Working Implementation Agreements for Open Systems Environment: Part 32 - Multimedia and Hypermedia: Model and Framework

Output from the June 1994 Open Systems Environment Implementors' Workshop (OIW)

Acting SIG Chair: **Jon Stewart, Consultant**
SIG Editor: **Mike Rubinfeld, NIST** Workshop Editor: **Brenda Gray, NIST**

Foreword

This part of the Working Implementation Agreements was prepared by the Multimedia Data and Document Interchange (MDDI) Special Interest Group (SIG) of the Open Systems Environment Implementors' Workshop (OIW).

All of the text in this part has been approved by the Plenary of the above-mentioned Workshop for inclusion in this document.

The generation history of this document is:

- Generated 9-14 February 1992 WG1 meeting in Australia.
- Modified 5-9 October 1992 WG 1 meeting in Germany.
- Modified 3-7 May 1993 WG 1 meeting in The Buckerell Lodge Hotel & Restaurant in Exeter England.
  Modified 6-9 September 1993 meeting at the Embassy Suites Hotel in St. Louis, Missouri.
  Modified as of 8 January 1994 as a result of the September meeting and associated research was performed since that time. All changes were performed as documented in the response to comments and started a background section to provide a context for the next model.

Future changes and additions to this version of these Implementor Agreements will be published as a new part. Deleted and replaced text will be shown as struckout. New and replacement text will be shown as shaded.

Table of Contents

Part 32 - Multimedia and Hypermedia: Model and Framework 1

0 Introduction 2
Audience 2
Background 2
Overview 2
Purpose and justification 2

1 Scope and field of applications 3

2 Normative references 4
2.1 ISO 4
2.2 ISO/IEC 4
2.3 Other References 6

3 Definitions and terminology 6
3.1 Action 6
3.2 Anchor 6
3.3 Atomic object 6
3.4 Bi–directional link 7
3.5 Bit combination 7
3.6 Class 7
3.7 Coded representation 7
3.8 Component object 7
3.9 Composite object 7
3.10 Computer mediated device 7
3.11 Content 7
3.12 Content object 8
3.13 Document 8
3.14 Document set 8
3.15 Event 8
3.16 Final form 8
3.17 Framework 8
3.18 Hypermedia 8
3.19 Hypermedia/Time–based Structuring Language: HyTime 8
3.20 Hyperobject 9
3.21 Hypertext 9
3.22 Interchange medium 9
3.23 Isochronous 9
3.24 Link 9
3.25 Medium (plural media) 9
3.26 Model 9
3.27 Monomedia 10
3.28 Monomedia object 10

3.29 Multimedia . . . 10
3.30 Multimedia object . . . 10
3.31 Multimedia representation . . . 10
3.32 Object . . . 10
3.33 Object class . . . 11
3.34 Object instance . . . 11
3.35 Perception medium . . . 11
3.36 Presentation . . . 11
3.37 Presentation medium . . . 11
3.38 Process . . . 11
3.39 Projection . . . 11
3.40 Rendition . . . 11
3.41 Representation . . . 12
3.42 Representation medium . . . 12
3.43 Representation of an object . . . 12
3.44 Script . . . 12
3.45 Scriptware (script software) . . . 13
3.46 Storage medium . . . 13
3.47 Structure . . . 13
3.48 Transmission medium . . . 13
3.49 Unidirectional link . . . 13
3.50 View . . . 13
3.51 Web . . . 13

4 Symbols and abbreviations . . . 14

5 Requirements . . . 14

6 Open Systems Environment Reference Model . . . 15
6.1 Open systems environment reference model Issues . . . 18
6.2 Information models. . . . 19
6.2.1 Content . . . 19
6.2.2 Structure . . . 20
6.3 Multimedia and hypermedia service areas . . . 20
6.3.1 User interface services . . . 21
6.3.2 Object identification services . . . 21
6.3.3 Object management services . . . 21
6.3.4 Information production services . . . 21
6.3.5 Application production services . . . 22
6.3.6 Intellectual property management services . . . 22
6.3.7 Interchange services . . . 23
6.3.8 Procedural language processing services . . . 23
6.3.9 Other services . . . 23
6.3.10 Open systems environment model entities . . . 23

7 A JTC1 Multimedia and Hypermedia Framework . . . 24
7.1 The Distributed Hyper–Document Model . . . 24
7.2 Characteristics of Content Objects (COs) . . . 24

7.3 Composite CO Documents . . . 25
7.4 Migration/Version Update Requirement . . . 26
7.5 Dynamic Linking Requirement . . . 26
7.6 Subsetting or Specializing the Reference Model . . . 27
7.7 Conclusions/Recommendations for Future Work on the Reference Model . . . 27

Annex A (informative)

Relationships to other JTC1 reference models . . . 28
A.1 Relationship to the computer graphics reference model . . . 28
A.2 Relationship to the open distributed processing reference model . . . 28
A.2.1 Overview . . . 28
A.2.2 Objective of ODP . . . 29
A.2.3 Reference points and conformance points . . . 31
A.2.4 Framework of abstractions: Viewpoints . . . 31
A.2.5 Structure of the RM–ODP and conclusion . . . 32

Annex B (informative)

Multimedia and Hypermedia Detailed Standards Descriptions . . . 33
B.1 Content Standards Descriptions . . . 33
B.2 Multimedia related Structure Standards Descriptions . . . 36
B.3 Multimedia and hypermedia service areas . . . 37
B.3.1 User interface services . . . 37
B.3.2 Object identification services . . . 40
B.3.3 Object management services . . . 40
B.3.4 Information production services . . . 41

B.3.5 Application production services . . . 42

B.3.6 Intellectual property management services . . . 42

B.3.7 Interchange services . . . 42

B.3.8 Procedural language processing services . . . 51

B.3.9 Other services . . . 53

Annex C (informative)

References . . . 55
C.1 References . . . 55
C.2 Multimedia and hypermedia model and framework input documents . . . 55

Annex D (informative)

Application scenarios 57
D.1 Multimedia CD creation and Use 57
D.2 Multimedia mail creation and receiving 57
D.3 Multimedia conferencing 57
D.4 Access to Multimedia Objects on the Internet 57
D.5 Multimedia Interactive Training 57
D.6 Multimedia Games and Entertainment 57

Annex E (informative)

Multimedia Related Acronym Definitions 58

Annex F (informative)

Computer Related Organizations Acronyms Definitions 59

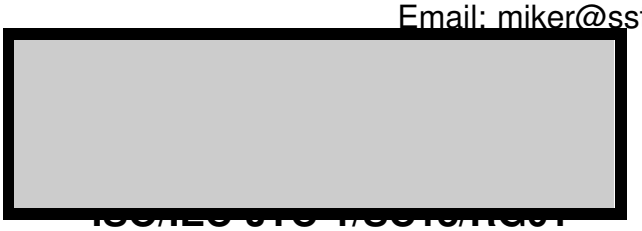
List of Figures

Figure 1 1
Figure 2: A Multimedia base model 17
Figure 3 - OSE Reference Model (Basis for the MHMF) 22

List of Tables

Part 32 - Multimedia and Hypermedia: Model and Framework

CONVENER:	Mr. Ed Wade Government of Canada 1974 Neepawa Ottawa, Ontario K2A 3L5 Canada	ISO/IEC JTC 1/SC18/RG01 N 1642 MHMF Rapporteur Group Editor: Mr. Mike Rubinfeld National Institute of Standards and Technology (NIST) Bldg. 225, B266 Gaithersburg, MD 20899 Telephone: +1 301 975 3064 Facsimile: +1 301 926 3696 Email: miker@sst.ncsl.nist.gov



MHMF Rapporteur Group

DATE:	June 12, 1994
TITLE:	Working Draft of the Technical Report on Multimedia and Hypermedia: Model and Framework
SOURCE:	SC18/RG01 (The MHMF Rapporteur Group)
STATUS:	Output of SC18/WG1 meeting 18 April 1994 - 20 April 1994 at the Nation Institute of Standards and Technology (NIST)
REQUESTED ACTION:	National bodies and affected subcommittees are required to

Figure 1

0 Introduction

Multimedia systems are a collection of technologies that combines the formats of text, voice, video, image, and animation for delivery to workstations, specialized platforms, or existing audiovisual equipment. While the individual functions incorporated in multimedia systems have been implemented in one form or another, no attention was paid to existing standards, and as a result, portability, interoperability and interchange functionality have been diminished within this new paradigm. This document is intended to define and summarize the current standards associated with various forms of multimedia information.

Audience

This technical report is intended for use by standards developers, standards managers and other members of the standards community. The model and framework could also be used as a reference by application developers and interested end users.

Background

Multimedia and hypermedia are current complex topics that could be better understood by depicting the associated domains in terms of an architectural framework. By defining this framework, existing applicable standards can more readily be associated with the model and those areas in need of standardization can easily be identified with respect to the model.

Overview

This document presents both an Open Systems model of Multimedia and Hypermedia and a framework based on the model showing how existing standards and standards activities interrelate.

Purpose and justification

The areas of multimedia and hypermedia are of significant importance in information processing and are currently the subject of considerable interest. There is also, however, a great deal of confusion regarding these subjects. This work is needed by JTC1 to permit the definition of relevant standards for the orderly development and implementation of multimedia and hypermedia applications.

The model identifies:

— functionality that is required by multimedia and hypermedia objects and hypermedia compound documents;

— interfaces, protocols, services, content notations, languages, supporting formats and encodings;

— hypermedia/multimedia issues which are: application specific, domain specific, or part of the base level hypermedia and multimedia technologies; and

— barriers to interoperability and portability for multimedia applications.

The model allows the relationships between various aspects of multimedia and hypermedia systems to be identified and understood. As an example, the model identifies features such as:

— presentation, i.e., the user system interface;

— database, i.e., the storage, access and retrieval of information;

— communication architecture and network, e.g., protocol/service;

— formats on interchangeable storage media, i.e., physical, file structure; and

— multimedia/hypermedia abstract information structure, i.e., the definition of the information nodes and the relationships among those nodes.

The framework identifies:

— existing standards needed by multimedia and hypermedia applications;

— how the standards interrelate;

— JTC1 standards needed by multimedia and hypermedia applications that need to be defined, extended, or modified;

— recommendations on where those standards could be developed or revised;

— areas under simultaneous development by two or more committees (for possible national body action); and

— definitions of the terms employed in multimedia and hypermedia standards.

1 Scope and field of applications

This technical report is intended to provide a descriptive model of the totality of multimedia and hypermedia systems (not constrained to JTC1 activities).

This document also provides and specifies a framework for current and future multimedia and hypermedia standardization within JTC1. A fundamental aspect of the work will be to develop an agreed definition of multimedia and hypermedia entities. Besides providing a reference framework for multimedia systems, an attempt is made to categorize and provide an overview of important related current standards.

2 Normative references

The following standards and technical reports contain provisions which, through reference in this text, constitute provisions of this International Technical Report. At the time of publication, the editions indicated were valid. All standards and technical reports are subject to revision, and parties to agreements based on this International Technical Report are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

2.1 ISO

[**1**] ISO 7498: 1984, *Information processing systems - Open systems interconnection - Basic reference model*;

[**2**] ISO 7498-2: 1989, *Information processing systems - Open systems interconnection - Basic reference model part 2: Security architecture*;

[**3**] ISO 7498-3: 1989, *Information processing systems - Open systems interconnection - Basic reference model part 3: Naming and addressing*;

[**4**] ISO 8613: 1989, *Information processing - Text and office systems - Office document architecture (ODA) and interchange format*;

[**5**] ISO 8649: 1987, *Information processing systems - Open systems interconnection - Service definition for association control service element*;

[**6**] ISO 8824: 1990, *Information processing Systems - Open systems interconnection - Specification of abstract syntax notation one (ASN.1)*;

2.2 ISO/IEC

[**1**] ISO/IEC 8879: 1986, *Information processing - Text and office systems - Standard generalized markup language (SGML)*;

[**2**] ISO/IEC 9066-1: 1989, *Information processing systems - Text communication - Reliable transfer - Part 1: Model and service definition*;

[**3**] ISO/IEC 9066-2: 1989, *Information processing systems - Text communication - Reliable transfer - Part 2: Protocol specification*;

[**4**] ISO/IEC 9069: 1988, *Information processing - SGML support facilities - SGML document interchange format (SDIF)*;

[**5**] ISO/IEC 9072-1:1989, *Information processing systems - Text communication - Remote operations - Part 1: Model, notation and service definition*;

[**6**] ISO/IEC 9072-2:1989, *Information processing systems - Text communication - Remote operations - Part 2: Protocol specification*;

[**7**] ISO/IEC 9541:199x, *Information processing - Font and character information interchange*;

[**8**] ISO/IEC 9594-2:1989, *Information processing systems - Open systems interconnection - The directory - Models*;

[**9**] ISO/IEC 9594-3:1989, *Information processing systems - Open systems interconnection - The directory - Abstract service definition*;

[**10**] ISO 9735: 1988, *Electronic data interchange for administration, commerce and transport (EDIFACT) - Application level syntax rules*;

[**11**] ISO/IEC 10021:1990, *Information processing systems - Text communication - MOTIF*;

[**12**] ISO/IEC 10031:1991, *Information technology - Text and office systems - Distributed-office-applications Model (DOAM)*;

[**13**] ISO/IEC DIS 10166, *Information technology - Text and office systems - Document filing and retrieval (DFR)*;

[**14**] ISO/IEC CD 10175, *Information technology - Text and office systems - Document printing application (DPA)*;

[**15**] ISO/IEC CD 10179, *Information Processing - Text and office systems - Document style semantics and specification language (DSSSL)*;

[**16**] ISO/IEC CD 10180, *Information Processing - Text and office systems - Standard page description language (SPDL)*;

[**17**] ISO/IEC TR xxxxx, *Information technology - Text and office systems - Operational model for text description and processing languages*;

[**18**] ISO/IEC JTC1/SC18, *Framework for future extensions to ODA*;

[**19**] ISO/IEC yyyyy, *Information technology - Text and office systems - Font services*;

[**20**] ISO/IEC TRTOSM-1, *Information technology - Text and office systems - Text and office systems reference model - Part 1:Basic reference model*;

[**21**] ISO/IEC TR9573:1988, *Information processing - SGML support facilities - Techniques for using SGML*;

[**22**] ISO/IEC 8824:1990, *Information processing - Open Systems Interconnection - Specification of abstract syntax notation one (ASN.1)*;

[**23**] ISO/IEC 10744: *Information processing - Hypermedia/time-based structuring language (HyTime)*;

[**24**] ISO/IEC 8613:1989, *Information Processing - Text and office systems - Office document architecture (ODA) and interchange format*;

[**25**] ISO/IEC 10918: *Information technology - Digital compression and coding of continuous tone still images (JPEG)*;

[**26**] ISO/IEC 11172: *Information technology - Coding of moving pictures and associated audio for digital storage media (MPEG)*;

[**27**] ISO/IEC JTC1/SC29, *Information technology - Coded representation of N186: 1992-06-9 Multimedia and hypermedia information objects: Part 1: Base notation (WD-56-MHEG)*;

[**28**] ISO/IEC 11544, *Information technology - Digital compression and coding of bi-level images (JBIG)*;

[**29**] ISO/IEC IS 9592, *Programmer's hierarchical interactive graphics system parts 1-4: (PHIGS)*;

[**30**] ISO/IEC IS 11072:1992, *Information technology - Computer graphics reference model*;

[**31**] ISO/IEC IS 7942, *Information technology - Graphical kernel system (GKS)*.

2.3 Other References

ODP reference
SQL1,2,3 reference
CGM reference
POSIX reference
Open System Environment reference
ISDN reference
CGI reference
(Appropriate JTAG reference)
(Other standards activities references)

3 Definitions and terminology

3.1 Action

A collection of statements that when executed causes state change.

3.2 Anchor

An object or collection of objects in a composite object that is referenced as an end of a link.

3.3 Atomic object

A representation of an independent piece of information that can be manipulated as a whole by applications and services and interchanged as one unit.

3.4 Bi-directional link

A connection between objects such that each anchor is both an originating and destination anchor. The link is triggered by a condition in one of the objects and calls for an action determined by a condition on the other object. Additional conditions and actions may be associated with the link independent of the anchors. Contrast with unidirectional link.

3.5 Bit combination

An ordered collection of bits (for example, a byte is a combination of 7 or 8 bits). A bit combination represents a character in character data, but can represent numeric or other values in non-character data.

3.6 Class

Any information category that has a specific and homogeneous "template," i.e. characteristics and behavior, as relevant to the contained information and to its functions.

3.7 Coded representation

A binary representation of the structure and the data within an object.

3.8 Component object

Any object embedded in a composite object.

3.9 Composite object

An object which contains information defining inter- and intra- object relations in time and space.

3.10 Computer mediated device

3.11 Content

The information conveyed by an object, other than the inter-object relations, and that is intended for human perception.

3.12 Content object

Encoded data from only one representation medium.

3.13 Document

The combination of a script and the content portion set the script is capable of accessing.

3.14 Document set

The set of content objects and structure, including scripts, that constitute an information model instance.

3.15 Event

The signal of a value change.

3.16 Final form

The property of not requiring a value or structure change for presentation.

> **NOTE -** The phrase "final form" is to be used as an adjective. The phrase must be used with a noun which provides context. An example is "final form document."

3.17 Framework

A suite of specifications to support the requirements of a specific domain.

3.18 Hypermedia

The ability to access monomedia and multimedia information with links.

> **NOTE -** The term hypermedia is an adjective. The term must be used with a noun which provides context.

3.19 Hypermedia/Time-based Structuring Language: HyTime

A standardized hypermedia structuring language for representing hypertext linking, temporal and spatial event scheduling, and synchronization. HyTime provides basic identification and addressing mechanisms and is independent of object data content notations, link types, processing and presentation functions and other application semantics. Links can be to documents that conform to HyTime and to those that do not, regardless of whether those documents can be modified. The full HyTime function supports "integrated open hypermedia" (IOH) - the "bibliographic model" of referencing allows links to anything, anywhere, at any time - but systems need support only the subset that is within

their present capabilities.

3.20 Hyperobject

A composite object containing links.

3.21 Hypertext

The ability to access text information with links.

3.22 Interchange medium

The type of means to interchange data; it can be either a storage medium, a transmission medium, or a combination.

3.23 Isochronous

Continuously and smoothly in synchronism with time.

3.24 Link

A connection among or within objects.

3.25 Medium (plural media)

A means by which information is perceived, expressed. stored or transmitted.

> **NOTE -** This is a broad (weak) definition. The term is to be avoided in its stand-alone form. To be unambiguous, the term should only be used in expressions such as: perception medium, representation medium, presentation medium, storage medium, and transmission medium.

3.26 Model

A generally accepted representation of a particular domain that allows people who are interested in that domain to agree on definitions and build a fundamental understanding within the scope of the domain.

3.27 Monomedia

The property of handling one type of content intended for presentation to a user.

> **NOTE -** The term monomedia is an adjective. The term must be used with a noun which provides the context.

3.28 Monomedia object

An object consisting of one type of content intended for presentation to a user.

3.29 Multimedia

The property of simultaneously handling various different types of related temporally and logically dependent content intended for presentation to a user.

> **NOTE -** The term multimedia is an adjective. The term must be used with a noun which provides the context. The term when used with regard to JTC1 Standards refers to digitally encoded data that is computer controlled.

3.30 Multimedia object

A composite object consisting of various different types of related temporal and logical content intended for presentation to a user.

> **NOTE -** Various types of related temporal and logical content includes contents such as audio, video, raster images, animation, and geometric images.

3.31 Multimedia representation

A composite object containing objects of at least two different representation media.

3.32 Object

A container for content and structure.

3.33 Object class

Any category of objects that have specific and homogeneous "template," i.e. characteristics and behavior, as relevant to the contained information an to its functions.

> **NOTE -** Examples of specific behaviors are "set volume control" for audio, "change position" for a text or a picture, "play backwards" for video, "zoom" for graphics . . .

3.34 Object instance

A given object in an object class is referred to as an "instance" of that class.

3.35 Perception medium

The nature of the information as perceived by the user.

> **NOTE -** Examples of auditory perception: speech, noise, and music. Examples of visual perception: text, drawings, and animation.

3.36 Presentation

A state which is ready for human perception.

3.37 Presentation medium

The type of physical mean which is used to reproduce information to the user (output device) or to acquire information from the user (input device). The presentation medium is in the platform's external environment.

3.38 Process

3.39 Projection

3.40 Rendition

3.41 Representation

A description of information structure and contents.

3.42 Representation medium

The type of the interchanged data, which defines the nature of the information as described by its coded form.

NOTE - Examples of representation media:

Nature of Information	Possible coded forms
characters or text	telex. ASCII, EBCDIC . . .
graphics	CEPT, NAPLPS or CAPTAIN videotext, CGM, . . .
audio	CCITT G711, MIDI, MPEG audio standard, . . .
still picture	FAX Group 3, JPEG standard, . . .
audiovisual sequence	CCIR REC 601 + associated audio, MPEG standard, . . .

NOTE - The representation medium is defined independently of the direction of interchange (i.e. to or from the user, or between equipment). Each representation medium may be used for input or output. For example, character-type representation may be used both for text display and for text input from a keyboard; graphics-type representation may be used both for graphic display and for graphic input (location) from a mouse. Audio-type or picture-type representations may be used both for reproduction and for capture.

3.43 Representation of an object

A description of the object structure and its contents.

3.44 Script

A specification of how objects are to be presented to a user and how input from the user is to be handled.

3.45 Scriptware (script software)

Software that defines a script.

3.46 Storage medium

The type of physical means to store data.

NOTE - Examples are electronic memory, floppy disk, hard disk, optical disk, magnetic tape . . .

3.47 Structure

A description of how information is organized.

3.48 Transmission medium

The type of physical means to transmit data.

NOTE - Examples are twisted pairs, coaxial cable, optical fibers, radio waves, . . .

3.49 Unidirectional link

A connection between objects such that the link is triggered by a condition in the originating anchor, and calls for an action determined by a condition in the destination anchor. Additional conditions and actions may be associated with link independent of the anchors. Contrast with bi-directional link.

3.50 View

A presentation of a web and the anchors that the web connects.

3.51 Web

A set of links that are used together.

4 Symbols and abbreviations

API: Application programming interface
CASE: Computer aided software engineering
CGI: Computer Graphics Interface
CGM: Computer Graphics Metafile
CGRM: Computer Graphics Reference Model
DBMS: Database management system
GKS: Graphical Kernel System
HM: Hypermedia
HyTime: Hypermedia/Time-based Structuring Language
IGES: Initial Graphic Exchange Specification
IRDS: Information Resource Dictionary System
ISO: International Organization for Standardization
MHMF: Multimedia and Hypermedia: Model and Framework, i.e. this technical report
MHM: Multimedia and Hypermedia
MHRM: Multimedia and Hypermedia: Reference Model
M: Multimedia
ODA/ODIF: Open Document Architecture/Open Document Interchange Format
ODL: Open Document Language
OSE: Open System Environment
OSI: Open System Interconnection
OSI-RM: Open System Interconnect Reference Model
PDES: Product Data Exchange using STEP
PHIGS: Programmer's Hierarchical Interactive Graphics System
POSIX: Portable Operating System Interface for Computer Environments
RDA: Remote Database Access
SGML: Standard Generalized Markup Language
SQL: Database Language SQL
STEP: Standard for the Exchange of Product Model Data
TFA: Transparent File Access

5 Requirements

This technical report describes a multimedia and hypermedia model and framework for open systems and other environments. The model and framework must:

— cover the totality of multimedia and hypermedia work;

— describe concepts and include definitions;

— reflect functions that are required by multimedia and hypermedia applications and systems;

— identify structural aspects, e.g., interfaces, protocols, formats, etc.;

— identify barriers to the use of multimedia and hypermedia that standards will help overcome;

— identify existing (including those under development) multimedia and hypermedia standards and the relationship among them;

— clarify the relationship among multimedia, hypermedia and conventional monomedia (e.g., audio, character, graphics) standards; and

— identify JTC1 standards that need to be defined and where those standards could be developed.

6 Open Systems Environment Reference Model

The MHMF already has a solid basis in the form of The Open Systems Environment Reference Model (OSE-RM). The primary reason for adopting the OSE-RM is based on the fact that it has presently attained a wide consensus with the formalization of the model as an IEEE standards activity that has been voted and adopted in the IEEE Computer Society Standards Working Group P1003.1. The OSE-RM is intended to serve as a baseline descriptive model of the totality of the MHMF Systems. Furthermore, consensus was reached in RG01 (formerly WG1/SWG01) that the OSE-RM as currently exists provides the structure for describing all aspects of multimedia and hypermedia systems.

The conceptual overview description for the Multimedia and Hypermedia Model and Framework is composed of the following parts:

- Application Software
- Application Programming Interface
- Application Platform
- External Environment Interface
- External Environment

Application Software is defined as software that is specific to an application in the multimedia and/or hypermedia domain and is composed of programs, data, and documentation.

Application Programming Interface Services (API) - The Application Programming Interface (API) is defined as the interface between the application software and the application platform across which all services are provided. The API specifies a complete interface between the application software and the underlying application platform, and is thought to be divided into the following parts:

- System Services API
- Communications Services API
- Information Services API
- Human/Computer Interaction Services API

The last three APIs listed are required to provide the application software with access to services associated with each of the external environment entities. The first API is required to provide access to services associated with the application platform internal resources, identified as the System Services API.

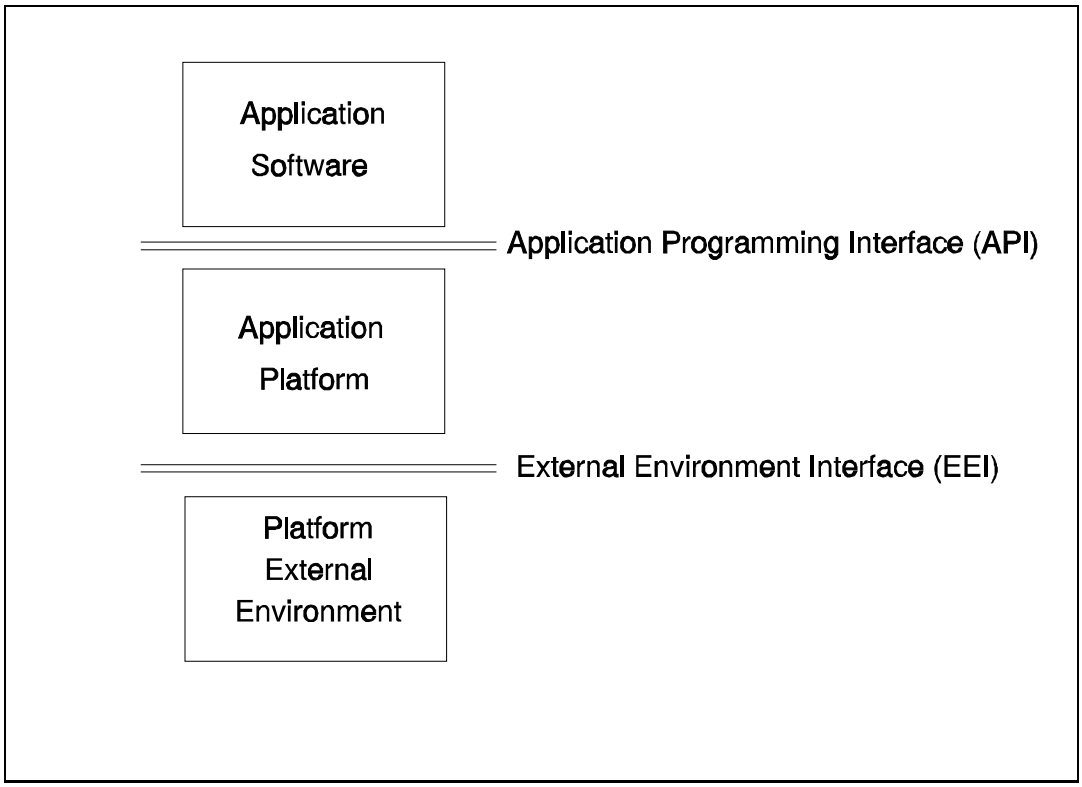


Figure 2: A Multimedia base model

The Platform is defined as a set of resources that support the service components on which an application or application software will run. It provides services at its interfaces that, as much as possible, make the implementation-specific characteristics of the platform transparent to the application software.

The External Environment contains the external entities with which the application platform exchanges information. These entities are classified into the general categories of Human Users, Information Interchange Entities, and Communication entities.

The External Environment Interface (EEI) is defined as the interface between the Application Platform and the External Environment across which information is exchanged. It is defined primarily in support of system and application software interoperability. User and data portability are directly provided by the EEI, but application software portability also is indirectly supported by reference to common concepts linking specifications at both the API and EEI. The EEI can be thought to be composed of the following set of services:

- Human/Computer Interaction Services Interface
- Information Services Interface
- Communication Services Interface

The Human/Computer Interaction EEI is the boundary across which physical interaction between the human being and the application platform takes place. Examples of this type of interface include CRT displays, keyboards, mice, and audio input/output devices.

The Information Services EEI defines a boundary across which external, persistent storage service is provided, where only the format and syntax is required to be specified for data portability and interoperability.

The Communication Services EEI provides access to services for interaction between internal application software and application platform external entities such as application software entities on other application platforms, external data transport facilities, and devices.

The multimedia and hypermedia reference model (MHMF) provides a context for understanding how the disparate technologies required as parts of a multimedia and hypermedia model interrelate. The MHRM reference model also provides a mechanism for identifying the key multimedia and hypermedia issues.

The MHRM views Multimedia HyperMedia, MHM, in terms of information models and services. An application processes an instance of the information model using services. The information models describe an MHM document set in terms of content and structure. The content is a set of atomic objects and the structure contains whatever other information is required by the application to process an instance of the document.

The distinction between content and structure within the MHRM does not imply that this distinction is necessarily visible in a specific application nor that there even exists any or all kinds of content or structure. For a given application and platform parts of the structure may be defined implicitly. A specific service or application may treat an object as though it were atomic if it does not process the object.

6.1 Open systems environment reference model Issues

The Open Systems Environment Reference Model (OSE-RM) allows consideration of OSE issues from five fundamental, integrated perspectives:

a) **Human/computer interaction issues** are associated with how applications within an OSE are delivered to the user, and the definition of a consistent look-and-feel for the dialogue between the human user and the application platform. User perspective embraces all aspects of human/computer interaction (e.g., window style guide, character representation, internationalization, commands, and input devices).

b) **Platform internal service issues** relate to how services provided by the application platform are delivered to application programs.

c) **Information interchange issues** deal with formats and related attributes required to support information interchange among application programs.

d) **Communication issues** focus on the functions required to handle a wide range of information interchange needs through basic network services and associated transfer syntax.

NOTE - Communication issues include distributed processing issues.

e) **Management issues** are pervasive across the OSE. These are the fundamental issues of OSE management (e.g., operational control, maintenance, service quality, security, etc.), and are supported by application software as the tools of the management process.The reference model described here has achieved substantial international consensus and recognition and is known as the OSE reference model.

There are two classes of interfaces, (see Figure 3) in the OSE reference model as described in the following paragraphs.

a) **Application Program Interface (API)** — The API is the interface, or set of function and subroutine calls, between the application software and the application platform. The primary function of the API in the model is to support portability of application software. An API is categorized according to the types of service accessible via that API. There are four types of API services:

1) Human/Computer Interaction Services

2) Information Interchange Services

3) Communication Services

4) Internal System Services

b) **External Environment Interface (EEI)** — The EEI is the interface which supports information transfer between the application platform and the external environment. Consisting chiefly of protocols and supporting data formats, it supports interoperability. An EEI is categorized according to the type of information transfer services provided. There are three types of information transfer services. These are transfer services to and from:

1) Human users

2) External data stores

3) Other application platforms

6.2 Information models.

6.2.1 Content

Examples of content are monomedia objects following a descriptive content notation, such as text, graphics, audio, video, etc., and other atomic objects follow content notations known to an application or service, such as procedural information, application specific structures, etc.

6.2.2 Structure

The structure contains any information that the services or the application itself may require in order to understand an instance of the information model.

Examples of structure are:

— declarative relationships between portions of content;

— descriptive attributes of portions of content;

— links between portions of content;

— layout information required to define the final form;

— ownership and access control information required by intellectual property control and security services;

— other logical information to aid the processing of the document.

6.3 Multimedia and hypermedia service areas

Multimedia and hypermedia service areas define the categories of functionality necessary for multimedia and hypermedia applications. Each of the service areas addresses specific components around which interface standards have been or may be defined.

A service area may be thought of as a module of functionality that provides services to applications and/or other service areas and may require functionality provided by other service area. Service areas may be further broken up into sub-service areas. This of course means that there is a complex web interconnecting the service areas. The service area component is defining a way of thinking about the functionality, not a way of implementing functionality or standardizing functionality.

The service areas are:

— user interface services;

— object management services;

— information production services;

— application production services;

— intellectual property management services;

— interchange services;

— procedural language processing services; and

— other services.

6.3.1 User interface services

User Interface Services include the two areas of user interaction (input) and display (output). The display services include those services transform or realize the content as visual feedback to the user, including text rendering, graphic rendering, animation, etc. Input devices include keyboards, microphones, mice, etc. The user interface services include the ability to attach/detach, control and mediate a variety of different devices that either supply input to the application or represent output from the application. The devices may process different media such as an analog magnetic tape media for sound, a video disc, or a television signal.

The user interface is required to display isochronous data.

6.3.2 Object identification services

Object Identification Services include those services for the recovery of location and descriptive information required to access an object. These services are required where objects are not accessible directly because of their definition through queries or other indirect references.

Parsing and entity management are some examples of services that are used to interpret an instance corresponding to the information model.

6.3.3 Object management services

Central to most systems is the management of objects that can be defined independent of the processes that create or use it, maintained indefinitely, and shared among many processes.

Object Management Services include those services for the physical act of storing, accessing and retrieving content objects.

6.3.4 Information production services

Producing a multimedia information product includes activities such as data collection, generation, editing, directing, and mixing.

6.3.5 Application production services

Professional system developers require tools appropriate to the development and maintenance of applications. These capabilities are provided by applications management services which can include the following:

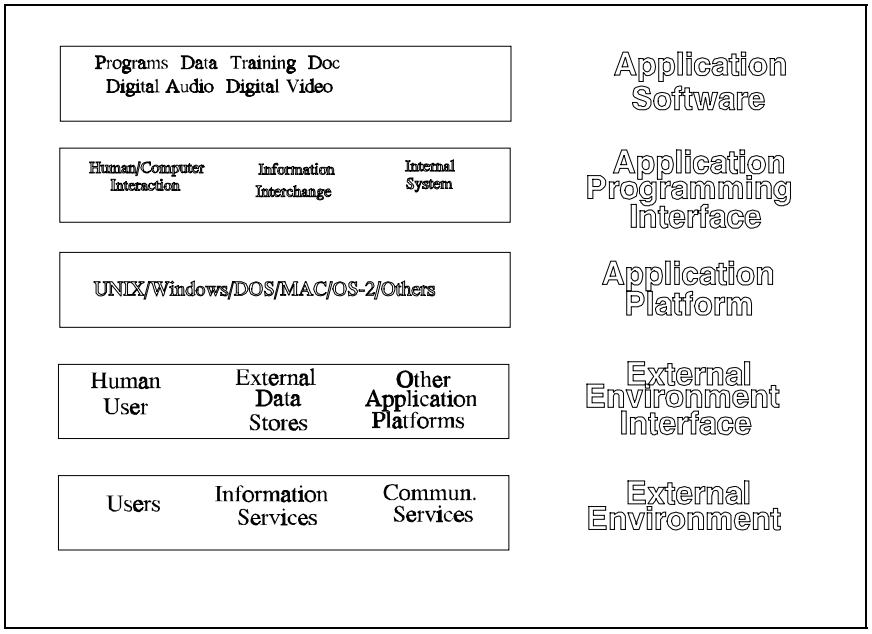


Figure 3 - OSE Reference Model (Basis for the MHMF)

— languages and language bindings;

— data formats and data format bindings;

— automated development and maintenance of applications; and

— audit trail services.

6.3.6 Intellectual property management services

Intellectual property management includes:

— security against unauthorized copying of protected (copyrighted) material

— tracking access/use of protected (copyrighted) material, so the authors can be correctly compensated.

6.3.7 Interchange services

Interchange services is the translation to an interchangeable notation.

6.3.8 Procedural language processing services

Procedural language processing services provide for the interpretation or execution of procedural information that may be included in an instance of an information model.

6.3.9 Other services

Other services is for functionality not covered in other parts of the model.

6.3.10 Open systems environment model entities

Figure 2 expands figure 1 to illustrate the component elements in the (1) application software, (2) application platform, and (3) platform external environment. These are not layers of functionality as described in other types of reference models, such as in the Open Systems Interconnection Reference Model (OSI RM). In effect, the elements are more closely akin to system components. The three classes of OSE reference model entities are described in the following:

a) **Application Software** — Most users consider application software to be the computing element supporting their particular business needs (e.g., the payroll, accounting, spreadsheets, and other systems that provide information to the users in the course of conducting business). The application software includes data, documentation, and training, as well as programs.

b) **Application Platform** — The application platform is composed of the collection of hardware and software components that provide the system services used by application programs. Application platforms facilitate portable application programs through services accessed by application programming interfaces (API) that make the specific characteristics of the platform transparent to the application. The application platform components include the hardware and the software that interface directly with the hardware (i.e., the hardware drivers) in supporting the application software.

c) **Platform External Environment** — The platform external environment consists of those system elements which are external to the application software and the application platform (e.g., systems and services executing on, or provided by, other platforms or peripheral devices).

Human users are a part of the platform external environment.

7 A JTC1 Multimedia and Hypermedia Framework

7.1 The Distributed Hyper-Document Model

The distributed hyper-document, as a whole, may be modelled as a hyper-linked structure of Content Objects (COs). This view is illustrated in Figure 1. Each CO is an information conveying element of the hyper-document, typically such information being presented to and perceived/interpreted by the human operator "in" the User Interface (UI) of the application. A CO may be a complete "document" in the usual word processing sense as shown in the document containers in Figure 1. Within such a document there may be "contextual links" that provide hierarchical and sequential structuring of the COs that make up the document; if a hyperlink to such a document is activated then there may be viewing/imaging steps automatically invoked by the contextual links, such as moving from page to page in a formatted (final form) document.

Figure 1: Distributed Hyper-document as Hyper-linked COs

NOTE - These are internal/contextual links known to a "native" processor for that particular format.

7.2 Characteristics of Content Objects (COs)

Multiple CO formats (data structures) are allowed to "interface" (be linked) to the to the hyper-document. NO "winning format" is chosen and all are allowed to "play" the hyper-document game if they follow the open rules of data interchange/access and conform to the semantics of the standardized methods defined for all content objects. Each unique content object type must provide its own methods for formatting (content layout), viewing, editing, placing and deleting link ends.

COs may be linked to as a whole or at addressable points within the substructure of the content (e.g., at a particular word within a paragraph). The requirement for "location or addressing models" for each content object format is the responsibility of the defining standard (and/or reference implementation) for that format. As a practical matter the "open protocol" must have a way of identifying the points to be linked to (link ends or anchors). Requirements for hyper-linking are thoroughly explored in the HyTime standard; and location models have been defined in the SGML/HyTime and HyperODA standardized environments.

COs are categorized as either Processable (PCO) or Formatted (FCO). The PCO is a source/revisable/computable form. It is a format supporting further revision and evaluation in authoring/editing and dynamic information processing environments (such as content based retrieval, linguistic analysis, use of spreadsheets, database accesses, modelling, etc.). The FCO is ready for presentation/display/imaging. Typically it is derived from the PCO for a particular imaging device (viewer/player) by formatting or otherwise transforming the PCO (e.g., plotting a graph from tabular numeric data, accessing records from a database by evaluating an SQL query and formatting them for viewing, etc.).

7.3 Composite CO Documents

Documents may be treated as Composite COs (CCOs in Figure 1). A processable (logical) document may be defined as a "logically structured set of PCOs. "To obtain a human perceivable "view" of the processable document requires a formatter or layout process to convert PCOs to FCOs ("content layout") and place them in an appropriate "layout structure" such as a set of pages or frames for display or printing. The result of such formatting (layout) process is a "final form" document that may be thought of as a "layout structured set of FCOs." Frequently this final form document is just a an sequence of FCOs that are "contextually linked" by order/placement. The CCO is effectively a container for COs and in an OOP implementation might very well be implemented as a container class. Note that revisable document formats do contain information that is never directly "viewed"by the human as a CO (e.g., management attributes, presentation styles).

BOTH processable and final form documents can be treated as composite COs in the hyper-document; it is thus possible for both the PCO and the FCO of the same object to co-exist in the same hyper-document (e.g., a PDF or MHEG final form document derived from an SGML/HyTime source) and be linked to avoid time consuming derivations such as formatting or other compute-intensive content transforms (e.g. changing color model, decompressing an image format). Figure 2 shows a hyper-document processing environment with three domains.

Figure 2: Distributed, Heterogeneous, Hyper/Multimedia Processing

NOTES

1 On "the open bus" all information interchanges are in an "open standard" format. This must be a self describing or self typing format; although, for efficiency, such "meta data" may be referred to if in an appropriate registry or sent only once. In this way a "context" for bulk data transfer could be setup, with the "data/content model" sent only once followed by repeated units of data of that type.

2 Each domain has its own method of object identification, object access, transfer, linking and execution/presentation control; for open interchange in a distributed application over many domains an application-wide object identification scheme must be adopted and a means of mapping this to the domain-specific scheme must be provided.

3 Each player may have its own delivery format, and its own delivery medium. Although there will be "standards" established for the players (i.e. "presentation engines") these are of a wide variety and changing rapidly in the marketplace (no clear winners!?).

4 There is critical need for content object standardization; although many defacto and formal standards exist there is still work to be done in relating these formats (so that lossless conversions are possible in open interchange) and adding standardized semantics.

5 Client/Server may be thought of as Consumer/Producer of COs and both may be "bound" in the same process on the same processor. To allow for the possibility of distribution, and the likelihood that one Client may wish to consume COs from many Servers on different processors the distinction is maintained throughout the model;

Figure 3 below provides more detail about any one of the domains that may participate in hyper-document processing. Note that there is a private "bus" for internal communications between the processes within the domain; but that whenever a request is made to the "outside" the request must go

out in a public protocol format and any response (status or data) comes back in the open standardized format.

Figure 3: Detail for One Domain from Figure 2

7.4 Migration/Version Update Requirement

It is often required that collections of content objects be interchanged or "migrated" between domains or even different versions of the "same" domain. An open specification that provides complete description of data structure and data types is required to achieve such open interchange and/or access. SGML/HyTime/SDIF is a particularly good candidate for revisable interchange, providing facilities for logical structuring, content description, specification of hyper-links, and with SMSL as the basis of overall flow control and invocation of methods. This format is also quite capable as the archive format for preserving the investment in the application design and the content and structuring of that content.

Note that many other content description standards are required since SGML/HyTime does not describe the substructure of non-SGML content elements (like PDF or CGM). There are other candidates for this "open data interchange" function -- for example, Apple Bento (part of the OpenDoc efforts). GDID, an ISO standard under development, is also intended to provide this data description and conveyance function. MIME has immediate application as a means for packaging a set of components into a single E-mail message.

7.5 Dynamic Linking Requirement

In some applications, dynamic linking to objects is required, but this imposes only the open interchange of specifications that allow proprietary implementations (servers) to "go inside" objects and perform the requested linking and access at run-time. This also implies a public method of object identification and location in the distributed environment.

Descriptions of the hyper-links should be maintained externally to the content objects themselves; and these preferably would include the timing and other control specifications (e.g., parameters for a content transform to be performed). These "content transform and timing specifications" may be considered as being invoked (carrying out processes, such as formatting, imaging, etc.) when the links are activated. Keeping "link sets" external means that the same content objects may be viewed in many different ways without changing the content objects simply by providing different external link sets, which also facilitates dynamic linking.

7.6 Subsetting or Specializing the Reference Model

The Multimedia/Hypermedia Reference Model should allow subsetting or specialization to describe applications where certain decisions have been fixed before runtime; as compilers and linkers turn source code into executable object code. Thus, for example, an interactive, stand-alone, CD-ROM application described in terms of its content objects and their relationships could be compiled/converted from its open source description (content objects, logical document structure, hyperlink structure, content transform and timing specs, etc.) into the proprietary on-disk structure (delivery format) for a particular player. The same revisable sources could support many different players (runtime environments) producing "equivalent" runtime behavior. Source portability would thus be achieved and design investment protected.

7.7 Conclusions/Recommendations for Future Work on the Reference Model

The Model as detailed above covers the following application scenarios:

1. Dynamic, distributed hypertext with many different servers providing content objects to a client process by a standard, open interchange/access format on the open interchange bus.

2. Stand-alone (no network) application. Could have been "compiled" from open sources, proprietary sources, or a mix. The execution delivery format is "proprietary."

3. Mix of 2 in 1. Some linking is dynamic, interactive; some formatting is done at runtime from revisable sources; the static on-CD links are fixed and cannot be changed. To allow the "stand-alone" application to play in the much wider distributed open environment requires a standardized "escape" mechanism to that environment, perhaps via a scripting language.

4. Complete hyper-document source migrations are covered; for example, to take an application from "proprietary domain #1" to "proprietary domain #2" by the OHMMIA bus. All COs are converted to "open" (on export) and then from "open" to proprietary on the receiving system. Source portability is achieved, execution efficiency is not compromised, since the "migration" of the application is itself not a time critical step.

5. Open, completely standardized hypertext solutions are covered. For example, where all documents/content objects are described by SGML/HyTime and content object standards, and control/presentation processes are governed by SMSL (scripting) and other standards. HyperODA is another domain that will be completely based on ISO/ITU-T standards. Two standardized domains such as SGML/HyTime and HyperODA could interrelate by using a consistent "lower level" standard such as MHEG as the final form for content objects.

Annex A (informative)

Relationships to other JTC1 reference models

A.1 Relationship to the computer graphics reference model

This subclause relates the multimedia and hypermedia model to the Computer Graphics Reference Model (CGRM). The computer graphics reference model describes the conceptual framework for computer graphics. Computer graphics is the creation of, manipulation of, and interaction with pictorial representations of objects and data using computers.

The main purpose of the CGRM is to define concepts that shall be used to develop computer graphics and standards. Additional purposes are to explain relations between ISO/IEC JTC1/SC24 Computer Graphics standards and to provide a forum whereby areas outside computer graphics can identify their relationships to computer graphics.

The overall structure of the computer graphics reference model is illustrated in Figure 3. Communication between a computer graphics system and the "outside world" is described in the CGRM in terms of the operator interface, application interface, data capture metafile interface, and the audit trail metafile interface.
The application interface in the CGRM performs a similar function to the application program interface in the MHMF model. The CGRM operator interface performs a similar function to the MHMF External Environment Interface, EEI. The CGRM audit trail metafile interface and data capture metafile interface would have both an API and EEI associated with the functionality in MHMF model. The API would allow the application to specify functionality such as record, playback, import, export, composition, collection store, token store aggregation store and environment state. The EEI would allow the associated data to be stored, retrieved, modified and deleted.

The computer graphics environments in the CGRM define functionality of services found in the MHMF model modelling services and application management service areas (See clauses 6.2.6 and 6.2.7). The CGRM operator corresponds to the MHMF model external environment. *Application* has the same meaning in both models.

A.2 Relationship to the open distributed processing reference model

A.2.1 Overview

ODP is the ISO/IEC standard-in-progress for Open Distributed Processing. The Reference Model standard defines the technical basis for ODP standards, using existing reference models and standards, such as OSI for communications and existing data management standards, and defining new areas of standardization.

ODP will define terms, concepts and the syntaxes and semantics for a set of prescriptive languages for the specification of distributed systems and applications. ODP is dependent on the OSI work to specify communications, but will also be concerned with the internal structure of real systems.

Distribution of information systems is a necessary consequence of real-world requirements, such as the need to replicate data and processes against failure, incremental growth, distribution to provide timely access, and potential parallel activity.

Openness allows for the integration of heterogeneous components. This may involve equipment from different vendors, running different operating systems, with applications written in different programming languages, running different database engines, and with different security authority levels.

An ODP system is an information processing system that conforms to ODP standards. As such, components of an ODP system may be heterogeneous, are capable of operating concurrently, can be physically and/or logically distributed, are portable, both statically and dynamically and are capable of working with one another. These systems are modular to enable incremental growth, and distributed management and security provisions.

The following goals for standardization are to be met by ODP standards:

- portability of applications in an ODP environment;
- exchange between ODP systems;
- distribution transparency in ODP systems.

Portability of distributed applications provides applications with the potential to migrate within a system without disrupting the service they provide, or the services they use. Migration includes both the static cases of reconfiguration and the reloading of a system and dynamic reconfiguration. Interoperability provides for systems exchanging information in a meaningful way, and using one another functionality successfully. Distribution transparency is the property of hiding the properties of distribution from applications.

There are special processing needs in an ODP system. There must be specialized components of the system that deal with the heterogeneous and distributed information processing structures, storage, user access, communications, identification, management, and security.

A.2.2 Objective of ODP

The objective of ODP is to enable distributed system components to exchange information seamlessly, despite heterogeneity in equipment, operating systems, networks, languages, data base models, or management authorities. An ODP system must supply the mechanisms which mask underlying heterogeneity from users and applications. These mechanisms will address a set of fundamental transparency properties, including:

- Access transparency, which masks differences in data (e.g. multimedia/hypermedia representation and invocation mechanisms between heterogeneous computer architectures, programming models and networking protocols.

- Location transparency, which masks changes in configuration of application components, and enables the transfer of configuration-independent interface references between components.

- Migration transparency, which masks dynamic relocation of components from both the components themselves and their clients.

- Concurrency transparency, which masks scheduling of invocations of operations that act on shared state.

- Federation transparency, which masks interworking boundaries between separate administration domains and heterogeneous technology domains.

- Likeness transparency, which masks the automated transfer of components from active to passive state or vice versa.

- Resource transparency, which masks variations in the ability of the local ODP infrastructure to proved the resources for an application component to engage in interactions with other remote components.

- Failure transparency, which masks recovery of failed components, thereby enhancing fault tolerance.

- Replication transparency, which masks replication of components, thereby enhancing performance and availability.

In order to achieve these ambitious goals, the RM-ODP must accomplish two things:

1. RM-ODP must prescribe an integrated set of functions that can provide the required transparencies. We refer to the realized set of software components that provide these transparencies as the ODP infrastructure. The Recommendations and International Standards that will be developed under the umbrella of the RM-ODP will standardize the ODP infrastructure components. Distributed application components will interoperate through the ODP infrastructure. The ODP infrastructure is the platform that will make multimedia/hypermedia network computing a reality.

RM-ODP must provide a technique for the specification of interfaces. The ODP infrastructure will allow client application components to access a server no matter where the clients are located in the network, no matter what programming languages were used for the clients or the server, no matter what local operating systems are involved, etc. The components of a distributed system might be developed by different teams, at different times, using differing technology - yet the components must work together. It is therefore essential for developers of a client component to have a precise specification of the server's interface; the specification must be unambiguous and implementation independent. The Recommendations and International Standards for components of the ODP infrastructure must be rigorously specified by defining their interfaces.

A.2.3 Reference points and conformance points

The problem of how to define interfaces has been central to the development the RM-ODP. A related problem is the broad categorization of interfaces based on their architectural placement. One of the tasks of any reference model is to specify reference points the architecture. The RM-ODP identifies four types of reference points, any or all of which may be specified as conformance points in a particular multimedia/hypermedia standard or specification. They are:

- Perceptual reference point, at which a human-computer interface can be established For example, a perceptual reference point might be established in a graphics standard. The perceptual reference point is of relevance to the presentation of multimedia/hypermedia information/objects.

- Interworking reference point, at which a communication interface can be established between two systems. OSI standards are based on the interconnection of interworking reference points (the physical medium). Interchange formats for multimedia/hypermedia information are in relation to the interworking reference points.

- Interchange reference point, at which an interface to an external physical storage medium can be established. An interchange conformance requirement is stated in terms of the behavior (access methods and formats) of some physical medium so that information can be recorded and then physically transferred, directly or indirectly, to be used on another system. The interchange reference point is of relevance to the formats of interchangeable storage media.

- Programmatic reference point, at which a programmatic interface can be established to allow access to a function. An interface at a programmatic reference point corresponds to the common notion of an API. The programmatic reference point is of relevance to the multimedia/hypermedia reference model. The concepts for the placement and definition of APIs should be considered carefully.

A.2.4 Framework of abstractions: Viewpoints

The purpose of the RM-ODP framework of abstractions is to position services relative the placement of boundaries upon ODP. The framework of abstractions is used to partition the concerns to be addressed when describing all facets of an ODP system, so that the description task is made simpler.To deal with the complexity of an ODP system, the system is considered from different viewpoints. Each viewpoint represents a different abstraction of the original system.

- The enterprise viewpoint is concerned with business policies, management policies and human user roles with respect to the systems and the environment with which they interact. The model constructed may well describe the constraints placed on the interaction of a number of organizations. The requirements to multimedia/hypermedia systems can be identified from this viewpoint.
- The information viewpoint is concerned with information modelling, providing a consistent common view covering information resources, sinks and information flows. The information viewpoint is of relevance to the multimedia/hypermedia information models.

- The computational viewpoint is concerned with the algorithms and data structures which provide the functions of the distributed system. The computational viewpoint can be characterized as "what should be done by the system". The computational viewpoint is of relevance to the multimedia/hypermedia

service areas, specific components and interfaces.

- The engineering viewpoint is concerned with distribution mechanisms and the provision of various transparencies needed to support distribution. The engineering viewpoint can be characterized as "how should it be done by the system". Especially multimedia/hypermedia engineering objects for communication, such as for continuous media transfer, synchronization, control and interceptors (format conversion) are positioned in the engineering viewpoint language.

- The technology viewpoint is concerned with the technical artifacts - realized - and software components - from which the distributed system is built.

The Basic Reference Model of ODP will be based on precise concepts and as far as possible on the use of formal description techniques for specification of the architecture.

A.2.5 Structure of the RM-ODP and conclusion

The structure of the ODP-RM is described below:

- Part 1: Overview and Guide to Use of the Reference Model: contains a motivational overview of ODP giving scope, justification and explanation of key concepts, and an outline of the ODP architecture. The informative part contains explanatory material on how the RM-ODP is intended to be understood and applied by its users, who may include standards writers and architects of open distributed systems. It also contains an enumeration of required areas of standardization expressed in terms of the reference points for conformance identified in Part 3. This part is not normative.

- Part 2: Descriptive Model: contains definition of the concepts and analytical framework for normalized description of (arbitrary) distributed processing system. This should only be to a level of detail sufficient to support Part 3 and to establish requirement for new specification techniques. This part is normative.

- Part 3: Prescriptive Model: contains specification of the required characteristics that qualify distributed processing as open. These are the constraints to which ODP standards must conform. It uses the descriptive techniques from Part 2. This part is normative.

- Part 4: Architectural Semantics: contains a formalization of ODP modeling concepts defined in Part 2, clauses 9 and 10. The formalization is achieved by interpreting each concept in terms of the construct of the different standardized formal description techniques. This part is normative.

From this standard Reference Model, specific ODP standards, such as the trader will be derived. From those standards, products will emerge to facilitate true Open Distributed Processing.

Two types of elements are used in the model:

— entities consisting of the application software, application platform, and platform external environment; and

interfaces including the application programming interface and external environment interface.

Annex B (informative)

Multimedia and Hypermedia Detailed Standards Descriptions

B.1 Content Standards Descriptions

Abbreviation:CGM

Name of Standard:Computer Graphics Metafile

Designator:ISO 8632:1992

Originating Group:JTC1 SC24

Field of application:Exchange of computer graphics pictures

Other standards ODA (Open Document Architecture) part 8
use of this standard:SGML, via external references
NITFS (National Imagery Transmission Format Standard)

Use of this standard:The storage and transmission of Vector Image Graphics Files

Standards used by
this standard:ISO 646

Brief Description: The Computer Graphics Metafile (CGM) is the scope International Standard for the exchange of computer graphics pictures. CGM files can contain both vector graphics information (e.g., polygons, circles, ellipses, NURBS) and raster graphics information (e.g., tile array). These primitives have an associated set of attributes (e.g., line width, colour, fill style). Several colour models (i.e., RGB, CMYK, CIELAB, and CIELUV) with their associated colour calibration information are aupported. The CGM package is appropriate for storing picture information from a wide variety of applications (e.g., engineering, cartography, graphics arts, business graphics) and for exchanging that information between similar and dissimilar application (e.g., from an engineering package to a graphics arts package or to a desktop publishing package) and between different platforms (e.,g., between PCs, Macintosh, and Unix).Current amendment work will make it possible to embed application structure information within the CGM element stream using facilities similar to the tagging mechanisms of SGML and HyTime.

Abbreviation:JPEG

Standard:Digital Compression and Coding of Continuous-tone Still Images

Designator/status:IS10918:1992

Originating group:ISO/IEC JTC 1/SC2 WG10

Field of application:This standard is applicable to continuous-tone - grayscale or colour - digital still image data. It is applicable to a wide range of applications which require use of compressed images. It is not applicable to bi-level image data.

Use of This Standard: A standardized way of compressing and storing both 24 bit color and gray scale images. With this compressed formatted image files are more efficiently transmitted across networks.

Other standards using this standard: SGML, ODA, Z39.50, ISO9660, MPEG, Huffman Encoding

Brief Description: There are three elements specified in this standard:

1) An *encoder* is an embodiment of an *encoding process*. An encoder takes as input *digital source image data* and *table specifications,* and by means of a specified set of *procedures* generates as output *compressed image data.*

2) A *decoder* is an embodiment of a *decoding process*. A decoder takes as input compressed image data and table specifications, and by means of a specified set of procedures generates as output *digital reconstructed image data*.

3) The *interchange format* is a compressed image data representation which includes all table specifications used in the encoding process. The interchange format is for exchange between *application environments.*

Abbreviation: RGCA

Standard: Raster Graphics Content Architecture

Designator/status: ISO 8613-7:1988/ International Standard (1988)

Originating group:

Field of application: The purpose of this International Standard is to facilitate the interchange of documents.

Use of this standard: ISO 8613-7 applies to documents that are structured according to the architecture defined in ISO 8613-2 that include raster graphics content, consisting of a descriptive representation of pictorial information provided by an array of picture elements (pels), encoded according to facsimile or bitmap encoding.

Other standards using this standard: SGML, ODA, CCITT Group 3 & 4, EDI

PART 32 - Multimedia and Hypermedia: Model and Framework June 1994 (Working)

Brief Description: This part of ISO 8613 defines: the raster graphics content architectures that can be used in conjunction with the document architecture defined in ISO 8613-2; the internal structure of content portions that are structured according to a raster graphics content architecture; those aspects of positioning and imaging applicable to the presentation of raster graphics contents in a basic layout object; a content layout process which, together with the document layout process defined in ISO 8613-2, specifies the method for determining the dimensions of basic layout objects for raster graphics content portions; the presentation and content portion attributes applicable to raster graphics content architectures. In the context of ISO 8613, documents are considered to be items such as memoranda, letters, invoices, forms and reports, which may include pictures and tabular material. The content elements used within the documents may include graphic characters, geometric graphics elements and raster graphics elements, all potentially within one document. ISO 8613 applies to the interchange of documents by means of data communications or the exchange of storage media.

Abbreviation: MPEG 1

Standard: Video Compression for multimedia applications

Designator/status: ISO 11172 - MPEG 1 (Coding of Moving Pictures and Associated Audio for Digital Storage Media up to about 1.5 Mb/s - part 2 Video and part 3 Audio)

Originating group: ISO-IEC/JTC1/SC2/WG11

Field of application: Addresses the compression of video signals at about 1.5Mbits, MPEG Audio is addressing the compression of a digital audio signal at the rates of 64, 128, and 192 kbits/s per channel.

Use of this standard: Compression of video and audio data which is stored on mass media, i.e., hard drives, optical media such as CD-ROM and writable CD, network servers and DAT Tape. MPEG compression techniques are geared to asymmetric applications where the decompression process is extremely faster than the compression process. Such applications as electronic publishing, video games, and delivery of movies make very good use of this asymmetric technology.

Other standards using this standard: All transmission standards, document structure standards and interchange standards

Brief Description: The MPEG video compression algorithm rely on two basic techniques: block-based motion compression for the reduction of temporal redundancy and transform domain-based compression for the reduction of spacial redundancy. Motion-compensated techniques are applied with both casual (pure predictive coding) and non-casual predictors (interpolative coding). The remaining signal (prediction error) is further compressed with spacial redundancy reduction (DCT). The information relative to motion is based on 16 x 16 blocks and is transmitted together with the spacial information. The motion information is

compressed using variable-length codes to achieve maximum efficiency.

Suggested additional standards associated with "*content*":

ISO 646 - 7 bit character set
ISO 2375 - register of ISO 2022 control and character sets
ISO 6937 - diacritical formed accented characters
ISO 8859 - Fully formed accented characters
ISO 9282 - Coded Representation of Computer Graphics Images
ISO 10646 - Multi-octet character set (Unicode)
ISO 2022 - 7 bit and 8 bit code extension techniques
ISO 6429 - Coding of controls
ISO 11544 - JBIG (Progressive Bi-level Image Compression)
MPEG 2 - (Generic Coding of Moving Picture and Associated Audio Information)
ISO 13522 - MHEG (specifically content encodings) including:
- H.261 - coding of videophone/conferencing signals
- T.4 - facsimile encoding (Group 3)
- T.6 - facsimile encoding (Group 4)
- G.711 - PCM Audio
- G.721 - adaptive PCM encoding of audio
- G.722 - 7 khz audio encoding
- G.723 - ADPCM at 24 and 40 khz
- G.725 - system aspects of audio encoding
- G.728 - LD-CELP (16 kb/s audio encoding)

ISO 9613 - ODA (Open Document Architecture)
- part 6 - character content
- part 7 - raster graphics content
- part 8 - geometric graphics content

B.2 Multimedia related Structure Standards Descriptions

Abbreviation:	ISO 9660
Standard:	CD-ROM Volume/File Structure
Designator/status:	ISO 9660:1988
Originating group:	JTC1 SC24
Field of application:	CD-ROM Publishing
Use of this standard:	As a standard publishing technique in the creation of CD-ROMs and CD Recordable media.
Other standards using this standard:	CD-ROM Physical Format Structure (ISO 10149) Z39.50 (NISO) Information Interchange

Brief Description: This standard describes volume/directory/file system on CD-ROM optical storage media as well as CD Recordable media. The standard makes use of a subset of the DOS Volume/File structure.

Other standards associated with "*Structure*":

ISO 8613 - ODA and HyperODA
ISO 8879 - SGML (Standard Generalized Markup Language)
ISO 10744 - Hytime (Hypermedia Time-based Structuring Language
SC18/WG5 - GDID (Generic Data Interchange/Interface for Documents
ISO 13522 - MHEG (Coded Representation of Multimedia and Hypermedia Information Objects)
SC18/WG8 - SMSL (Standard Multimedia/Hypermedia Scripting Language) (semantics)
ISO 11172 - MPEG 1 (Coding of Moving Pictures and Associated Audio for Digital Storage Media up to about 1.5 Mb/s - part 1:system)
ISO 9281 - Identification of picture coding methods
ISO 8824/5 - ASN.1 (and its application)

B.3 Multimedia and hypermedia service areas

B.3.1 User interface services

Abbreviation: GKS

Standard: Computer Graphics - Graphical Kernel System (GKS)

Designator/status: ISO 7942:1985 and ANSI X3.124-1985/International Standard (1985) and American National Standard (1985)

Originating group: X3

Field of application: Computer Graphics, CAD/CAM,

Use of this standard: IEEE Computer Graphics and Applications. The Graphical Kernel System (GKS) is a set of basic functions for computer graphics programming usable by many graphics producing applications. Use of this standard allows graphics application programs to be easily transported between installations, aids graphics applications programmers in understanding and using graphics methods, and guides device manufacturers on useful graphics capabilities.

Other standards using this standard:

Brief Description: This American National Standard specifies a set of functions for computer

graphics programming, the Graphical Kernel System (GKS). GKS is a basic graphics system for applications that produce computer generated two dimensional pictures on line graphics or raster graphics output devices. It supports operator input and interaction by supplying basic functions for graphical input and picture segmentation. It allows storage and dynamic modification of pictures. A fundamental concept in GKS is the workstation, consisting potentially of a number of input devices and a single output device. Several workstations can be used simultaneously. The application program is allowed to adapt its behavior at a workstation to make best use of workstation capabilities. This standard includes functions for storage on and retrieval from an external graphics file. Last, but not least, the functions are organized in upward compatible levels with increasing capabilities. For certain parameters of the functions, GKS defines value ranges as being reserved for registration or future standardization. The meanings of these values will be defined using the procedures established in an International Standard under development (Procedures for registration of graphical items).

The Graphical Kernel System (GKS) provides a functional interface between an application program and a configuration of graphical input and output devices. The functional interface contains all basic functions for interactive and non-interactive graphics on a wide variety of graphics equipment. The interface is at such a level of abstraction that hardware peculiarities are shielded from the application program. As a result a simplified interface presenting uniform output primitives, and uniform input classes is obtained. A central concept both for structuring GKS and for realizing device independence is introduced, called the workstation. The facilities for picture manipulation and change are introduced via the segment facilities, the dynamic attributes and the transformations. The concept of multiple workstations allows simultaneous output to and input from various display systems. Facilities for internal and external storage are provided by special workstations together with the possibility of transferring graphical entities directly from the special workstation for internal storage to other workstations. Not every GKS implementation needs to support the full set of functions. Twelve levels are defined to meet the different requirements of graphics systems. Each GKS implementation provides at least the functions of one level. The levels are upward compatible.

Abbreviation: PHIGS

Standard: Computer Graphics - Programmer's Hierarchical Interactive Graphics System

Designator/status: ISO DIS 9592:1988 and ANSI X3.144:1988/International Standard (1988) and American National Standard (1988)

Originating group: ANSI X3

Field of application: This American National Standard specifies a set of functions for computer graphics programming, the Programmer's Hierarchical Interactive Graphics System (PHIGS). PHIGS is a graphics system for applications that produce computer generated pictures on line graphics or raster graphics output devices.

It supports operator input and interactions by supplying basic functions for graphical input and hierarchical picture definition. It allows for storage, and dynamic modification of pictures.

Use of this standard: This Standard allows graphics application programs to be easily transported between installations; aids graphics applications programmers in understanding and using graphics methods; guides device manufacturers on useful graphics capabilities; performs many functions currently performed by graphics applications; thus, off-loading the graphics application development effort. This Standard defines an application level programming interface to a hierarchical interactive and dynamic graphics system. Hence it contains functions for:

- displaying graphical primitives.
- controlling the appearance of graphical primitives with attributes.
- controlling graphical workstations.
- controlling 2D & 3D transformations and coordinate systems.
- generating, modifying, and controlling groups of primitives called structures.
- modifying the hierarchical relationship of structures.
- obtaining graphical input.
- archiving and retrieving structures and structure hierarchies.
- inquiring the capabilities and states of the graphics system.
- handling errors."

Other standards using this standard:

Brief Description: The Programmer's Hierarchical Interactive Graphics System (PHIGS) provides a functional interface between an application program and a configuration of graphical input and output devices. The functional interface contains basic functions for dynamic interactive hierarchical graphics on a wide variety of graphics equipment. The interface is at such a level of abstraction that hardware peculiarities are shielded from the application program. PHIGS defines only a language independent nucleus of a graphics system. For integration into a language, PHIGS is embedded in a language dependent layer containing the language conventions, for example, parameter and name assignment. A fundamental concept in PHIGS is the workstation, consisting of a number of input devices and a single output device. Several workstations can be used simultaneously. he application program is allowed to adapt its behavior at a workstation to make best use of workstation capabilities. A second fundamental concept is the centralized structure store, where graphical information is stored and edited. This American National Standard includes functions for storage on and retrieval from an external graphics file.

Other standards associated with "*User Interface Services*":

ISO 9636 - CGI (Computer Graphics Interface)

SC24/WG6 PREMO (Presentation environment for Multimedia Objects)
ISO 13522 - MHEG (selection and modification objects)
SC18/WG9 - Human computer interface standards and recommendations

B.3.2 Object identification services

ISO 8613 - ODA and HyperODA
ISO 8879 - SGML (entity management)
ISO 13522 - MHEG (naming and addressing techniques)
ISO 10744 - HyTime (naming and addressing techniques)
SC21 - Open distributed processing (ODP) Trader

B.3.3 Object management services

Abbreviation: IGES

Standard: Initial Graphics Exchange Specification

Designator/status: ASME/ANSI Y14.26M-1987 Digital Representation for Communication of Product Definition Data (Based on Version 3.0 of the Initial Graphics Exchange Specification published as NBSIR 86-3359.)/American National Standard (1987)

Originating group:

Field of application: CAD/CAM, Computer Graphics, Electronic Publishing

Use of this standard: IGES is used "to describe and communicate the essential engineering characteristics of physical objects as manufactured products. Such products are described in terms of their physical shape, dimensions, and information which further describes or explains the product. The processes which generate or utilize the product definition data typically include design, engineering analysis, production planning, fabrication, material handling, assembly, inspection, marketing, and field service."

Other standards using this standard:

Brief Description: This document establishes information structures to be used for the digital representation and communication of product definition data. Use of the specification established herein permits the compatible exchange of product definition data used by various (CAD/CAM) Computer Aided Design and Computer Aided Manufacturing) systems.

This specification defines a file structure format, a language format, and the representation of geometric, topological, and non-geometric product definition

data in these formats. Product definition data represented in these formats will be exchanged through a variety of physical media. The specific features and protocols for the communications media are the subject of other standards. The methodology for representing product definition data in this specification is extensible and independent of the modeling methods used. Chapter 1 is general in nature and defines the overall purpose and objectives of this specification. Chapter 2 defines the communications file structure and format. It explains the function of each of the sections of a file. The geometry data representation in Chapter 3 deals with two- and three-dimensional edge-vertex models and with simple surface representations. Chapter 4 specifies non-geometric representations, including common drafting practices, data organization methods, and data definition methods. In Chapters 3 and 4, the product is described in terms of geometric and non-geometric information, with non-geometric information being divided into annotation, definition, and organization. The geometry category consists of elements such as points, curves, and surfaces that model the product. The annotation category consists of those elements which are used to clarify or enhance the geometry, including dimensions, drafting notation, and text. The definition category provides the ability to define specific properties or characteristics of individual or collections of data entities. The organization category identifies groupings of elements from geometric, annotation, or property data which are to be evaluated and manipulated as single items.

B.3.4 Information production services

Abbreviation:	SR
Standard:	Documentation - Search and Retrieve Service Definition
Designator/status:	ISO/IS 10162
Originating group:	NISO
Field of application:	Document Search and Retrieval
Use of this standard:	The purpose of this standard is to provide a set of Application Layer communication-related services which can be used to perform bibliographic search activities in an Open Systems Interconnection (OSI) environment defined by ISO 7498.
Other standards using this standard:	ISO 10163, Z39-50, SGML, ODA
Brief Description:	The model of SR application and service utilizes a number of key concepts, namely, service user roles, a model of a database, records, record composition, result-set, result-set bounds, positional retrieval, query type, and a model of query processing.

B.3.5 Application production services

This area still needs to be developed.

B.3.6 Intellectual property management services

This area still needs to be developed.

B.3.7 Interchange services

Abbreviation:	SGML
Standard:	Standard Generalized Markup Language (SGML)
Designator/status:	ISO 8879:1986/International Standard (1986)
Originating group:	ISO/IEC/JTC1/SC18
Field of application:	
Use of this standard:	SGML is specifically designed for the world of publishing and the management and control of the information which may take form in many types of documents. SGML can be used for publishing in its broadest definition, ranging from single medium conventional publishing to multi-media data base publishing. SGML can also be used in office document processing when the benefits of human readability and interchange with publishing systems are required.
Other standards using this standard:	
Brief Description:	This International Standard specifies an abstract syntax known as the Standard Generalized Markup Language (SGML). The language expresses the description of a document's structure and other attributes, as well as other information that makes the markup interpretable. This International Standard specifies a reference concrete syntax that binds the abstract syntax to specific characters and numeric values, and criteria for defining variant concrete syntaxes. This International Standard defines conforming documents in terms of their use of components of the language. This International Standard defines conforming systems in terms of their ability to process conforming documents and to recognize markup errors in them. Specifies how data not defined by this International Standard (such as images, graphics, or formatted text) can be included in a conforming document. (ISO 8879 "1 Scope") SGML was designed to interchange documents without regard to how the information is formatted. This allows for the use of the information in many different

formats. SGML was designed to be application independent, and as such can be used in conjunction with a database application. The user is allowed to interact with and to modify the logical structures which are a primary part of his application. An SGML document may be processed by any formatter (for a formatting application) which has been suitably enabled with an SGML parser and other entity-management software. The SGML notation may be used to describe both logical and layout structures, if the format of the document is also to be interchanged. A set of standardized formatting semantics are to be provided by DSSSL.

Abbreviation: SDIF

Standard: Information Processing - SGML Support Facilities - SGML Document Interchange Format (SDIF)

Designator/status: ISO IS 9069:1987/International Standard (1987)

Originating group:

Field of application: Document Interchange

Use of this standard: The SGML Document Interchange Format shall be used solely for the interchange of SGML documents, as defined in ISO 8879, among SGML systems. Interchange can be by means of data communications in Open Systems Interconnection or other environments, or by the exchange of storage media.

Other standards using this standard:

Brief Description: This International Standard specifies a data structure known as the SGML Document Interchange Format (SDIF). The SDIF data stream represents one or more SGML document entities, and zero or more SGML sub-document, SGML text, and data entities, as defined in ISO 8879. SDIF enables a document conforming to ISO 8879, which might be stored in several entities, to be packed into a data stream for interchange in a manner that will permit the recipient to reconstitute the separate entities. SDIF also allows related documents to be included in the data stream, such as covering letters, transmittal forms, catalog cards, formatting procedures, font resources, or the 'document profile' required by a document architecture.

Abbreviation: GGCA

Standard: Geometric Graphics Content Architectures (GGCA),

Designator/status: ISO 8613-8:1988/International Standard (1988)

Originating group:

Field of application:	Image processing, storage, retrieval and transmission
Use of this standard:	The purpose of this International Standard is to facilitate the interchange of documents.
Other standards using this standard:	CGM, CCITT Group 3 & 4, SGML, ODA, EDI
Brief Description:	In the context of ISO 8613, documents are considered to be items such as memoranda, letters, invoices, forms and reports, which may include pictures and tabular material. The content elements used within the documents may include graphic characters, geometric graphics elements and raster graphics elements, all potentially within one document. ISO 8613 applies to the interchange of documents by means of data communication or the exchange of storage media. (ISO 8613-8 "1 Scope"). This standard defines a geometric graphics content architecture that can be used in conjunction with the document architecture defined in ISO 8613-2; defines an interface which allows the use of content structured [sic] according to ISO 8632 within documents structured according to ISO 8613-2; defines those aspects of positioning and imaging applicable to the presentation of this geometric graphics content architecture in a basic layout object; defines the presentation attributes applicable to this geometric graphics content architecture; describes a content layout process, which together with the document layout process described in ISO 8613-2, describes the layout of geometric graphics content in basic layout objects and determines the dimensions of these basic layout objects. ISO 8613-8 applies to documents that are structured according to the architecture defined in ISO 8613-2 that include geometric graphics content, consisting of a descriptive representation of picture description information as an ordered set of elements such as lines, arcs, polygons, attributes for these drawing elements,elements that structure the content portion, etc. using the Computer Graphics Metafile (CGM) and its binary encoding defined in ISO 8632-1 and ISO 8632-3, respectively.
Abbreviation:	Group 4
Standard:	CCITT Group 4 Facsimile
Designator/status:	International Standard (1984)
Originating group:	CCITT
Field of application: Networks	Telecommunications, Electronic Data Transfer, Wide Area and Local Area
Use of This Standard:	Image compression, storage and transfer across telecommunications networks and computer communications networks
Other standards	Data Transfer Protocols such as X.25, Synchronous and Bi-synchronous

using this standard: communications.

Brief Description: The Group 4 Facsimile Standard has two parts. Recommendation T.5 defines the general aspects of Group 4 facsimile apparatus. The Group 4 facsimile coding scheme and facsimile control functions are defined in Recommendation T.6. (CCITT Red Book "Recommendation T.5 2 Scope").

The Group 4 apparatus provides the means for direct document transmission from any subscriber to any other subscriber. All apparatus participating in the international Group 4 facsimile service has to be compatible with each other at the basic level defined in this Recommendation. Additional operational functions may be invoked. The range of data rates is described in Section 6. Detailed arrangements on a national level are left to the Administrations concerned, as it is recognized that national implementation of the Group 4 facsimile service on various types of networks may involve national operation at different data throughput rates. The page is the basis for facsimile message formatting and transmission. Both A4 and North American paper formats are taken into account. Facsimile coding schemes are applied in order to reduce the redundant information in facsimile signals prior to transmission. The apparatus must have the ability to reproduce facsimile messages. The content, layout and format of facsimile messages must be identical at the transmitting and receiving apparatus. The reproducible area is defined within which facsimile messages are assured to be reproduced. The Group 4 facsimile apparatus should provide means for automatic reception. In addition Class II/III apparatus should provide means for automatic reception of Telex and mixed mode documents. All Classes of Group 4 facsimile apparatus shall incorporate the functions defined as basic for the Group 4 facsimile service in Section 3.2 below. In addition, optional functions can be incorporated. In this Recommendation, the optional functions are divided into CCITT standardized options and nationally and/or privately specified options. (CCITT Red Book "Recommendation T.5 3.1 Basic Characteristics") Facsimile coding schemes consist of the basic facsimile coding scheme and optional facsimile coding schemes. Facsimile coding schemes are specified assuming that transmission errors are corrected by control procedures at a lower level. "The basic facsimile coding scheme is the two-dimensional coding scheme which is in principle the same as the two-dimensional coding scheme of Group 3 facsimile specified in Recommendation T.4. Optional facsimile coding schemes are specified not only for black and white images but also for grey scale images and colour images. Facsimile coding control functions are used in facsimile user information in order to change facsimile parameters or to invoke the end of facsimile block. (CCITT Red Book, "Recommendation T.6 1.2.1 Facsimile coding schemes and coding control functions") USE: Group 4 facsimile is used mainly on public data networks (PDN) including circuit-switched, packet-switched, and the integrated services digital network (ISDN). The apparatus may also be used on the public switched telephone network (PSTN) where an appropriate modulation process will be utilized. "The procedures used with Group 4 facsimile apparatus enable it to transmit and reproduce image coded information essentially without transmission errors. Group 4

facsimile apparatus has the means for reducing the redundant information in facsimile signals prior to transmission. The basic image type of the Group 4 facsimile apparatus is black and white. Other image types, e.g. grey scale image or colour image, are for further study. There are three classes of Group 4 facsimile terminals:

- Class I - Minimum requirement is a terminal able to send and receive documents containing facsimile encoded information.

- Class II - Minimum requirement is a terminal able to transmit documents which are facsimile encoded. In addition, the terminal must be capable of receiving documents which are facsimile coded, Teletex coded, and also mixed-mode documents.

- Class III - Minimum requirement is a terminal which is capable of generating, transmitting and receiving facsimile coded documents, Teletex coded documents, and mixed-mode documents. (CCITT Red Book "Recommendation T.5 1 General") Group 4 facsimile apparatus shall be capable of handling:

a. the basic end-to-end control procedures as defined in Recommendation T.62;

b. document interchange protocol as defined in Recommendation T.73;

c. the basic facsimile coding scheme as defined in Recommendation T.6;

d. the control functions associated with the basic facsimile coding scheme as defined in Recommendation T.6.

"All classes of Group 4 apparatus shall have the following provisions for facsimile messages:

a. provision for scanning the documents to be transmitted;

b. provision for receiving and presenting hard or soft copies of the documents.

In addition Group 4 Class II apparatus shall have provision for receiving and displaying basic Teletex and mixed mode documents. In addition to the requirements for Group 4 Class II apparatus, Class III apparatus shall have provisions for generating and transmitting basic Teletex and mixed mode documents. Basic page formatting functions are as follows:

a. vertical page orientation;

b. paper size of ISO A4;

c. reproducible area/printable area is defined taking into account ISO A4 and North American paper formats and ISO standard 3535. (CCITT Red Book

"Recommendation T.5 3.2 Basic Functions").

Abbreviation:	EDI
Standard:	Electronic Data Interchange (EDI)
Designator/status:	ANSI X12.3:1986 Data Element Dictionary ANSI X12.6:1986 Application Control Structure ANSI X12.20:1986 Functional Acknowledgment ANSI X12.22:1986 Data Segment Directory
Originating group:	
Field of application:	Computer Network Communications
Use of This Standard:	File transfer across computer communications networks.
Other standards using this standard:	CCITT Group 3 &4, JPEG, ASN1, X25, SGML, ODA
Brief Description:	Transaction set standards define the procedural format and data content requirements for specified business transactions, such as purchase orders. The data dictionary defines the precise content for data elements used in building transaction sets. The segment directory provides the definitions and formats of the data segments used in building transaction sets. The transmission control standards define the formats for the information required to interchange data. These controls are already in use by some industry groups. EDI is used to standardize the format and content of data to be interchanged between two computers. Subsets of the total EDI Standards package will be selected based on the needs of the potential interchange partners.
Abbreviation:	ODA
Standard:	Office Document Architecture (ODA)
Designator/status:	ISO 8613:1988/International Standard (1988)
Originating group:	ISO-IEC/JTC1/SC18
Field of application:	The purpose of this international standard is to facilitate the interchange of documents. In the context of ISO 8613, documents are considered to be items such as memoranda, letters, invoices, forms and reports, which may include pictures and tabular material. The content elements used in the documents may include graphic characters, geometric graphics elements, and raster graphics elements, all potentially within one document.

Use of this standard: ODA/ODIF is specifically designed for the interchange and replication of office documents in exact format. The design strives to be content-independent in order to allow for future content architectures such as audio information or possible mathematical and scientific equations.

Other standards using this standard: ODIF, SGML, ASN1

Brief Description: ODA was developed to allow the interchange of documents from one word processor to another. Page layout is handled according to some precise semantics which strive to be content independent. The page or sets of pages are specified denoting margins, columns, character path, line progression, etc., which detail the placement of rectangular "blocks," with content, specifically characters, image, and graphics to be poured in to occupy various areas on the page.

The parts of the standard are as follows:

1. General Introduction
2. Document Structures
3. Document Profile
4. Office Document Interchange Format (ODIF) (see ODIF, ODL, and SDIF)
5. Character Graphics Content Architectures
6. Raster Graphics Content Architectures (see Raster and TRIF)
7. Geometric Graphics Content Architectures (see GGCA)

NOTE - ISO 8613 is designed to allow for extensions, including typographical features, colour, spreadsheets and additional types of content such as sound." (ISO 8613-1:1988 (E) "1.1)

Abbreviation: ODIF

Standard: Office Document Interchange Format

Designator/status: ISO 8613-5:1988/ International Standard (1988)

Originating group: ISO-IEC/JTC1/SC18

Field of application: The purpose of this international standard is to facilitate the interchange of documents. In the context of ISO 8613-5, documents are considered to be items such as memoranda, letters, invoices, forms and reports, which may include pictures and tabular material. The content elements used in the documents may include graphic characters, geometric graphics elements, and raster graphics elements, all potentially within one document.

Use of this standard: The purpose of this International Standard is to facilitate the interchange of documents.In the context of ISO 8613, documents are considered to be items such as memoranda, letters, invoices, forms and reports, which may include

pictures and tabular material. The content elements used within the documents may include graphic characters, geometric graphics elements and raster graphics elements, all potentially within one document. ISO 8613 applies to the interchange of documents by means of data communication or the exchange of storage media. A document structured in accordance with ISO 8613 may be represented for interchange by the Office Document Interchange Format (ODIF).Since ODIF is a data structure specified using ASN.1, it is intended for use in an OSI environment.

Other standards using this standard: ODA

Brief Description: This part of ISO 8613 defines the format of the data stream used to interchange documents structured in accordance with ISO 8613-2; defines the representation of the constituents which may appear in an interchanged document. ODIF is an abstract data syntax in which the constituents and attributes of the document are represented by a hierarchy of data structures and data items, specified using the abstract syntax notation ASN.1 defined in ISO 8824. The coded representation of each data structure or data item is obtained by applying a set of encoding rules. The ODIF data stream is described in terms of a set of data structures, called 'interchange data element', which represent the constituents (document profile, object descriptions, object class descriptions, presentation styles, layout styles and content portion descriptions) of a document. The formats of the interchanged data element according to ODIF are defined using the Abstract Syntax Notation One (ASN.1) specified in ISO 8824.

Abbreviation: ODL

Standard: Office Document Language

Designator/status: ISO 8613-5:1988/International Standard (1988)

Originating group: ISO-IEC/JTC1/SC18

Field of application: The purpose of this international standard is to facilitate the interchange of documents. In the context of ISO 8613-5, documents are considered to be items such as memoranda, letters, invoices, forms and reports, which may include pictures and tabular material. The content elements used in the documents may include graphic characters, geometric graphics elements, and raster graphics elements, all potentially within one document.

Use of this standard: In the context of ISO 8613, documents are considered to be items such as memoranda, letters, invoices, forms and reports, which may include pictures and tabular material. The content elements used within the documents may include graphic characters, geometric graphics elements and raster graphics elements, be all potentially within one document. ISO 8613 applies to the interchange of documents by means of data communication or the exchange of storage media.

A document structured in accordance with ISO 8613 may be represented for interchange by the Office Document Language (ODL) in conjunction with the SGML Document Interchange Format (SDIF). ODL is particularly appropriate for systems that share information through marked-up text files, especially where human users can access the markup directly.

Other standards using this standard: ODA, SGML, ASN1, ODIF

Brief Description: This part of ISO 8613 defines the format of the data stream used to interchange documents structured in accordance with ISO 8613-2; defines the representation of the constituents which may appear in an interchanged document. ODL uses the Standard Generalized Markup Language (SGML) specified in ISO 8879. It consists of a standard set of SGML names and markup conventions for representing the constituents and attributes of a document.

Abbreviation: TRIF

Standard: Tiled Raster Interchange Format (TRIF)

Designator/status: Proposed to ANSI X3V1 as an Extension to ISO 8613

Originating group: ANSI X3V1 by the Tiling Task Group, February 1988.

Field of application: The purpose of this International Standard is to facilitate the interchange of documents.

Use of this standard: The tiling scheme developed provides a format that supports operation on a subset of an image without requiring other portions of the image to be accessed. For large format documents this provides a way to interchange images between systems of various capabilities. Further, the tile format was developed for interchange that could also reasonably be used for storage and retrieval without necessarily requiring translation. ISO 8613 applies to the interchange of documents by means of data communications or the exchange of storage media.

Other standards using this standard: ISO 8613

Brief Description: This part of ISO 8613 defines: the tiled raster graphics content architectures that can be used in conjunction with the document architecture defined in ISO 8613-2; the internal structure of content portions that are structured according to a tiled raster graphics contents in a basic layout object; those aspects of positioning and imaging applicable to the presentation of tiled raster graphics contents in a basic layout object; a content layout process which together with the document layout process defined in ISO 8613-2, specifies the method for determining the dimensions of basic layout objects for tiled raster graphics content portions; the presentation and content portion attributes applicable to

tiled raster graphics content architectures. The following restrictions for use were made to ease user implementation: This interchange format deals only with bi-tonal (black and white) data. Pixels are assumed to be square. A tile is a rectangular region in a page in which all regions have the same dimensions (are regular) and no part of any region overlaps any other region. They are positioned in a fixed grid, determined by partitioning the page into tile-sized areas. For the purposes of this interchange format, the application profile restricts all tiles to being square. Square tiles have the desirable attribute of being easily rotated. Tiles are allowed to be absent A single tile size is desirable to limit the burden on implementors of the interchange standard. The tile size is specifically 512 by 512 pels. Only one page (one single raster image) is allowed per document. Any given tile is to be encoded as T.6 compressed data, as bitmap data, or is specified as all foreground or all background. In the context of ISO 8613, documents are considered to be items such as memoranda, letters, invoices, forms and reports, which may include pictures and tabular material. The content elements used within the documents may include graphic characters, geometric graphics elements and raster graphics elements, all potentially within one document.

B.3.8 Procedural language processing services

Abbreviation: SPDL

Standard: Standard Page Description Language (SPDL)

Designator/status: ISO IS 10180

Originating group: ISO-IEC/JTC1/SC18/WG8

Field of application: Document Interchange and Electronic Publishing

Use of this standard: This International Standard is intended for use in a wide variety of application environments, including: electronic publishing (including production publishing, work group publishing, desktop publishing, database publishing, electronic pre-press, etc.), office systems, information networks, and demand printing.

Other standards using this standard: DSSSL, SGML, ODA, ODL, Hytime

Brief Description: This International Standard provides a straightforward and efficient method of representing documents which are generated by ODA systems to presentation devices. It also provides a capability for similarly representing documents generated by SGML systems whose formatting is described by DSSSL This International Standard allows for document presentation to be disjoint in both time and place from the document creation and formatting processes. It is specifically intended that SPDL document descriptions will be: sent directly to presentation systems which are accessed via a local connection sent to

proximate or remote presentation systems via OSI or non-OSI networks, and stored or interchanged for the purpose of presentation at other times or at other locations. The Standard Page Description Language is capable of representing all content types for fully composed, non-revisable documents. Any combination of the following types of content can be represented; any content may in [sic] black-and-white, gray-scale, or full colour; and content types may be intermixed in any way in the same document. In addition to specifying how document images are represented, this International Standard specifies how additional information called printing instructions affects the document image. Printing instructions may be supplied with the request to print the document by means of a print access protocol.

Abbreviation: DSSSL

Standard: Document Style Semantics and Specification Language (DSSSL)

Designator/status: ISO/IEC IS 10179

Originating group: ISO/IEC/JTC1/SC18/WG8

Field of application: Information Technology - Text and Office Systems

Use of this standard: To provide a formal and rigorous means of expressing the range of document production specifications, including high-quality typography, required by the graphic arts industry.

Other standards using this standard: SGML, ODA, ODL, ODIF, Hytime, HGML,SMSL, SPDL

Brief Description: This standard defines the semantics and syntax of a language for the specification of document processing. The semantics of DSSSL include a document architecture for typographic presentation style and other document processing specifications, typically associated with traditional text processing languages. DSSSL also incorporates a Specification Language which describes how to apply DSSSL semantics to SGML documents. DSSSL semantics may also be used in conjunction with ODA documents represented in the Office Document Language (ODL). DSSSL includes:

a) transformations independent of a particular type of semantic processor;

b) provisions for specifying the relationships between SGML logical elements, as expressed in the source Document Type Definition, and the semantic-specific result document, e.g., the output of a formatter. The semantic-specific result document may be an ISO 10180 SPDL document; it may be used as input to a formatter; or it may be a document in some other, possibly proprietary, form;

c) formatting and style semantics to describe the typographic style and

layout of a document, including functions that allow the transformation of documents from one representation to another;

d) a specification language that describes how DSSSL semantics apply to SGML documents;

e) definitions of a syntax for the representation of a DSSSL specification and its various components;

f) provisions for creating new DSSSL attributes and attribute values defined in terms of DSSSL properties. These attributes and permitted values are declared in the semantic-specific language declaration;

Other standards associated with "*Procedural language processing services*":

ISO 646 - 7 bit character set
ISO 2375 - register of ISO 2022 control and character sets
ISO 6937 - diacritical formed accented characters
ISO 8859 - Fully formed accented characters
ISO 9282 - Coded Representation of Computer Graphics Images
ISO 10646 - Multi-octet character set (Unicode)
ISO 2022 - 7 bit and 8 bit code extension techniques
ISO 6429 - Coding of controls

B.3.9 Other services

Abbreviation: POSIX OSE

Standard: POSIX Open Systems Environment

Designator/status: P1003.0/D15

Originating group: Technical Committee on Operating Systems and Application Environments of the IEEE Computer Society

Field of application: Open system concepts and their application

Use of this standard: The POSIX OSE brings together many different standards necessary to address the scope of an entire information processing system. P1003 identifies standards that can be used when constructing a complete information processing system.

Other standards using this standard:

Brief Description: The POSIX Open Systems Environment comprises a reference model used to

classify information processing standards. The reference model categorizes standards as two types of interfaces:

a) Application Programming Interface (API) Standards - These standards affect how application software interacts with the computer system. These standards affect application portability.

b) External Environment Interface (EEI) Standards - these standards affect how an information processing system interacts with its external environment. These standards affect system interoperability, user interface look and feel, and data portability.

Annex C (informative)

References

C.1 References

Interactive Multimedia Association. (1991). IMA compatibility project proceedings. Vol. 1. Issue 1., IMA Compatibility Project Headquarters, 9 Randall Court, Annapolis, MD. 21401. September.

Interactive Multimedia Association. (1992). IMA compatibility project proceedings. Vol. 2. Issue 1., IMA Compatibility Project Headquarters, 9 Randall Court, Annapolis, MD. 21401. March.

ACM.(1992) *Communications of the ACM.* Vol 35. No. 1 January.

Note: [Special issue on hypermedia.**]**

IEEE. (1991) *Computer Graphics and Applications.* Vol. 11. No. 4. July.

Note: [Special issue on multimedia.**]**

C.2 Multimedia and hypermedia model and framework input documents

ISO/IEC JTC1/SC18/WG1-N1347 "Multimedia and Hypermedia Model/Framework Input."

All the documents listed in ISO/IEC JTC1/SC18/WG1-N1347 "Multimedia and Hypermedia Model/Framework Input."

ISO/IEC JTC1/SC18/WG1-N1346 "Revised Modelling for Multimedia and Hypermedia."

ISO/IEC JTC1/SC18/WG1-N1345 "Japanese Comments on WG1 N1285; US Proposal for Multimedia/Hypermedia Model/Framework."

ISO/IEC JTC1/SC18/WG1-SD4 "Integrated Video Services (IVS) Baseline Document: Annex 6, Multimedia Service Support."

ISO/IEC JTC1/SC18/WG1-SD2 "INSPEC Articles Jan 24, 1991" (abstracts of "Hypertext: Concepts Systems and Applications. Proceedings of the First European Conference on Hypertext 1990 P25-37" and "Hypertext" Concepts, Systems and Applications. Proceedings of the First European Conference of Hypertext 1990 P81-94.)"

ISO/IEC JTC1/SC18/WG1-SD2 "LINKS: What Link Characteristics are Supported?"

ISO/IEC JTC/SC18/WG1-SD5 "Creators, Vendors Form Multimedia Alliances."

ISO/IEC JTC1/SC18/WG1-SD3 "INSPEC articles Jan 24, 1991" (abstract of "Toolkit hypermedia facilities. Hypertext: Concepts, Systems and Applications. Proceedings of the First European Conference on Hypertext 1990 P13-24.").

ISO/IEC JTC1/SC 18/WG1 N1409 "Liaison Contribution from JTC1/SC29 to JTC1/SC18 on Multimedia and Hypermedia Model/Framework."

ISO/IEC JTC1/SC18/WG1 N1399 "Information between Open System by Interchangeable Storage Media - Basic Reference Model."

ISO/IEC JTC1/SC18/WG1 N1392 "Liaison Statement from CCITT study group I to JTC1/SC18 on Audiovisual interactive service activity in CCITT SG1."

ISO/IEC JTC1/SC18/WG1 N1408 "Liaison Document ISO/IEC JTC1/SC29 N156 SC29 Position on NP MM Model."

ISO/IEC JTC1/SC18/WG1 N1393 "Liaison Statement to JTC1/SC18 from JTC1/SC15 responding to SC18 liaison statement on Multimedia and Hypermedia Model/Framework."

Annex D (informative)

Application scenarios

D.1 Multimedia CD creation and Use

D.2 Multimedia mail creation and receiving

D.3 Multimedia conferencing

D.4 Access to Multimedia Objects on the Internet

D.5 Multimedia Interactive Training

D.6 Multimedia Games and Entertainment

Annex E (informative)

Multimedia Related Acronym Definitions

Annex F (informative)

Computer Related Organizations Acronyms Definitions