

튜 토 리 얼 Ⅲ

객체지향 멀티미디어 DB-ObjectStore

▷ 연 사 : 박 정 호 박사 (Datec)

▷ 사 회 : 나 연 목 교수 (단국대)

ObjectStore ODBMS와 ActiveX interface를 이용한 멀티미디어 DBMS 구현



데이텍 주식회사

박정호

jungho@datec.co.kr

Agenda

1. 멀티미디어 데이터 관리를 지원하기 위해
요구되는 ODBMS의 기능
2. ObjectStore ODBMS의 주요 특징과 제품군
3. ObjectStore ODBMS와 ActiveX Interface(OSAX)를
이용하여 Visual Basic에서 멀티미디어 DB 구축
어플리케이션 작성하기

1. 멀티미디어 데이터 관리를 지원하기 위해 요구되는 ODBMS의 기능

- 멀티미디어 데이터 관리를 위한 DBMS의 기술적 수준
- 멀티미디어 데이터베이스의 기본적인 이용 형태
- 멀티미디어 DBMS로서 ODBMS가 제공해야 할 역할

멀티미디어 데이터 관리를 위한 DBMS의 기술적 수준

• Data Repositories

- 보안이나 데이터 백업 지원을 목적으로 하는 간단한 DBMS 장치
- 저장되는 데이터의 포맷을 이해할 필요가 없음.
- 저장되는 객체들은 대개 BLOBs로 저장되기 때문에 단순 객체로 존재
- 질의는 meta data에 대해서만 가능하고, multimedia data에 대해서는 불가능

• Intelligent Data Management

- 관리하는 데이터를 이해할 수 있기 때문에 단순히 meta data에 대해서가 아니라 multimedia 객체에 대해서 질의가 가능

• Presentation Environment

- Multimedia 객체에 대한 질의는 물론 시간적, 공간적 구속을 가지는 복합 Multimedia data를 실시간적으로 배달하고 지휘 감독할 수 있음.

멀티미디어 데이터베이스의 기본적인 이용

<input type="checkbox"/> Read	데이터와 프레젠테이션을 retrieve & view
<input type="checkbox"/> Update	새로운 멀티미디어 데이터를 만들고 수정함
<input type="checkbox"/> Compose	기본적인 멀티미디어 데이터들을 이용하여 composition과 presentation을 만들
<input type="checkbox"/> Query	멀티미디어 데이터를 찾음. Meta data를 통해 간접적으로 질의하거나 멀티미디어 데이터 자체에 대해 직접적으로 질의함
<input type="checkbox"/> Interaction	멀티미디어 데이터와 함께 사용자의 ODBMS interaction을 포함. 데이터는 사용자 어플리케이션, 시스템 상호작용을 지원하기 위해 동적으로 만들어짐

멀티미디어 DBMS로서 ODBMS가 제공해야 할 역할

- Data types: 단순한 Data Repository : BLOBs 지원
Intelligent Data Management : 각종 멀티미디어 데이터 형 지원
- Data size: 크기가 큰 데이터에 대한 지원
멀티미디어 데이터는 일반적으로 크기가 상당히 큼
(MPEG-1으로 압축된 2시간 짜리 영화의 크기가 보통 4~5Gbytes)
- Viewing: Retrieve된 멀티미디어 객체를 화면에 묘사하거나 재생할 수 있는 기능 지원
Concurrent user 환경에서 적절한 QoS 레벨을 만족시켜야 함
- Querying: 멀티미디어 데이터에 대한 직접적인 질의를 지원하기 위해서 저장된 데이터에 대한 해독이 가능해야 하고 정교한 인덱싱 스키마와 내용에 특정 동작이나 기능에 대한 설명을 만들어 낼 수 있는 이미지/오디오 검색 알고리즘이 요구됨.
- Throughput: 각각의 멀티미디어 데이터 타입이 요구하는 시간적 구속을 만족시킬 수 있도록 S/W와 H/W를 최적화해야 함.

멀티미디어 DBMS로서 ODBMS가 제공해야 할 역할

- Resource scheduling : 다수의 사용자가 같은 디스크로부터 다른 데이터를 동시에 요청하는 경우, 멀티미디어 재생 및 기록 장치의 충돌이 일어나지 않도록 스케줄되어야 함.
- Memory, Bus, CPU : 멀티미디어 데이터를 처리하기에 충분한 정도로 메모리 공간, 버스 속도, 다중 프로세서 처리등을 지원해야 함.
- Special chip sets and Cards : 컴퓨터 장비 제조업체에서 데이터 캡처, 프레젠테이션, 변환 압축 및 복원, crop과 rotate와 같은 멀티미디어 동작을 위한 전용 chip set이나 board를 만들어 내고 있는데, 경우에 따라 이런 H/W 지원 디바이스 드라이버와 멀티미디어 데이터베이스간에 연동이 보장되어야 함. (멀티미디어 표준화 작업에 대한 지원)
- Storage : 멀티미디어 데이터를 위한 리퍼지토리가 디스크 어레이나 CD 주크박스와 같은 대형 스토리지 어레이를 포함해야 함. 또한, 비디오의 경우에는 throughput requirements로 인해, 2차 저장장치나 multiple request를 처리할 수 있을 만큼 빨라야 함.
- Networks : Throughput과 신뢰성을 제공할 수 있는 네트워크 인프라, 프로토콜 지원

객체 지향 데이터베이스

2. ObjectStore의 주요 특징과 제품군

ObjectStore의 주요 특징

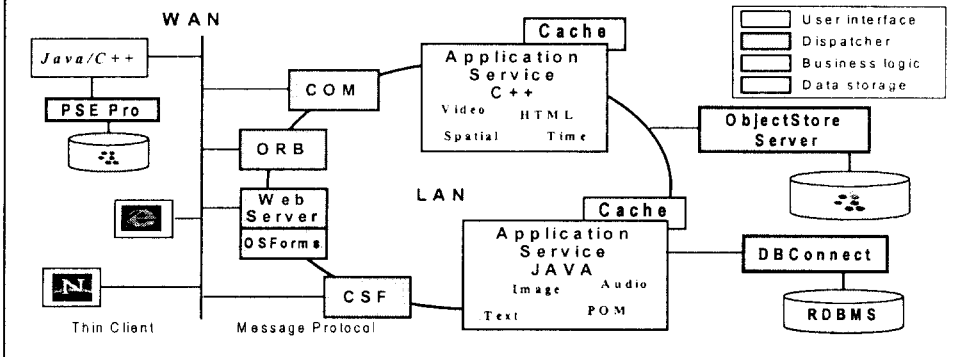
The Database for Component Based Computing
The Object Database Leader
Cache-Forward Architecture:
Delivers Component-Ready Data

Object Design, Inc. 제품군

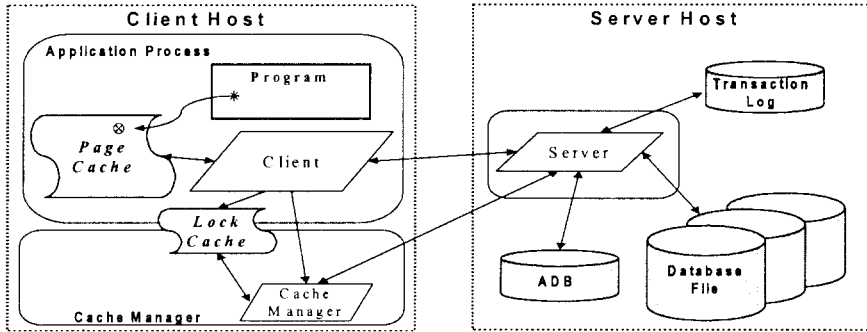
ObjectStore 5.1:
ObjectStore PSE Pro/PSE :
Rapid Development Tools
Enterprise Integration Tools



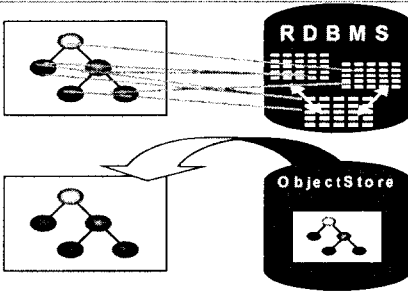
The Database for Component based computing



ObjectStore *Cache-Forward Architecture (CFA)*



Cache-Forward Architecture: Delivers Component-Ready Data



- RDBMS requires complex mapping code from tables to objects
- *Cache-Forward Architecture* delivers entire object in correct format

- No mapping code!
- Smaller component footprint
- Direct support for Java, C++

Result: Faster development and execution because CFA eliminates up to 60% of code

Object Design, Inc. 제품군

- **Object Storage Environments**
 - Object-Oriented Database Management System (OODBMS)
 - C++, Java and ActiveX interfaces
 - Persistent Storage Engines (PSE Pro / PSE)
 - C++, Java and ActiveX interfaces
- **Rapid Database Development Tools**
 - ObjectStore Inspector
 - ObjectStore Database Designer (OSDD)
 - ObjectStore Object Managers (OM)
 - ObjectStore Performance Expert (OPE)
- **Enterprise Integration Tools**
 - Component Server Framework (CSF)
 - Active toolkit (ATK)
 - ObjectForms (OF)
 - DBconnect

ObjectStore 5.1: The Object Database Leader

- Shipping since 1990
- APIs for Java, C++, ActiveX
- Robust, mission-critical database functionality:
 - Distributed database support
 - Asynchronous replication
 - Distributed backup/recovery
 - Automatic failover
 - Distributed archive logging
 - Event notification
- Platforms: NT, Win95, Solaris, SGI, HP, AIX, OS/2, Digital Unix, Alpha NT, and others

Designed for

 Microsoft®
 BackOffice®

BYTEBEST

1997



1991-1997

ObjectStore PSE Pro/PSE : Award-Winning Single-User DBMS

- Component data storage- Java, C++, ActiveX
- Small footprint (Under 450K)
- API compatible with ObjectStore ODBMS
- Licensed by industry leaders:



JavaWorld
 Editors' Choice
 1997



Microsoft



SYMANTEC

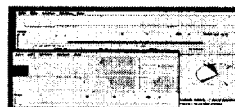
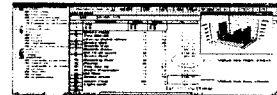
Borland



SUPERCEDE

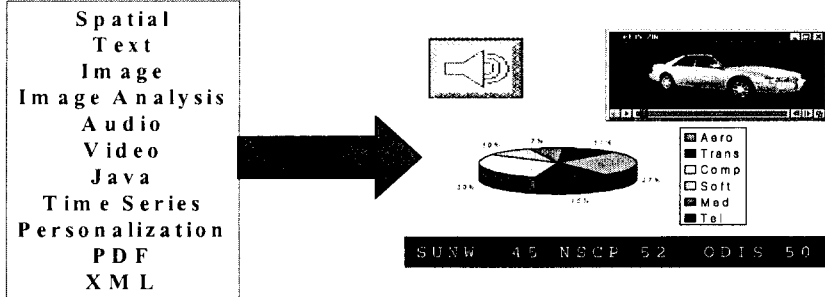
ObjectStore Rapid Development Tools

- Blueprint
 - Rational Rose plug-in generates schema
- Inspector
 - Graphical database browser & query tool
- Performance Expert
 - Performance monitoring & analysis
- Object Managers
 - High-performance data type components

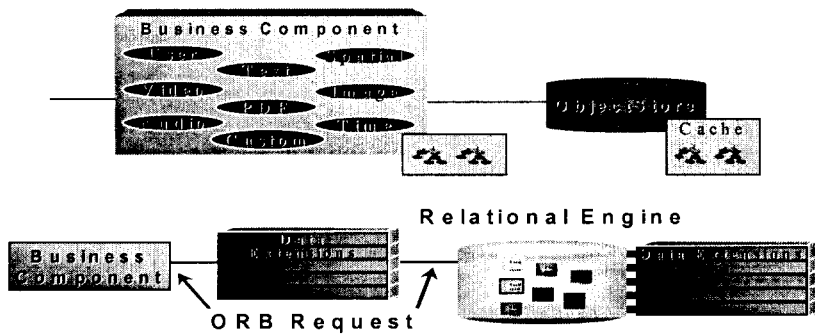


Object Managers: Ready to Use and Easy to Extend

Extensible class libraries

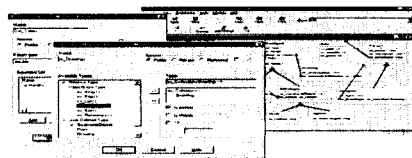


ObjectStore Object Managers: Productive, Fast and Reliable



ObjectStore Rapid Development Tools

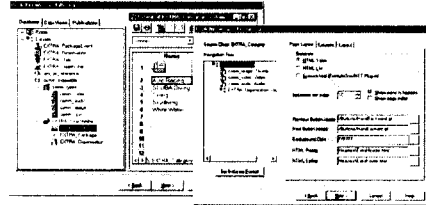
- ObjectStore Database Designer (OSDD)
 - Graphical designer for ObjectStore schema
 - Schemata translate to VC++ and Unix header/makefiles or JAVA Class Descriptions



- ObjectStore Component Wizard
 - Code Generation for ObjectStore Functions
 - Create, Retrieve, Update, Delete
 - VC++ Source
 - JAVA Source
 - Unix C++ Source
 - Code Generation for IDL
 - COM, JavaBeans, CORBA

ObjectStore Enterprise Integration Tools

- DBconnect:
 - Full connectivity with relational databases
- ObjectForms:
 - Integrates ObjectStore with any Web server
- Active Toolkit:
 - Full Integration With Microsoft standards



3. Visual Basic 과 ObjectStore 를 이용한 Multimedia DB 구축

프로그램 개발 환경

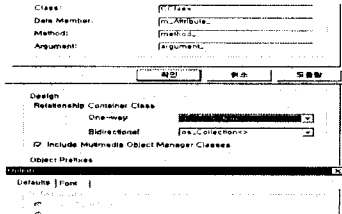
- 하드웨어 요구사항 :
펜티엄 90MHz 이상, 메인메모리 64M Bytes 이상,
하드디스크 300M Bytes 이상
- 소프트웨어 요구사항 :
ObjectStore ODBMS 5.1, OSAX 3.0, OSDD 1.0
ObjectStore Inspector 3.0, Virage Image ObjectManager 2.3
Video ObjectManager 2.3, Audio ObjectManager 2.3,
MS-Visual C++ 5.0 Service Pack 3,
MS-Visual Basic 5.0 Service Pack 3
- 플랫폼
 - Server : 한글 MS-Windows NT Server 4.0 Service Pack 3
 - Client : 한글 MS-Windows NT Workstation 4.0 or
한글 MS-Windows 95 / 98

ObjectStore ActiveX 컴포넌트를 사용하여 Multimedia DB Application 개발 순서

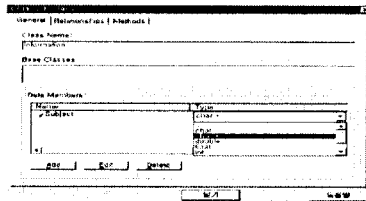
1. OSDD를 이용하여 어플리케이션 스키마를 디자인 한다.
2. Visual C++에서 OSDD를 사용하여 만들어진 데이터베이스 스키마를 이용하여 ActiveX 컴포넌트를 만든다. 이때, Visual C++에 설치된 ObjectStore Component Wizard를 사용하므로서 코딩을 하지 않고 곧바로 ActiveX 컴포넌트를 컴파일하여 생성한다.
3. Visual C++에서 만든 ActiveX를 등록하고 Visual Basic에서 이 ActiveX 컴포넌트를 이용하여 응용프로그램을 만든다.

OSDD 를 이용한 어플리케이션 스키마 디자인

- 어플리케이션 스키마 디자인
 - OSDD를 실행. File메뉴에서 New 항목 선택
 - Tools메뉴에서 Option 항목을 선택하여 Include Multimedia Object Manager Classes 체크박스를 체크한다. <그림 1>



<그림 1> OSDD 환경설정



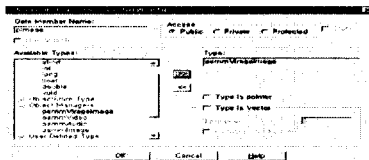
<그림 2> 클래스 타입 지정

OSDD 를 이용한 어플리케이션 스키마 디자인

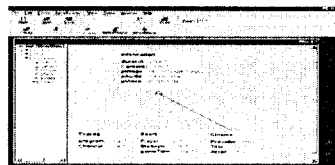
- 클래스 추가
Class 메뉴에서 Add Class Command
Class Name : Information
- 데이터 멤버 추가 (char * Name)
Data Members 그룹의 Name 아래를 클릭하고 멤버 이름을 입력하고 Type 을 클릭하면 멤버 타입 리스트 박스가 생기며, char *를 선택한다. <그림 2>
Data Members Name에 Content, Type은 char *으로 추가한다.
- 데이터 멤버에 이미지 타입 지정
Data Members 그룹에서 Add 버튼을 누른다.
"Information" Class : Data Member 다이얼로그 박스가 나타나며, Data Member Name 에 plmage 입력, Available Types 리스트 박스에서 Object Managers를 더블클릭하고, osmmViragelImage를 선택후 >> 버튼을 누른 다음 다이얼로그 박스 하단의 OK 버튼을 누른다. <그림 3>

OSDD 를 이용한 어플리케이션 스키마 디자인

같은 방법으로 pVideo, pAudio 멤버를 정의하도록 한다. 다시 닫기 버튼을 누르면 Information Class 가 만들어졌으며, 화면에 그림으로 표시된다.



<그림 3> 이미지 타입 지정



<그림 4> 상속관계 설정

OSDD 를 이용한 어플리케이션 스키마 디자인

• Cinema Class 생성

위와 같은 순서로 Class Name에는 Cinema, Data Member Name에는 Provider, Title, Actor, Type은 모두 char *로 입력한다. <그림 4>

상속 관계 설정 : Cinema Class는 Information Class로부터 상속
클래스에서 Inheritance 버튼 클릭 Cinema Class를 Shift-Click 한 다음
Information Class로 Drag한다.
Drag시 Information Class의 Information이 있는데 까지 Drag해야
상속이 설정된다.

OSDD 를 이용한 어플리케이션 스키마 디자인

• Sport Class 생성 및 상속 설정

같은 방법으로 Class Name에는 Sport, Data Member Name에는
Player, Stadium, gameTime, type은 모두 char *로 한 후 Sport Class를
Information으로부터 상속 관계 설정한다.
(Inheritance 클릭, Sport Shift-Click, Information 클래스로 Drag)

• TVprog Class 생성 및 상속 설정

같은 방법으로 Class Name에는 TVprog, Data Member Name에는
program, Channel, type은 모두 char *로 한 후 TVprog Class를 Information
으로부터 상속 관계 설정한다.
(Inheritance 클릭, TVprog Shift-Click, Information 클래스로 Drag)

• Action Class 생성

클래스를 만드는 같은 방법으로
Class Name : Action, Data Member Name : MovieKind Type : char *로
만들다.

OSDD 를 이용한 어플리케이션 스키마 디자인

• Relationship 정의

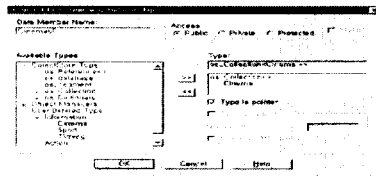
Action에 Cinema가 속해 있으므로 일대다의 관계가 성립한다.

이 관계를 One-way Relation이라고 하며 다음과 같이 설정한다

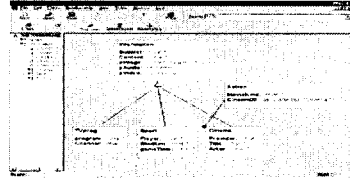
- 1)클래스에서 One-way 버튼 선택
- 2)Action Class를 Shift-Click 한 다음 Cinema Class로 Drag한다.
- 3)Data Member Name 에디트 박스에 Cinema00입력후 OK버튼 <그림 5>

Action Class에 Cinema 인스턴스의 collection인 Cinema00를 만들었다.
비록 간단한 스키마지만 이 스키마는 상속과 일대다 관계를 가지고 있다.
이와 유사한 방법으로 Sport에 대한 클래스를 만들어서 일대다 관계를 가지는
종목별 정보 제공 같은 것을 관리 할 수도 있을 것이다.

OSDD 를 이용한 어플리케이션 스키마 디자인



<그림 5> 관계설정



<그림 6> 관계설정후의 스키마

스키마 표시 방법은 OMT 표기법이며, View 메뉴에서 UML 표기법을 선택할 수도 있다. <그림 6> 이 스키마가 완성되었으므로 스키마를 저장한다. 스키마 파일은 .dbs로 저장된다. 아무이름으로 저장해도 관계없으나 Information.dbs로 저장한다. (MSVC 에서 사용)

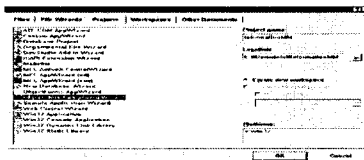
ObjectStore Component Wizard 이용한 ActiveX 컴포넌트 생성

• ActiveX Server 컴포넌트 생성

ActiveX Server 컴포넌트는 ObjectStore Component Wizard 를 이용하여 다음과 같이 간단하게 만들 수 있다.

- 1) Visual C++ 실행시킨다.
- 2) File 메뉴 New 항목 선택 New Dialog box에서 Project tab 선택한다.
- 3) OSAX를 설치하고 나면 여기에 ObjectStore App Wizard와 ObjectStore Component Wizard가 추가된다 여기서 ObjectStore Component Wizard를 선택하고 Project Name에는 InformationMM을 입력하고 Location을 정한다 음 OK 버튼을 누른다. <그림 7>
- 4) 다음 화면에서 Open Database Design 버튼을 누르고 OSDD를 이용하여 디자인한 스키마 파일을 지정한다. <그림 8>

ObjectStore Component Wizard 이용한 ActiveX 컴포넌트 생성



<그림 7> 오브젝트스토아 워저드 실행

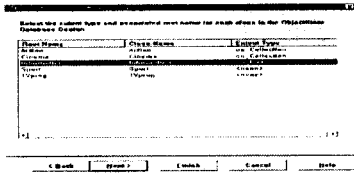


<그림 8> .dbs파일 지정

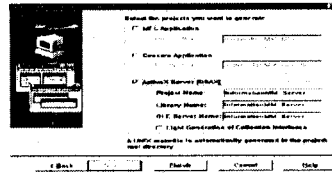
- 5) 다음 스텝에서 데이터 베이스의 Root명 과 Extent Type 을 설정한다. 데이터베이스 root명은 설정하지 않으면, 클래스 이름으로 설정되며, Extent Type은 디폴트로 os_Collection으로 설정된다. Root명은 그대로 두고 Extent Type만 바꾼다. 바꾸고자 하는 Extent Type에서 마우스를 클릭하면 리스트 박스가 나타나며 여기서 선택하면 된다. Sport와 TVprog는 none, Information은 os_List, 그리고 Action와 Cinema는 os_Collection 그대로 둔다. <그림 9>

ObjectStore Component Wizard 이용한 ActiveX 컴포넌트 생성

- 6) 다음 스텝에서 ActiveX Server (OSAX)를 체크하고 Finish 버튼을 누르면 소스코드가 생성된다. <그림 10>
이제 ActiveX Server 컴포넌트를 만들기 위한 소스코드가 모두 생성되었다. ObjectStore Component Wizard에 의해서 생성된 소스를 수정없이 그대로 컴파일 하면 ActiveX Server Component가 만들어진다. 물론 프로그래머의 취향에 맞추어서 ActiveX 인터페이스를 더 추가해도 무방하다.



<그림 9> Root 및 타입 지정



<그림 10> ActiveX Component 선택

ObjectStore Component Wizard 이용한 ActiveX 컴포넌트 생성

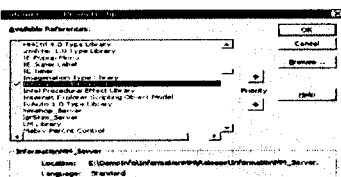
- 7) Project Workspace에서 InformationMM_ServerClass에 오른쪽 버튼을 누른 뒤 Set as active project를 클릭한 다음 F7을 눌러 InformationMM_Server를 Build한다. 컴파일이 정상적으로 끝나고 링크가 되고 나면, Visual C++가 InformationMM_Server.dll을 Registry에 자동으로 등록한다.

지금까지는 Visual Basic에서 사용할 ActiveX Server Component를 만들었다. 만들어진 DLL은 COM (Component Object Model) 인터페이스로서 DCOM (Distributed COM)과 함께 분산 네트워크 환경에서 ObjectStore를 사용할 수 있다.

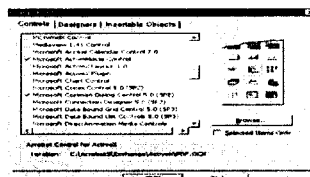
또한 마이크로소프트의 MTS나 ASP에서도 사용할 수가 있으므로 Web application에서 직접 ObjectStore의 모든 기능들을 사용할 수 있으며, 특히 대용량의 데이터와 접속자수가 많은 웹 어플리케이션에 최적의 성능을 제공한다. 멀티미디어 데이터를 간편히 저장, 조회할 수 있을 뿐만 아니라, RDB에서처럼 Table Join에 의해서 데이터를 검색하지 않고 오브젝트 포인터로 데이터를 검색하기 때문에 RDB보다 훨씬 빠른 속도로 저장, 검색, 조회가 가능하다.

Visual Basic에서 ActiveX 컴포넌트를 이용하여 ObjectStore용 DB 응용 프로그램 만들기

- 응용 프로그램 작성
 - 1) 비주얼 베이직 실행
 - 2) 새 프로젝트에서 표준 EXE를 선택한다.
 - 3) 멀티미디어 ActiveX 컴포넌트 등록
프로젝트 메뉴의 참조 항목에서 InformationMM_Server를 체크한다. <그림 11>
프로젝트 메뉴의 구성요소 항목에서 Microsoft Common Dialog Control과 Microsoft ActiveMovie Control 체크 <그림 12>



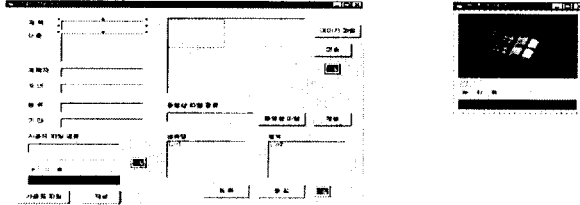
<그림 11> VB에서 ActiveX Component 선택



<그림 12> Common Dialog 선택

Visual Basic 에서 ActiveX 컴포넌트를 이용하여 ObjectStore 용 DB 응용 프로그램 만들기

- 4) 비주얼 베이직 폼디자인
비주얼 베이직 폼을 위와 같이 디자인 하고 각 컴포넌트 이름을 다음과 같이 부여한다.



<그림 13> 비주얼 베이직 폼

Visual Basic 에서 ActiveX 컴포넌트를 이용하여 ObjectStore 용 DB 응용 프로그램 만들기

컴포넌트	컴포넌트명	컴포넌트	컴포넌트명
제목 입력 Edit Box	Text1	제목 리스트 박스	List2
내용 입력 Edit Box	Text2	이미지가져오기	Command1
제작자 입력 Edit Box	Text3	동영상가져오기	Command3
주연 입력 Edit Box	Text4	재생	Command4
분류 입력 Edit Box	Text5	사운드가져오기	Command6
기타 입력 Edit Box	Text6	재생	Command5
Picture box	Picture1	종료	Command8
Image	Image1	사진읽슴	Command2
분류명 리스트박스	List1	CommonDialog	CommonDialog1
등록	Command7	Commondialog	cdAV1
		Commondialog	MpAudio

Visual Basic 에서 ActiveX 컴포넌트를 이용하여 ObjectStore 용 DB 응용 프로그램 만들기

- 비주얼 베이직에서 사용되는 오브젝트스토어 OSAX 컴포넌트

ObjectStore.OpenDatabase (Name as String, [ReadOnly],[Create Mode],[SchemaDatabase]) as IOSAXDatabase : 오브젝트스토어 데이터베이스 오픈

Name : Database Name,
ReadOnly : Boolean, 1=ReadOnly, 0=ReadWrite
CreateMode : int, 0 = Open only, 1 = Open if exist, create if not exist
Return : IOSAXDatabase 데이터 오브젝트

ObjectStore.BeginTrans : 오브젝트스토어 오브젝트 트랜잭션 시작, 모든 오브젝트에 대한 동작은 트랜잭션안에서 일어나야 한다.

ObjectStore.CommitTrans : 트랜잭션 커밋
ObjectStore.Rollback : 트랜잭션 취소, 취소한 트랜잭션에 의해 변경된 오브젝트들은 그 이전의 상태로 되돌려진다.

Visual Basic 에서 ActiveX 컴포넌트를 이용하여 ObjectStore 용 DB 응용 프로그램 만들기

IOSAXDatabase : 오브젝트스토어 데이터베이스 클래스
 IOSAXDatabase.Close () : 데이터 베이스 닫기

IAction : 비주얼 베이직에서 사용하기 위한 Action 클래스 OSDD에서 작성한 클래스의 이름앞에 I가 붙어있다. 오브젝트스토어 워자드에서 생성
 ICinema : Cinema 클래스
 CAction : Action 클래스의 타입
 CCinema : Cinema의 클래스 타입
 IActionCollection : ActionCollection에 대한 클래스
 IActionCollection.evaluate : Collection IActionCollection에 대한 Query
 Query string은 C++문법을 따른다.

- 글로벌변수 선언
 Dim osDB as IOSAXDatabase
 Dim theAct as IAction

Visual Basic 에서 ActiveX 컴포넌트를 이용하여 ObjectStore 용 DB 응용 프로그램 만들기

여기서 osDB 오브젝트는 IOSAXDatabase 클래스의 인스턴스이며, theAct오브젝트는 IAction 클래스의 인스턴스이다.

- 데이터 베이스 오픈
 Private Sub Form_Load()
 Set osDB = ObjectStore.OpenDatabase("information db", 0, 1)
 End Sub
- 데이터베이스로부터 오브젝트 읽어오기
 ObjectStore.BeginTrans
 Set actCol = osDB.Value("Action", CActionCollection)
 For Each theAct In actCol
 List1.AddItem theAct.Subject
 Next theAct
 ObjectStore.CommitTrans

Visual Basic 에서 ActiveX 컴포넌트를 이용하여 ObjectStore 용 DB 응용 프로그램 만들기

osDB.Value "Action" 데이터베이스 Root, 데이터베이스 Root는 데이터 베이스에 대한 Entrypoint이다 모든오브젝트는 데이터베이스 Root값과, 타입을 통해서 Entrypoint를 얻는다 즉 actCol은 "Action" Root에 Action에 대한 Collection타입으로 맵핑되며, actCol은 Collection으로 되었으므로 Collection내부의 오브젝트에 대해서 반복해서 Collection의 마지막 오브젝트까지 Subject을 List1 리스트 박스에 추가한다.

- 데이터베이스 Query
 오브젝트의 멤버에 대한 질의를 할수있다 질의 문법은 C++에서와 똑같이 한다. 질의 문장이 String으로 전달되므로 다음과같이 C++문법을 사용한다.

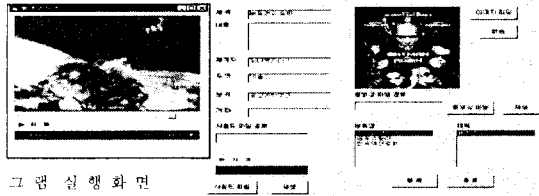
QueryString = "Action*" & strcmp(Subject, "" & List1 text & "")

질의 결과는 다시 Collection으로 Return되며, 이 Collection에 질의 결과 오브젝트가 반환된다.

Visual Basic 에서 ActiveX 컴포넌트를 이용하여 ObjectStore 용 DB 응용 프로그램 만들기

예제 프로그램의 동작 실행 :

주어진 입력 에디트 박스에 제목, 내용, 제작자, 주연 배우, 분류, 기타를 입력하고 각각의 멀티미디어 자료인 이미지, 동영상, 사운드를 가져오기 버튼으로 표시한 다음, 등록 버튼을 누르면, 그 내용들이 ObjectStore에 저장되며, 아래의 리스트 박스에 표시가 된다. 리스트 박스에서 분류명과 제목을 클릭하면, 선택된 영화에 대한 정보를 ObjectStore에서 검색해서 이미지와 함께 화면에 표시되고, 재생 버튼을 클릭하면, 동영상이나, 사운드가 재생된다.



<그림 15> 프로그램 실행 화면

Visual Basic 에서 ActiveX 컴포넌트를 이용하여 ObjectStore 용 DB 응용 프로그램 만들기

ObjectStore ActiveX 컴포넌트를 사용하면 ...

C++에서의 마찬가지로 ObjectStore에 있는 영속 객체를 사용할 수 있을 뿐만 아니라 COM과 DCOM을 이용한 분산 객체 환경에서 ODB Application의 개발이 가능.

ObjectStore Component Wizard를 이용하여 만들어진 ActiveX Server Component는 COM 인터페이스로서 수정 없이 MTS에서 사용 가능하므로 Multi-tier 구조로 DB Application을 확장시킬 수 있으며, ASP에서도 사용 가능하므로 멀티 미디어 데이터 많이 다루는 Web Application에 ObjectStore를 쉽게 적용시켜, 다양하고 효율 높은 멀티 미디어 응용 프로그램의 개발이 가능.

ObjectStore: More Information ...

- World Wide Web
 - ODI : www.odi.com, www.objectdesign.com
 - DATEC : www.datec.co.kr
- Mailing List Subscription
 - majordomo@odi.com
- Newsgroup
 - comp.datebases.object
- Performance Demo
 - <http://demo.str.com/javademo/ODIFrame.html>