State of the Media", pp. 275-282, VOL. 15, NO. 12, 1990, BYTE.
 17. L. Brett Glass, "Digital Video Interactive", pp. 283-289, VOL. 14, NO. 5, 1989, BYTE.
 18. "VideoLogic Guide to Multimedia Development Tools and Resources", pp. 1-25, VideoLogic Inc.
 19. "VideoLogic Training Course Note", 1990. VideoLogic Inc.
 20. "LaserDisc LaserVision System Guidebook", 1988, Pioneer.
 21. Ben Shneiderman & Greg Kearsley, "Hypertext Hands-on", 1989, Addison-Wesley Publishing Company.

22. 최양희, “멀티미디어 입문”, 1991.1.
23. 양유길 외, “다중매체 데이터베이스 시스템에 관한 연구”, 1990.12,
한국과학기술연구원 부설 시스템공학연구소
24. “멀티미디어 기술동향 조사보고서”, 1990.12. 한국데이터통신주식회사.
25. 한상기, “멀티미디어 : 현황과 전망”, pp.31-41, 삼성종합기술원
26. “차세대 멀티미디어 시장 노린 야심작 ‘매킨도시 II si’”,
pp.175-178, 1990.12, 컴퓨터.
27. “멀티미디어 응용분야”, pp.80-94, 1990.10, 소프트웨어.
28. “90 년대의 PC 응용”, pp.162-168, 1990.2, 컴퓨터월드.
29. “한국형 저작도구 시사회”, pp.6-24, 1991.2, 현대전자산업주식회사.
30. “멀티미디어에 관한 연구 동향”, pp.5-11, 1988.4, 정보통신기술.
31. “다가오는 멀티미디어의 세계”, pp.168-175, 1990.11, 컴퓨터월드.
32. “멀티미디어란 무엇인가?”, pp.68-116, 1990.10, 소프트웨어.
33. 이주현, “90 년대의 신상품, Multi-Media 의 개념과 활용사례”,
1991.2, 멀티마인드.
34. 김봉일 외, “대전 엑스포 ‘93 전산 기초조사 및 기본계획 보고서”,
1990.10, 한국과학기술연구원 부설 시스템공학연구소.
35. 이재용, “멀티미디어 통신 시스템”, pp.171-207, 1990.12, “멀티미디어 통신 단기강좌 논문집”, 한국정보과학회 정보통신연구회.

부록 : 라이브러리 함수

| | |
|---------------------|-----|
| 1. 헤더 파일 | 213 |
| 2. 분야별 함수 | 220 |
| 가. 비디오 인터페이스 | 220 |
| 나. 오디오 인터페이스 | 235 |
| 다. 사용자 인터페이스 | 241 |
| 라. 한글 그래픽 윈도우 | 250 |
| 마. 다중매체 정보검색 | 272 |

1. 헤더 파일

```
/*-----*/  
/*                                           */  
/*           MULTI.H                       */  
/*           ( header file )               */  
/*                                           */  
/*-----*/
```

```
/* single key define */
```

```
#define LF      10  
#define CR      13  
#define Space  32  
  
#define BkSp    8  
#define Tab     9  
#define Enter  13  
#define Esc    27  
  
#define CtrA    1  
#define CtrB    2  
#define CtrC    3  
#define CtrD    4  
#define CtrE    5  
#define CtrF    6  
#define CtrG    7  
#define CtrH    8  
#define CtrI    9  
#define CtrJ   10  
#define CtrK   11  
#define CtrL   12  
#define CtrM   13  
#define CtrN   14  
#define CtrO   15  
#define CtrP   16  
#define CtrQ   17  
#define CtrR   18  
#define CtrS   19  
#define CtrT   20  
#define CtrU   21  
#define CtrV   22  
#define CtrW   23  
#define CtrX   24  
#define CtrY   25  
#define CtrZ   26
```

/* double key define */

```
#define Left      375
#define Right     377
#define Up        372
#define Down      380
#define Home      371
#define End       379
#define PgUp      373
#define PgDn      381
#define Ins       382
#define Del       383
#define BkTab     315

#define F1        359
#define F2        360
#define F3        361
#define F4        362
#define F5        363
#define F6        364
#define F7        365
#define F8        366
#define F9        367
#define F10       368

#define AltA      330
#define AltB      348
#define AltC      346
#define AltD      332
#define AltE      318
#define AltF      333
#define AltG      334
#define AltH      335
#define AltI      323
#define AltJ      336
#define AltK      337
#define AltL      338
#define AltM      350
#define AltN      349
#define AltO      324
#define AltP      325
#define AltQ      316
#define AltR      319
#define AltS      331
#define AltT      320
#define AltU      322
#define AltV      347
#define AltW      317
#define AltX      345
#define AltY      321
#define AltZ      344
```

```

/* ERROR Code */

#define    INVALID_DATA_TYPE    -11
#define    INCORRECT_REC_NO     -10
#define    MEMORY_FAULT         -9
#define    SYNTAX_ERROR         -8
#define    D/B_NOT_INITIATED    -7
#define    FONT_FILE_NOT_FOUND  -6
#define    PST_FILE_NOT_EXIST   -5
#define    KWD_FILE_NOT_EXIST   -4
#define    SRC_FILE_NOT_EXIST   -3
#define    NDX_FILE_NOT_EXIST   -2
#define    DEVICE_ERROR         -1
#define    OK                    0

/* playVideo, PlayMovie */

#define    NORMAL                0x00
#define    REVERSE               0x01
#define    SLOW                  0x02
#define    FAST                  0x04
#define    VFAST                 0x08
#define    MUTE                  0x10
#define    CUT_BY_CUT            0x20

#define    CURRENT                20
#define    NO                     21

/* VideoControl, FadeWindow, FadeAll */

#define    ON                     30
#define    OFF                    31

/* ReadStatus */

#define    COORDS                 40
#define    POSITION                41
#define    SIZE                   42

/* MoveWindow, SizeWindow */

#define    LD                     50
#define    HD                     51

/* FadeWindow */

#define    END                    60

```

```

/* Touch Screen */

#define CROUP_OFFSET      2058L      /* 0x080a */
#define PAD_OFFSET       2568L      /* 0x0a08 */
#define usr_bar( x, y )  bar( x * 127, y * 119, x * 127 + 4, y * 119 + 4 )

/* OpenHan */

#define KS5601            70
#define COMBINATION      71

/* RestWindow */

#define QUICK            80
#define SPLIT           81

/* EditWindow */

#define SHOW             90
#define EDIT            91
#define TRIM            92
#define UPPER           93
#define NO_CLEAR        94
enum COLOR {
    /* dark colors */
    BLACK, BLUE, GREEN, CYAN, RED, MAGENTA, BROWN, DARKGRAY,
    /* light colors */
    LIGHTGRAY, LIGHTBLUE, LIGHTGREEN, LIGHTCYAN, LIGHTRED,
    LIGHTMAGENTA, YELLOW, WHITE
};

typedef struct {
    int y1; /* start row */
    int x1; /* start col */
    int y2; /* end row */
    int x2; /* end col */
    void *p; /* pointer to window buffer */
} FRAME;

struct data
{
    char x1lsb;
    char x1msb;
    char y1lsb;
    char y1msb;
    char x2lsb;
    char x2msb;
    char y2lsb;
    char y2msb;
}

```

```

    char padname[13];
    int group;
    char group_str[4];
    char x1[6];
    char y1[6];
    char x2[6];
    char y2[6];
} pad[256];

```

```

/*-----*/
/*   Video Interface   */
/*-----*/

```

```

int   VideoCmd( char command );
void  InitVideo();
void  EndVideo();
int   SetCoords( int xsiz, int ysiz );

```

```

void  PlayVideo( unsigned int sframe, unsigned int eframe, unsigned int option );
void  VideoControl( int vid, int aud, int ind );

```

```

char  *myncpy( char *dest, char *src, int nbyte );
int   assignValue( int *x, int *y );
int   ReadStatus( int status, int *x, int *y );

```

```

int   MoveWindow( int status, int x, int y );
int   SizeWindow( int status, int x, int y );

```

```

int   SaveImage( char *filename );
int   DisplayImage( char *filename );
int   FadeWindow( int status, int time, int color, ... );
int   FadeAll( int status, int time );

```

```

/*-----*/
/*   Audio Interface   */
/*-----*/

```

```

void  VoiceRecord( unsigned int segment, unsigned int offset, unsigned int length );
void  Voiceplay( unsigned int segment, unsigned int offset, unsigned int length );
void  VoiceStop();
void  VoiceLoad( char *file, unsigned char *buffer, unsigned int length );
void  VoiceSave( char *file, unsigned char *buffer, unsigned int length );

```

```

/*-----*/
/*   User Interface(Touch Screen) */
/*-----*/

```

```

void  InitPadDump ( char argc, char *argv[] );

```

```

char    *Upcase ( char *c );
void    InitCalibration ( int mode );

int     ReadXY( int *xc, int *yc );
int     ReadXYBuffer ( int *xc, int *yc );
int     XCalibration ( int mm );
int     YCalibration ( int nn );

void    cal_init();
void    FlushBuffer();

/*-----*/
/*    Hangeul Interface    */
/*-----*/

/* Session Level Funtion */

int     OpenScreen();
void    CloseScreen();
int     OpenHan( int code_system );
Void    CloseHan();

/* Basic I/O */

int     GetKey();
void    DispLine( int row, int col, char *string, int color1, int color2 );
int     EditLine( int row, int col, char *string, int length, int color1,
                 int color2 );

/* Window Manipulation */

int     MakeFrame( FRAME *frame, int row1, int col1, int row2, int col2 );
void    KillFrame( FRAME *frame );

int     SaveWindow( FRAME *frame );
void    RestWindow( FRAME *frame, int option );
void    ClearWindow( FRAME *frame, int color );
int     EditWindow( FRAME *frame, char **string_tbl, int lines, int width,
                  int color1, int color2, int option, int *bel );
int     ShowWindow( FRAME *frame, char **string_tbl, int lines, int width,
                  int color1, int color2 );
int     SelectWindow( FRAME *frame, char **string_tbl, int lines, int width,
                    int color1, int color2, int color3, color4, int *break_point );

```

```

/*-----*/
/*      Query Information      */
/*-----*/

/* Create Database Files */

int      CreateNDX( char *dbid );

/* Basic Query Function */

int      InitQuery( char *dbid );
int      ExecQuery( char *query_command, unsigned int *result_tbl, int limit );
void     EndQuery();

/* Function for result manipulation */

int      BrowseAll( FRAME *frame, int color1, int color2, int color3, int color4,
                  int *break_point );
int      BrowseResult( unsigned int *result_tbl, int num, FRAME *frame, int color1,
                    int color2, int color3, int color4, int *break_point );
int      GetRecordInfo( unsigned int rec_no, unsigned int *data_type, int *page );
int      ShowText( unsigned int rec_no, FRAME *frame, int color1, int color2,
                 int *page );

int      ShowImage( unsigned int rec_no, FRAME *frame, int color, int *page );
int      PlaySound( unsigned int rec_no, int *page );
int      PlayMovie( unsigned int rec_no, FRAME *frame, int color, int *option,
                  int *page );

```

2. 분야별 함수

가. 비디오 인터페이스

| 함수명 | 서식 | 기능 |
|--------------|---|---------------------------------------|
| VideoCmd | int VideoCmd(command); char *command; | 비디오 보드가 제공하는 명령을 직접 수행 |
| InitVideo | void InitVideo(); | 비디오 보드를 초기화 |
| EndVideo | void EndVideo(); | 화면을 지우고 레이저 디스크를 드라이버에서 제거 |
| SetCoords | int SetCoords(xsiz, ysiz); int xsiz, ysiz; | 출력화면의 좌표계를 설정 |
| PlayVideo | void PlayVideo(sframe, eframe, option); unsigned int sframe, eframe; unsigned int option; | 레이저 디스크의 내용을 다양한 형태로 출력 |
| VideoControl | void VideoControl(vid, aud, ind); int vid, aud, ind; | 레이저 디스크의 비디오, 오디오, 인덱스의 On 또는 Off |
| ReadStatus | int ReadStatus(status, x, y); int status; int *x, *y; | 현재 화면의 좌표계, 윈도우 위치, 크기에 관한 정보 제공 |
| MoveWindow | int MoveWindow(status, x, y); int status, x, y; | 이미지의 윈도우 위치를 임의의 위치에 지정 |
| SizeWindow | int SizeWindow(status, x, y); int status, x, y; | 이미지의 윈도우 크기를 임의의 크기로 지정 |
| SaveImage | int SaveImage(filename); char *filename; | 레이저 디스크의 비디오 이미지를 하드 디스크에 저장 |
| DisplayImage | int ShowImage(filename); char *filename; | 하드 디스크에 저장되어 있는 정지 화상을 화면에 전화면 크기로 출력 |
| FadeWindow | int FadeWindow(status, time, color, ...); int status, time, color; | Foreground 상의 특정한 색상을 Fade On 또는 Off |
| FadeAll | int FadeaAll(status, time); int status, time; | 모든 Foreground 상의 색상을 Fade On 또는 Off |

VideoCmd()

1. 함수명 : VideoCmd

2. 문 법 : #include <multi.h>
int VideoCmd(command)
char *command;

3. 기 능 : 비디오 보드가 제공하는 명령어를 직접 수행한다.

4. 인 수 :
. command : 비디오 보드가 제공하는 명령어 및 그 명령어에 부합되는
인수들

5. 반환값 :
. 정상종료 : OK
. 실패시 :
- DEVICE_ERROR

6. 관련함수 : ReadStatus()

7. 사용예 :

```
#include <multi.h>
.
main()
{
    char command[64];

    /* 비디오 보드를 초기화, InitVideo() 함수와 동일 */
    VideoCmd( "init 4000" );

    /* 비디오, 오디오, 인덱스를 모두 On */
    VideoControl( ON, ON, ON );

    /* 레이저 디스크의 1000-2000 프레임의 비디오를 (100, 100) 위치에
    출력 */
    Play( 1000, 2000, NORMAL );
    MoveWindow( LD, 100, 100 );

    /* 화면을 지우고 레이저 디스크를 드라이버에서 제거 */
    EndVideo();
}
```

InitVideo()

1. 함수명 : InitVideo
2. 문 법 : #include <multi.h>
void InitVideo()
3. 기 능 : 비디오 보드를 초기화 시킨다.
4. 인 수 : 없음
5. 반환값 : 없음
6. 관련함수 : EndVideo()
7. 사용예 : PlayVideo 참조

EndVideo()

1. 함수명 : EndVideo
2. 문 법 : #include <multi.h>
void EndVideo()
3. 기 능 : 전환면을 클리어 하고 레이저 디스크를 드라이버에서 제거한다.
4. 인 수 : 없음
5. 반환값 : 없음
6. 관련함수 : InitVideo()
7. 사용예 : PlayVideo 참조

SetCoords()

1. 함수명 : SetCoords

2. 문 법 : #include <multi.h>
int SetCoords(xsiz, ysiz)
int xsiz, ysiz;

3. 기 능 : 출력 화면의 좌표계를 설정한다.

4. 인 수 :
· xsiz : 출력 화면의 가로 좌표계
(단 40 <= xsiz <= 2000)
· ysiz : 출력 화면의 세로 좌표계
(단 25 <= ysiz <= 2000)

5. 반환값 :
· 정상종료 : OK
· 실패시 :
- DEVICE_ERROR

6. 관련함수 : 없음

7. 사용예 :

```
#include <multi.h>

main()
{
    InitVideo();

    /* 비디오, 오디오, 인덱스를 모두 On */
    VideoControl( ON, ON, ON );

    /* 화면을 가로 80, 세로 25로 좌표계를 설정한 경우 */
    SetCoords( 80, 25 );

    /* 레이저 디스크의 1000-2000 프레임의 비디오를 (10, 10) 위치에
    출력 */
    Play( 1000, 2000, NORMAL );
    MoveWindow( LD, 10, 10 );
}
```

PlayVideo()

1. 함수명 : PlayVideo

2. 문 법 : #include <multi.h>
void PlayVideo(sframe, eframe, option)
unsigned int sframe, eframe, option;

3. 기 능 : 레이저 디스크의 내용을 출력하는 명령으로 다음과 같이 다양한 형태로 출력이 가능하다.

- . 현재 프레임으로 부터의 내용을 출력
 - . 지정한 프레임으로 부터의 내용을 출력
 - . 지정한 프레임으로 부터 지정한 프레임까지의 내용을 출력
 - . 현재 프레임으로 부터 지정한 프레임까지의 내용을 출력
- 이경우 아래의 option중에서 reverse, reverse와 vfast 조합, reverse와 slow 조합 기능은 사용 불가능하다.

이때 option을 지정하여 각각의 출력에 대하여 다음과 같은 기능을 조합할 수 있다.

- . normal : 정상적으로 비디오, 오디오 출력
- . reverse : 역방향으로 비디오 출력, 오디오 출력 안됨
- . fast : 보통속도 보다 빠르게 비디오 출력, 오디오 출력 안됨
- . slow : 보통속도 보다 느리게 비디오 출력, 오디오 출력 안됨
- . vfast : 보통속도 보다 매우 빠르게 비디오 출력, 오디오 출력 안됨
- . reverse와 vfast 조합
- . reverse와 slow 조합

4. 인 수 :

- . sframe : 출력을 시작할 프레임
 - 프레임 번호
 - CURRENT : 현재의 프레임
- . eframe : 출력을 끝낼 프레임
 - 프레임 번호
 - NO : 프레임 번호를 지정하지 않을 경우
- . option : 여러가지 다양한 기능 제공
 - NORMAL : option을 선택하지 않고 출력
 - REVERSE : reverse 기능을 추가
 - FAST : fast 기능을 추가
 - SLOW : slow 기능을 추가
 - VFAST : vfast 기능을 추가
 - REVERSE | VFAST : reverse 기능과 vfast 기능을 동시에 추가
 - REVERSE | SLOW : reverse 기능과 slow 기능을 동시에 추가

5. 반환값 : 없음

6. 관련함수 : VideoControl()

7. 사용예 :

```
#include <multi.h>

main()
{
    Initvideo();

    /* 비디오, 오디오, 인덱스를 모두 On */
    VideoControl( ON, ON, ON );

    /* 현재 프레임으로 부터의 내용을 option선택없이 출력할 경우 */
    PlayVideo( CURRENT, NO, NORMAL );

    /* 5000번 프레임으로 부터의 내용을 reverse로 출력할 경우 */
    PlayVideo( 5000, NO, REVERSE );

    /* 1000번 프레임으로 부터 3000번 프레임까지의 내용을 reverse와
       slow 기능을 동시에 추가하면서 출력할 경우 */
    PlayVideo( 1000, 3000, REVERSE | SLOW );

    /* 현재 프레임으로 부터 30000번 프레임까지의 내용을 vfast 기능을
       추가하면서 출력할 경우 */
    PlayVideo( CURRENT, 30000, VFAST );

    EndVideo();
}
```

VideoControl()

1. 함수명 : VideoControl
2. 문 법 :

```
#include <multi.h>
void VideoControl( vid, aud, ind )
int vid, aud, ind;
```
3. 기 능 : 비디오, 오디오, 인덱스의 순서로 각각 on, off를 결정한다.
해당되는 인수에 "ON", "OFF", "MUTE" 등을 사용하므로써 각각의 기능을 임의로 조합할 수 있다.
단 비디오 off 상태에서는 인덱스를 on 하여도 인덱스가 표시되지 않는다.
PlayVideo() 함수와 적절히 조합하면 다양한 기능이 가능하다.
4. 인 수 :
 - . vid : 비디오의 출력 여부
 - ON : 출력함
 - OFF : 출력하지 않음
 - . aud : 오디오의 출력 여부
 - ON : 출력함
 - MUTE : 출력하지 않음
 - . ind : 해당되는 프레임의 인덱스 출력 여부
 - ON : 출력함
 - OFF : 출력하지 않음
5. 반환값 : 없음
6. 관련함수 : PlayVideo()
7. 사용예 : PlayVideo 참조

ReadStatus()

1. 함수명 : ReadStatus

2. 문 법 : #include <multi.h>
int ReadStatus(status, x, y)
int status;
int *x, *y;

3. 기 능 : 현재의 화면 좌표계에 대한 정보나 윈도우 위치, 크기에 관한 정보를 시스템으로 부터 Query하여 사용자가 이용할 수 있도록 한다.

4. 인 수 :

- . status : 상수 COORDS, POSITION 또는 SIZE 중 선택
 - COORDS : 현재 화면에서 사용되고 있는 좌표계의 값
 - POSITION : 현재 윈도우의 좌측상단의 좌표값
 - SIZE : 현재 윈도우의 우측하단의 좌표값
- . x : 각 status에 따라 x축의 좌표값을 돌려주는 인수
- . y : 각 status에 따라 y축의 좌표값을 돌려주는 인수

5. 반환값 :

- . 정상종료 : OK
- . 실패시 :
 - DEVICE_ERROR

6. 관련함수 : VideoCmd()

7. 사용예 :

```
#include <multi.h>

main()
{
    int x, y ;

    /* 현재 윈도우의 좌측상단의 좌표값을 Query하여 출력 */
    ReadStatus( POSITION, &x, &y );
    printf( "%d\t%d", x, y );
}
```


MoveWindow()

1. 함수명 : MoveWindow

2. 문 법 : #include <multi.h>
int MoveWindow(status, x, y)
int status, x, y;

3. 기 능 : 디스플레이할 윈도우의 좌측 상단의 위치를 지정해 준다.

4. 인 수 :

- . status : 상수 LD 또는 HD 중의 하나를 선택
 - LD : 레이저 디스크의 비디오를 제어
 - HD : 하드 디스크의 정지 화상을 제어
- . x : 새로운 윈도우를 디스플레이하고자 하는 위치의 x좌표 값
- . y : 새로운 윈도우를 디스플레이하고자 하는 위치의 y좌표 값

5. 반환값 :

- . 정상종료 : OK
- . 실패시 :
 - DEVICE_ERROR

6. 관련함수 : SizeWindow()

7. 사용예 :

```
#include <multi.h>

main()
{
    InitVideo();
    VideoControl( ON, ON, ON );
    PlayVideo( 1000, 1500, NORMAL );

    /* 하드 디스크에 sample.img 파일로 이미지를 저장함 */
    SaveImage( "sample.img" );

    /* 하드 디스크에 저장된 sample.img 파일을 화면에 출력함 */
    DisplayImage( "sample.img" );

    /* 화면에 출력된 이미지를 (200, 200) 크기의 윈도우로 지정 */
    SizeWindow( HD, 200, 200 );

    /* 윈도우의 좌측 상단의 위치를 (150, 100) 으로 지정 */
    MoveWindow( HD, 150, 100 );
}
```

SizeWindow()

1. 함수명 : SizeWindow
2. 문 법 :

```
#include <multi.h>
int SizeWindow( status, x, y )
int status, x, y;
```
3. 기 능 : 디스플레이할 윈도우의 가로, 세로 크기를 지정한다.
4. 인 수 :
 - . status : 상수 LD 또는 HD 중 선택
 - LD : 레이저 디스크의 비디오를 제어
 - HD : 하드 디스크의 정지 화상을 제어
 - . x : 새로운 윈도우를 디스플레이하고자 하는 위치의 x좌표의 우측 값
 - . y : 새로운 윈도우를 디스플레이하고자 하는 위치의 y좌표의 하단 값
5. 반환값 :
 - . 정상종료 : OK
 - . 실패시 :
 - DEVICE_ERROR
6. 관련함수 : MoveWindow()
7. 사용예 : MoveWindow 참조

SaveImage()

1. 함수명 : SaveImage
2. 문 법 :

```
#include <multi.h>
int SaveImage( filename )
char *filename;
```
3. 기 능 : 레이저 디스크로부터 화면에 출력된 비디오 이미지를 하드 디스크에 파일로 저장한다.
4. 인 수 :
 - . filename : 비디오 이미지를 하드 디스크에 저장할 때의 파일명
항상 전화면 이미지로 저장됨.
5. 반환값 :
 - . 정상종료 : OK
 - . 실패시 :
 - DEVICE_ERROR
6. 관련함수 : DisplayImage()
7. 사용예 : MoveWindow 참조

DisplayImage()

1. 함수명 : DisplayImage
2. 문 법 :

```
#include <multi.h>
int DisplayImage( filename )
char *filename;
```
3. 기 능 : 하드 디스크에 저장된 정지 화상을 화면에 출력한다.
4. 인 수 :
 - . filename : 화면에 출력하고자 하는 하드 디스크에 저장되어 있는 정지 화상의 화일명
5. 반환값 :
 - . 정상종료 : OK
 - . 실패시 :
 - DEVICE_ERROR
6. 관련함수 : SaveImage()
7. 사용예 : MoveWindow 참조

FadeWindow()

1. 함수명 : FadeWindow
2. 문 법 :

```
#include <multi.h>
int FadeWindow( status, time, color, ... )
int status, time, color;
```
3. 기 능 : 비디오 보드의 비디오 이미지는 항상 Background에 디스플레이 되며, 그래픽 이미지는 Foreground에 디스플레이 된다. FadeWindow() 함수는 Foreground상의 각종 색상중에서 특정 색상의 부분만을 Transparent하게 하여 Background의 이미지를 볼 수 있게 한다.
4. 인 수 :
 - . status : 특정 색상의 Transparent ON 또는 OFF
 - OFF : Foreground상의 특정 색을 Transparent
 - ON : Transparent되었던 색상의 복귀
 - . time : FadeWindow() 함수가 실행되면서 부터 Fade가 완료되는데 까지 걸리는 시간 (단위 : 1/100 초)
 - . color : Foreground의 색상 가운데서 Transparent하고자 하는 색상 사용자가 원하는 색상을 1개 이상 몇 개 까지라도 가능하며 끝에는 반드시 END라는 인수를 써야함.
5. 반환값 :
 - . 정상종료 : OK
 - . 실패시 :
 - DEVICE_ERROR
6. 관련함수 : FadeAll()
7. 사용예 :

```
#include <multi.h>

main()
{
    /* Foreground 상의 BLUE와 GREEN이 Fade되는데 1초가 소요 */
    FadeWindow( OFF, 100, BLUE, GREEN, END );
}
```

FadeAll()

1. 함수명 : FadeAll
2. 문 법 :

```
#include <multi.h>
int FadeAll( status, time )
int status, time;
```
3. 기 능 : 모든 Foreground 색상을 Transparent하게 하여 Background의 이미지를 볼 수 있게 하여준다.
4. 인 수 :
 - . status : 모든 색상의 Transparent ON 또는 OFF
 - OFF : Foreground상의 모든 색상을 Transparent
 - ON : Transparent되었던 모든 색상의 복귀
 - . time : FadeWindow() 함수가 실행되면서 부터 Fade가 완료되는데 까지 걸리는 시간 (단위 : 1/100 초)
5. 반환값 :
 - . 정상종료 : OK
 - . 실패시 :
 - DEVICE_ERROR
6. 관련함수 : FadeWindow()
7. 사용예 :

```
#include <multi.h>

main()
{
    /* Foreground 상의 모든 색상이 Fade되는데 1초가 소요 */
    FadeAll( OFF, 100 );
}
```

나. 오디오 인터페이스

| 함수명 | 서 식 | 기 능 |
|-------------|---|---------------------------|
| VoiceRecord | void VoiceRecord(segment, offset, length); unsigned int segmenu; unsigned int offset; unsigned int length; | 지정된 메모리에 음성 신호를 저장 |
| VoicePlay | void VoicePlay (segment, offset, length); unsigned int segmenu; unsigned int offset; unsigned int length; | 지정된 메모리에 로드되어 있는 음성신호를 실행 |
| VoiceStop | void VoiceStop (); | 출력중인 음성을 중지 |
| VoiceLoad | void VoiceLoad (file, buffer, length); char *file; unsigned char *buffer; unsigned int length; | 음성신호가 저장된 파일로부터 버퍼로 로드 |
| VoiceSave | void VoiceSave (file, buffer, length); char *file; unsigned char *buffer; unsigned int length; | 음성신호를 파일에 저장 |

VoiceRecord()

1. 함수명 : VoiceRecord

2. 문 법 : #include <multi.h>
void VoiceRecord(segment, offset, length)
unsigned int segment, offset, length;

3. 기 능 : 지정된 메모리에 음성신호를 저장한다.

4. 인 수 :
· segment : 음성신호를 저장할 메모리의 segment 주소
· offset : 음성신호를 저장할 메모리의 offset
· length : 음성신호를 저장할 메모리의 바이트 수

5. 반환값 : 없음

6. 관련함수 : VoicePlay()

7. 사용예 :

```
#include <multi.h>

main()
{
    while (1) {
        key = getch();
        switch(key)
        {
            case 114:          /* recording */
                VoiceRecord( segment, offset, length );
                break;
            case 112:          /* playing */
                VoicePlay( segment, offset, length );
                break;
            case 108:          /* loading */
                VoiceStop();
                VoiceLoad( file, buffer, length );
                break;
            case 115:          /* saving */
                VoiceStop();
                VoiceSave( file, buffer, length );
                break;
            default:           /* any key */
                VoiceStop();
        }
    }
}
```


VoicePlay()

1. 함수명 : VoicePlay
2. 문 법 : #include <multi.h>
void VoicePlay(segment, offset, length)
unsigned int segment, offset, length;
3. 기 능 : 지정된 메모리에 로드되어 있는 음성신호를 출력한다.
4. 인 수 :
 - . segment : 음성신호가 저장된 메모리의 segment 주소
 - . offset : 음성신호가 저장된 메모리의 offset
 - . length : 음성신호가 저장된 메모리의 바이트 수
5. 반환값 : 없음
6. 관련함수 : VoiceRecord
7. 사용예 : VoiceRecord 참조

VoiceStop()

1. 함수명 : VoiceStop
2. 문 법 : #include <multi.h>
void VoiceStop()
3. 기 능 : 출력중인 음성을 중지한다.
4. 인 수 : 없음
5. 반환값 : 없음
6. 관련함수 : 없음
7. 사용예 : VoiceRecord 참조

VoiceLoad()

1. 함수명 : VoiceLoad
2. 문 법 :

```
#include <multi.h>
void VoiceLoad( file, buffer, length )
char *file;
unsigned char *buffer;
unsigned int length;
```
3. 기 능 : 음성신호를 저장된 파일로부터 버퍼로 로드한다.
4. 인 수 :
 - . file : 음성신호가 저장되어 있는 파일
 - . buffer : 음성신호를 저장할 메모리 버퍼
 - . length : 음성신호를 저장할 메모리 버퍼의 바이트 수
5. 반환값 : 없음
6. 관련함수 : VoiceSave
7. 사용예 : VoiceRecord 참조

VoiceSave()

1. 함수명 : VoiceSave
2. 문 법 :

```
#include <multi.h>
void VoiceSave( file, buffer, length )
char *file;
unsigned char *buffer;
unsigned int length;
```
3. 기 능 : 음성신호를 파일에 저장한다.
4. 인 수 :
 - . file : 음성신호를 저장할 파일
 - . buffer : 음성신호가 저장되어 있는 메모리 버퍼
 - . length : 음성신호가 저장되어 있는 메모리 버퍼의 바이트 수
5. 반환값 : 없음
6. 관련함수 : VoiceLoad
7. 사용예 : VoiceRecord 참조

다. 사용자 인터페이스

| 함수명 | 서 식 | 기 능 |
|-----------------|--|----------------------------------|
| InitPadDump | void InitPadDump(argc, argv[]); char argc; char *argv[]; | 터치스크린의 초기화 |
| Uppcase | char *Uppcase(c); char *c; | 입력된 패드명의 영문 소문자를 대문자로 변환 |
| InitCalibration | void InitCalibration (mode); int mode; | 좌표보정의 초기화 |
| ReadXY | int ReadXY (xc, yc); int *xc; int *yc; | 터치 스크린에서 좌표값을 읽는 함수 |
| ReadXYBuffer | int ReadXYBuffer (xc, yc); int *xc; int *yc; | 스크린을 Touch하면 버퍼로부터 좌표값을 읽는 함수 |
| XCalibration | int XCalibration (mm); int mm; | 보정된 X 좌표값을 계산 |
| YCalibration | int YCalibration (nn); int nn; | 보정된 Y 좌표값을 계산 |
| FlushBuffer | void FlushBuffer(); | 버퍼의 데이터를 Clear |

InitPadDump()

1. 함수명 : InitPadDump

2. 문 법 : #include <multi.h>
void InitPadDump(argc, argv[])
int argc;
char *argv[];

3. 기 능 : 터치 스크린 드라이버 파일에 padman을 사용하여 정의된 pad(zone)을 저장하여 터치 스크린을 터치할 때의 해당 pad명을 알려준다.

4. 인 수 :
· argc : 입력할 인수의 갯수
· argv[] : 입력된 pad명
이때 group명은 GROUP_OFFSET, pad명은 PAD_OFFSET 번지에 저장됨

5. 반환값 : 없음

6. 관련함수 : 없음

7. 사용예 :

```
#include <multi.h>

main()
{
    int x, y;
    int argc;
    char *argv[];

    InitPadDump ( argc, argv );

    InitCalibration(1);
    FlushBuffer();

    while(1)
    {
        ReadXY (&x, &y);
        if ( x != 0 && y != 0 )
            exit(0);
    }
}
```

Uppcase()

1. 함수명 : Uppcase
2. 문 법 :

```
#include <multi.h>
char *Uppcase( c )
char *c;
```
3. 기 능 : 입력된 pad명의 영문 소문자를 대문자로 변환하여 준다.
4. 인 수 :
. c : 입력된 영문 소문자열
5. 반환값 : 변환된 문자열의 포인터
6. 관련함수 : 없음
7. 사용예 :

```
#include <multi.h>

main()
{
    char *c = "kist seri";

    /* "KIST SERI" 출력 */
    printf( "%s", Uppcase(c) );
}
```

InitCalibration()

1. 함수명 : InitCalibration
2. 문 법 :

```
#include <multi.h>
void InitCalibration( mode )
int mode;
```
3. 기 능 : 화면을 640 x 480의 좌표로 보정한다.
4. 인 수 :
 - mode : 좌표모드
 - 0 : 좌표 보정을 하지 않고, 이미 보정된 값을 calib.dat 화일에서 읽어 온다.
 - 1 : 좌표보정 모드로 calib.dat 화일에 새로운 보정값을 기록한다.
5. 반환값 : 없음
6. 관련함수 : 없음
7. 사용예 : InitPadDump 참조

ReadXY()

1. 함수명 : ReadXY
2. 문 법 :

```
#include <multi.h>
int ReadXY( xc, yc );
int *xc, *yc;
```
3. 기 능 : 터치 스크린에서 읽어들이 좌표값을 변수로 가져온다.
4. 인 수 :
 - . xc : x 좌표값
 - . yc : y 좌표값
5. 반환값 :
 - . 0 : 화면을 터치하지 않았을 때 돌려주는 값
6. 관련함수 : ReadXYBuffer
7. 사용예 : InitPadDump 참조

ReadXYBuffer()

1. 함수명 : ReadXYBuffer

2. 문 법 : #include <multi.h>
int ReadXYBuffer(xc, yc)
int *xc, *yc;

3. 기 능 : 원형 버퍼에 데이터가 없으면 데이터가 입력될때 까지 기다렸다가 보정된 좌표값을 읽어온다.

4. 인 수 :
. xc : x 좌표값
. yc : y 좌표값

5. 반환값 :
0 : 화면을 터치하지 않았을 때 돌려주는 값

6. 관련함수 : ReadXY()

7. 사용예 :

```
#include <multi.h>

main()
{
    int *xc, *yc;
    ReadXYBuffer( &xc, &yc );
}
```

XCalibration()

1. 함수명 : XCalibration
2. 문 법 :

```
#include <multi.h>
int XCalibration( xx )
int xx;
```
3. 기 능 : 보정된 x 좌표값을 계산한다.
4. 인 수 :
. xx : 보정된 x 좌표값
5. 반환값 : 없음
6. 관련함수 : YCalibration()
7. 사용예 :

```
#include <multi.h>

main()
{
    int xx, xcal;

    xcal = XCalibration( xx );

    /* 보정된 x 좌표값을 출력 */
    printf( "%d", xcal );
}
```

YCalibration()

1. 함수명 : YCalibration
2. 문 법 :

```
#include <multi.h>
int YCalibration( yy );
int yy;
```
3. 기 능 : 보정된 Y 좌표값을 계산한다.
4. 인 수 :
. yy : 보정된 y 좌표값
5. 반환값 : 없음
6. 관련함수 : XCalibration
7. 사용예 :

```
#include <multi.h>

main()
{
    int yy, ycal;

    ycal = YCalibration( yy );

    /* 보정된 y 좌표값을 출력 */
    printf( "%d", ycal );
}
```

FlushBuffer()

1. 함수명 : FlushBuffer
2. 문 법 : #include <multi.h>
void FlushBuffer();
3. 기 능 : 원형 버퍼의 데이터를 클리어 한다.
4. 인 수 : 없음
5. 반환값 : 없음
6. 관련함수 : 없음
7. 사용예 : InitPadDump 참조

라. 한글 그래픽 윈도우

| 함수명 | 서식 | 기능 |
|-------------|--|-------------------------------------|
| OpenScreen | int OpenScreen(); | 현재 마이크로 컴퓨터의 비디오 어댑터에 따라 그래픽 모드 설정 |
| CloseScreen | void CloseScreen(); | 스크린 모드를 텍스트 모드로 |
| OpenHan | int OpenHan(code_system); int code_system | 한글 폰트 파일을 열어 한글 폰트 이미지를 메모리로 읽어들임 |
| CloseHan | void CloseHan(); | 메모리 상의 한글 폰트 버퍼를 제거 |
| GetKey | int GetKey(); | 사용자가 누른 Key를 검사 |
| DispLine | void Displine(row, col, string, color1, color2); int row, col, color1, color2; char *string; | 지정된 행, 열에 한글 문자열 출력 |
| EditLine | int EditLine(row, col, string, length, color1, color2); int row, col, length, color1, color2; char *string; | 한글 문자열을 입력 받음 |
| MakeFrame | int MakeFrame(frame, row1, col1, row2, col2); FRAME *frame; int row1, col1, row2, col2; | FRAME 구조체에 윈도우의 위치 및 크기 정보를 저장 |
| KillFrame | void KillFrame(frame); FRAME *frame; | FRAME 구조체의 정보를 소거하고 메모리를 개방 |
| SaveWindow | int SaveWindow(frame); FRAME *frame; | 화면의 윈도우 이미지를 버퍼에 저장 |
| RestWindow | void RestWindow(frame, option); FRAME *frame; int option; | SaveWindow()에 의해 저장된 이미지를 화면에 다시 표시 |
| ClearWindow | void ClearWindow(frame, color); FRAME *frame; int color; | 윈도우의 현재 이미지를 지우고 지정 색상을 Painting |

| 함수명 | 서식 | 기능 |
|--------------|---|---|
| EditWindow | <pre>int EditWindow(frame, string_tbl, lines, width, color1, color2, option, break_point); FRAME *frame; char **string_tbl; int width, lines, color1, color2; int option, *break_point;</pre> | 화면 상의 지정 위치에 한글 입출력이 가능한 그래픽 텍스트 에디터 제공 |
| ShowWindow | <pre>int ShowWindow(frame, string_tbl, width, lines, color1, color2); FRAME *frame; char **string_tbl; int width, lines, color1, color2;</pre> | 윈도우에서 한글 문자열을 조회할 수 있는 기능 제공 |
| SelectWindow | <pre>int SelectWindow(frame, string_tbl, lines, width, color1, color2, color3, color4, break_point); FRAME *frame; char **string_tbl; int lines, width; int color1, color2, color3, color4; int *break_point;</pre> | 윈도우 상에서 한글 문자열을 조회하고 선택할 수 있는 기능 제공 |

OpenScreen()

1. 함수명 : OpenScreen
2. 문 법 :

```
#include <multi.h>
int OpenScreen()
```
3. 기 능 : 스크린 모드를 현재 마이크로 컴퓨터에 장착되어 있는 비디오 어댑터에 따라 640*480 (VGA) 또는 640*350 (EGA)의 그래픽 모드로 설정한다.
4. 인 수 : 없음
5. 반환값 :
 - . 정상종료 : OK
 - . 실패시 :
 - DEVICE_ERROR : EGA 또는 VGA 비디오 어댑터가 장착되어 있지 않은 경우
6. 관련함수 : CloseScreen()
7. 사용예 : EditLine 참조

CloseScreen()

1. 함수명 : CloseScreen
2. 문 법 : #include <multi.h>
void CloseScreen()
3. 기 능 : 스크린 모드를 텍스트 모드로 되돌린다.
4. 인 수 : 없음
5. 반환값 : 없음
6. 관련함수 : OpenScreen()
7. 사용예 : EditLine 참조

OpenHan()

1. 함수명 : OpenHan

2. 문 법 : #include <multi.h>
int OpenHan(code_system)
int code_system;

3. 기 능 : 디스크 상에 존재하는 한글 폰트 화일을 열어 한글 폰트 이미지를 메모리로 읽어들이므로써 화면에 한글 문자가 표시될 수 있는 환경을 마련한다. 이 때, 사용하고자 하는 한글 코드 시스템이 KS5601 완성형인지 상용조합형인지를 파라미터로 지정한다.

KS5601 완성형을 선택하였을 경우에는 외부 폰트 화일인 ASCII.FNT와 HAN1.FNT를 읽어들이고, 상용조합형을 선택하였을 경우에는 ASCII.FNT와 HAN2.FNT를 읽어들인다.

4. 인 수 :

. code_system 완성형 한글과 조합형 한글의 선택 옵션

| 상수명 | 의 미 |
|-------------|--------------------|
| KS5601 | KS 5601 완성형 코드를 선택 |
| COMBINATION | 조합형 한글 코드를 선택 |

5. 반환값 :

. 정상종료 : OK

. 실패시 :

- FONT_FILE_NOT_FOUND : 필요한 폰트 화일이 없는 경우

- MEMORY_FAULT : 폰트 이미지를 읽어들이기 메모리가 부족한 경우

6. 관련함수 : CloseHan()

7. 사용예 : EditLine 참조

CloseHan()

1. 함수명 : CloseHan
2. 문 법 : #include <multi.h>
void CloseHan()
3. 기 능 : 메모리 상의 한글 폰트 버퍼를 제거한다.
4. 인 수 : 없음
5. 반환값 : 없음
6. 관련함수 : OpenHan()
7. 사용예 : EditLine 참조

GetKey()

1. 함수명 : GetKey
2. 문 법 : #include <multi.h>
int GetKey()
3. 기 능 : 사용자가 누른 키를 검사하여, 단일 코드를 반환하는 키에 대해서는 ASCII CODE 값을 반환하고, 그 이외의 키에 대해서는 "multi.h"에서 정의된 값을 반환한다.
4. 인 수 : 없음
5. 반환값 :
. ASCII 코드 값 또는 "multi.h"에 정의된 키 값
6. 관련함수 : 없음

DispLine()

1. 함수명 : DispLine
2. 문 법 :

```
#include <multi.h>
void DispLine( row, col, string, color1, color2 )
int row, col, color1, color2;
char *string;
```
3. 기 능 : 지정된 행, 열에 주어진 한글 문자열을 출력한다.
4. 인 수 :
 - . row, col : 문자열이 나타날 시작 위치
 - . string : 나타날 문자열을 담고있는 문자열의 포인터
 - . color1 : 문자 색상
 - . color2 : 배경 색상
5. 반환값 : 없음
6. 관련함수 : ShwoWindow()
7. 사용예 : EditLine 참조

EditLine()

1. 함수명 : EditLine

2. 문 법 : #include <multi.h>
int EditLine(row, col, string, length, color1, color2)
int row, col, length, color1, color2;
char *string;

3. 기 능 : 지정된 행, 열에서 주어진 길이만큼 한글 문자열을 입력받는다.
좌우 커서 이동 키와 BS, DEL, INS 키를 편집 키로 사용할 수 있다.

4. 인 수 :

- . row, col : 문자열이 나타날 시작 위치
- . string : 나타날 문자열을 담고있는 문자열의 포인터
- . length : 입력 문자열의 최대 길이
- . color1 : 문자 색상
- . color2 : 배경 색상

5. 반환값 :

| 상수명 | 의 미 |
|-------|-----------------------------------|
| Enter | 사용자가 Enter로 편집 종료. string에 데이터 저장 |
| F1 | 사용자가 F1 키로 편집 종료. string에 데이터 저장 |
| F2 | 사용자가 F2 키로 편집 종료. string에 데이터 저장 |
| F3 | 사용자가 F3 키로 편집 종료. string에 데이터 저장 |
| F4 | 사용자가 F4 키로 편집 종료. string에 데이터 저장 |
| F5 | 사용자가 F5 키로 편집 종료. string에 데이터 저장 |
| F6 | 사용자가 F6 키로 편집 종료. string에 데이터 저장 |
| F7 | 사용자가 F7 키로 편집 종료. string에 데이터 저장 |
| F8 | 사용자가 F8 키로 편집 종료. string에 데이터 저장 |
| F9 | 사용자가 F9 키로 편집 종료. string에 데이터 저장 |
| F10 | 사용자가 F10키로 편집 종료. string에 데이터 저장 |
| Esc | 사용자가 Esc로 편집 종료. string을 변경하지 않음 |

6. 관련함수 : EditWindow()

7. 사용예 :

```
#include <multi.h>

main()
{
    char string[81]
    int last_key;

    OpenScreen();

    OpenHan( KS5601 );

    DispLine( 9, 0, "Input Command .... ", YELLOW, RED );

    last_key = EditLine( 10, 0, string, WHITE, MAGENTA );

    switch( last_key ) {
        .....
        .....
        .....
    }

    CloseHan();

    CloseScreen();
    .....
    .....
}
```

MakeFrame()

1. 함수명 : MakeFrame
2. 문 법 :

```
#include <multi.h>
int MakeFrame( frame, row1, col1, row2, col2 )
FRAME *frame;
int row1, row2, col1, col2;
```
3. 기 능 : 윈도우 정보를 담은 FRAME 구조체에 윈도우의 위치 및 크기 정보를 저장한다.
4. 인 수 :
 - . frame : 윈도우의 위치 및 크기 정보를 담은 FRAME 구조체의 포인터
 - . row1, col1 : 윈도우 좌측, 상단의 좌표
 - . row2, col2 : 윈도우 우측, 하단의 좌표
5. 반환값 :
 - . 정상종료 : OK
 - . 실패시 :
 - MEMORY_FAULT : 윈도우 이미지를 저장할 버퍼의 MEMORY ALLOCATION이 실패하였을 경우
6. 관련함수 : KillFrame()
7. 사용예 : EditWindow 참조

KillFrame()

1. 함수명 : KillFrame
2. 문 법 :

```
#include <multi.h>
void KillFrame( frame )
FRAME *frame;
```
3. 기 능 : 윈도우 정보를 담고 있는 FRAME 구조체의 정보를 소거하고 메모리를 개방한다.
4. 인 수 :
. frame : 소거할 윈도우에 관한 정보를 담은 FRAME 구조체의 포인터
5. 반환값 : 없음
6. 관련함수 : MakeFrame()
7. 사용예 : EditWindow 참조

SaveWindow()

1. 함수명 : SaveWindow
2. 문 법 :

```
#include <multi.h>
int SaveWindow( frame )
FRAME *frame;
```
3. 기 능 : 화면에 나타나 있는 윈도우 이미지를 버퍼에 저장한다.
4. 인 수 :
 - . frame : 저장할 윈도우의 위치 및 크기 정보를 담은 FRAME 구조체의 포인터
5. 반환값 :
 - . 정상종료 : OK
 - . 실패시 :
 - MEMORY_FAULT : FRAME 구조체에 정상적인 값이 세트되어 없을 경우
6. 관련함수 : RestWindow(), ClearWindow()
7. 사용예 : EditWindow 참조

RestWindow()

1. 함수명 : RestWindow
2. 문 법 : #include <multi.h>
void RestWindow(frame, option)
FRAME *frame;
int option;
3. 기 능 : SaveWindow()에 의해 저장된 윈도우 이미지를 화면에 다시 표시한다.
4. 인 수 :
 - . frame : 복원할 윈도우의 위치 및 크기 정보를 담은 FRAME 구조체의 포인터
 - . option : 본래의 윈도우 이미지를 화면에 나타내는 방식에 대한 선택 옵션

| 상수명 | 기 능 |
|-------|------------------------------------|
| QUICK | 한 순간에 전 화면을 복원 |
| SPLIT | 화면이 가운데에서부터 양쪽으로 갈라지면서 원래의 이미지를 복원 |

5. 반환값 : 없음
6. 관련함수 : SaveWindow(), ClearWindow()
7. 사용예 : EditWindow 참조

| |
|---------------|
| ClearWindow() |
|---------------|

1. 함수명 : ClearWindow
2. 문 법 :

```
#include <multi.h>
void ClearWindow( frame, color )
FRAME *frame;
int color;
```
3. 기 능 : 윈도우의 현재 이미지를 지우고 지정 색상을 PAINTING
4. 인 수 :
 - . frame : 클리어할 윈도우의 위치 및 크기 정보를 담은 FRAME 구조체의 포인터
 - . color : 배경 색상
5. 반환값 : 없음
6. 관련함수 : SaveWindow(), RestWindow()
7. 사용예 : EditWindow() 참조

EditWindow()

1. 함수명 : EditWindow

2. 문 법 : #include <multj,h>
int EditWindow(frame, string_tbl, lines, width, color1, color2,
option, break_point)
FRAME *frame;
char **string_tbl;
int lines, width, color1, color2, option;
int *break_point;

3. 기 능 : 화면상의 지정 위치에 한글 입출력이 가능한 그래픽 텍스트 에디터를 제공한다.

이 에디터 윈도우는 텍스트 데이터의 편집 및 검색 질의어 입력 등에 사용할 수 있다.

이 에디터의 편집 기능과 이를 위한 기능 키는 다음과 같다.

. 커서 이동

- Up : 한 줄 위로 이동. 첫 줄에서 사용시 1 줄 scroll down
- Down : 한 줄 아래로 이동. 마지막 줄에서 사용시 1 줄 scroll up
- Left : 한 칸 좌로 이동. 좌우 scroll은 지원하지 않는다.
- Right : 한 칸 우로 이동. 좌우 scroll은 지원하지 않는다.
- Tab : 다음 줄 첫 칸으로 이동
- BackTab : 위 줄 첫 칸으로 이동

. 페이지 넘김

- PgUp : 앞 페이지로
- PgDn : 다음 페이지로
- Home : 처음 페이지로
- End : 마지막 페이지로

. 문자의 삽입 및 삭제

- Ins : 삽입 모드 설정/해제 (toggle key)
- Del : 커서 위치의 1 문자 삭제
- BackSpace : 커서 앞 위치의 1 문자 삭제

. 자판 변경

- * : 한글/영문 모드 변환 (toggle key)

. 편집 종료

- Enter, F1 - F10 : 정상 종료
- Esc : 편집 취소

4. 인수 :

- . frame : 에디터 윈도우의 위치 및 크기 정보를 담고 있는 FRAME 구조체의 포인터
- . string_tbl : 에디터와 데이터를 주고 받기 위한 텍스트 버퍼의 포인터
에디터에 초기 데이터를 줄 필요가 없을 때에는 NULL
스트링을 지정한다. (*string = NULL)
- . lines : string_tbl에 선언되어 있는 스트링 line의 수
(-> 이 에디터를 통해 편집할 수 있는 line의 수)
- . width : string_tbl에 선언되어 있는 스트링의 한 line의 길이
- . color1 : 문자의 색상
- . color2 : 배경의 색상
- . option : 편집 모드와 입력 스트링 조작에 관한 옵션
옵션 지정은 다음의 상수를 사용하고, 두 개 이상의 상수는
bit manipulation의 ORing으로 전달한다.
- . break_point : 조회 종료시 마지막 화면의 페이지와 커서의 위치를 담기
위한 정수 변수의 포인터

| 상수명 | 기능 |
|----------|-----------------------------|
| SHOW | 텍스트를 출력하기만 하고 편집은 허용하지 않는다. |
| EDIT | 텍스트의 편집을 허용한다. |
| TRIM | 편집 완료시 텍스트에 중복 사용된 공백을 제거. |
| UPPER | 편집 완료시 영문 소문자를 대문자로 변환. |
| NO_CLEAR | 에디터 영역의 화면을 지우지 않고 편집 개시. |

5. 반환값 :

| 상수명 | 의미 |
|-------|---------------------------------------|
| Enter | 사용자가 Enter로 편집 종료. string_tbl에 데이터 저장 |
| F1 | 사용자가 F1 키로 편집 종료. string_tbl에 데이터 저장 |
| F2 | 사용자가 F2 키로 편집 종료. string_tbl에 데이터 저장 |
| F3 | 사용자가 F3 키로 편집 종료. string_tbl에 데이터 저장 |
| F4 | 사용자가 F4 키로 편집 종료. string_tbl에 데이터 저장 |
| F5 | 사용자가 F5 키로 편집 종료. string_tbl에 데이터 저장 |
| F6 | 사용자가 F6 키로 편집 종료. string_tbl에 데이터 저장 |
| F7 | 사용자가 F7 키로 편집 종료. string_tbl에 데이터 저장 |
| F8 | 사용자가 F8 키로 편집 종료. string_tbl에 데이터 저장 |
| F9 | 사용자가 F9 키로 편집 종료. string_tbl에 데이터 저장 |
| F10 | 사용자가 F10키로 편집 종료. string_tbl에 데이터 저장 |
| Esc | 사용자가 Esc로 편집 종료. string_tbl을 변경하지 않음 |

6. 관련함수 : MakeFrame(), ShowWindow(), SelectWindow, SaveWindow(), RestWindow()

7. 사용예 :

```
#include <multi.h>

main()
{
    FRAME txt_win;
    char string[10][81]
    int break_point;
    int last_key;

    OpenScreen();

    OpenHan( KS5601 );

    .....
    .....

    MakeFrame( &txt_win, 22, 0, 24, 79 );

    ClearWindow( &txt_win, MAGENTA );

    while(1) {
        last_key = EditWindow( txt_win, string, 10, 81, WHITE,
                               MAGENTA, EDIT|NO_CLEAR, &break_point );

        SaveScreen( &txt_win );

        switch( last_key ) {
            .....
            .....
            .....
        }

        RestScreen( &txt_win, QUICK );
    }

    KillFrame( &txt_win );

    CloseHan();

    CloseScreen();
    .....
    .....
}
```

ShowWindow()

1. 함수명 : ShowWindow

2. 문 법 : #include <multi.h>
int ShowWindow(frame, string_tbl, lines, width, color1, color2,
break_point)
FRAME *frame;
char **string_tbl;
int lines, width, color1, color2;
int *break_point;

3. 기 능 : 2 차원 문자 테이블에 담긴 여러 행의 한글 문자열을 화면상에 만들어진 창(윈도우) 속에서 조회할 수 있는 기능을 제공한다.

윈도우 내에서의 텍스트 조회를 위한 기능 키는 다음과 같다.

. 커서 이동

- Up : 한 줄 위로 이동. 첫 줄에서 사용시 1 줄 scroll down
- Down : 한 줄 아래로 이동. 마지막 줄에서 사용시 1 줄 scroll up
- Left : 한 칸 좌로 이동. 좌우 scroll은 지원하지 않는다.
- Right : 한 칸 우로 이동. 좌우 scroll은 지원하지 않는다.
- Tab : 다음 줄 첫 칸으로 이동
- BackTab : 윗 줄 첫 칸으로 이동

. 페이지 넘김

- PgUp : 앞 페이지로
- PgDn : 다음 페이지로
- Home : 처음 페이지로
- End : 마지막 페이지로

. 조회 종료

- Enter, F1-F10, Esc

4. 인 수 :

- . frame : 윈도우의 위치 및 크기 정보를 담고 있는 FRAME 구조체의 포인터
- . string_tbl : 조회하고자 하는 문자 배열(string_tbl)의 포인터
- . lines : string_tbl에 선언되어 있는 스트링 line의 수
- . width : string_tbl에 선언되어 있는 스트링의 한 line의 길이
- . color1 : 문자의 색상
- . color2 : 배경의 색상
- . break_point : 조회 종료시 마지막 화면의 페이지와 커서의 위치를 담기 위한 정수 변수의 포인터

5. 반환값 :

| 상수명 | 의 미 |
|-------|--------------------|
| Enter | 사용자가 Enter로 조회 종료. |
| F1 | 사용자가 F1 키로 조회 종료. |
| F2 | 사용자가 F2 키로 조회 종료. |
| F3 | 사용자가 F3 키로 조회 종료. |
| F4 | 사용자가 F4 키로 조회 종료. |
| F5 | 사용자가 F5 키로 조회 종료. |
| F6 | 사용자가 F6 키로 조회 종료. |
| F7 | 사용자가 F7 키로 조회 종료. |
| F8 | 사용자가 F8 키로 조회 종료. |
| F9 | 사용자가 F9 키로 조회 종료. |
| F10 | 사용자가 F10키로 조회 종료. |
| Esc | 사용자가 Esc로 조회 종료. |

6. 관련함수 : MakeFrame(), EditWindow(), SelectWindow, SaveWindow(), RestWindow()

7. 사용예 :

```
#include <multi.h>

main()
{
    FRAME txt_win;
    int break_point;
    int last_key;
    char string[5][81] = { "하나", "둘", "셋", "넷", "다섯" };

    OpenScreen();
    OpenHan( KS5601 );
    .....

    MakeFrame( &txt_win, 22, 0, 24, 79 );

    last_key = ShowWindow( txt_win, string, 5, 81, WHITE, MAGENTA,
        &break_point );

    switch( last_key ) {
    case Enter:
        .....
    case F1:
        .....
    case F7:
        .....
    case F8:
        .....
    case Esc:
        .....
    }
    .....
}
```

SelectWindow()

1. 함수명 : SelectWindow

2. 문 법 : #include <multi.h>
int SelectWindow(frame, string_tbl, lines, width, color1, color2,
color3, color4, break_point)
FRAME *frame;
char **string_tbl;
int lines, width, color1, color2, color3, color4;
int *break_point;

3. 기 능 : 2 차원 문자 테이블에 담긴 여러 행의 한글 문자열을 화면상에 만들어진 창(윈도우) 속에서 조회할 수 있는 기능과 함께, 그 중 한 행의 정보를 선택할 수 있는 기능을 제공한다. 이 함수는 Pull Down Menu의 구현을 위해 쓰일 수 있다.

윈도우 내에서의 텍스트 조회 및 정보 선택을 위한 기능 키는 다음과 같다.

. 커서 이동

- Up : 한 줄 위로 이동. 첫 줄에서 사용시 1 줄 scroll down
- Down : 한 줄 아래로 이동. 마지막 줄에서 사용시 1 줄 scroll up
- Left : 한 칸 좌로 이동. 좌우 scroll은 지원하지 않는다.
- Right : 한 칸 우로 이동. 좌우 scroll은 지원하지 않는다.
- Tab : 다음 줄 첫 칸으로 이동
- BackTab : 윗 줄 첫 칸으로 이동

. 페이지 넘김

- PgUp : 앞 페이지로
- PgDn : 다음 페이지로
- Home : 처음 페이지로
- End : 마지막 페이지로

. 조회 종료

- Enter : 특정 행의 선택
- Esc : 아무것도 선택하지 않음

4. 인 수 :

- . frame : 윈도우의 위치 및 크기 정보를 담고 있는 FRAME 구조체의 포인터
- . string_tbl : 조회하고자 하는 문자 배열(string_tbl)의 포인터
- . lines : string_tbl에 선언되어 있는 스트링 line의 수
- . width : string_tbl에 선언되어 있는 스트링의 한 line의 길이
- . color1 : 문자의 색상
- . color2 : 배경의 색상

- . color3 : 커서가 위치한 라인의 문자 색상
- . color4 : 커서가 위치한 라인의 배경 색상
- . break_point : 조회 종료시 마지막 화면의 페이지와 커서의 위치를 담기 위한 정수 변수의 포인터

5. 반환값 :

- . 사용자가 선택한 정보의 라인 번호 (base: 1) 또는 0 (Esc 키로 종료하였을 경우)

6. 관련함수 : MakeFrame(), ShowWindow(), EditWindow(), SaveWindow(), RestWindow()

7. 사용예 :

```
#include <multi.h>

main()
{
    FRAME txt_win;
    int break_point;
    int select;
    char string[5][81] = { "하나", "둘", "셋", "넷", "다섯" };

    OpenScreen();
    OpenHan( KS5601 );

    .....
    .....

    MakeFrame( &txt_win, 22, 0, 24, 79 );

    select = SelectWindow( txt_win, string, 5, 81, WHITE, MAGENTA,
                          &break_point );

    switch( select ) {
    case 0: /cancel */
        .....
    case 1:
        .....
    case 2:
        .....
    case 3:
        .....
    case 4:
        .....
    case 5:
        .....
    }
    .....
    .....
}
```

마. 다중매체 정보 검색

| 함수명 | 서식 | 기능 |
|---------------|--|---|
| CreateNDX | int CreateNDX(dbid); char *dbid; | 검색을 위한 화일 생성 |
| InitQuery | int InitQuery(dbid); char *dbid; | 검색을 위한 초기화 작업을 수행 |
| ExecQuery | int ExecQuery(query_command, result_tbl, limit); char *query_command; unsigned int *result_tbl; int limit; | 현재 사용중인 데이터베이스에서 검색조건에 맞는 레코드를 찾는 작업 수행 |
| EndQuery | void EndQuery(); | 검색작업 종료시 인덱스화일 Close |
| BrowseAll | int BrowseAll(frame, color1, color2, color3, color4, break_point); FRAME *frame; int color1, color2, color3, color4; int *break_point; | 데이터베이스 상의 전체 레코드를 열람하고, 특정 레코드를 선택할 수 있는 기능 제공 |
| BrowseResult | int BrowseResult(result_tbl, num, frame, color1, color2, color3, color4, break_point); unsigned int *result_tbl; int num; FRAME *frame; int color1, color2, color3, color4; int *break_point; | 검색결과 얻어진 레코드의 내용을 열람하고, 특정 레코드를 선택할 수 있는 기능 제공 |
| GetRecordInfo | int GetRecordInfo(rec_no, data_type, page); unsigned int rec_no, *data_type; int *page; | 레코드의 번호를 입력하여 데이터 타입과 페이지 수에 관한 정보를 가져 오는 기능 제공 |
| ShowText | int ShowText(rec_no, frame, color1, color2, page); unsigned int rec_no; FRAME *frame; int color1, color2, *page; | 지정 레코드의 문자정보를 원하는 윈도우에 보여주는 기능 제공 |

| 함수명 | 서식 | 기능 |
|-----------|---|--|
| ShowImage | int ShowImage(rec_no, frame, color, page); unsigned int rec_no; FRAME *frame; int color, *page; | 특정 레코드의 데이터 중 정지화상을 페이지 단위로 화면에 출력 |
| PlaySound | int PlaySound(rec_no, page); unsigned int rec_no; int *page; | 특정 레코드의 데이터 중 음성 데이터를 페이지 단위로 음향장치에 출력 |
| PlayMovie | int PlayMovie(rec_no, frame, color, option, page); unsigned int rec_no; FRAME *frame; int color, *option, *page; | 특정 레코드의 데이터 중 동화상 데이터를 페이지 단위로 화면 및 음향장치에 출력 |

CreateNDX()

1. 함수명 : CreateNDX

2. 문 법 : #include <multi.h>
int CreateNDX(dbid)
char *dbid;

3. 기 능 : 소스파일(확장자 SRC)인 파일을 입력으로 하여 검색을 위한 파일
(확장자 NDX, KWD, PST)을 생성하며 입력이 잘못된 레코드는 확장자가
RJT인 파일로 생성한다.

모든 생성되는 파일의 이름은 주어진 파일명과 같으며, 단지 확장자만
다르다.

소스 파일과 인덱스 파일의 기능은 다음과 같다.

- . *.src : 사용자가 편집한 다중매체 정보 Configuration File
- . *.ndx : Configuration File의 내용에 대한 인덱스 테이블
- . *.kwd : Configuration File의 Text 부문에서 추출된 키 워드 리스트
- . *.pst: kwd 파일과 ndx 파일의 정보를 연계시켜 주는 인덱스

4. 인 수 :
. dbid : 소스 파일 및 인덱스 파일의 파일 이름을 담은 스트링의 포인터

5. 반환값 :
. 정상종료 : OK 또는 입력에러가 발생한 레코드의 건수
. 실패시 :
- SRC_FILE_NOT_EXIST : 입력파일이 없다
- DEVICE_ERROR : 결과파일을 생성할 수 없다

6. 관련함수 : InitQuery(), ExecQuery(), EndQuery()

7. 사용예 :

```
#include <multi.h>
main()
{
    char filename[80];

    printf( "Input File Name: " );
    gets( filename );

    if( CreateNDX( filename ) == OK )
        printf( "Successfully Created!" );
    .....
    .....
}
```

InitQuery()

1. 함수명 : InitQuery
2. 문 법 : #include <multi.h>
int InitQuery(dbid)
char *dbid;
3. 기 능 : 검색을 위한 초기화작업을 수행(확장자가 SRC, NDX, KWD, PST인 검색에 필요한 파일들을 오픈함)
4. 인 수 :
. dbid : 검색할 대상 데이터베이스 파일명
5. 반환값 :
. 정상종료 : OK

. 실패시 :
- NDX_FILE_NOT_EXIST
- KWD_FILE_NOT_EXIST
- PST_FILE_NOT_EXIST
- SRC_FILE_NOT_EXIST
6. 관련함수 : ExecQuery(), EndQuery()
7. 사용예 :

```
#include <multi.h>

main()
{
    char filename[80];
    int rv;

    printf( "Input File Name: " );
    gets( filename );

    rv = InitQuery( filename );

    if( rv == OK ) printf( "Database %s is Connected!" );
    .....
    .....
    .....
}
```

ExecQuery()

1. 함수명 : ExecQuery
2. 문 법 :

```
#include <multi.h>
int ExecQuery( query_command, result_tbl, limit )
char *query_command;
unsigned int *result_tbl;
int limit;
```
3. 기 능 : 현재 사용중인 데이터베이스에서 주어진 검색조건에 해당되는 레코드를 찾는 작업을 수행함
4. 인 수 :
 - . query_command : 검색조건을 전달하기위한 스트링의 포인터
 - . result_tbl : 검색결과(레코드의 번호)를 담기 위한 정수배열의 시작번지
 - . limit : 검색 결과 배열의 최대값
5. 반환값 :
 - . 정상종료 : 검색결과 건수
 - . 실패시 :
 - D/B_NOT_INITIATED : 이 함수의 사용에 앞서 InitQuery()를 호출하지 않은 경우
 - SYNTAX_ERROR : 검색조건문의 문법이 잘못되었을 경우
 - MEMORY_FAULT : 작업수행시 필요한 메모리가 부족한 경우
6. 관련함수 : InitQuery(), EndQuery()
7. 사용예 :

```
#include <multi.h>

main()
{
    unsigned int result_tbl[ 100 ];
    int rv;

    InitQuery( "test" );

    while(1) {
        gets( user_command );
        if( strcmp(strupr( user_command ), "QUIT" ) == 0 ) break;
        rv = ExecQuery( user_command, result_tbl, 100 );
        .....
        .....
    }
    .....
    EndQuery();
}
```


EndQuery()

1. 함수명 : EndQuery
2. 문 법 : #include <multi.h>
void EndQuery()
3. 기 능 : 검색 작업 종료시, 또는 다른 분야의 정보를 검색하고자 할 때, 현재
사용중이던 각종 인덱스 화일을 Close 함.
4. 인 수 : 없음
5. 반환값 : 없음
6. 관련함수 : InitQuery(), ExecQuery()
7. 사용예 : ExecQuery 참조

BrowseAll()

1. 함수명 : BrowseAll
2. 문 법 :

```
#include <multi.h>
int BrowseAll( frame, color1, color2, color3, color4, break_point )
FRAME *frame;
int color1, color2, color3, color4, *break_point;
```
3. 기 능 : 데이터베이스 상의 전체 레코드의 내용을 열람하고 그 가운데 특정 레코드를 선택할 수 있는 기능을 제공한다.
4. 인 수 :
 - . frame : 현재의 윈도우에 관한 정보를 갖는 FRAME 구조체의 포인터
 - . color1 : 문자 색상
 - . color2 : 배경 색상
 - . color3 : 커서가 위치한 라인의 문자 색상
 - . color4 : 커서가 위치한 라인의 배경 색상
 - . break_point : 마지막 화면과 커서의 위치에 관한 정보를 담기 위한 기억장소의 포인터
5. 반환값 :
 - . 정상종료 :
 - Enter Key로 종료하였을 경우 : 사용자가 선택한 레코드의 일련번호
 - Esc Key로 종료하였을 경우 : CANCEL(0)
 - . 실패시 :
 - D/B_NOT_INITIATED : 이 함수의 사용에 앞서 InitQuery()를 호출하지 않았을 경우
6. 관련함수 : BrowseResult()
7. 사용예 :

```
#include <multi.h>

main()
{
    FRAME browse_window;
    int select, break_point;

    MakeFrame( &browse_window, 0, 0, 10, 79 );

    InitQuery( "test" );

    select = BrowseAll( &browse_window, CYAN, BLUE, WHITE, RED,
        &break_point );

    .....
    .....
}
```

BrowseResult()

1. 함수명 : BrowseResult

2. 문 법 : #include <multi.h>
int BrowseResult(result_tbl, number, frame, color1, color2,
color3, color4, break_point)
unsigned int *result_tbl;
int number;
FRAME *frame;
int color1, color2, color3, color4, *break_point;

3. 기 능 : 검색 결과 얻어진 다수의 레코드의 내용을 열람하고 그 가운데 특정 레코드를 선택할 수 있는 기능을 제공한다.

4. 인 수 :
. result_tbl : 검색결과(레코드의 번호)를 담고있는 정수 배열의 포인터
. number : result_tbl에 담겨 있는 검색 결과의 건수
. frame : 현재의 윈도우에 관한 정보를 갖는 FRAME 구조체의 포인터
. color1 : 문자 색상
. color2 : 배경 색상
. color3 : 커서가 위치한 라인의 문자 색상
. color4 : 커서가 위치한 라인의 배경 색상
. break_point : 마지막 화면과 커서의 위치에 관한 정보를 담기 위한 기억장소의 포인터

5. 반환값 :
. 정상종료 :
- Enter Key로 종료하였을 경우 : 사용자가 선택한 레코드의 일련번호
- Esc Key로 종료하였을 경우 : CANCEL(0)
. 실패시 :
- D/B_NOT_INITIATED : 이 함수의 사용에 앞서 InitQuery()를 호출하지 않았을 경우
- INCORRECT_REC_NO : result_tbl의 값이 잘못된 경우

6. 관련함수 : BrowseAll()

7. 사용예 :

```
#include <multi.h>

main()
{
    FRAME browse_window;
    unsigned int result_tbl[100];
    int select, number, break_point;

    MakeFrame( &browse_window, 0, 0, 10, 79 );

    InitQuery( "test" );

    EditLine( 24, 0, user_command, 80, WHITE, MACENTA );

    number = ExecQuery( user_command, result_tbl, 100 );

    select = BrowseResult( result_tbl, number, &browse_window,
                          CYAN, BLUE, WHITE, RED, &break_point );
    .....
    .....
}
```

GetRecordInfo()

1. 함수명 : GetRecordInfo

2. 문 법 : #include <multi.h>
int GetRecordInfo(rec_no, data_type, page)
unsigned int rec_no, *data_type;
int *page;

3. 기 능 : 레코드의 번호 입력하여 데이터 타입과 페이지 수에 관한 정보를 가져온다. 여기서 페이지란 유관한 화상과 음성을 동시에 조작하기 위한 동기 번호를 의미한다.

4. 인 수 :

- . rec_no : 조회하고자 하는 레코드의 고유번호
- . data_type : 지정된 레코드의 데이터 타입에 관한 정보를 담은 정수 변수의 포인터.
특정 유형의 데이터가 존재할 경우 변수에 해당 값이 세트됨

| 상수명 | 실제 값 | 타입 |
|-------|------|--------|
| TEXT | 1 | 문자 정보 |
| IMAGE | 2 | 정지 화상 |
| VOICE | 4 | 음성 데이터 |
| MOVIE | 8 | 동 화상 |

of) 특정 레코드가 문자정보, 정지화상, 음성정보를 포함할 경우

```
*data_type & TEXT == TRUE  
*data_type & IMAGE == TRUE  
*data_type & VOICE == TRUE
```

- . page : 해당 레코드 내에서 유관한 영상과 음성이 조합된 데이터 셋의 건 수(페이지 수)를 담은 정수 변수의 포인터

5. 반환값 :

. 정상종료 : OK

. 실패시 :

- D/B_NOT_INITIATED : 이 함수의 사용에 앞서 InitQuery()를 호출하지 않았을 경우
- INCORRECT_REC_NO : rec_no의 값이 잘못된 경우

6. 관련 함수 : ShowText(), ShowImage(), PlaySound(), PlayMovie()

7. 사용 예 :

```
#include <multi.h>

main()
{
    char user_command[80];
    unsigned int result_tbl[100], data_type;
    int number;
    int page, i;

    InitQuery( "test" );

    EditLine( 24, 0, user_command, 80, WHITE, MAGENTA );

    number = ExecQuery( user_command, result_tbl, 100 );

    for( i=0; i<number; i++ ) {
        GetRecordInfo( result_tbl[i], &data_type, &page )
        if( data_type & TEXT ) ....
        if( data_type & IMAGE ) ....
        if( data_type & VOIDE ) ....
        if( data_type & MOVIE ) ....
    }
    .....
    .....
}
```

ShowText()

1. 함수명 : ShowText

2. 문 법 : #include <multi.h>
int ShowText(rec_no, frame, color1, color2, page)
unsigned int rec_no;
FRAME *frame;
int color1, color2, *page;

3. 기 능 : 지정 레코드의 문자 정보를 원하는 윈도우에 보여준다.

4. 인 수 :

- . rec_no : 보여줄 레코드의 번호
- . frame : 현재의 윈도우에 관한 정보를 갖는 FRAME 구조체의 포인터
- . color1 : 문자 색상
- . color2 : 배경 색상
- . page : 표시된 (또는 표시할) PAGE의 번호를 담은 정수 변수의 포인터
문자 정보의 페이지는 윈도우의 사이즈에 의해 가변적으로 정해짐

5. 반환값 :

- . 정상종료 : OK
- . 실패시 :
 - D/B_NOT_INITIATED : 이 함수의 사용에 앞서 InitQuery()를 호출하지 않았을 경우
 - INCORRECT_REC_NO : rec_no의 값이 잘못된 경우

6. 관련함수 : GetReoordInfo(), ShowImage(), PlaySound(), PlayMovie()

7. 사용예 :

```
#include <multi.h>

main()
{
    char    user_command[80];
    unsigned int    result_tbl[100], data_type;
    int    number;
    FRAME txt_window, pic_window;
    int    page, txt_page, av_page;
    int    key, i;

    .....

    number = ExecQuery( user_command, result_tbl, 100 );

    .....

    GetRecordInfo( result_tbl[i], &data_type, &page )

    txt_page = av_page = 0;
    exit_flag = 0;

    while( !exit_flag ) {
        if( data_type & TEXT )
            ShowText( result_tbl[i], &txt_window,
                WHITE, GREEN, &txt_page );
        if( data_type & IMACE )
            ShowText( result_tbl[i], &pic_window,
                RED, &av_page );
        if( data_type & VOICE )
            ShowText( result_tbl[i], &av_page );
        if( data_type & MOVIE )
            ShowText( result_tbl[i], &pic_window,
                RED, &option, &av_page );

        switch( GetKey() ) {
            case F5:
                txt_page--;
                break;
            case F6:
                txt_page++;
                break;
            case F7:
                av_page--;
                /* 범위를 초과하는 page 값은 */
                /* 하위 모듈에서 조정하므로 */
                /* 범위 check의 필요가 없음 */
            case F8:
                av_page++;
                break;
            case F10:
                exit_fag = 1;
                break;
        }
    }

    .....
    .....
}
```


ShowImage()

1. 함수명 : ShowImage
2. 문 법 :

```
#include <multi.h>
int ShowImage( rec_no, frame, color, page )
unsigned int rec_no;
FRAME *frame;
int color, *page;
```
3. 기 능 : 선택된 특정 레코드의 데이터 중 정지 화상(Still Image)을 페이지 단위로 화면에 출력한다.
여기서 페이지란 유관한 화상과 음성을 동시에 조작하기 위한 동기 번호를 의미한다.
4. 인 수 :
 - . rec_no : 보여줄 대상 레코드의 번호
 - . frame : 정지화상이 보여질 윈도우에 관한 정보를 갖고있는 구조체의 포인터
 - . color : 정지화상이 출력될 부분의 바탕 색상 (RED, GREEN, BLUE, WHITE)
 - . page : 표시된 (또는 표시할) PACE의 번호를 담은 정수 변수의 포인터
5. 반환값 :
 - . 정상종료 : OK
 - . 실패시 :
 - D/B_NOT_INITIATED : 이 함수의 사용에 앞서 InitQuery()를 호출하지 않았을 경우
 - INCORRECT_REC_NO : rec_no의 값이 잘못된 경우
 - INVALID_DATA_TYPE : 정지화상 데이터를 포함하지 않은 레코드를 지정하였을 경우
 - DEVICE_ERROR : 비디오 인터페이스 장비의 조작에서 에러가 발생한 경우
6. 관련함수 : GetReoordInfo(), ShowText(), PlaySound(), PlayMovie()
7. 사용예 : ShowText 참조

PlaySound()

1. 함수명 : PlaySound

2. 문 법 : #include <multi.h>
int PlaySound(rec_no, page)
unsigned int rec_no;
int *page;

3. 기 능 : 선택된 특정 레코드의 데이터 중 음성 데이터를 페이지 단위로 음향 장치에 출력한다.
여기서 페이지란 유관한 화상과 음성을 동시에 조작하기 위한 동기 번호를 의미한다.

4. 인 수 :
· rec_no : 보여줄 대상 레코드의 번호
· page : 표시된 (또는 표시할) PAGE의 번호를 담은 정수 변수의 포인터

5. 반환값 :
· 정상종료 : OK
· 실패시 :
- D/B_NOT_INITIATED : 이 함수의 사용에 앞서 InitQuery()를 호출하지 않았을 경우
- INCORRECT_REC_NO : rec_no의 값이 잘못된 경우
- INVALID_DATA_TYPE : 음성 데이터를 포함하지 않은 레코드를 지정하였을 경우
- DEVICE_ERROR : 오디오 인터페이스 장비의 조작에서 에러가 발생한 경우

6. 관련함수 : GetReocrdInfo(), ShowText(), ShowImage(), PlayMovie()

7. 사용예 : ShowText 참조

PlayMovie()

1. 함수명 : PlayMovie

2. 문 법 : #include <multi.h>
int PlayMovie(rec_no, frame, color, option, page)
unsigned int rec_no;
FRAME *frame;
int color, *option, *page;

3. 기 능 : 선택된 특정 레코드의 데이터 중 동화상 데이터와 데이터 페이지 단위로 화면 및 음향장치에 출력한다.
여기서 페이지란 유관한 화상과 음성을 동시에 조작하기 위한 동기 번호를 의미한다.

4. 인 수 :

- . rec_no : 보여줄 대상 레코드의 번호
- . frame : 동화상이 보여질 윈도우에 관한 정보를 갖고있는 구조체의 포인터
- . color : 정지화상이 출력될 부분의 바탕 색상 (RED, GREEN, BLUE, WHITE)
- . option : 동화상이 보여진 (또는 보여질) 방식을 담은 정수 변수의 포인터
 - NORMAL : 정상속도로 화면재생
 - MUTE : 음향을 배제하고 영상만 재생
 - SLOW : 느린속도로 화면재생
 - FAST : 빠른속도로 화면재생
 - VFAST : 매우 빠른속도로 화면재생
 - CUT_BY_CUT : 한 장면씩 재생
- . page : 표시된 (또는 표시할) PACE의 번호를 담은 정수 변수의 포인터

5. 반환값 :

- . 정상종료 : OK
- . 실패시 :
 - D/B_NOT_INITIATED : 이 함수의 사용에 앞서 InitQuery()를 호출하지 않았을 경우
 - INCORRECT_REC_NO : rec_no의 값이 잘못된 경우
 - INVALID_DATA_TYPE : 음성 데이터를 포함하지 않은 레코드를 지정하였을 경우
 - DEVICE_ERROR : 레이저 디스크 인터페이스 장비의 조작에서 에러가 발생한 경우

6. 관련함수 : GetReocrdInfo(), ShowText(), ShowImage(), PlaySound()

7. 사용예 : ShowText 참조

주 의

1. 이 보고서는 과학기술처에서 시행한 특정연구개발사업의 연구보고서이다.
2. 이 연구개발내용을 대외적으로 발표할 때에는 반드시 과학기술처에서 시행한 특정연구개발사업의 연구결과임을 밝혀야 한다.