Part V
Content Description

MPEG-7

motivation

- exponentially increasing amount of audiovisual information is becoming available in digital form (digital archives, WWW, broadcast data streams, personal and professional DBs, etc.)
- new ways to produce, offer, filter, search, and manage digitized multimedia information
- broadband is being offered with increasing audio and video quality and speed of access.
- information value often depends on how easy information can be found, retrieved, accessed, filtered and managed.

MPEG-7

motivation (continued)

- users will be confronted with such a large number of contents that efficient and accurate access to this content will seem to be unimaginable.
- identifying and managing content efficiently is becoming more difficult, because of the sheer volume.
- problem not restricted to database retrieval applications such as digital libraries, but extends to areas like broadcast channel selection, multimedia editing, and multimedia directory services

MPEG-7

solution to this problem: MPEG-7

- ISO/IEC standard; formally called "Multimedia Content Description Interface"
- Both human users and automatic systems that process audiovisual information are within the scope of MPEG-7.
- aims at offering a comprehensive set of audiovisual description tools that form the basis for applications enabling the needed quality access to content
- active people: broadcasters, electronics manufacturers, content creators and managers, publishers and intellectual property rights managers, telecommunication service providers and academia

MPEG-7 Main Goals

- description of multimedia content,
- flexibility in data management, and
- globalization and interoperability of data resources.

MPEG-7 aims to standardize

- a set of Description Schemes and Descriptors to describe data,
- a language to specify Description Schemes, such as the Description Definition Language(DDL), and
- a scheme for coding the Description.

Terminology

Definition: A *Feature* is a distinctive characteristic of the data that signifies something to somebody.

- Features require a meaningful Feature representation (Descriptor) and its instantiation (Descriptor Value) for a given data set.
- examples: the color of an image, fundamental frequency of a speech segment, rhythm of an audio segment, camera motion in a video, genre of a piece of music, title of a movie, and actors in a movie.

Terminology

Definition: A *Descriptor* is a representation of a Feature. A Descriptor defines the syntax and semantics of the Feature representation.

- A descriptor must precisely define the semantics of the feature, the associated data type, legal values, and an interpretation of the Descriptor Values.
- An example might be `Color: string`. The data type may be composite, meaning that it can be formed by concatenating multiple instances of a data type. An example of this would be `RGB-Color: [integer, integer, integer]`.

Descriptor and Descriptor Value

- several descriptors may represent a single feature—that is, address different relevant requirements.
- examples of multiple descriptors for a single feature include enumerated lists, color moments, and histograms that represent color.
- the syntax of a Descriptor might look like this:

```
<name, typekind, spec> value </name>
```

Definition: A *Descriptor Value* is an instantiation of a Descriptor for a given data set (or subset thereof). Descriptor Values are combined via a Description Scheme to form a Description.

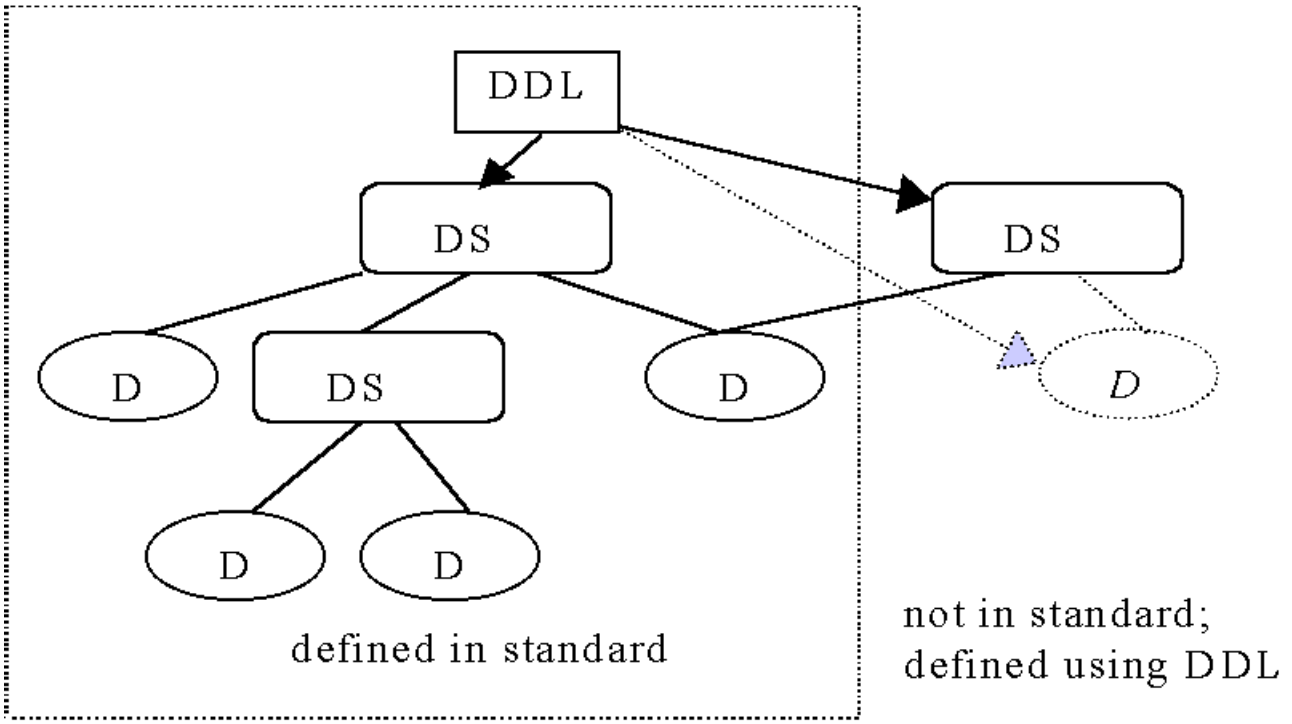
Descriptor Items

Item	Description
Name	Descriptor ID
Typekind	Data type of Descriptor Value: free text (plus language identifier), structured text (plus structure identification), integer, real, date, time/time index, version, enumeration (such as RGB values, vector), relation (Descriptor-Descriptor, Descriptor-Description Scheme, Description-Description), complex (a bitmap, histogram)*, function*, and possibly trigger (executable code)*
Spec	Contains a specification for the data type. For example, “free text” might specify the language the text is written in (Dutch, English, Japanese, and so on).
Value	An instantiation of the Descriptor is assigned to the Feature as pertaining to the data. Descriptor Values are combined via the mechanism of a Description Scheme to form a Description

*Value could contain the actual data or a link

Terminology

Definition: A *Description Scheme* specifies the structure and semantics of the relationships between its components, which may be both Descriptors and Description Schemes.



Description Scheme

- The last figure presents an abstract representation of possible relations between Descriptors and Description Schemes. The arrows from DDL to Description Scheme signify that the Description Schemes are generated using the DDL.

- The distinction between a Description Scheme and a Descriptor is that a Descriptor is concerned with the representation of a Feature whereas the Description Scheme deals with the structure of a Description.

Description Scheme

- A simple Description Scheme for describing technical aspects of a shot (elements written in bold represent other Description Schemes):

Shot_Technical_Aspects

Lens providing information about type (wide-angle), movement (zooms), state (deep focus), masking, and so on.

Camera providing information about distance (close-up), angle (overhead), move-ment (pan_left), position (viewpoint of the frame), and so on.

Speed

Color

Granularity

Contrast

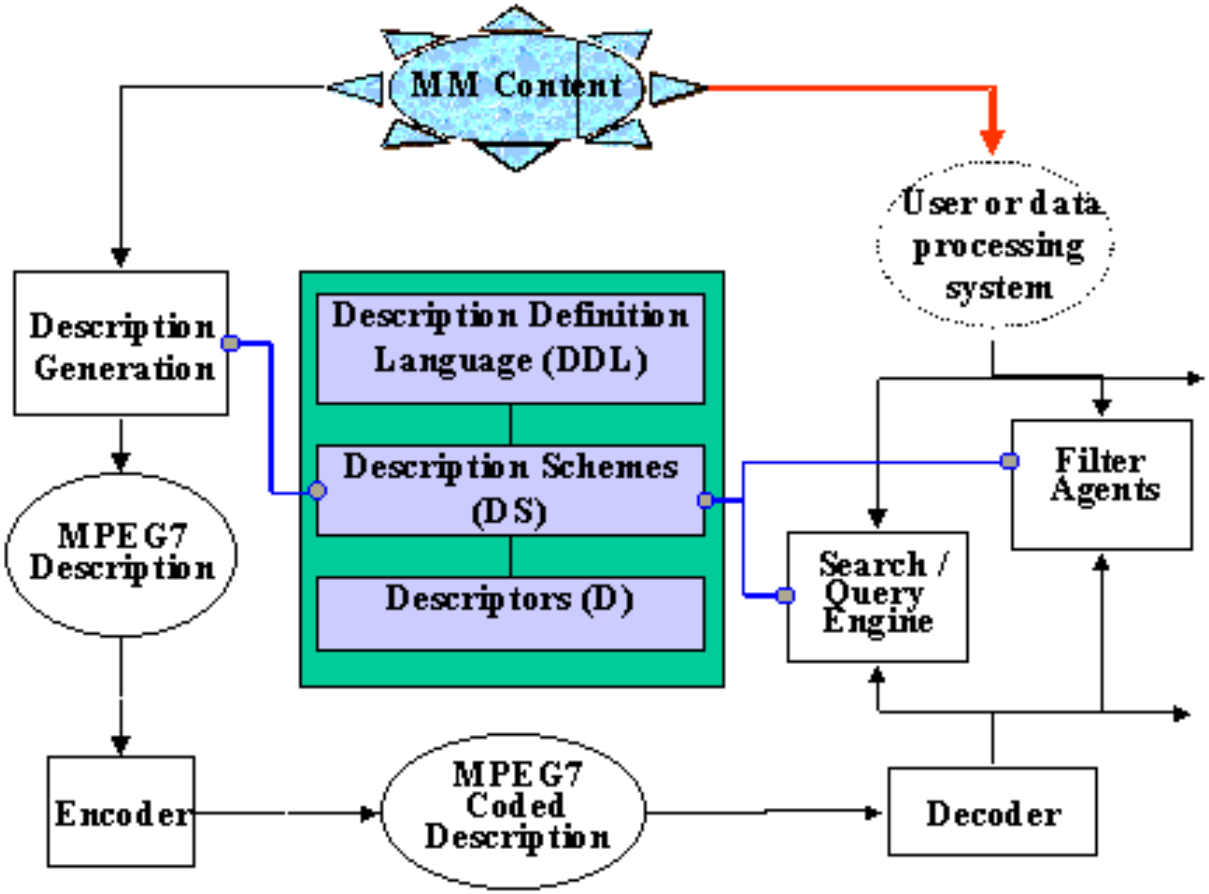
Terminology

Definition: A *Description* consists of a Description Scheme (structure) and the set of Descriptor Values (instantiations) that describe the data. A Description contains or refers to a fully or partially instantiated Description Scheme.

Definition: A *coded description* is a Description that has been encoded to fulfill relevant requirements such as compression efficiency, random access, and so on.

Definition: The *Description Definition Language (DDL)* allows the creation of new Description Schemes and, possibly, Descriptors. It also allows the extension and modification of existing Description Schemes.

MPEG-7 in Practice



The left side portrays how data is annotated, whereas the right side demonstrates how described data can be retrieved.

Square boxes describe processing tools (e.g. encoding or decoding). Circular boxes describe static elements, such as a Description.

For example, the “description generation” box is a description generation engine that produces an “MPEG-7 description” as output, shown here as a rectangular box.

Requirements

The DDL's design forms a core part of the work within MPEG-7. The DDL provides a solid descriptive (for example, SGML-based) underpinning for users to create their own Description Schemes and Descriptors. For this purpose, MPEG-7 has identified a number of requirements the DDL should cover:

- *Compositional capabilities*. The DDL supplies the ability to compose new Description Schemes and Descriptors, where a Description Scheme might be composed from multiple Description Schemes. A newly created Description Scheme must allow the creation of MPEG-7 compliant Descriptions.

Requirements

- *Transformational capabilities*. The DDL allows the reuse, extension, and inheritance of existing Descriptors and Description Schemes.
- *Unique identification*. The DDL provides mechanisms to uniquely identify Description Schemes and Descriptors so that they can be referred to unambiguously.
- *Data types*. The DDL provides a set of primitive data types, such as e.g. text, integer, real, date, time/time index, version to succinctly describe composite data types that might arise from processing digital signals (such as histograms, graphs, RGB values). Also, the DDL must provide a mechanism to relate Descriptors to data of multiple media types of inherent structure (e.g. audio, video, audio-visual presentations, etc.)

Requirements

- *Relationships within a Description Scheme and between Description Schemes*. The DDL provides the capability to express relationships between Description Schemes and among elements of a description Scheme. The DDL expresses the semantics of these relations, such as spatial, temporal, structural, and conceptual relations.
- *Relationship between Description and data*. The DDL shall supply a rich model for links and/or references between one or several Descriptions and the described data.

Parts of the Standard

The MPEG-7 Standard consists of the following parts:

1. MPEG-7 Systems - the tools that are needed to prepare MPEG-7 Descriptions for efficient transport and storage, and to allow synchronization between content and descriptions. Tools related to managing and protecting intellectual property
2. MPEG-7 Description Definition Language - the language for defining new Description Schemes and perhaps eventually also for new Descriptors.
3. MPEG-7 Audio - the Descriptors and Description Schemes dealing with (only) Audio descriptions

Parts of the Standard

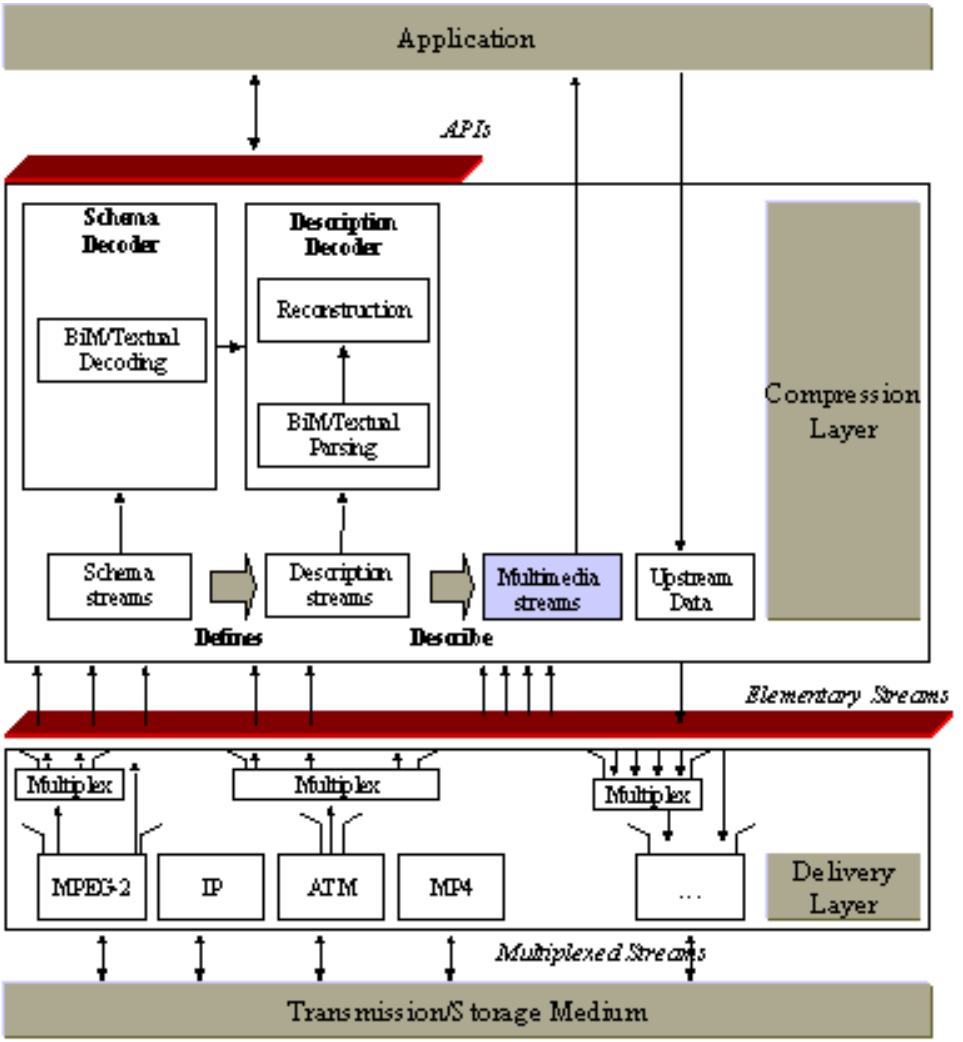
4. MPEG-7 Visual - the Descriptors and Description Schemes dealing with (only) Visual descriptions
5. MPEG-7 Multimedia Description Schemes - the Descriptors and Description Schemes dealing with generic features and multimedia descriptions
6. MPEG-7 Reference Software - a software implementation of relevant parts of the MPEG-7 Standard
7. MPEG-7 Conformance - guidelines and procedures for testing conformance of MPEG-7 implementations.

MPEG-7 Systems

MPEG-7 Systems defines the terminal architecture and the normative interfaces:

- *Terminal*—the entity that makes use of coded representation of the multimedia content; may correspond to a stand-alone application or be part of an application system:
  - application
  - compression layer
  - delivery layer
  - transmission/storage medium

Terminal Architecture



Terminal Architecture

- transmission/storage medium—refers to the lower layers of the delivery infrastructure (network layer and below as well as storage); these layers deliver multiplexed streams to the DL.
- delivery layer (DL)—encompasses mechanisms allowing synchronization, framing and multiplexing of MPEG-7 content.
  - MPEG-7 content may be delivered independently or together with the content they describe. Not all MPEG-7 streams have to be downstream (server to the client).
  - provides to the Compression layer MPEG-7 elementary streams. MPEG-7 elementary streams consist in consecutive individually accessible portion of data named *Access Units*.

Terminal Architecture

delivery layer (continued)

- An *access unit* is the smallest data entity to which timing information can be attributed
- MPEG-7 elementary streams contain information of different nature:
  - Schema information: this information defines the structure of the MPEG-7 description;
  - Descriptions information: this information is either the complete description of the multimedia content or fragments of the description.

Terminal Architecture

- compression layer—the flow of Access Units (either textual or binary encoded) is parsed, and the content description is reconstructed.
  - MPEG-7 does not mandate the reconstruction of a textual representation as an intermediate step of the decoding process.
  - The MPEG-7 binary stream can be either parsed by the BiM parser, transformed in textual format and then transmitted in textual format to further reconstruction processing, or the binary stream can be parsed by the BiM parser and then transmitted in proprietary format to further processing.

MPEG-7 Visual

MPEG-7 visual description tools consist of b*asic structures* and descriptors that cover the following basic visual features:

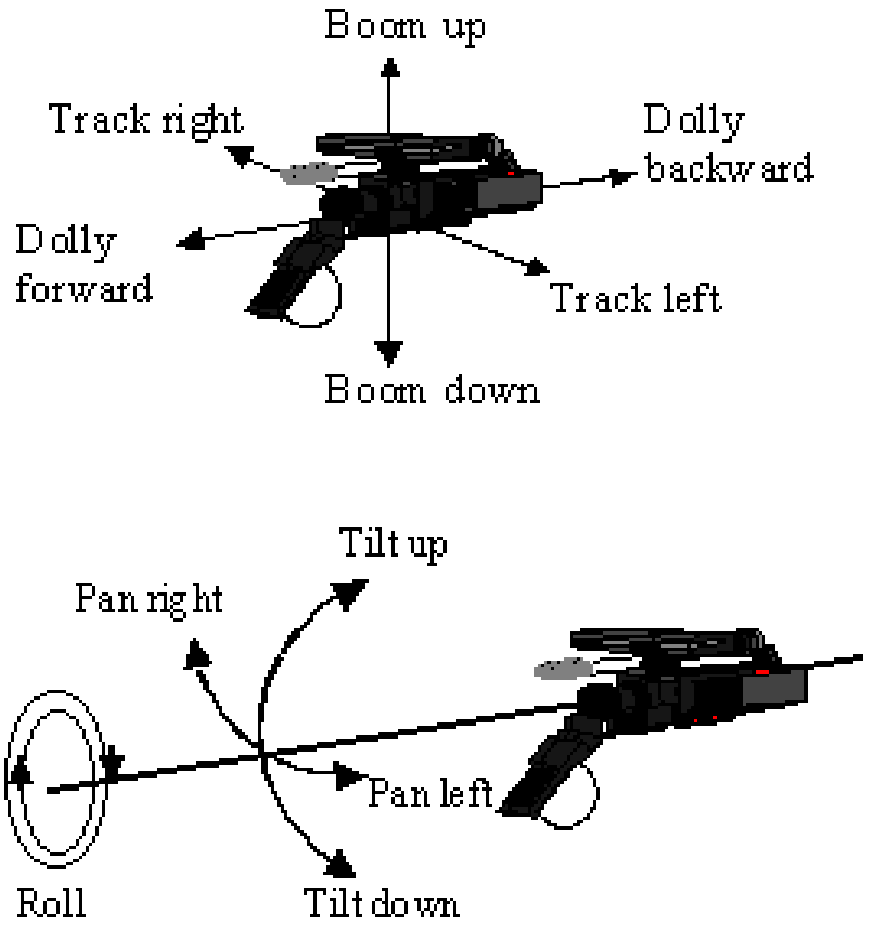
- Color
- Texture
- Shape
- Motion
- Localization
- Others

Each category consists of *elementary* and *sophisticated descriptors*

Descriptors

- 5 Basic structures: the Grid Layout, the Time Series, MultiView, the Spatial 2D Coordinates, and Temporal Interpolation.
- 8 Color Descriptors: Color space, Dominant Colors, Color Quantization, GoF/GoP Color, Color-Structure, Color Layout and Scalable Color Histogram
- 3 Texture Descriptors: Homogeneous Texture, Texture Browsing and Edge Histogram
  - Texture Browsing Descriptor—useful for representing homogeneous texture for browsing type applications, and requires only 12 bits (maximum). It provides a perceptual characterization of texture (similar to a human) in terms of regularity, coarseness and directionality

Camera Motion Descriptors



Motion Descriptors

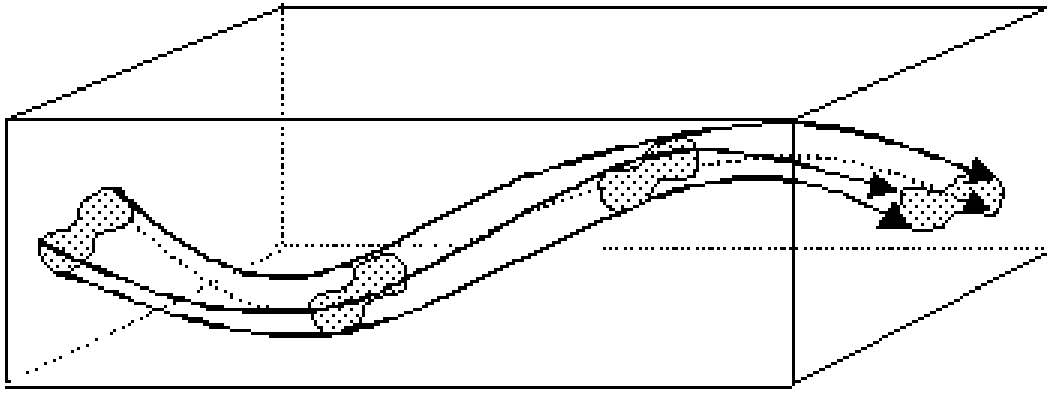
- motion trajectory of an object—a simple, high level feature, defined as the localization in time and space, of one representative point of this object.
  - the descriptor is essentially a list of keypoints (x,y,z,t) along with a set of optional interpolating functions that describe the path of the object between keypoints, in terms of acceleration. The speed is implicitly known by the keypoints specification.
  - The keypoints are specified by their time instant and either their 2-D or 3-D Cartesian coordinates, depending on the intended application. The interpolating functions are defined for each component x(t), y(t), and z(t) independently.

Motion Descriptors

- Parametric motion—has been extensively used within various related image processing and analysis areas, including motion-based segmentation and estimation, global motion estimation, mosaicing and object tracking. Parametric motion models have been already used in MPEG-4, for global motion estimation and compensation and sprite generation
- activity descriptor—captures the intuitive notion of 'intensity of action' or 'pace of action' in a video segment. Examples of high 'activity' include scenes such as 'goal scoring in a soccer match', 'scoring in a basketball game', 'a high speed car chase' etc..

Localization

- Region Locator—enables localization of regions within images or frames by specifying them with a brief and scalable representation of a Box or a Polygon.
- Spatio Temporal Locator—describes spatio-temporal regions in a video sequence, such as moving object regions, and provides localization functionality (for hypermedia and CBR)



Others

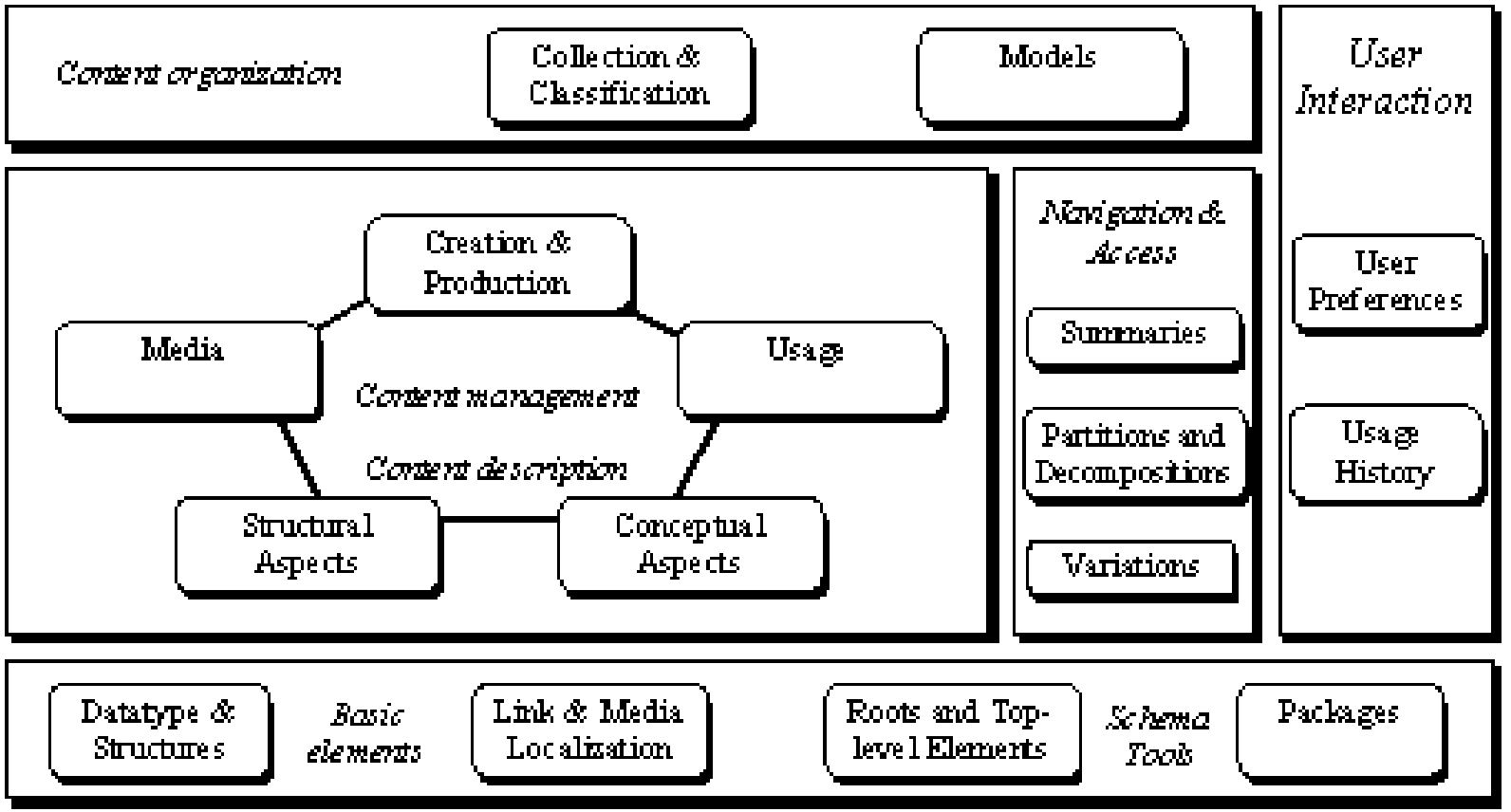
- Face Recognition descriptor—can be used to retrieve face images which match a query face image.
  - The descriptor represents the projection of a face vector onto a set of basis vectors which span the space of possible face vectors. The Face Recognition feature set is extracted from a normalized face image.
  - This normalized face image contains 56 lines with 46 intensity values in each line. The centers of the two eyes in each face image are located on the 24th row and the 16th and 31st column for the right and left eye respectively.

MPEG-7 Multimedia Description Schemes

Organization of MDS tools

- Basic Elements,
- Schema Tools,
- Content Description,
- Content Management,
- Content Organization,
- Navigation and Access, and
- User Interaction

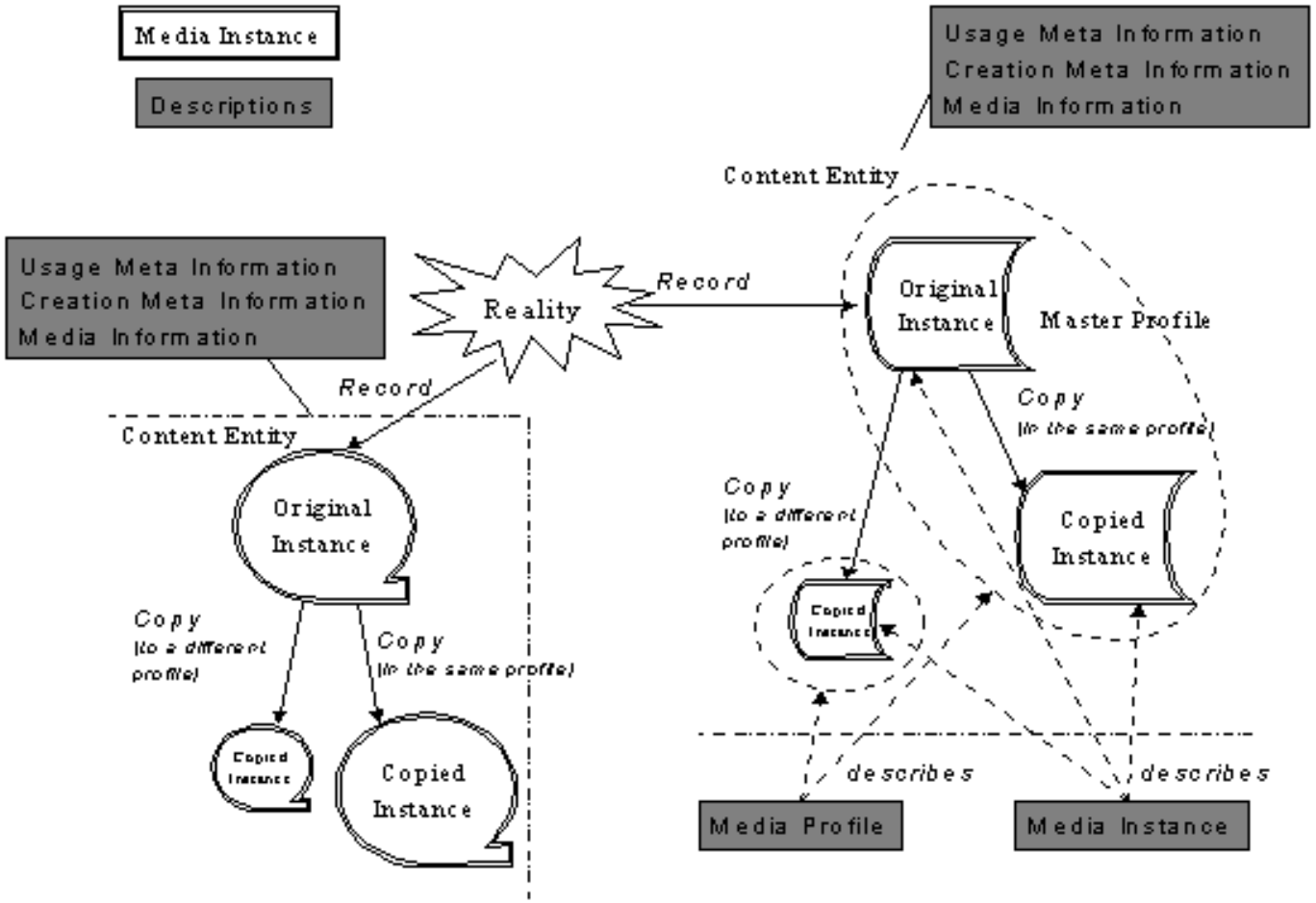
Organization of MDS Tools



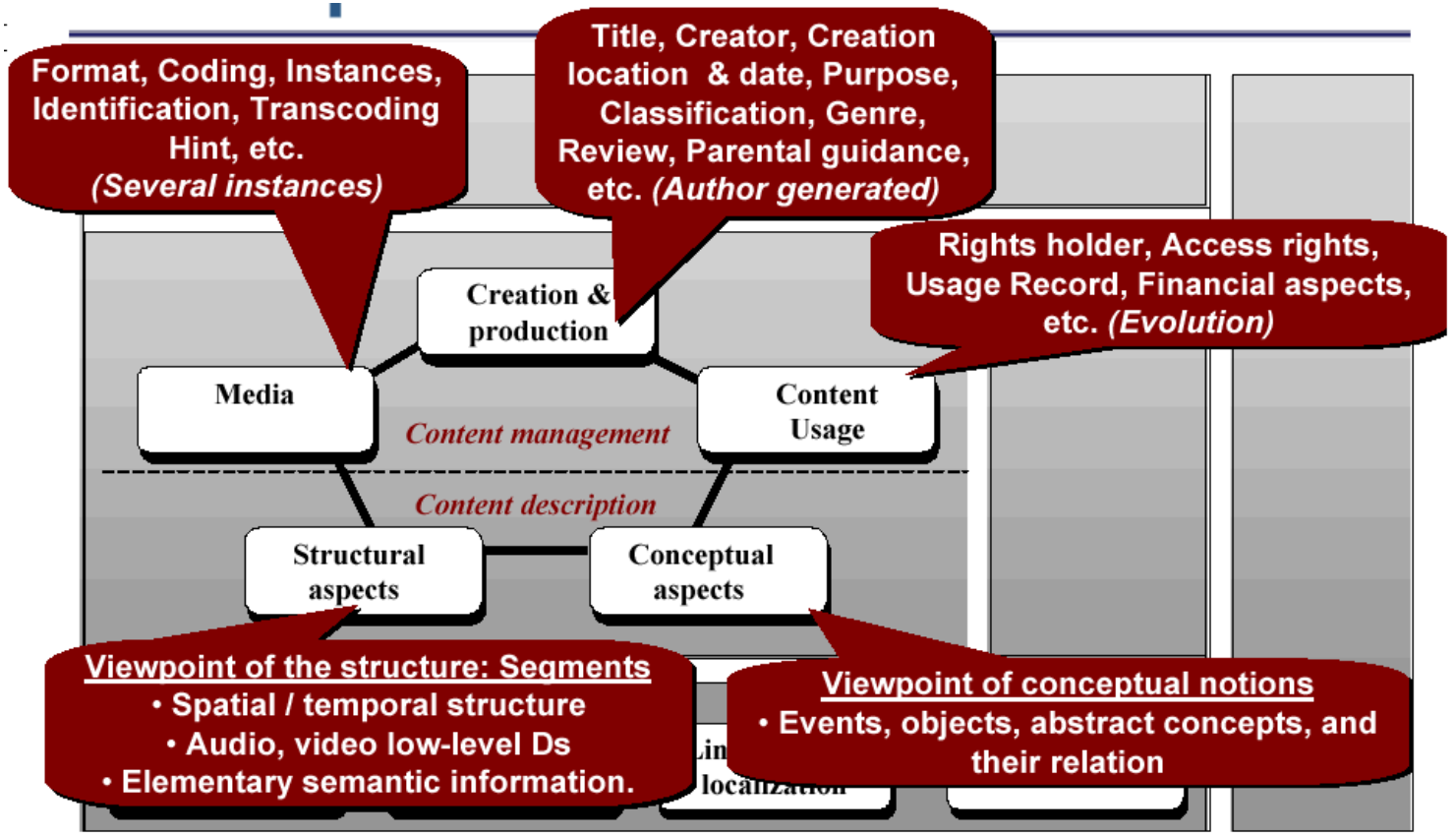
Content Management (CM)

- CM description tools allow the description of the life cycle of the content, from content creation to consumption.
- The content described by MPEG-7 descriptions can be available in different modalities, formats, coding schemes, and there can be several instances.
  - For example, a concert can be recorded in two different modalities: audio and audio-visual. Each of these modalities can be encoded by different encoding schemes. This creates several media profiles.

Content Management



Content Management and Description



Content Description

Description of the content structural aspects

- core element of this part is the *Segment DS* (description of the physical and logical aspects of audio-visual content)
- Segment DSs may be used to form segment trees.
- MPEG-7 also specifies a Graph DS that allows the representation of complex relations between segments. It is used to describe spatio-temporal relationships, between segments that are not described by the tree structures
- A segment represents a section of an audio-visual content item.

Content Description—Segment DS

- *Segment DS* is an abstract class (in the sense of oop). It has five subclasses: AudioVisual Segment DS, Audio Segment DS, Still Region DS, Moving Region DS and Video Segment DS. Therefore, it may have both spatial and temporal properties.
- examples: A *temporal segment* may be a set of samples in an audio sequence, represented by an *Audio Segment DS*, a set of frames in a video sequence, represented by a *Video Segment DS* or a combination of both audio and visual information described by an *Audio Visual Segment DS*. A *spatial segment* may be a region in an image or a frame in a video sequence, represented by a *Still Region DS*.
- a segment can be decomposed into sub-segments through the Segment Decomposition DS.

Segment DS

- A segment is not necessarily connected, but may be composed of several non-connected components.
- Connectivity refers to both spatial and temporal domains. A temporal segment (Video Segment, Audio Segment and AudioVisual Segment) is said to be temporally connected if it is a sequence of continuous video frames or audio samples.
- A spatial segment (Still Region) is said spatially connected if it is a group of connected pixels.
- A spatio-temporal segment (Moving Region) is said spatially and temporally connected if the temporal segment where it is instantiated is temporally connected and if each one of its temporal instantiations in frames is spatially connected (Note: different from the classical connectivity in a 3D space).

Segment Discription

Any segment may be described by

- Creation information
- Usage information
- Media information
- Textual annotation
- Specific features

Specific Features for Segment Discription

Features	Video Segment	Still Region	Moving Region	Audio Segment
Time	X	-	X	X
Shape	-	X	X	-
Color	X	X	X	-
Texture	-	X	-	-
Motion	X	-	X	-
Camera motion	X	-	-	-
Mosaic	X	-	-	-
Audio features	-	-	X	X

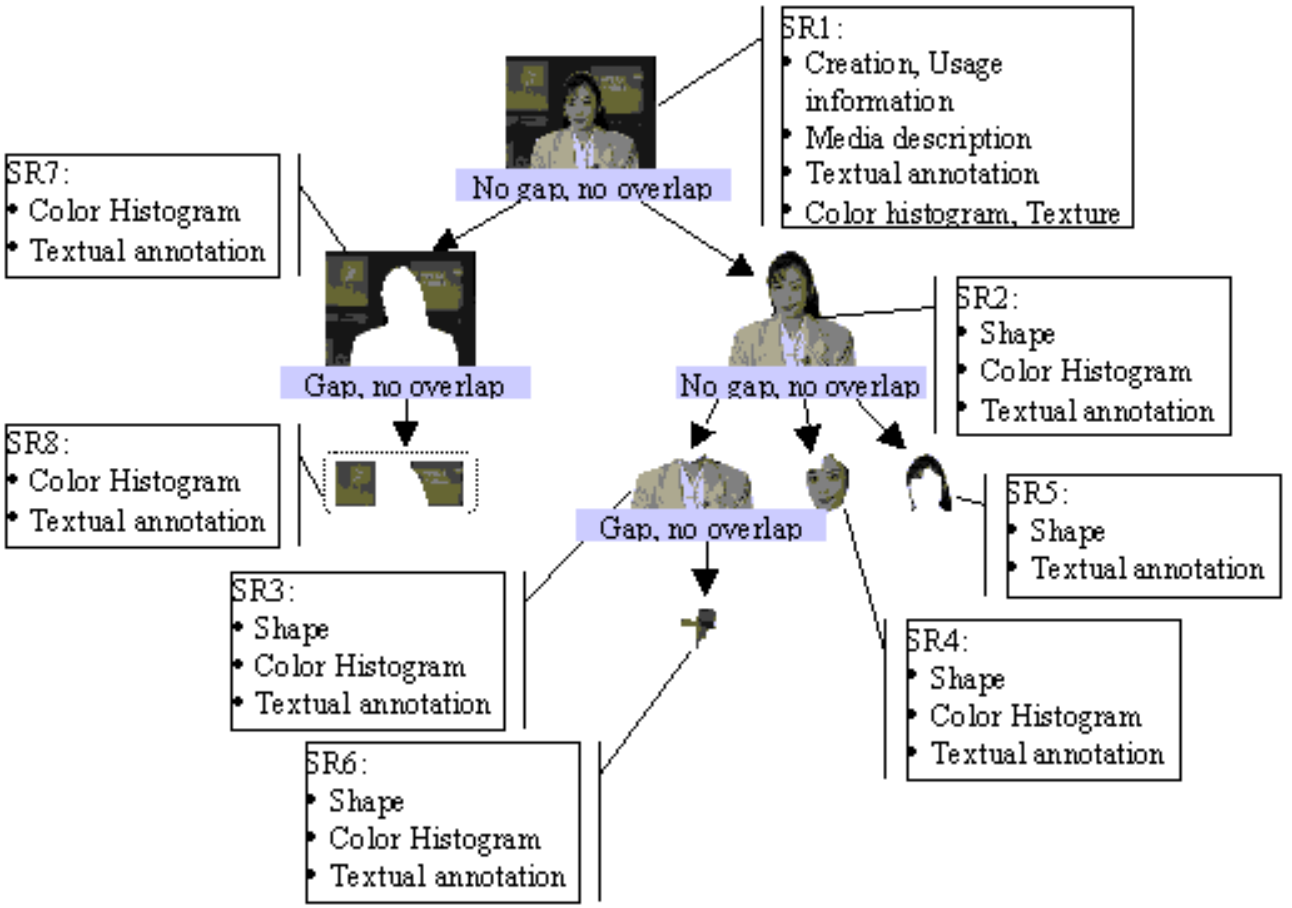
Example of Segment Description

Example — Still image, discribed as a tree

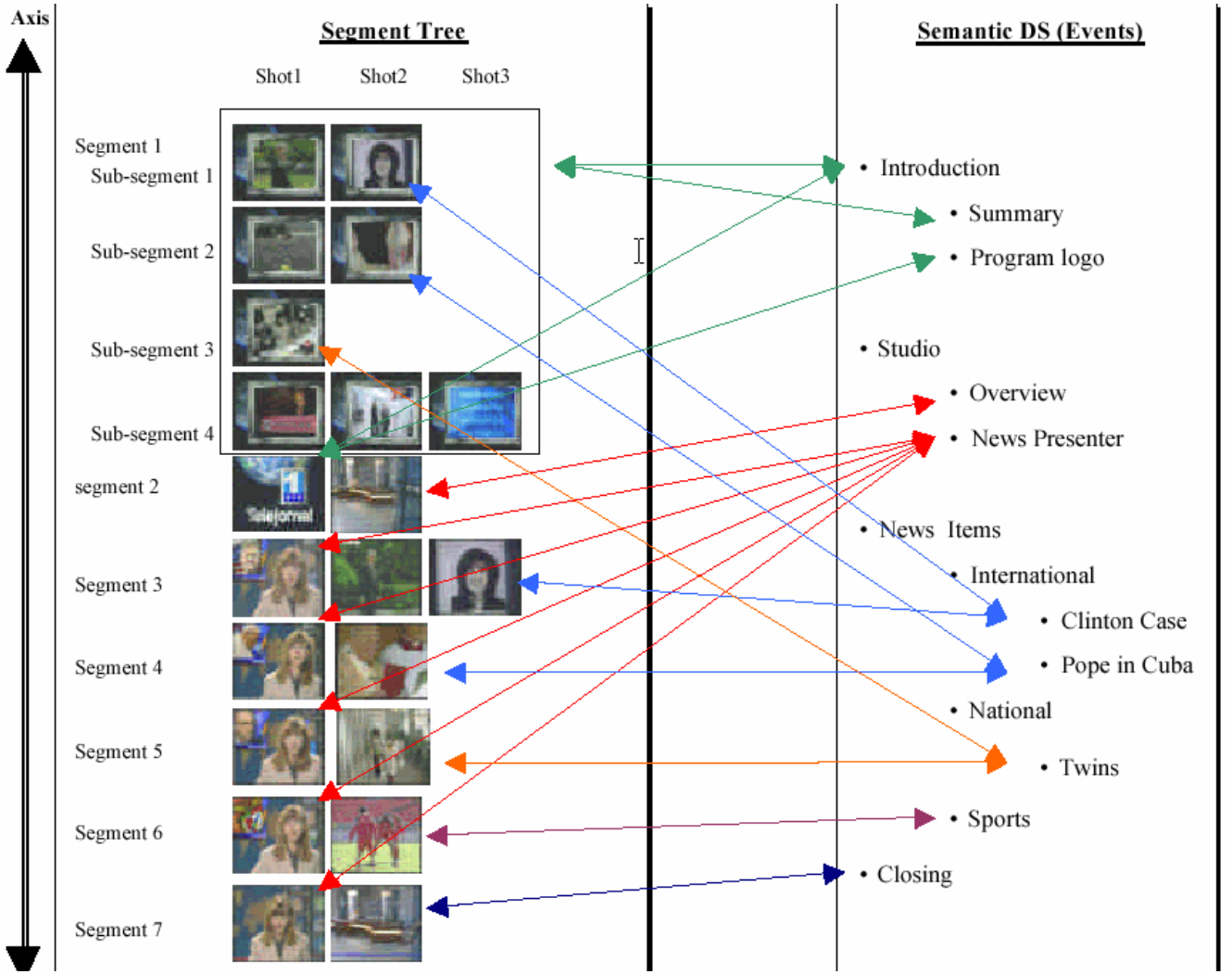
- Original image — discribed as still region SR1: creation (title, creator), usage information (copyright), media information (file format), textual annotation (summarizing the image content), color histogram, texture descriptor
- Segment Tree — composed of 8 still regions, each of these are assigned types of features.

It is not necessary to repeat the creation, usage information and media information in the tree hierarchy, since the children segments are assumed to inherit their parent value

Example of Image Description



Segment Tree



Segment Graph

Hierarchical structures as trees are adequate for efficient access, retrieval and scalable description. But for certain applications they imply constraints that may make them inappropriate. In such cases we use —

- Segment Graph DS; defined by
  - Set of Nodes - representing segments
  - Set of edges - specifying the relationship between the nodes

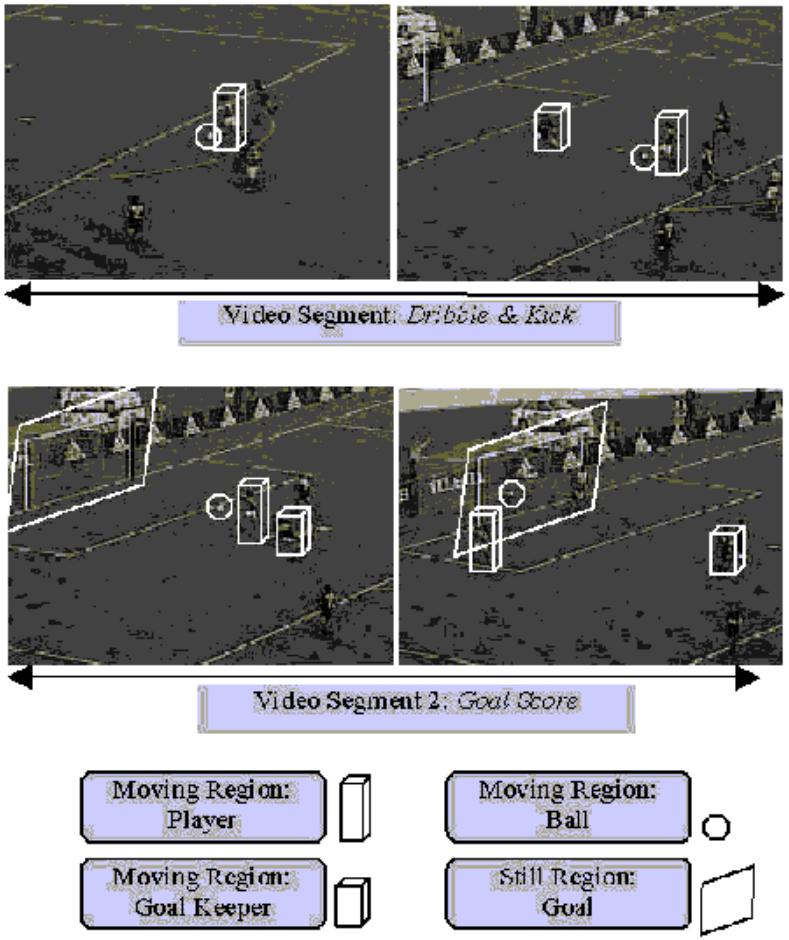
Example of Segment Relationship Graph

Example — Excerpt of a football match

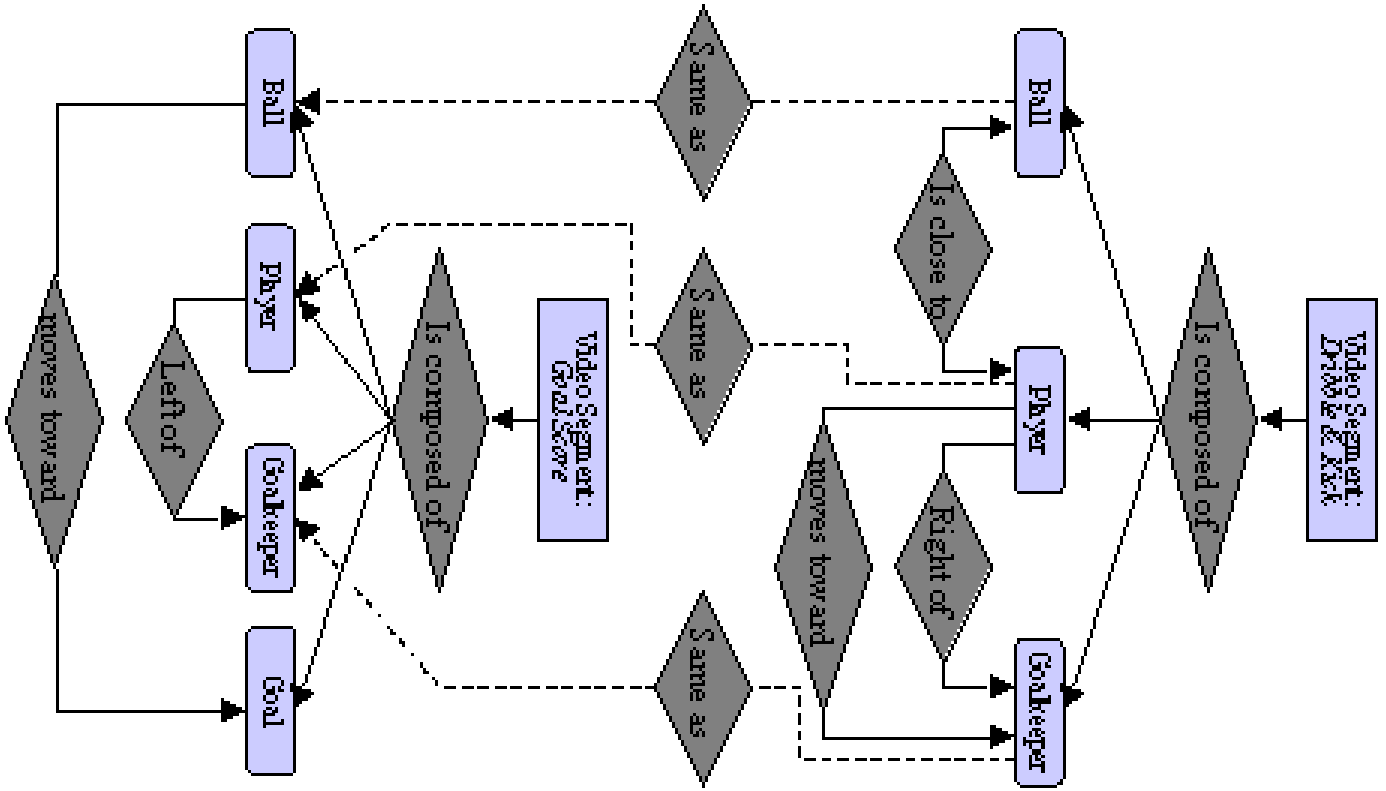
Two Video segments, one still region, three moving regions have been defined

- Video Segment *Dribble&Kick* — involves the moving regions *Player*, *Goalkeeper*, *Ball*. The *Ball* remains close to the *Player* who is moving towards the *Goalkeeper*. The *Player* appears on the *Right of* the *Goalkeeper*.
- Video Segment *Goal Score* — involves the moving regions *Player*, *Goalkeeper*, *Ball* and the still region *Goal*. *Player Left of* the *Goalkeeper*, the *Ball* moves towards the *Goal*.

Example of Segment Relationship Graph



Example of Segment Relationship Graph

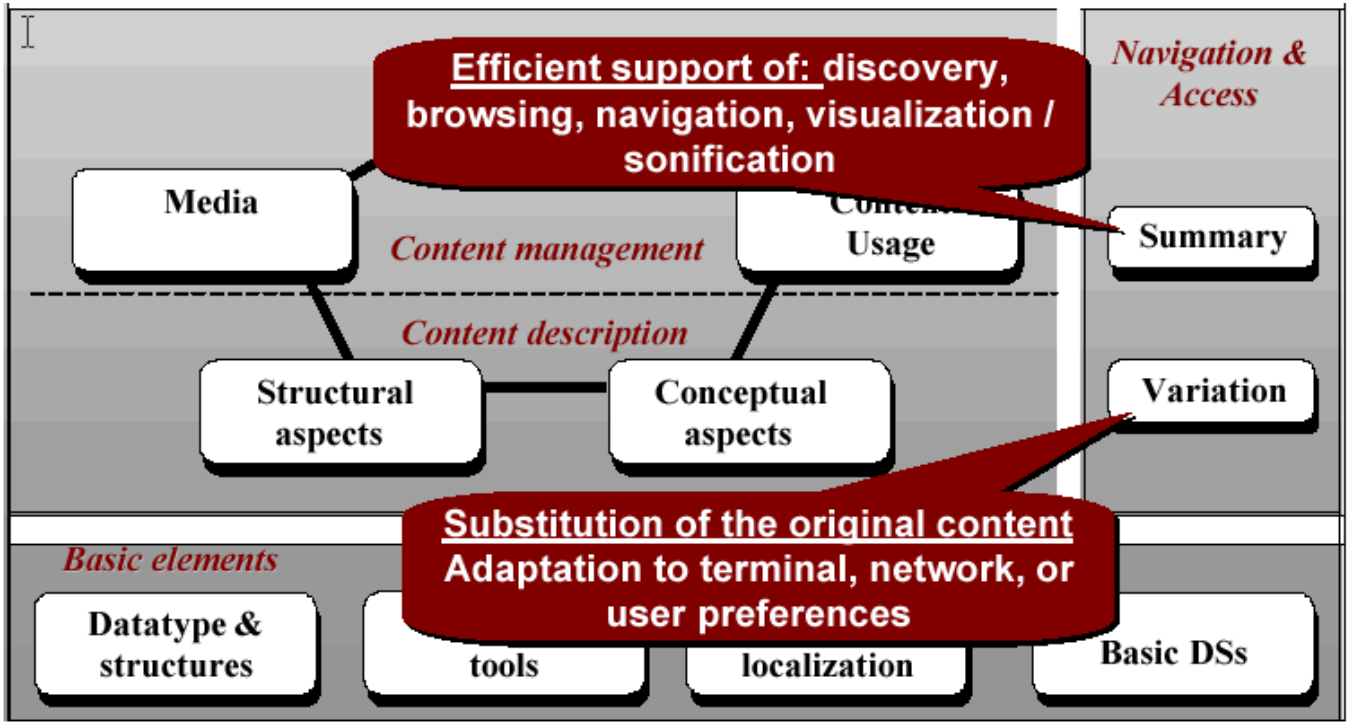


Navigation and Access

MPEG-7 provides DSs that facilitate navigation and access of audio-visual content by specifying

- Summaries — enable efficient browsing, navigation, discovery, visualization, sonification
- Views and Partitions — provide views of the av-data in the space or frequency domain, which allows multi-resolution and progressive access
- Variations — specify the relation between different of av-material, which allows adative selection of the different variations of the content under different delivery conditions

Navigation and Access



Summary

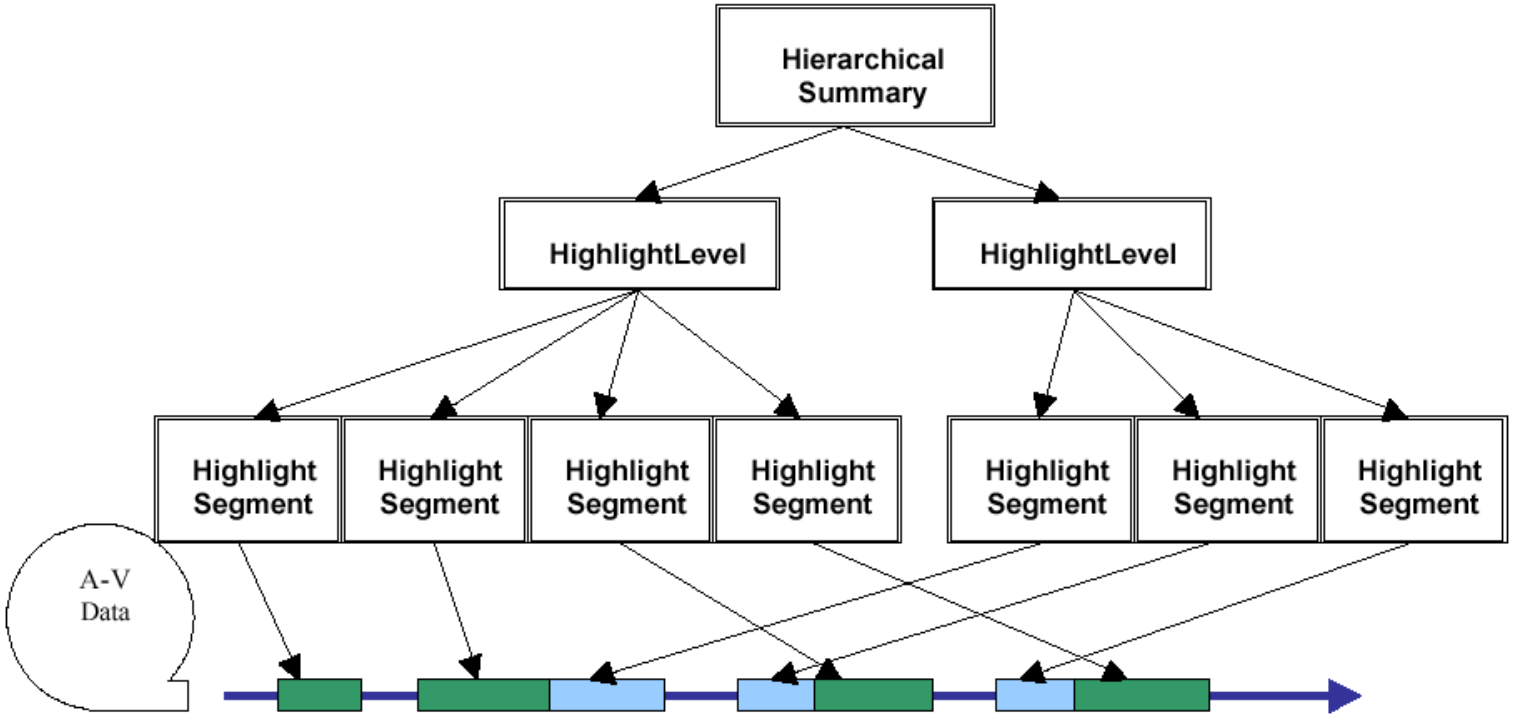
Summary DSs allow the av-content to be navigated in either a hierarchical or sequential fashion —

- Hierarchical Summary
- Sequential Summary

Hierarchical Summary

- Hierarchical Summary DS — organizes the content into successive levels that discribe the av-content at different levels of detail from coarse to fine. The hierarchy forms a tree.
- Highlightlevel DS — specifies the elements of the Hierarchical Summary DS by providing a summary at a particular level of detail.It refers to a sequence of av-segments (image , key-frame, video or audio clip)

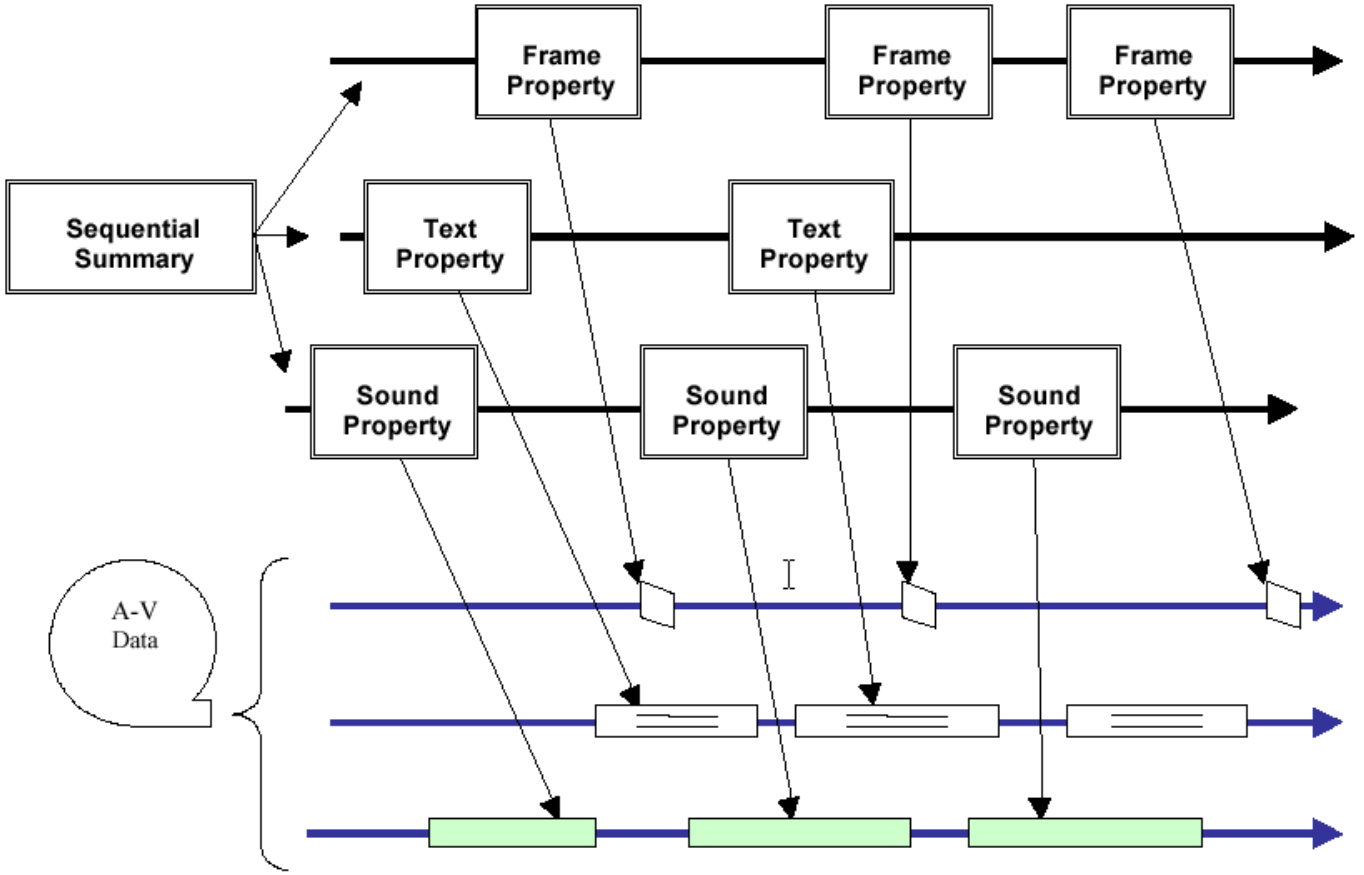
  Example: news program
  - level1 - local news, sports, weather, society...
  - more detailed level - detailed highlights from sport segment corresponding to the events of a football game


Hierarchical Summary
HighlightLevel
HighlightLevel
Highlight Segment
Highlight Segment
Highlight Segment
Highlight Segment
Highlight Segment
Highlight Segment
Highlight Segment
A-V Data

Sequential Summary

- Sequential Summary DS — It specifies a summary consisting of a sequence of audio clips, a sequence of images or video frames, possibly synchronized with audio or text, that compose a slide-show or a audio-visual skim
  - Sequential Summary may be stored separately from the original av-content - allows fast navigation and access.
  - Alternatively, the Sequential Summaries may link directly to the original av-content - reduce storage.

Sequential Summary

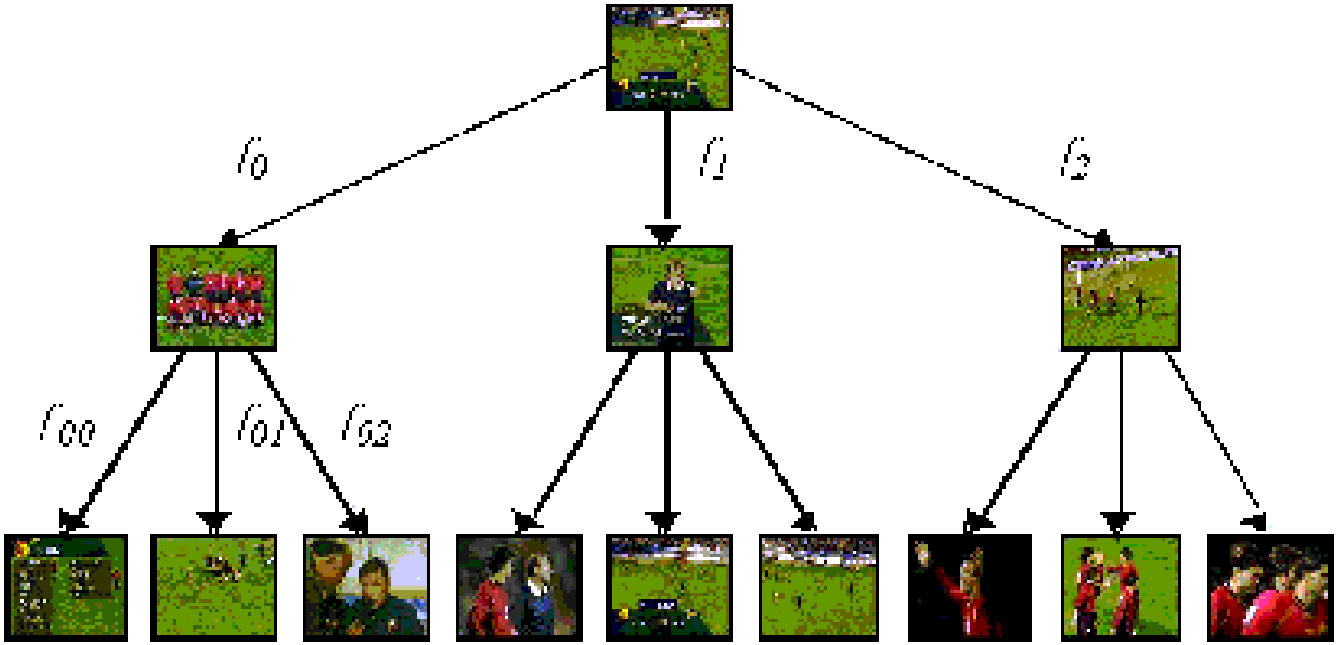


Example: Hierarchical Summary

Example — Football game

- Root — the whole game summarized into a single frame
- Next level — three frames that summarize different segments of the video (team, referee, goal score)
- Bottom level — additional frames, depicting in more detail the scenes depicted in the segments.

Example: Hierarchical Summary



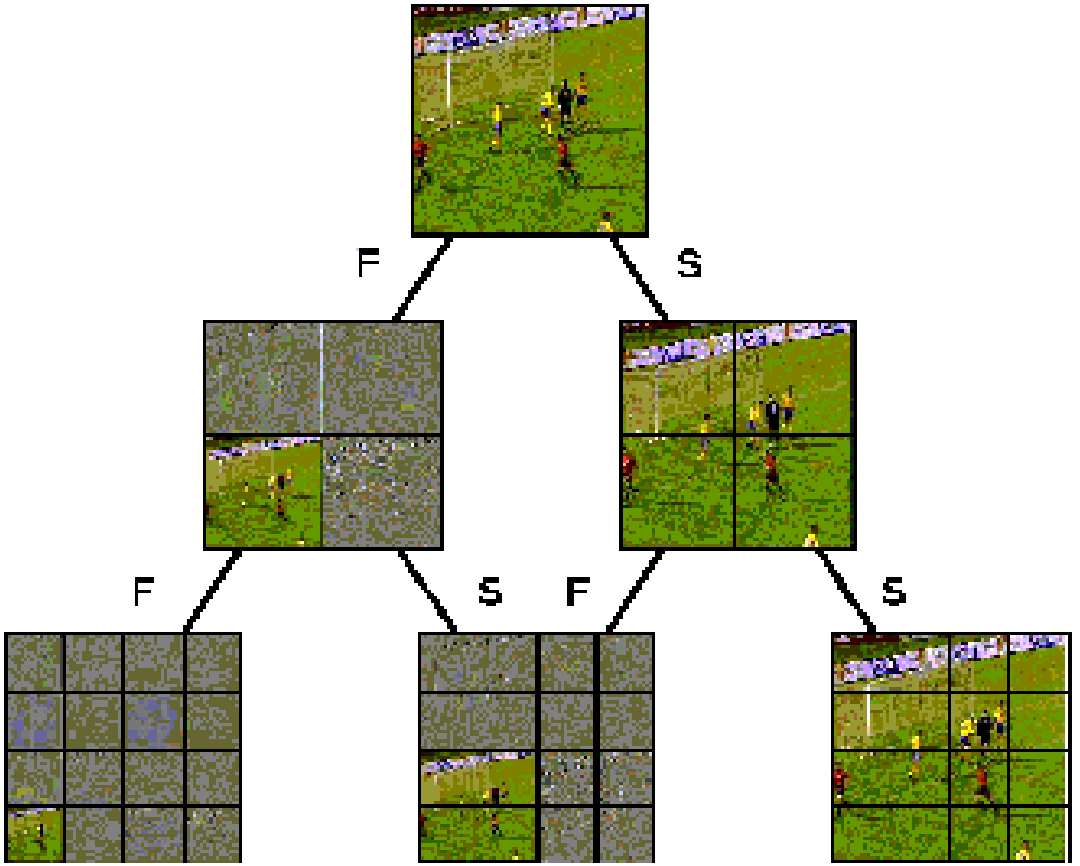
Partitions and Decomposition

Example —The Space and Frequency Graph of an image

Nodes correspond to the different space and frequency views of the image —

- views in space - spatial segments (transitions S)
- views in frequency - wavelet subbands (transitions F)
- views in space and frequency - wavelet subbands of spatial segments (transition S+F)

This kind of decomposition enables efficient multi-resolution access and progressive retrieval of the image data


F
S
F
S
F
S

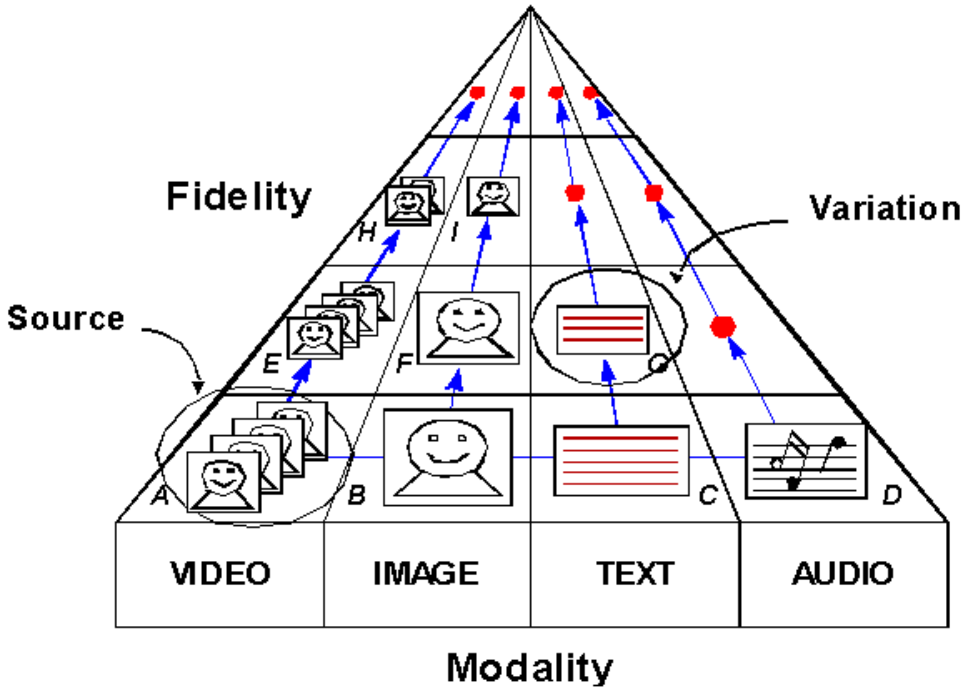
Variation of the Content

a server, proxy or terminal is to select the most suitable variation of av-content (capabilities of terminal devices,network conditions, user preferences.....)

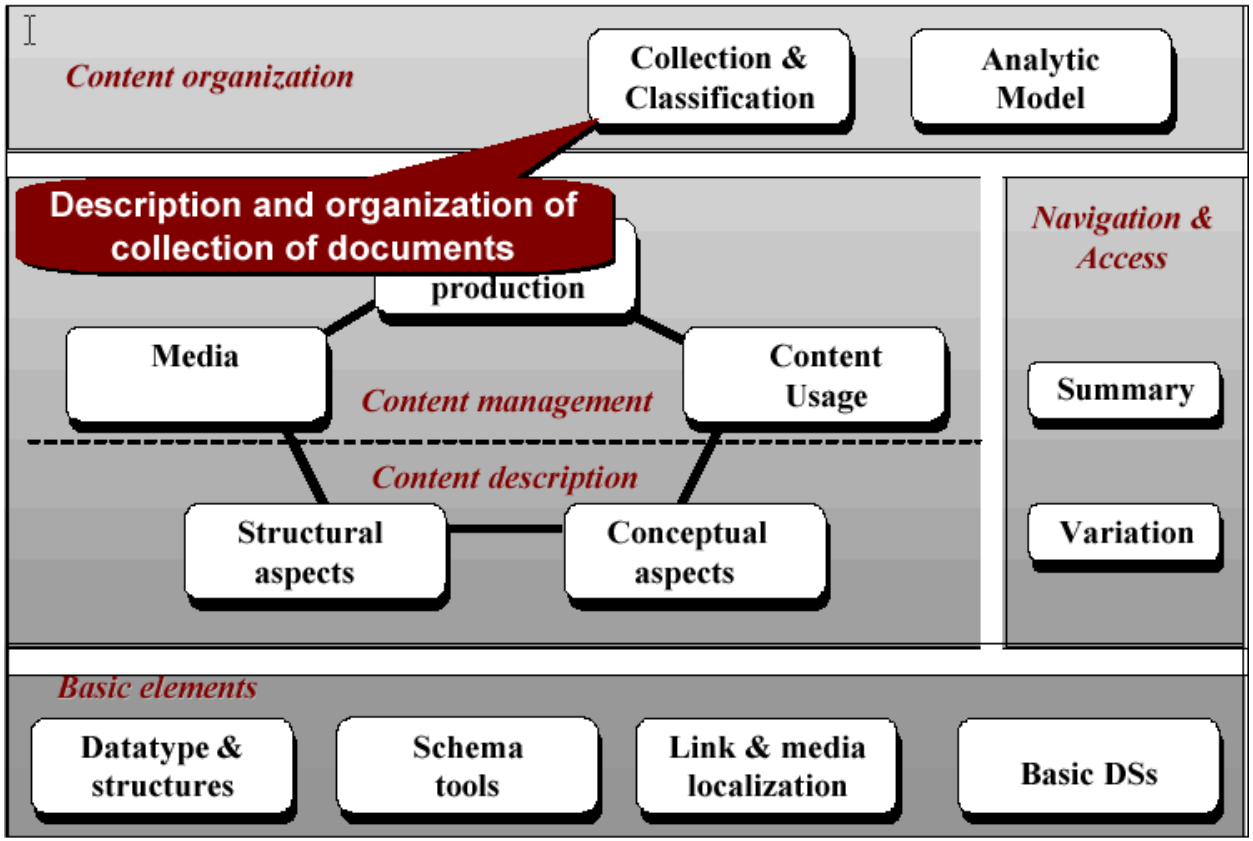
- Variation DS — specifies the different variations of an av-content, such as summaries, compressed or low resolution version, different languages and modalities (audio, video, image, text)
  - Variation fidelity value - quality of variation compared to the original
  - Variation type attribute - summary, abstract, compression, modality translation, color reduction, language translation..

Variation

Universal Multimedia Access: Adapt delivery to network and terminal characteristics (QoS)



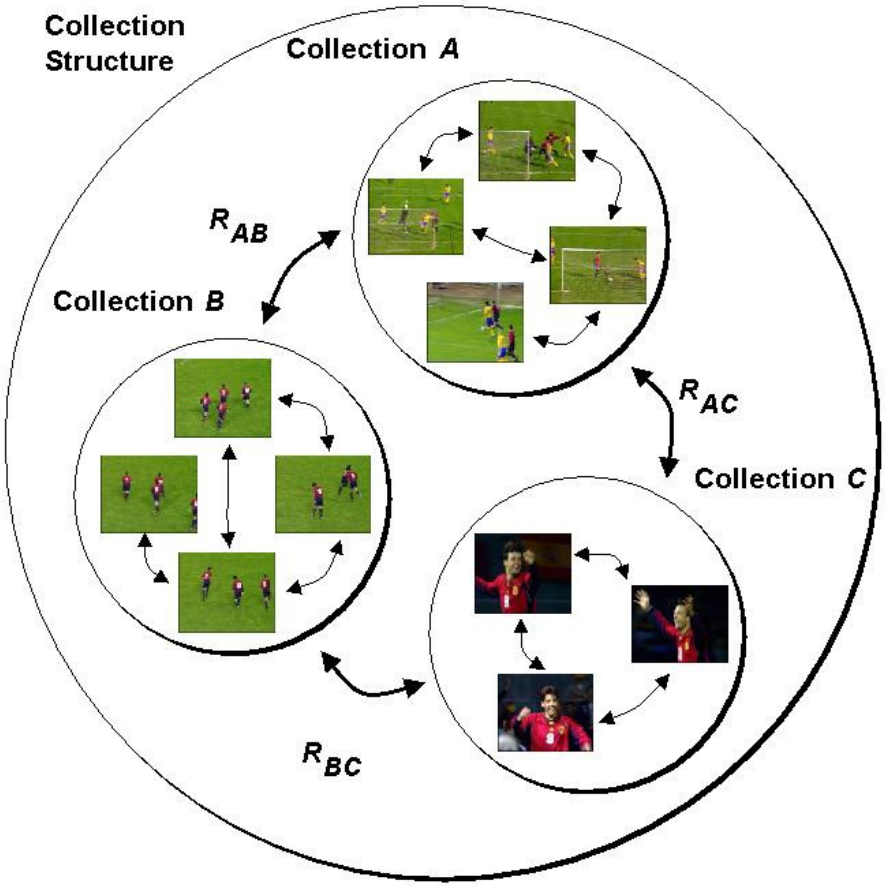
Content Organization



Collection

- Collection Structure DS — groups av-content, segments, events, or objects into collection clusters and specifies properties that are common to the elements;
  discribes relationships among collection clusters
- Example: Football game —
  each collection consists of a set of images with common properties (similiar events in the game)
  - relationships within each collection - degree of similarity of images
  - relationships across the collections - degree of similarity of collections

Collection


Collection Structure
Collection A
R_{AB}
Collection B
R_{AC}
Collection C
R_{BC}

User Interaction

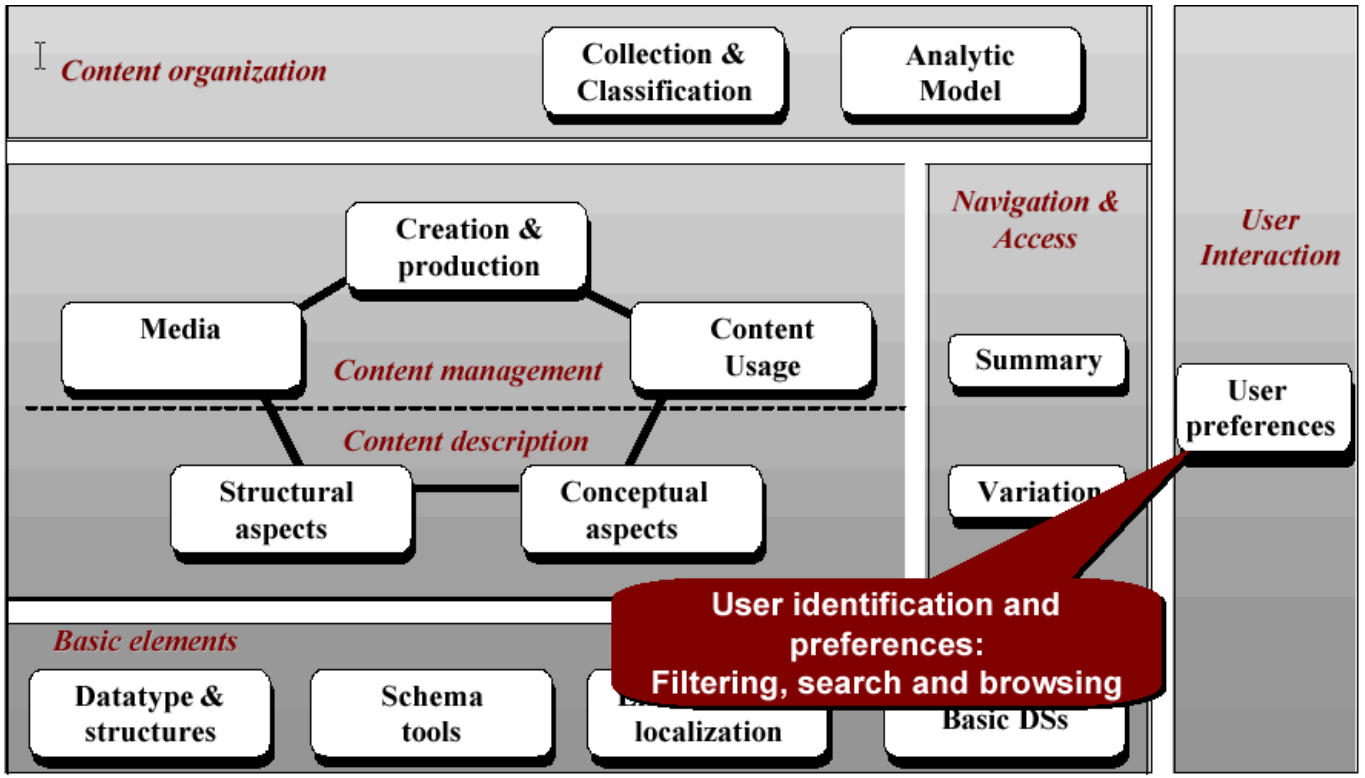
User Preference DSs

MPEG-7 av-content descriptions can be matched to the User-Preference DS descriptions in order to select and personalize av-content for more efficient access, presentation and consumption

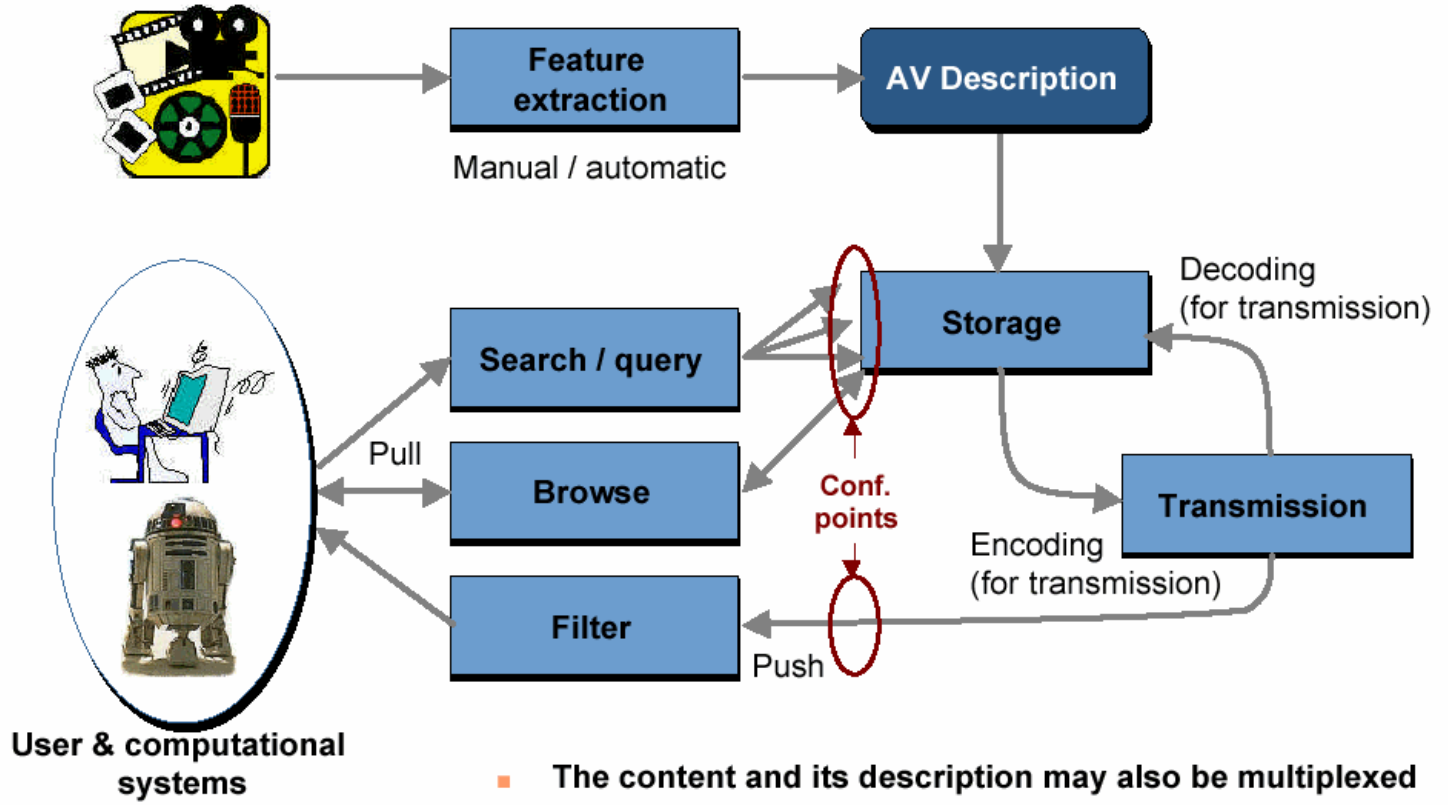
UserPreference DS allows —

- specification of preferences for different types of content and modes of browsing,including context dependency in terms of time and place
- weighting of the relative importance of different preferences
- specification of the privacy characteristics of preferences

User Interaction



Summary—Information Flow



- The content and its description may also be multiplexed

Summary—Further Information

- major MPEG-7 documents are public:
  www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm

Part VI MPEG-4

- Principles
- Architecture
- Systems
- Video
- Audio

Motivation

motivation

a blurring of borders between three distinct service models: communications, interactivity and broadcasting;

objective

- to standardize algorithms for audiovisual coding in MM applications,
- allowing for interactivity, high compression, scalability of video and audio content and
- support for natural and synthetic audio video content

MPEG-4 Overview

- formally ISO/IEC international standard 14496
- defines a *multimedia system* for interoperable communication of complex scenes containing audio, video, synthetic audio, and graphics material.
- combines some typical features of other MPEG standards,
- but aims to provide a set of technologies to satisfy the needs of *authors*, *service providers,* and *end users*.

MPEG-4 Overview

- For authors, MPEG-4 enables the production of content with greater *reusability and flexibility*; also, it permits better management and protection of content owner rights.
- For network service providers, MPEG-4 offers transparent information, interpreted and translated into the appropriate native signaling messages of each network.
- For end users, MPEG-4 enables many functionalities potentially accessible on a single compact terminal and higher levels of interaction with content, within the limits set by the author.

MPEG-4 Principles

- *Audio-visual scenes* made of audio-visual objects composed together according to a scene description:
  - allows interaction with elements within the audio-visual scene,
  - coding scheme can differ for individual objects,
  - allows easy reuse of audio-visual content.

MPEG-4 Principles

- Audio-visual objects can be of different nature:
  - audio (single or multi-channel) or video (arbitrary shape or rectangular),
  - natural (natural audio or video) or synthetic (text & graphics, animated faces, synthetic music),
  - 2D (Web like pages) or 3D (spatialized sound, 3D virtual world),
  - streamed (video movie) or downloaded (audio jingle).

MPEG-4 Principles

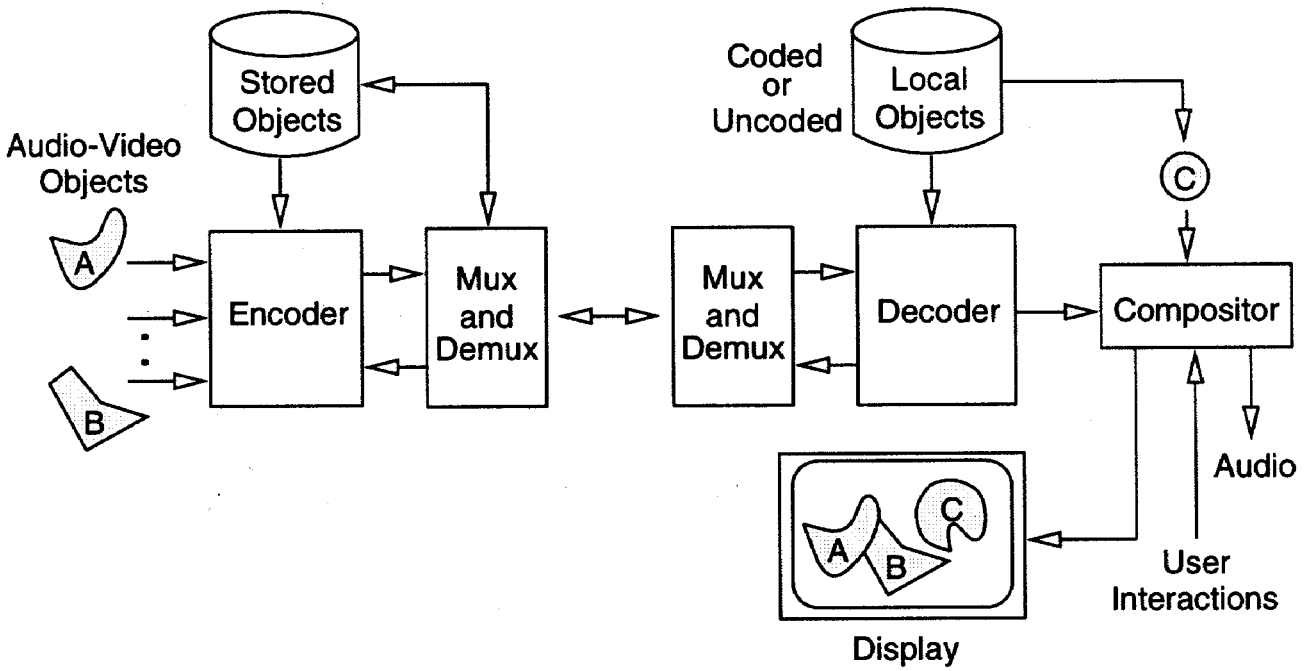
- Scene Description provides:
  - the spatial/temporal relationship between the audiovisual objects (2D, 3D, mixed 2D and 3D scene description),
  - the behavior and interactivity of the audio-visual objects and scenes,
  - protocols to modify and animate the scene in time
- These principles are independent of the bit rate.
- All this information is delivered in a compressed format.

MPEG-4 Principles

MPEG-4 provides

- Coding—representing units of audio, visual, or audiovisual content, called media objects.
- Composition—describing the composition of these objects to create compound media objects that form audiovisual scenes.
- Multiplex—multiplexing and synchronizing the data associated with media objects for transport over network channels providing appropriate QoS.
- Interaction—interacting with the audiovisual scene at the receiver's end or, via a back channel, at the transmitter's end.

MPEG-4 Architecture



MPEG-4 Standard Structure

- Systems
- Visual
- Audio
- Conformance Testing
- Reference Software
- Delivery Multimedia Integration Framework (DMIF).

MPEG-4 Systems

- The Systems subgroup defined the framework for integrating the natural and synthetic components of complex multimedia scenes.
- The Systems level integrates the elementary decoders for media components specified by other MPEG-4 subgroups
- provides the specification for the parts of the system related to composition and multiplex.
  - Composition information consists of the representation of the hierarchical structure of the scene.
  - composition of elementary media objects inspired by the existing Virtual Reality Modeling Language (VRML)

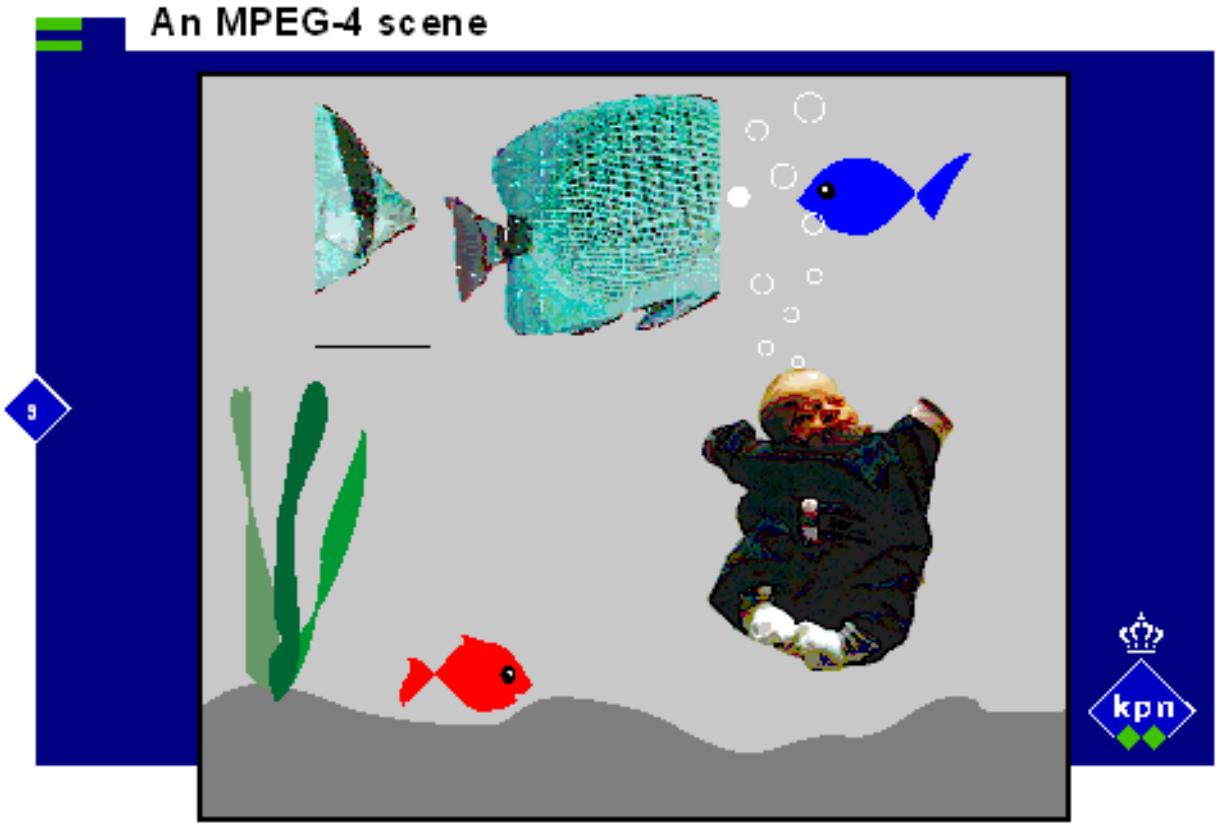
MPEG-4 Systems—Composition

- model to describe the composition of a complex multimedia scene relies on the concepts VRML
- main areas featuring new concepts
  - dealing with 2D-only content, for a simplified scenario where 3D graphics is not required;
  - interfacing with streaming media (video, audio, streaming text, streaming parameters for synthetic objects); and
  - adding synchronization capabilities.
- authors can generate this description in textual format, possibly through an authoring tool.

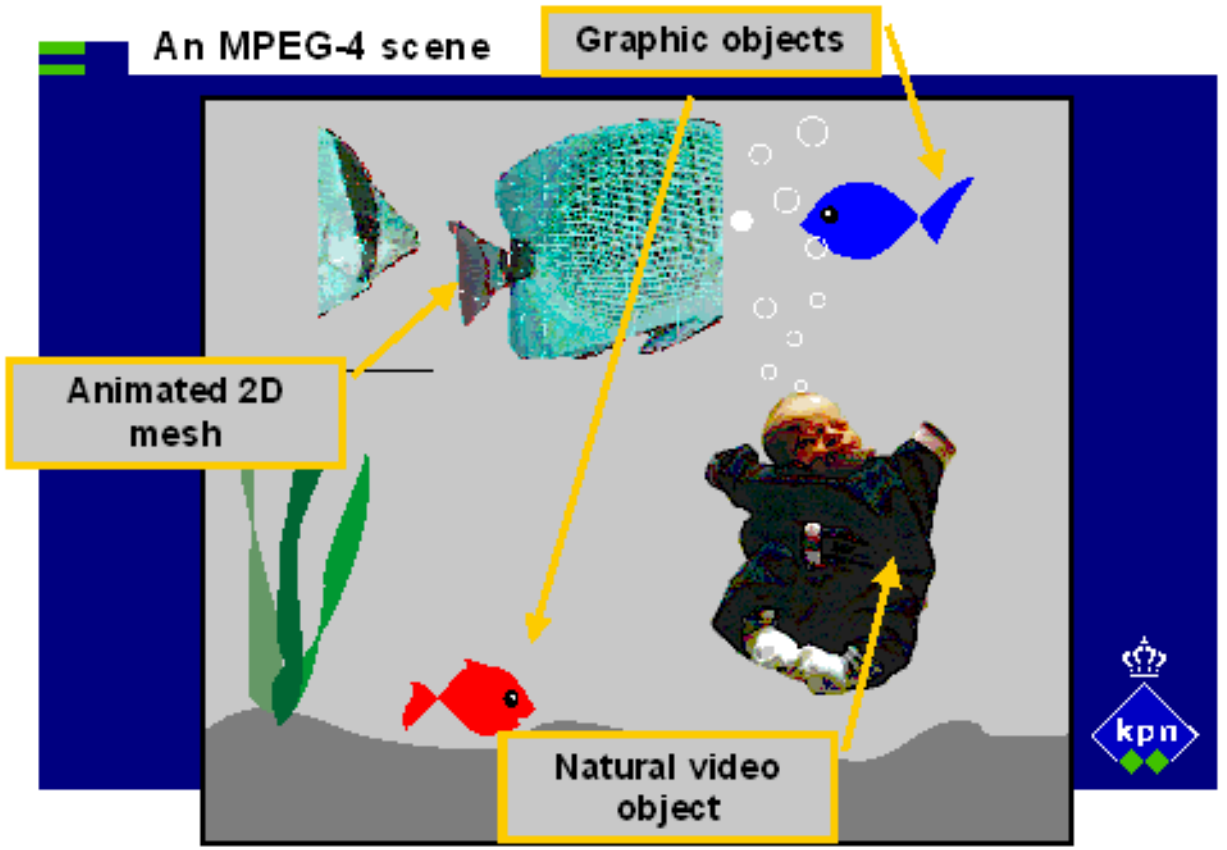
MPEG-4 Systems—Composition

- For efficiency, the standard defines a way to encode the scene description in a binary representation—Binary Format for Scene Description (BIFS).
- Multimedia scenes are conceived as hierarchical structures represented as a graph. Each leaf of the graph represents a media object.
- The initial snapshot of the scene is sent or retrieved on a dedicated stream.
- All the nodes and graph leaves that require streaming support to retrieve media contents are logically connected to the decoding pipelines.

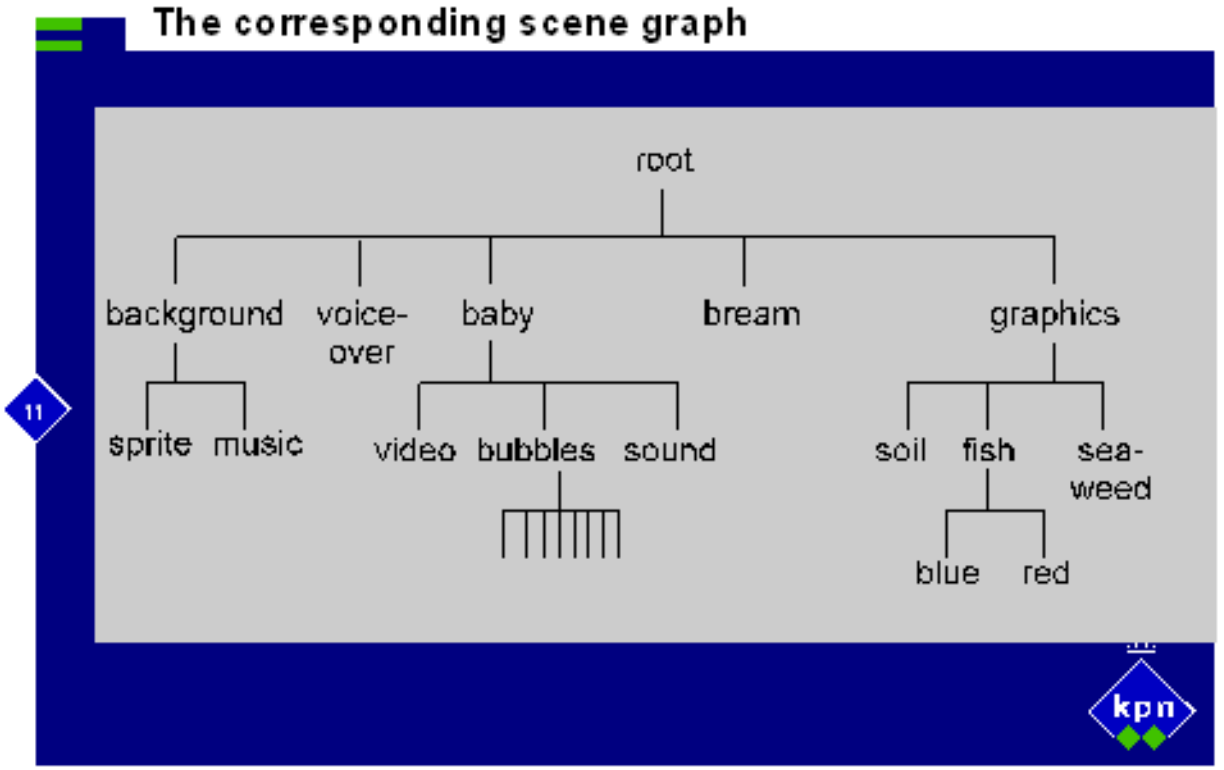
Example Scene



Example Scene



Scene Graph


The corresponding scene graph
root
background
voice-over
baby
bream
graphics
sprite
music
video
bubbles
sound
soil
fish
sea-weed
blue
red
11
kpn

Spatial Composition

- The *composition stream* is treated differently from others because it provides the information required by the terminal to set up the scene structure and map all other elementary streams to the respective media objects.
- Spatial relationships—Each elementary media object is represented by a leaf in the scene graph and has its own local coordinate system. The mechanism to combine the scene graph's nodes into a single global coordinate system uses spatial transformations associated to the intermediate nodes, which group their children together.

Temporal Composition

- The composition stream (BIFS) has its own time base.
- Even if the time bases for the composition and for the elementary data streams differ, they must be consistent except for translation and scaling of the time axis.
- Time stamps attached to the elementary media streams specify at what time the access unit for a media object should be ready at the decoder input (*DTS, decoding time stamp*), and at what time the composition unit should be ready at the compositor input (*CTS, composition time stamp*).
- Time stamps associated to the composition stream specify at what time the access units for composition must be ready at the input of the composition information decoder.

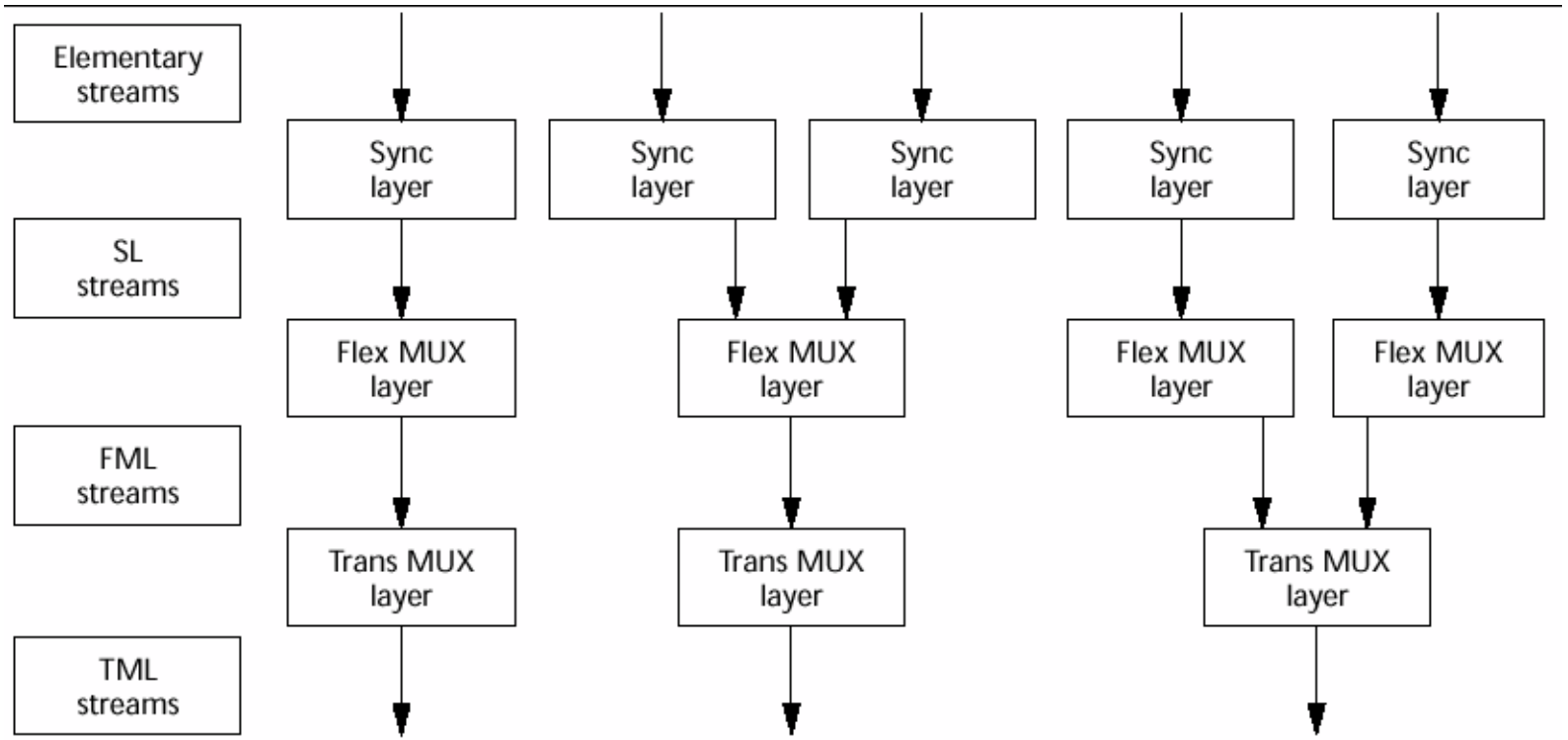
Temporal Composition

- In addition to the time stamps mechanism (derived from MPEG-1 and MPEG-2), fields within the scene description also carry a time value. They indicate either a *duration in time* or an *instant in time*.
  - represent a relative time with respect to the time stamp of the BIFS elementary stream.
  - For example, the start of a video clip represents the relative offset between the composition time stamp of the scene and the start of the video display.

Multiplex

- Because MPEG-4 is intended for use on a wide variety of networks with widely varying performance characteristics, it includes a three-layer multiplex separating the functionality of
  - adding MPEG-4-specific information for timing and synchronization of the coded media (synchronization layer);
  - multiplexing streams with very different characteristics, such as average bit rate and size of access units (flexible multiplex layer); and
  - adapting the multiplexed stream to the particular network characteristics in order to facilitate the interface to different network environments (transport multiplex layer).

Multiplex Structure



Multiplex Structure

- Elementary streams are packetized, adding headers with timing information (clock references) and synchronization data. They make up the *synchronization layer* (SL) of the multiplex.
- Streams with similar QoS requirements are then multiplexed on a content multiplex layer, termed the *flexible multiplex layer* (FML). It interleaves data from a variable number of variable bit-rate streams.
- A service multiplex layer, known as the transport *multiplex layer* (TML), can add a variety of levels of QoS and provide framing of its content and error detection

Synchronization Layer

- The header attached by sl contains fields specifying:
  - *Sequence number*, *Instantaneous bit rate*, *OCR (object clock reference), DTS (decoding time stamp), CTS (composition time stamp).*
- The information contained in the SL headers maintains the correct time base for the elementary decoders and for the receiver terminal, plus the correct synchronization in the presentation of the elementary media objects in the scene.
- The clock references mechanism supports timing of the system, and the mechanism of time stamps supports synchronization of the different media.

Flexible multiplex layer

- wide range of possible bit rates associated to the elementary streams—ranging from 1 Kbps to more than 100 Mbps
  - intermediate multiplex layer provides more flexibility
- The intermediate (optional) flexible multiplex layer provides a way to group together several low-bit-rate streams for which the overhead associated to a further level of packetization is not necessary or introduces too much redundancy
- With conventional scenes, like the usual audio plus video, this optional multiplex layer can be skipped;

Transport multiplex layer

- The multiplex layer closest to the transport level depends on the specific transmission or storage system on which the coded information is delivered.
- The Systems part of MPEG-4 doesn't specify the way SL packets (when no FML is used) or FML packets are mapped on TML packets. The specification simply references several different transport packetization schemes.
- The "content" packets may be transported directly using e.g. an ATM Adaptation Layer 2 (AAL2) scheme, an MPEG-2 transport stream packetization or transport control TCP/IP.

MPEG-4 Video

- MPEG-4 Video supports:
  - *Content-based interactivity*
    - content-based multimedia data access tools,
    - content-based manipulation and bit-stream editing,
    - hybrid natural and synthetic data coding, and
    - improved temporal random access.
  - *Compressio*n.
    - improved coding efficiency and
    - coding of multiple concurrent data streams.
  - *Universal access*
    - robustness in error-prone environments and
    - content-based scalability.

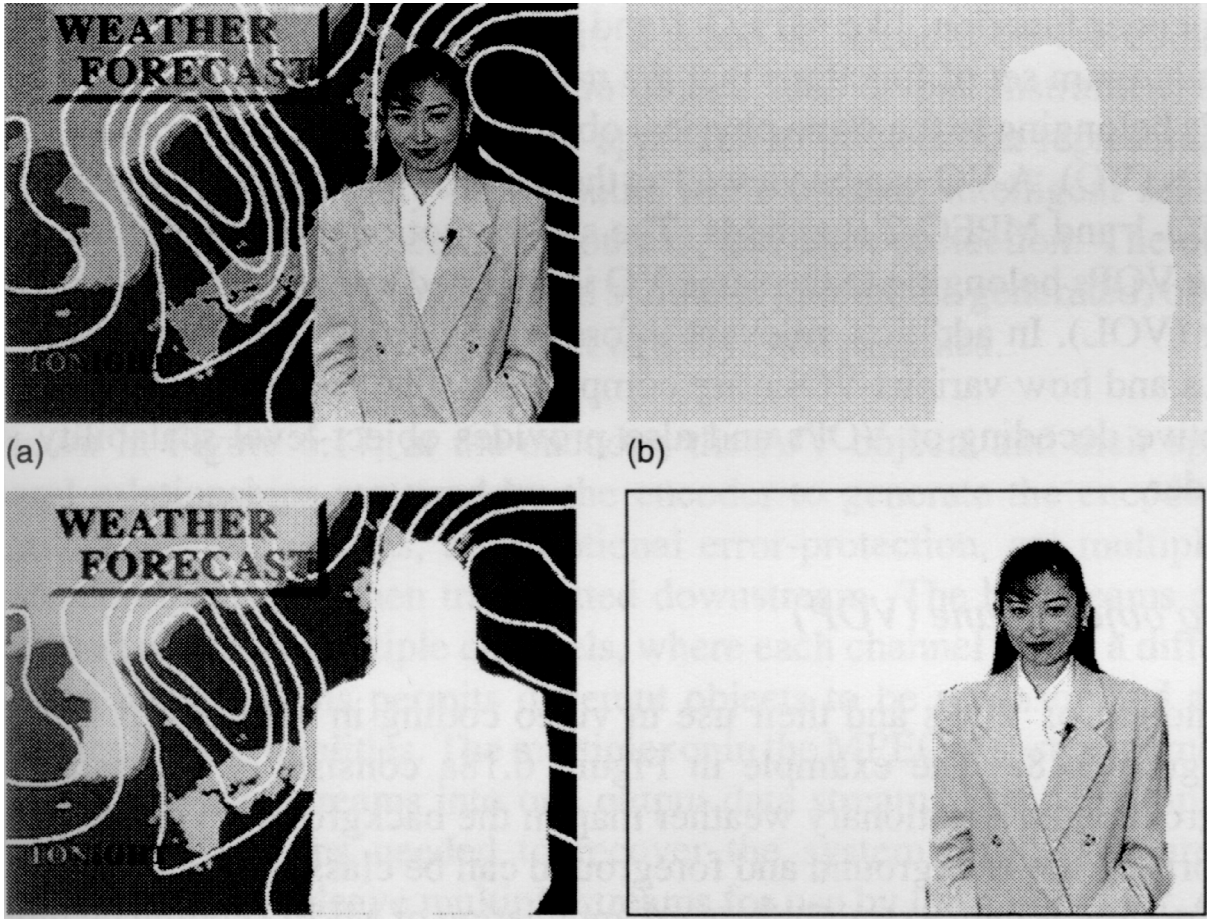
Video Codec Structure

- syntax allows coding of *rectangular* as well as *arbitrarily shaped video objects* in a scene and supports both nonscalable and scalable coding.
- The scalability syntax enables reconstructing useful video from pieces of a bit stream by structuring the total bit stream in two or more layers, starting from a *stand-alone base layer* and adding a number of *enhancement layers*.
- The ability to access individual objects requires achieving a coded representation of their shape. A natural video object consists of a sequence of 2D representations (VOPs).

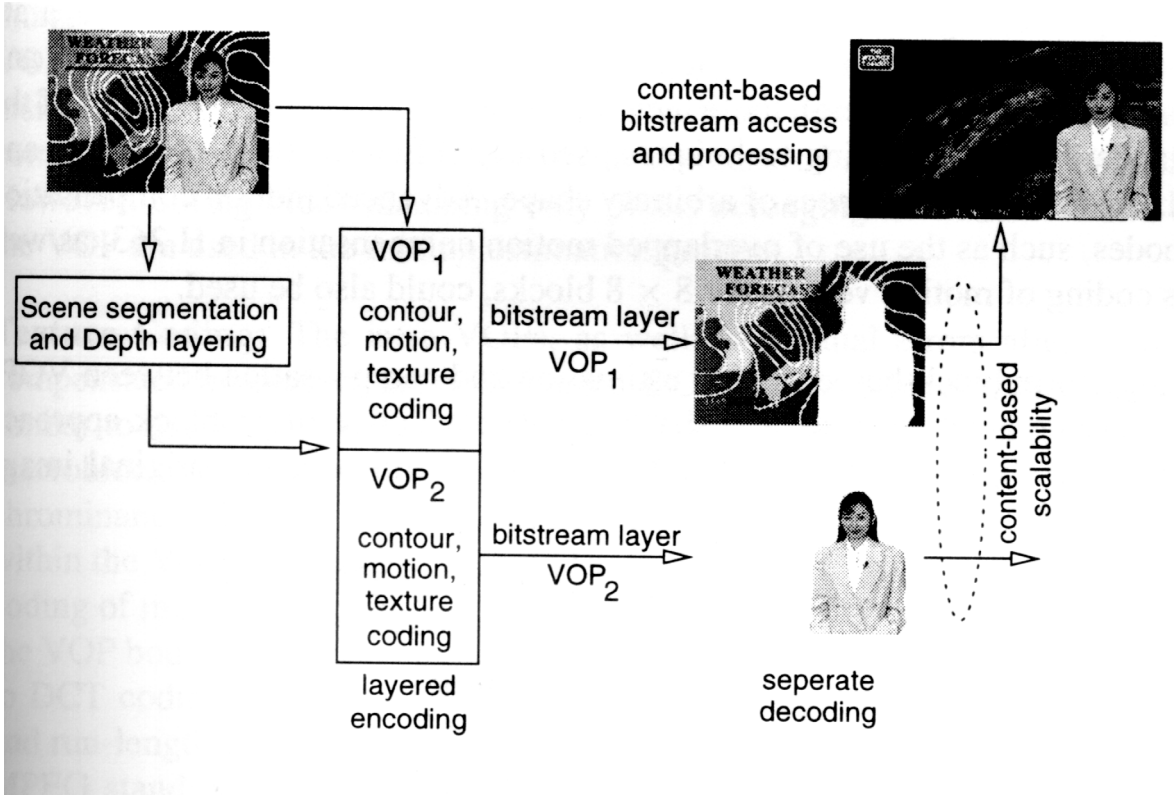
MPEG-4 Video Coding

- each video frame is segmented into a number of arbitrary shaped regions, called video object planes (VOP).
- successive VOPs belonging to the same physical object in a scene are referred to as video objects (VO).
- the shape, motion and texture information of the VOPs belonging to the same VO is encoded into a separate video object layer (VOL).
- relevant information needed to identify each of the VOLs and how various VOLs are composed is also encoded. This allows for selective decoding and provides object-level scalability.

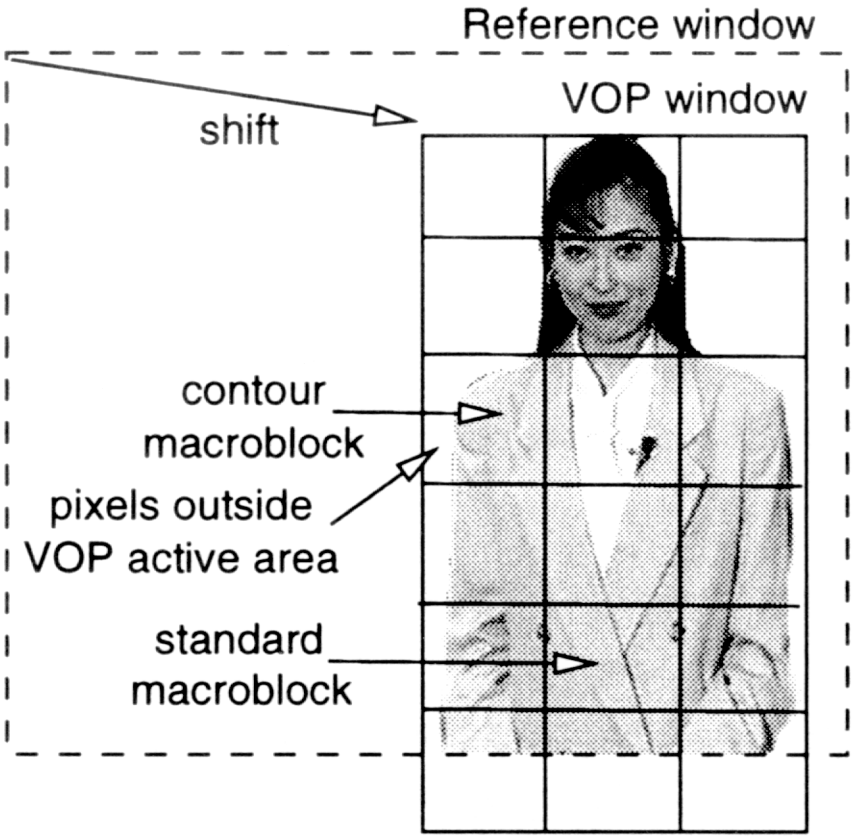
Video Object Plane (VOP)



VOP Coding Process



Macroblock Grid for MC



Interactivity

- important feature: approach doesn't explicitly define a temporal frame rate; encoder and decoder can function in different frame rates, which don't even need to stay constant
- Interactivity between user and encoder or decoder:
  - at the encoding level, e.g. in coding control to distribute the available bit rate between different video objects or to influence the multiplexing to change parameters such as the composition script at the encoder.
  - In case no back channel is available, or when the compressed bit stream already exists, the user may interact with the decoder by acting on the compositor to change either the position of a video object, its display depth order or by requesting the processing of a portion of the bit stream only.

Media Integration of Text and Graphics

- MITG provides a Layout node to specify the placement, spacing, alignment, scrolling, and wrapping of scene objects.
- *Still images* or *video objects* can be placed in a scene graph in many ways, and they can be texture-mapped on any 2D object.
- The most common way, though, is to use the Bitmap node to insert a rectangular area in the scene in which pixels coming from a video or still image can be copied.
- The 2D scene graphs can contain *audio sources* by means of the Sound2D nodes.
- *Text* can be inserted in a scene graph by the Text node. Text characteristics (font, size, style, spacing, and so on) can be customized by means of the Font Style node.

Media Integration of Text and Graphics

- *3D graphics*—BIFS 3D nodes—an extension of the ones defined in VRML—allow the creation of virtual worlds.
- Like in VRML, it's possible to add behavior to objects through Script nodes.
- MPEG-4 allows the creation of much more complex scenes than VRML, of 2D/3D hybrid worlds where contents are not downloaded once but can be streamed to update the scene continuously.

Face Animation

- Focuses on parameters for face animation and definition. It has a very tight relationship with hybrid scalable text-to-speech synthesis for speech-driven avatars.
- the face animation work is the first attempt to define in a standard way the sets of parameters for synthetic anthropomorphic models.
- Face animation is based on the development of two sets of parameters:
  - facial animation parameters (FAPs) and
  - facial definition parameters (FDPs).

Face Animation

- FAPs allow having a single set of parameters regardless of the face model used
- Most FAPs describe atomic movements of facial features; others (expressions and visemes) define much more complex deformations.
- Visemes define the position of the mouth (lips, jaw, tongue) associated with phonemes.
- In the context of MPEG-4, the expressions mimic the facial expressions associated with human primary emotions like joy, anger, fear, surprise, sadness, and disgust.
- Animated avatars' animation streams fit very low bit-rate channels (about 4 Kbps).

Face Animation / Texture Coding

- FAPs can be encoded either with arithmetic encoding or with discrete cosine transform (DCT).
- FDPs are used to calibrate (that is, modify or adapt the shape of) the receiver terminal default face models or to transmit completely new face model geometry and texture.
- Texture coding—MPEG-4 supports an ad-hoc tool for encoding textures and still images based on a wavelet algorithm that provides spatial and quality scalability, content-based (arbitrarily shaped) object coding, and very efficient data compression over a large range of bit rates.
- Texture scalability comes through many (up to 11) different levels of spatial resolutions, allowing progressive texture transmission and many alternative resolutions

MPEG-4 Text-to-Speech

- MPEG-4 doesn't define a specific text-to-speech technique but rather the binary representation of a TTS stream and the interfaces of an MPEG-4 text-to-speech (M-TTS) with the other parts of an MPEG-4 decoder.
- An M-TTS stream may contain many different information types about the synthetic voice apart from text: gender, age, speech rate, language code, prosody, and lip shape information.
- It may contain fields that allow trick mode (fast forwarding, pausing, playing, or rewinding the synthetic speech).
- An M-TTS can also carry the International Phonetic Alphabet (IPA) coded phonemes with their time duration.

MPEG-4 Text-to-Speech

- Handed to the face animation engine in the MPEG-4 player, the coded phonemes can produce speech-driven face animation.
- In this case the face animation system doesn't receive a FAP stream from the MPEG-4 demultiplexer; instead it converts phonemes into visemes and uses them to perform the face model deformations.
- The phoneme duration synchronizes model animation and speech.
- Interestingly, applications require a tiny channel bandwidth—from 200 bps to 1.2 Kbps.

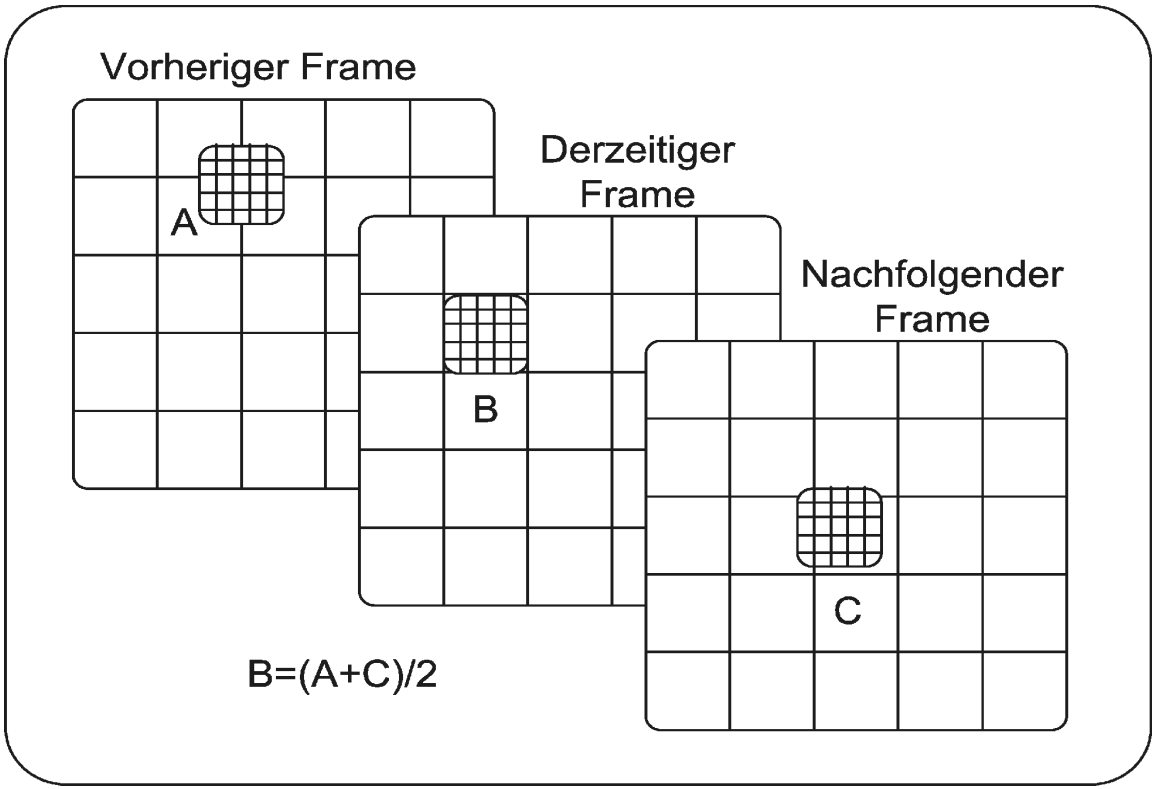
Allgemeine MPEG-4 Video Konzepte

- 4 Modi für B-Framekodierung
- Adaptive Quantisierung
- Bewegungsvektoren mit bis zu Viertel-Pixel Auflösung
- Globaler Bewegungausgleich
- Wahl der Quantisierungsmethode
- Variable Blockgröße
  - 1 MV Mode – 1 Vektor/16x16 Block (MPEG-2)
  - 4 MV Mode – 4 Bewegungsvektoren/16x16 Block, entspricht 1 Vektor/8x8 DCT-Block

MPEG B-Frames

- MPEG-2 bietet 3 Möglichkeiten der B-Frame Kodierung
  - Vorwärts Modus – Referenzframe ist vorhergegangener I- oder P-Frame, kodiert wird ein Vektor + Fehlerbild (Fehlerbild = Prädiktionsfehler)
  - Rückwärts Modus – Referenzframe ist nachfolgender I- oder P-Frame, kodiert wird ein Vektor + Fehlerbild
  - Interpolationsmodus – Beide Vektoren (aus Rückwärts- und Vorwärtsprädiktion) werden übertragen, übertragendes Fehlerbild resultiert aus Interpolation beider Referenzwerte

MPEG-2 B-Frame Interpolationsmodus



MPEG-4 B-Frame: Direkter Modus

- MPEG-4 bietet zusätzlich vierten Modus: „Direkter Modus“
- In die Kodierung werden wie im Interpolationsmodus sowohl Rückwärts- wie auch Vorwärtsprädiktionswerte mit einbezogen
- Die beiden Vektoren können allerdings von einem einzigen Vektor abgeleitet werden:
  - Bewegungsvektor im nachfolgendem Frame, gleiche Position wie Makroblock im betrachteten B-Frame
  - Vektor wird linear skaliert, je nach zeitlichen Abständen der 3 Frames (vorh. Frame – B-Frame – nachfolg. Frame)
  - Kodiert wird nur ein Delta Vektor (entsp. Fehlerkorrektur)

Adaptive Quantisierung

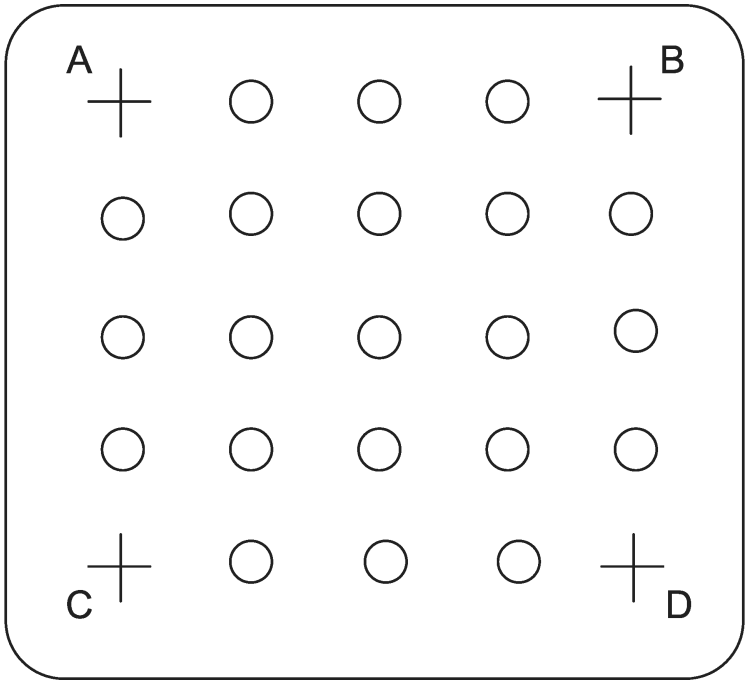
- Adaptive Quantisierung (MPEG-4) – Jedem Makroblock wird unter Berücksichtigung psychovisueller Aspekte individuelle Quantisierung zugeordnet
- Einfaches Beispiel (bei Xvid angewandt)
  - Artefakte an sehr hellen/dunklen Bildstellen für menschliches Auge weniger sichbar
  - Entsprechende Blöcke werden stärker quantisiert
- Adapitive Quantisierung heisst bei DivX „Psychovisuelle Verbesserung“

Viertel-Pixel Bewegungskompensation

- Bewegungskompensation mittels Blockmatching
  - Encoder sucht Makroblock im Referenzbild
  - Kodiert wird Bewegungsvektor (x-, y-Koordinaten) und das Fehler-Bild (Prädiktionsfehler)
  - Je genauer Bewegungsvektor, desto kleiner der Fehler
- Bewegungsvektor mit „Ein-Pixel“- Genauigkeit
  - Nur ganzzahlige Vektorkomponenten möglich
  - Reale Bewegung nur unzureichend beschrieben
  - Hoher Prädiktionsfehler, schlechte Kompression
  - Verwendet in H.261

Viertel-Pixel Bewegungskompensation

- (Virtuelle) Viertel-Pixel Werte mittels Interpolation berechnet
- Bewegungsvektoren werden im Viertel-Pixel Netz ermittelt



Viertel-Pixel Bewegungskompensation

- Bewegungsvektor mit „Halb-Pixel“ - Genauigkeit
  - Vektorkomponenten sind ganze oder halbe Pixelwerte
  - Halbpixelwerte durch Interpolation errechnet
  - Kleiner Prädiktionsfehler
  - Verwendet in MPEG-1, MPEG-2 und H.263
- MPEG-4 bietet Möglichkeit der Viertel-Pixel Genauigkeit
  - Realitätsnahe Beschreibung von Bewegung möglich
  - Signifikante Qualitätssteigerung bei niederen Datenraten
  - Interpolationsalgorithmen komplex

Globaler Bewegungsausgleich

- Für ein VOP werden bis zu 4 globale Bewegungsvektoren berechnet und im Header kodiert
- Effektive Kompression bei Kamerabewegungen
  - Geradliniger Schwenk – Ein Globaler Bewegungsvektor
  - Allgemeine Bewegung – Globaler Bewegungsausgleich mittels 4 Bewegungsvektoren (an jeder Ecke einer)
- Für jeden Makroblock wird zusätzlich Bewegungsvektor berechnet
- Encoder entscheidet, für welchen Makroblock welcher Bewegungsvektor (oder -kombination) kodiert wird

Alternative Scan Modi

- Alternativer horizontaler Scan

0	1	2	3	10	11	12	13
4	5	8	9	17	16	15	14
6	7	19	18	26	27	28	29
20	21	24	25	30	31	32	33
22	23	34	35	42	43	44	45
36	37	40	41	46	47	48	49
38	39	50	51	56	57	58	59
52	53	54	55	50	61	62	63

Alternative Scan Modi

- Alternativer vertikaler Scan

0	4	6	20	22	36	38	52
1	5	7	21	23	37	39	53
2	8	19	24	34	40	50	54
3	9	18	25	35	41	51	55
10	17	26	30	42	46	56	60
11	16	27	31	43	47	57	61
12	15	28	32	44	48	58	62
13	14	29	33	45	49	59	63

Wahl der Quantisierungsmethode

- MPEG-4 bietet User Möglichkeit, Quantisierungsmethode selbst zu wählen
  - Wahl zwischen H.263, MPEG oder Erstellen eigener MPEG-Quantisierungstabellen (Inter- und Intramatrix)