Multimedia authoring and user interface

Multimedia systems are different from other systems in two main respects the variety of information objects used in application and the level of integration achieved in using these objects in complex interconnected applications. In multimedia applications, the user creates and controls the flow of data and determines the expected rendering of it. For this reason, applications that allow users to create multimedia objects and link or embed them in other compound objects such as documents or database records are called authoring systems.

Multimedia authoring systems

Authoring systems for multimedia applications are designed with the following two primary target users in mind: professionals who prepare documents, audio or soundtracks, and full motion video clips for wide distribution and average business users preparing documents, audio recording, or full motion video clips for stored messages or presentations.

Design issues of multimedia authoring

Display resolution
Data formats for captured data
Compression algorithms
Network interfaces
Storage formats

A number of design issues must be considered for handling different display outputs such as Level of standardization on display resolutions

Display protocol standardization

Corporate norms for service degradations

Corporate norms for network traffic degradations as they relate to resolutions issues.

File format and data compression issues

The primary concern with very large objects is being able to locate them quickly and being able to play them back efficiently. In almost all cases the objects are compressed in some form. There is, however, another aspect of storage that is equally important from a design perspective. It is useful to have some information about the object itself available outside the object to allow a user to decide if they need to access the object data.

1.compression type
2.Estimated time to decompress and display or play back the object [for audio and full motion video]
3.Size of the object [for images or if the user wants to download the object to a notebook]
4.Object orientation [for images]
5.Annotation markers and history[for images and sound or full motion video]
6.Index markers [for sound full motion video]
7.Data and time of creation
8.Source file

Design approaches to authoring

Designing an authoring system spands a number of critical design issues,including
The following;

1.hypermedia application design specifics
2.user interface aspects
3.embedding / linking streams of objects to a main document or presentation
4.storage of and access to multimedia objects.
5. Playing back combined streams in a synchronized manner.

Hypermedia applications bring together a number of design issues not commonly encountered in other types of applications. However as in any other application type, a good user interface design is crucial to the success of a hypermedia application. The user interface presents a window to the user for controlling storage and retrieval , inserting objects in the document and specifying the exact point of insertion, and defining index marks for combining different multimedia streams and the rules for playing them back.

Types of multimedia authoring systems

Dedicated authoring systems
Timeline-based authoring
Structured multimedia authoring
Programmable authoring systems
Multisource multi-user authoring system
Telephone authoring systems

Hypermedia application design considerations

Multimedia applications are based on a totally new metaphor that combines the television, VCR and window-based application manager in one screen. The user interface must be highly intuitive to allow the user to learn the tools quickly and be able to use them effectively.

A good designers needs to determine the strategic points during the execution of an application
where user feedback is essential or very useful.

The following steps for good hypermedia design

1. Determining the type of hypermedia application
2. Structuring the information
3. Determining the navigation throughout the application
4. Methodologies for accessing the information
5. Designing the user interface

Integration of applications

Depending on the job function of the knowledge worker, the computer may be called upon to run a diverse set of applications, including some combination of the following.

Electronic mail
Word processing
Graphics and formal presentation preparation software

Spreadsheet
Access to a relational
Customized applications directly related to job function

Common UI and application integration

The Microsoft has different user interface for a large number of applications by providing standardization at the following levels.
Overall visual look and feel of the application windows
Menus
Dialog boxes
Buttons
Help feature
Scroll bars
Tool bars
File open and save

Data Exchange

The MS clipboard allows exchanging data in any format. The clipboard can be used to exchange multimedia objects as well, including cutting or copying a multimedia object in one document and pasting it
in another.

Distributed data access

Application integration succeeds only if all applications required for a compound object can access
the sub objects that they manipulate. Fully distributed data access implies that any application at any client
workstation in the enterprise-wide WAN must be able to access any data object s if it were local.

User interface design

User interface design for multimedia applications is more involved than for other applications due to the number of types of interactions with the user.
Four kinds of user interface design is available such as
Media editors
An authoring application
Hypermedia object creation
Multimedia object locator and browser
A media editor is an application responsible for the creation and editing of a specific multimedia object such as an image, voice or video object.

Designing user interface

The correctness of a user interface is a perception of a user.
Guidelines

Planning the overall structure of the application
Planning the content of the application
Planning the interactive behavior
Planning the look and feel of the application

Special metaphors for multimedia applications

Multimedia applications bring together two key technologies; entertainment and business computing.

The organizer metaphor

The multimedia aspects of the organizer are not very obvious until one begins to associate the concept of embedding multimedia objects in the appointment diary or notepad for future filling.

The telephone metaphor

The telephone, until very recently, was considered an independent office appliance. The advent of voice mail systems was the first step in changing the role of the telephone.

Aural user interface

The common approach for speech-recognition based user interfaces has been to graft the speech recognition interface into existing graphical user interfaces. This is a mix of conceptually mismatched media that makes the interface cumbersome and not very efficient.

The real challenge in designing AUI systems is to create an aural desktop that substitutes voice and ear for the keyboard and display, and be able to mix and match them.

The VCR metaphor

The easiest user interface for functions such as video capture, channel play and stored video playback is to emulate the camera, television, and VCR on screen.

Audio and video indexing functions

Audio tape indexing has been used by a large number of tape recorders since the early 1950s.

Index marking on tape is a function that has been available in many commercial VCRs. Index marking on

tape left a physical index mark on the tape. These index marks could be used in fast forwarded and rewind

searches.

Hypermedia messaging

E-mail based document interchange, generally known as messaging services. Messaging is one of the major multimedia applications. Mobile messaging represents a major new dimension in the user's interaction with the messaging system. Handheld and desktop devices, an important growth area for messaging, require complementary back-end services to effectively manage communications for a large organization. An answering service can take multiple messages simultaneously irrespective of line usage. The roles of telephone carries and local cable companies are starting to blur.

Hypermedia message components:

A hypermedia message may be a simple message in the form of text with an embedded graphics, soundtrack, or video clip. The components of hypermedia messages are handled through the following
three steps.

- The user may have watched some video presentation on the material and may want to attach a part of that clip in the message.
- Some pages of the book are scanned as images. The images provide an illustration or a clearer analysis of the topic.
- The user writes the text of the message using a word processor.

When the message is fully composed, the user signs it and mails the message to the addressee. The messaging system must ensure that the images and video clips referenced in the message are also transferred to a server local to recipient.

Message types
Text messages
Rich-text messages
Voice messages
Full-motion video management

Hypermedia linking and embedding

Linking and embedding are two methods for associating multimedia objects with documents.

- Linking as in hypermedia applications. Hypertext systems associate keywords in a document with other documents.
- Linking multimedia objects stored separately from the document and the link provides a pointer
to its storage. An embedded object is a part of the document and is retrieved when the document is retrieved.
- Linking and embedding in a context specific to Microsoft object linking and embedding.

When a multimedia object is incorporated in a document, its behavior depends on whether it is linked or embedded. The difference between linking and embedding stems from how and where the actual source data that comprises the multimedia object resides.

Linking objects:

When an object is linked, the source data object, called the link source, continues to reside wherever it was as the time the link was created. This may be at the object server where it was created, or where it may have been copied in a subsequent replication.

Embedding objects:

When the multimedia object is embedded, a copy of the object is physically stored in the hypermedia document. Graphics and images can be inserted in a rich-text document or embedded using such techniques as OLE.

Design issues:

For users who have a requirement for component documents, OLE represents an important advancement in systems and application software on distributed platforms. OLE will create significant support headaches for users if there is incomplete link tracking between documents that have been mailed between PCs and the application which created those objects.

Creating hypermedia messages:

A hypermedia message can be a complex collection of a variety of objects. It is an integrated message consisting of text, binary files, images, bitmaps, voice and sound.

Procedure:

Planning

Creating each component

Integrating components

Integrated multimedia message standards:

As text-based technologies have progressed and have become increasingly integrated with messaging systems, new standards are being developed to address interoperability of application from different software vendors.

Vendor-independent messaging

Vendor independent messaging interface is designed to facilitate messaging between VIM-enabled electronic mail systems as well as other applications. A VIM interface makes mail and messaging services available through a well-defined interface. A messaging service enables its clients to communicate with each other in a store-and –forward manner. VIM defines messaging as the data exchange mechanism between VIM aware applications.

VIM mail message is a message of a well defined type that must include a message header and may include note parts, attachments, and other application-defined components.

VIM services

The VIM interface provides a number of services for creating and mailing a message such as,

Electronic message composition and submission

Electronic message sending and receiving

Message extraction from mail system

Address book services

The developers of VIM targeted four areas in which VIM could fit into the business process: mail

enabling existing applications, creating alert utilities, creating scheduling applications, and helping workflow

applications. The benefits of implementing applications in each of these four areas vary significantly.

MAPI

MAPI is to provide a messaging architecture rather than just providing a messaging API in windows. MAPI provides a layer of functionality between applications and underlying messaging systems.

Goals

Separate client applications from the underlying messaging services
Make basic mail-enabling a standard feature for all applications
Support messaging- reliant workgroup applications

Telephony API

The TAPI standard has been defined by Microsoft and Intel, and has been upgraded through successive release to stay abreast of on going technology changes.

X 400 Message handling service

The MHS describe a functional model that provides end users the ability to send and receive electronic messages. A user agent is an entity that provides the end user function for composing and sending messages as well as for delivering messages. Most user agent implementations also provide local mail management functions such as storage of mail, sorting mail in folders, purging and forwarding. When a user composes a message and sends it the UA communicates the message to and MTA. If there is no local MTA, the message is forwarded to the MTA in a submission envelope based on one of a set of message protocol data units(MPDU) defined in the submission and delivery protocol. The delivery protocol is designed to use a remote operations service and optionally, a reliable transfer service to submit message to the MTA.

A collection of MTAs and UAs constitutes a management domain. Administrative management domains are public services such as AT&T, Sprint and so on.

Distributed multimedia systems:

A multimedia system consists of a number of components , which are distributed and dedicated function with different locations.

Components:

Application software
Container object store
Image and still video store
Audio and video component store
Object directory service agent
Component service agent
User interface service agent
Networks

The application software is the multimedia application that creates, edits or renders multimedia objects.

The container object store is used to store container objects in a network object server.

A image or still video store is a mass storage component for images and still video.

An audio/video component store is the storage resource used for storing audio and video objects.

An object directory service agent is responsible for assigning identification for all multimedia object types managed by that agent.

A component service agent is responsible for locating each embedded or linked component object of a multimedia container, and managing proper sequencing for rendering of the multimedia objects.

A user interface service agent is responsible for managing the display windows on a user workstation, interacting with the user, sizing the display windows, and scaling the decompressed object to the selected window size.

The network as used in this context refers to the corporate wide network consisting of all LAN and WAN interfaces required for supporting a particular application for a specific group of users.

Distributed client-server operation:

While the client server architecture has been used for some time for relational databases such as Sybase and Oracle. Most client-server systems were designed to connect a client across a network to a server that provided database functions. The clients in this case were custom-designed for the server.

Client in distributed workgroup computing

The client systems interact with the data servers in any of the following ways:

1. Request specific textual data
2. Request specific multimedia objects embedded
3. Require activation of rendering server application to display
4. Create and store multimedia objects on servers
5. Request directory information on locations of objects on servers.

Servers in distributed workgroup computing

1. Provide storage for a variety of object classes
2. Transfer objects on demand to clients
3. Provide hierarchical storage for moving unused objects to near-line media
4. System administration functions for backing up stored data
5. Direct high-speed LAN and WAN server-to-server transport for copying multimedia objects.

Database operations

Search

Browse

Retrieve

Create and store

Update

Middleware in distributed workgroup computing

1. Provide the user with a local index, an object directory, for objects with which a client is concerned
2. Provide automatic object directory services for locating available copies of objects
3. Provide protocol and data format conversations between the client requests and the stored formats in the server
4. Provide unique identification throughout the enterprise wide network for every object through time.

Multimedia object servers:

The resources where information objects are stored so that they remain sharable across the network are called servers.

Types of multimedia servers
Data processing servers
Document database servers
Document imaging and still video servers
Audio and voice mail servers
Full-motion video servers.
Network topologies for multimedia object servers
Centralized multimedia server
Dedicated multimedia servers
Distributed multimedia servers
Multimedia network topologies
Traditional LAN
Extended LANs
High-speed LANs
WANs
Distributed multimedia database
A multimedia database consists of a number of different types of multimedia objects.
Database organization for multimedia applications
Data independence
Common distributed database architecture
Multiple data servers
Transaction management for multimedia systems
Managing hypermedia records as objects
Multimedia objects need not always be embedded in the database record or a hypermedia document; instead, a reference can be embedded, and the multimedia object can reside separately in its own database, potentially optimized for that type of multimedia object.
Managing distributed objects
The issues are
How objects are located, and once located , how retrieval is managed in a multi-user environment, replication, archival, load balancing and purging.
The above issues are addressed with the following concepts
Interserver communications
Object server architecture
Object identification
Object revision management
Optimizing network location of objects
Object directory services