MPEG-7

Ing. Marco Bertini - Università degli Studi di Firenze
Via S. Marta 3 - 50139 - Firenze - Italy
Tel. +39-055-4796262
Fax +39-055-4796363
E-mail bertini@dsi.unifi.it
Web: http://viplab.dsi.unifi.it/~bertini

Introduction

How Much Information?

- The world's total yearly production of print, film, optical, and magnetic content would require roughly 1.5 billion GB (1.5EB) of storage.

  How Much Information Report
  http://www.sims.berkeley.edu/how-much-info

- This is equivalent to 250MB per person for every man, woman, and child on earth.

SI prefixes

Factor	Name	Symbol	Factor	Name	Symbol
10^{24}	yotta	Y	10^{-1}	deci	d
10^{21}	zetta	Z	10^{-2}	centi	c
10^{18}	exa	E	10^{-3}	milli	m
10^{15}	peta	P	10^{-6}	micro	μ
10^{12}	tera	T	10^{-9}	nano	n
10^{9}	giga	G	10^{-12}	pico	p
10^{6}	mega	M	10^{-15}	femto	f
10^{3}	kilo	k	10^{-18}	atto	a
10^{2}	hecto	h	10^{-21}	zepto	z
10^{1}	deka	da	10^{-24}	yocto	y

Digital Information

- Increasingly, individuals produce their own content
  - E.g. YouTube
- Of all information produced in the world
  - 93% is stored in digital form
  - HD in stand-alone PCs account for 55% of total storage shipped each year
- Over 80 billion photographs are taken annually
  - >400 petabytes
  - > 80 million times storage required for text

*Peta = 10^{15}

Information: Individuals

ITEM	AMOUNT	TERABYTES*
Photos	80 billion images	410,000
Home Video	1.4 billion tapes	300,000
X-Rays	2 billion images	17,200
Hard disks	200 million installed	13,760
TOTAL		740,960

*Tera = 10^{12}

Information: Published

ITEM	AMOUNT	TERABYTES
Books	968,735	8
Newspapers	22,643	25
Journals	40,000	2
Magazines	80,000	10
Newsletters	40,000	0.2
Office Documents	7.5E9	195
Cinema	4,000	16
Music CDs	90,000	6
Data CDs	1,000	3
DVD-video	5,000	22
TOTAL		285

Information: Film

ITEM	UNITS	DIGITAL CONVERSION	TOTAL PETABYTES
Photography	82E9	5Mb/photo	410
Motion Pictures	4,000	4Gb/movie	0.016
X-Rays	2.16E9	8Mb/radiograph	17.2
ALL FILM TOTAL			427.216

*Peta = 10^{15}

MPEG family of standards

MPEG Family of Standards (1)

- MPEG-1(**1992**): for the storage and retrieval of moving pictures and audio on storage media.
- MPEG-2 (**1995**): for digital television, the response for the satellite broadcasting and cable television industries in their transition from analog to digital formats.

Summary MPEG-1

- MPEG-1(1988-1993)

  -Goal: coding of moving picture with associated audio at rate up to *1.5Mbps*

  -Entertainment and storage

  -Video storage with VHS quality

  -Video CD is full application of MPEG-1

  -3 layers in audio coding.

  -Audio Layer 3 is the famous MP3

Summary MPEG-2

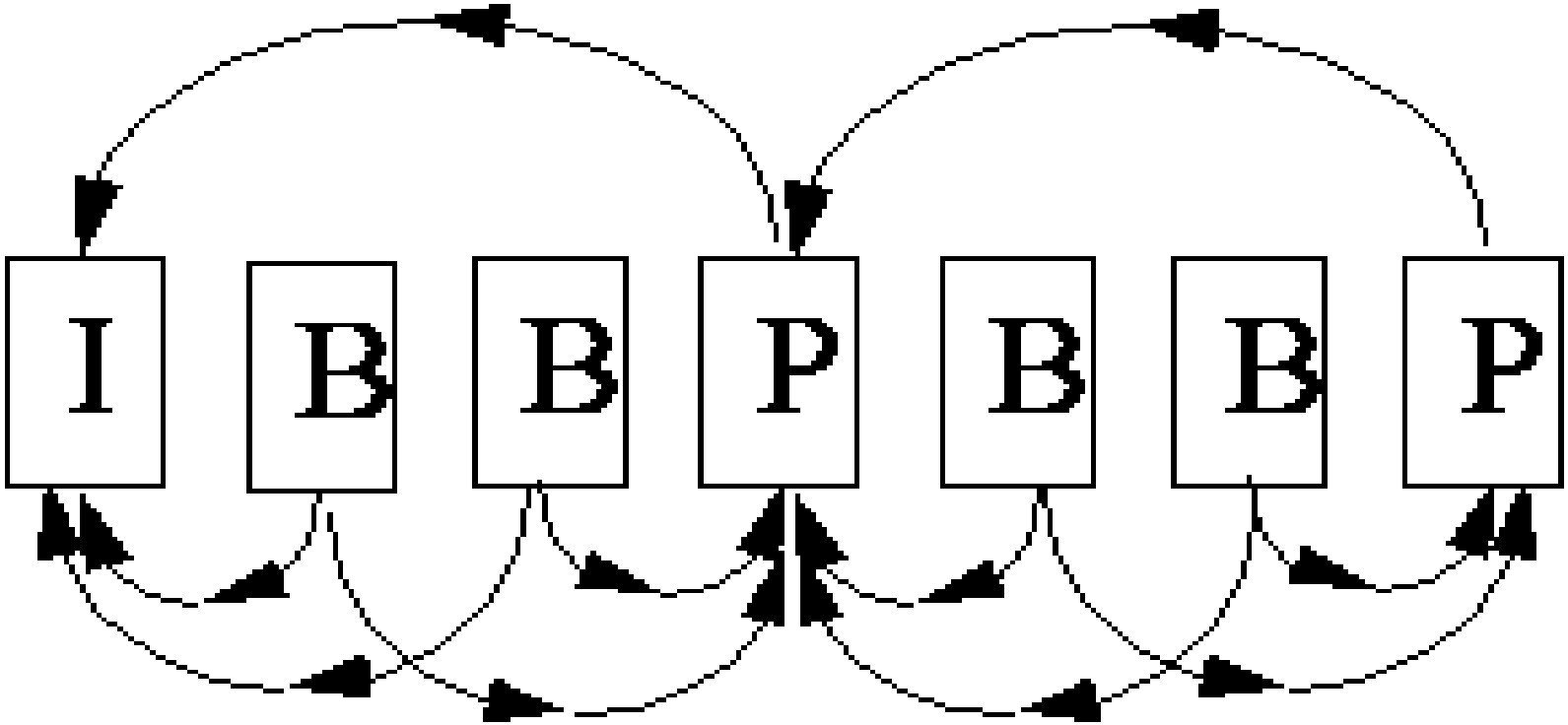
- MPEG-2(1990-1995)
  - Higher bit rate at *80Mbps*
  - mainly broadcast: interlaced images
  - Application in DVD, HDTV etc.
  - Backward compatibility with MPEG-1

MPEG Compression

MPEG encoding produces 3 types of frames

- I-Frame “Intracoded”
  - One complete video image
  - No other images needed to view
  - Contains the most data of any type
- P-Frame “Forward Predicted”
  - Encodes the changes from a previous frame
  - Previous frame needed to ‘view’
- B-Frame “Bidirectionally Predicted”
  - Encodes changes from a previous or future frame
  - Contains the least data (25% of an “I”)
  - Previous or future frames needed to ‘view’

Frame Dependency



MPEG Family of Standards (2)

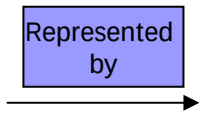
MPEG-4 **(1998 v.1, 1999 v.2)**

- First real multimedia representation standard
- Encodes content as independent objects
- Enables those objects to be manipulated individually or collectively on an audio visual scene
- Allows interactivity

Summary MPEG-4

- MPEG-4 embraces several areas:
  - Coding of visual objects
  - Coding of audio objects
  - Scene description and user interaction
- Most important: object coding
- This set the stage for MPEG-7.

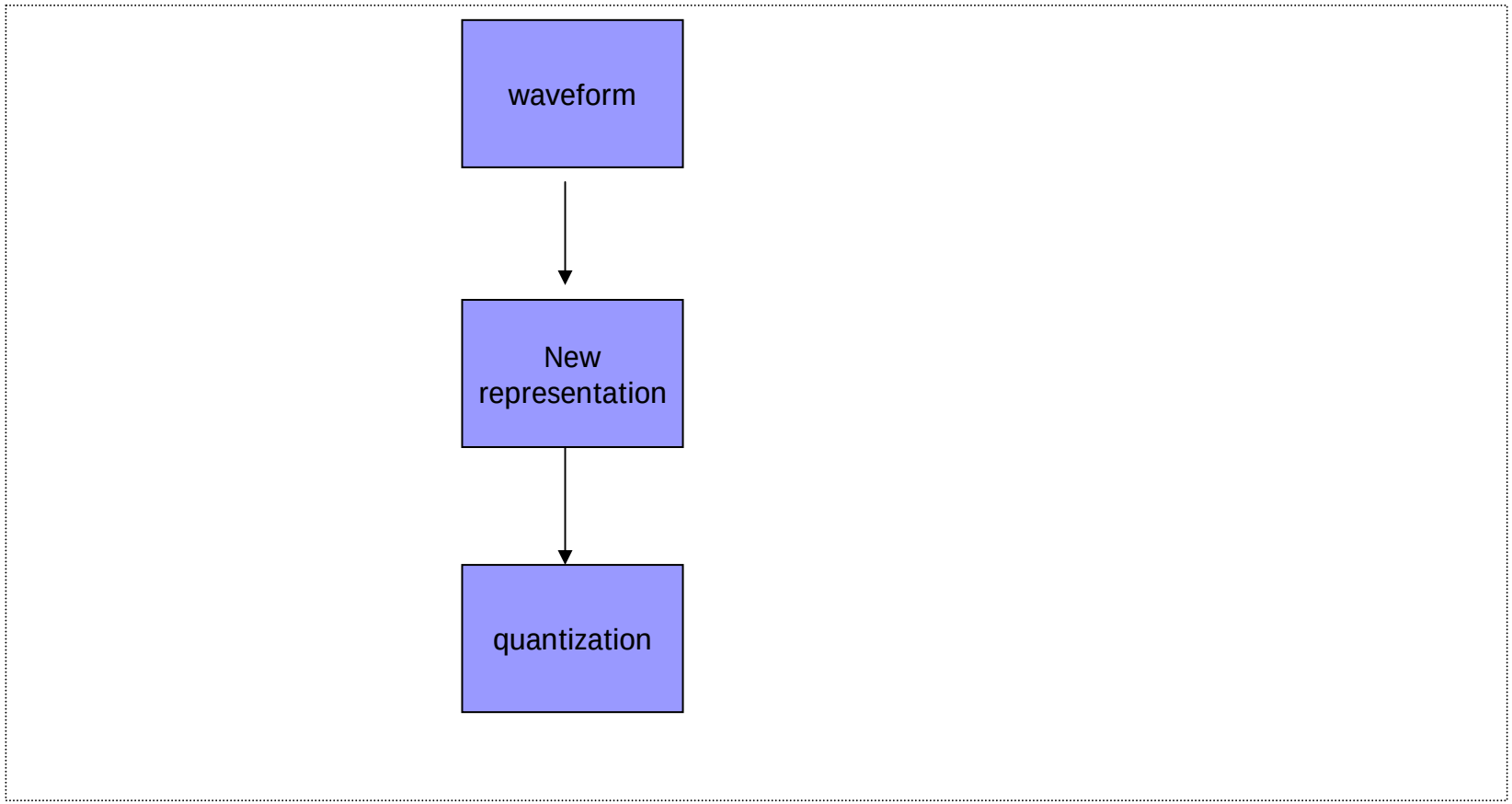
Why MPEG-7?

- Almost all standards up to this point involve either some form of *waveform coding* (e.g. linear prediction etc) or *transform coding* (e.g. DCT etc).
- Now, given the input to the filter and the filter coefficients we can reconstruct the waveform, albeit with some distortion.
- {Waveform } 

{input, filter coeffs}

- Hence, we have a different (more efficient) *representation* of the waveform.

Big picture so far.



Extension in Purpose

- MPEG-1, -2, and -4
  - Make content available
- MPEG-7
  - Lets you find the content you need
  - May be co-located with media data
    - Same data stream or storage system
- MPEG-21
  - Describes “big picture” across wide range of networks and devices

MPEG-3, -5, and –6 ???

- MPEG-3 existed to enable HDTV
  - Accomplished with tools of MPEG-2
  - Work item abandoned
- -1, -2, -4, -5 or -8 ???
  - MPEG decided NOT to follow either logical expansion
  - Chose number 7 instead

MPEG-21 ????

- Comprehensive and flexible framework for the 21st Century
  - Quality of Service
  - Rights Management
  - E-Commerce
- Efficient multimedia resource use across networks and devices
- Key concern is processor loading in network terminals
- Draft committee stage expected by Dec 2001

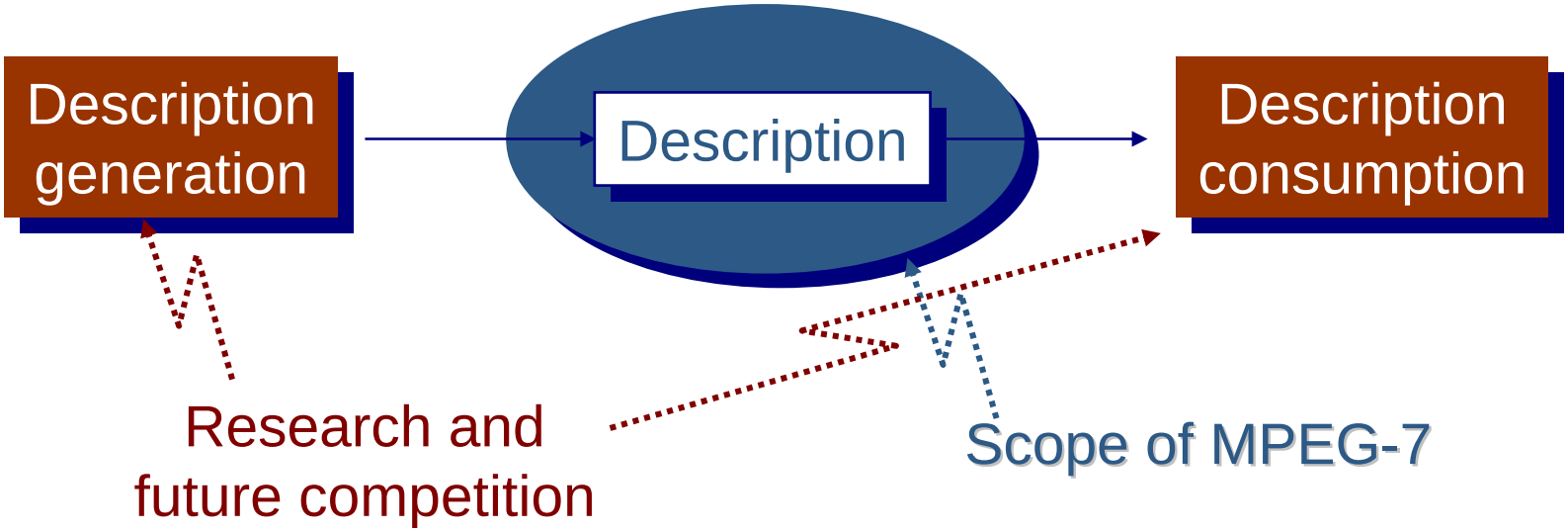
MPEG-47 ???

- It’s just a nickname:
  - Use MPEG-4 to do content CODEC and distribution and MPEG-7 to describe the content
  - MPEG-4 may contain an MPEG-7 stream
    - Also the MPEG-2 transport stream

Frameline 47 uses the MPEG-47 nickname

MPEG-7 standard

MPEG-7 ISO 15938

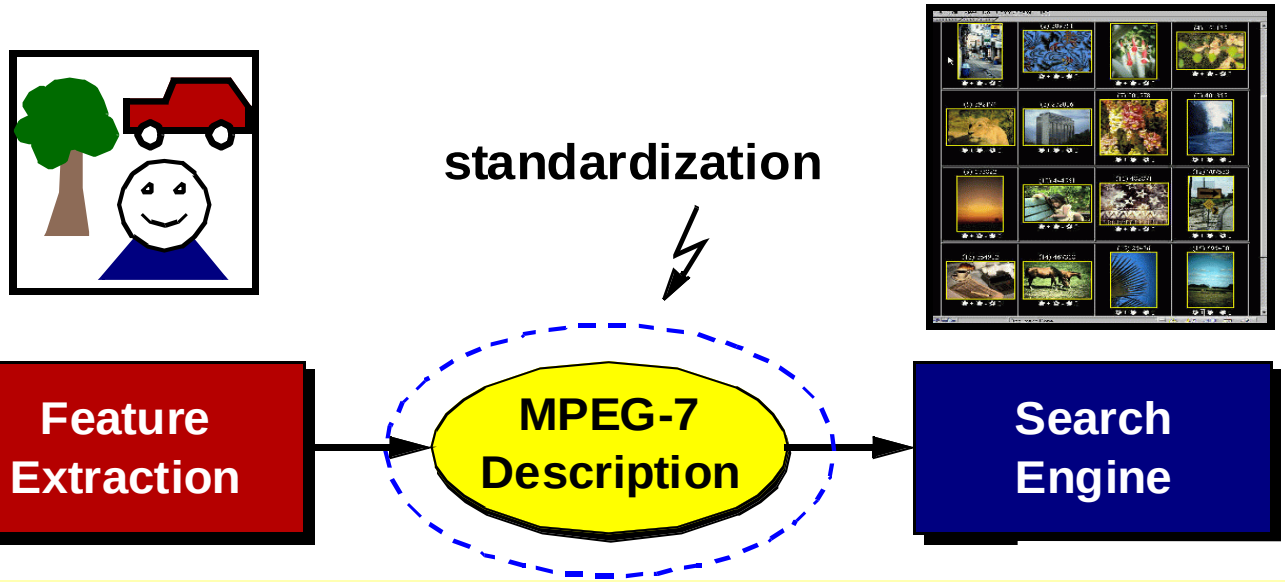


A standard for describing features of multimedia content.

MPEG-7 does NOT …

- Standardize the extraction of Audiovisual descriptions/features
- Specify the software programs that can use the descriptions
- Specify how to measure similarity between contents

MPEG-7 ISO 15938



Feature Extraction:
Content analysis (D, DS)
Feature extraction (D, DS)
Annotation tools (DS)
Authoring (DS)

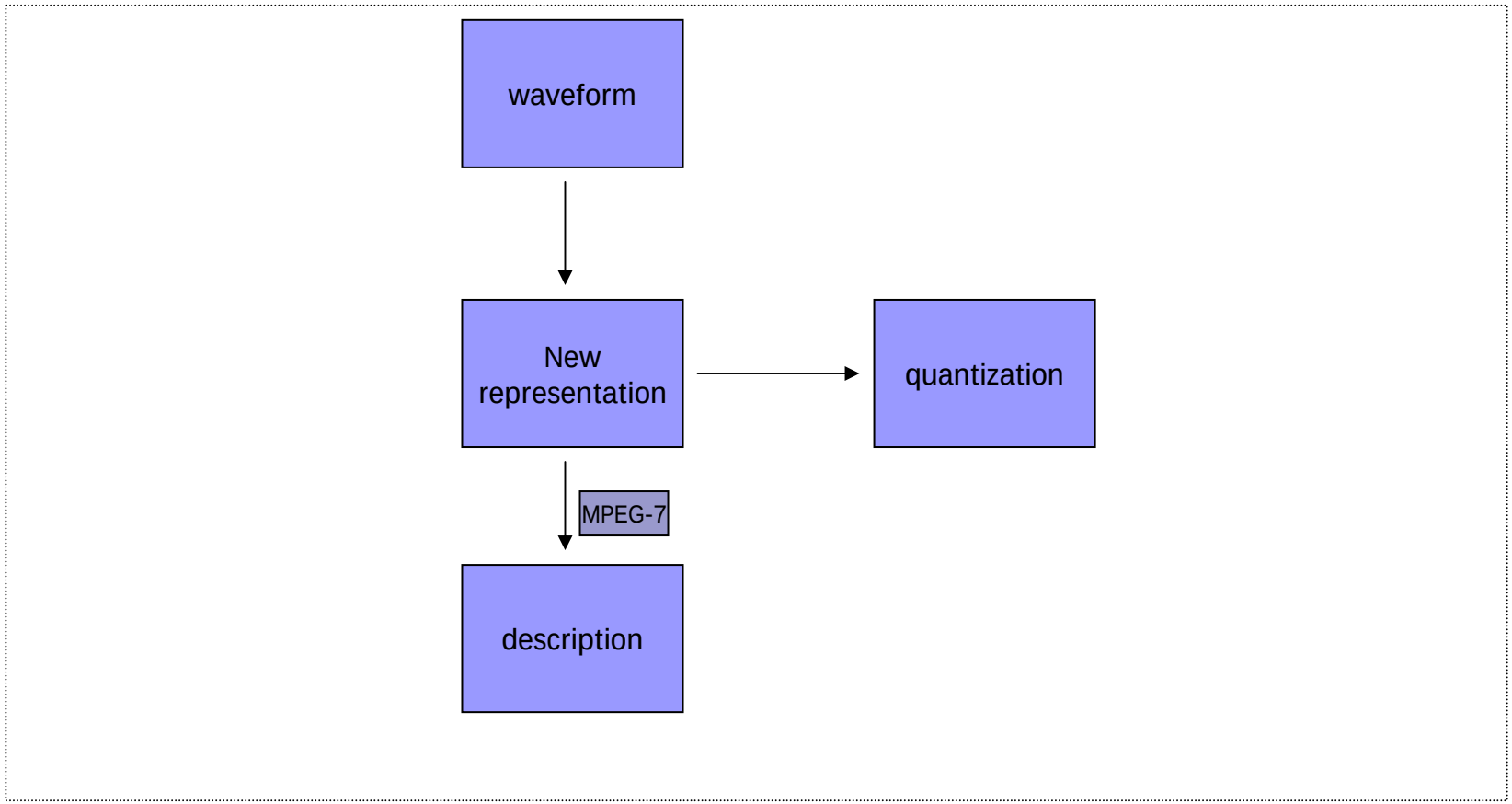
MPEG-7 Scope:
Description Schemes (DSs)
Descriptors (Ds)
Language (DDL)
Ref: MPEG-7 Concepts

Search Engine:
Searching & filtering
Classification
Manipulation
Summarization Indexing

Goals and Objectives

- Provide interoperability among systems and applications used in generation, management, distribution and consumption of audio-visual content descriptions.
- Help users or applications to identify, retrieve, or filter audiovisual information with descriptions of streamed or stored media.

Why MPEG-7?


waveform
New representation
quantization
MPEG-7
description

Link between MPEG-4 and MPEG-7

- MPEG-4 codes contents as objects.
- But an object can be described in many different ways, just like how we can describe the object ‘apple’ in for example French, English, Russian etc.
- MPEG-7 defines the ‘universal language’ as to how these objects are described and the ‘grammar’ as to how ‘sentences’ of these objects can be made.

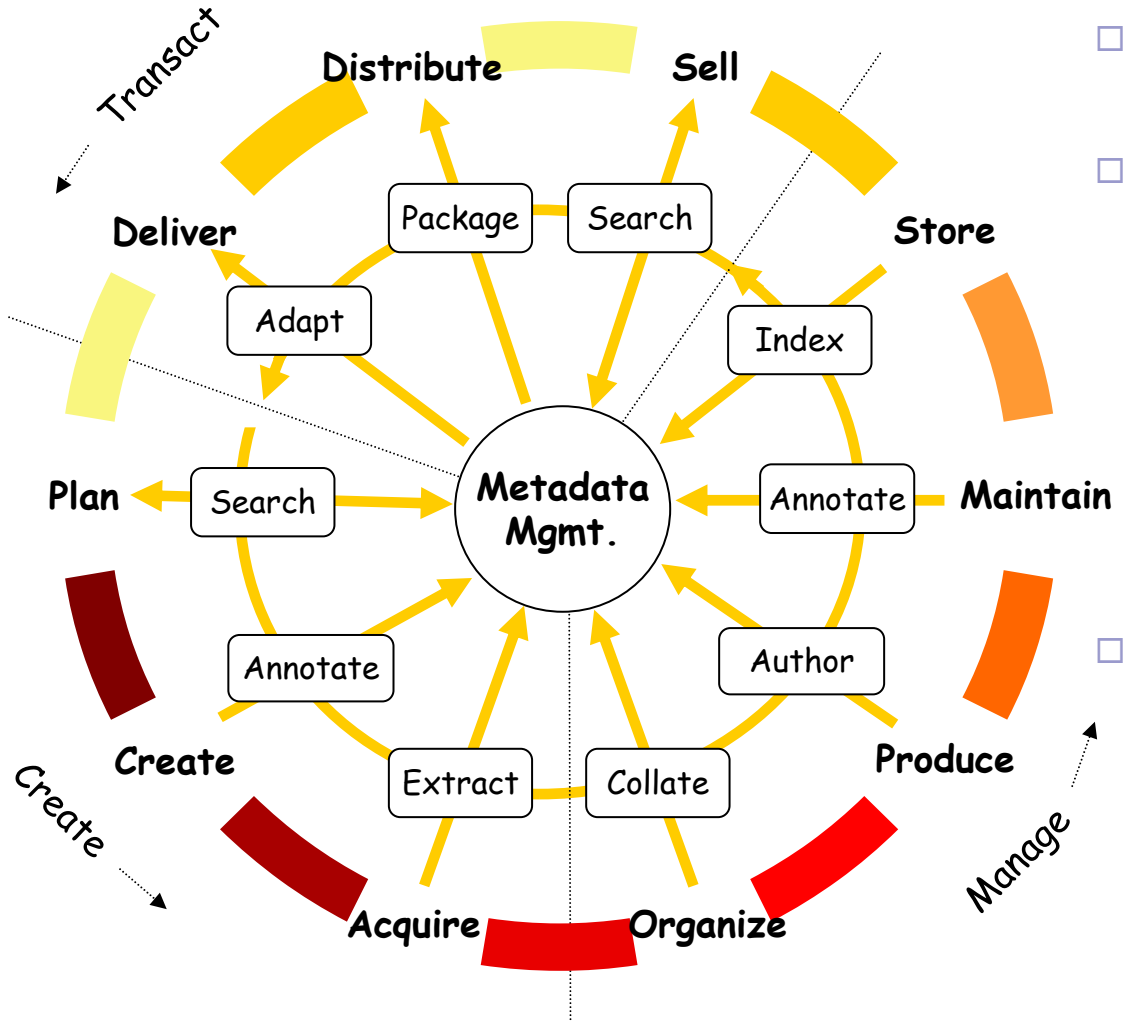
MPEG-7 Context

- Audiovisual information used to be consumed directly by human beings
- Increasingly created, exchanged, retrieved, re-used by computational systems
- Representations that allow some degree of interpretation of the information's meaning can be accessed and processed by computer

MPEG-7(1998-2001)

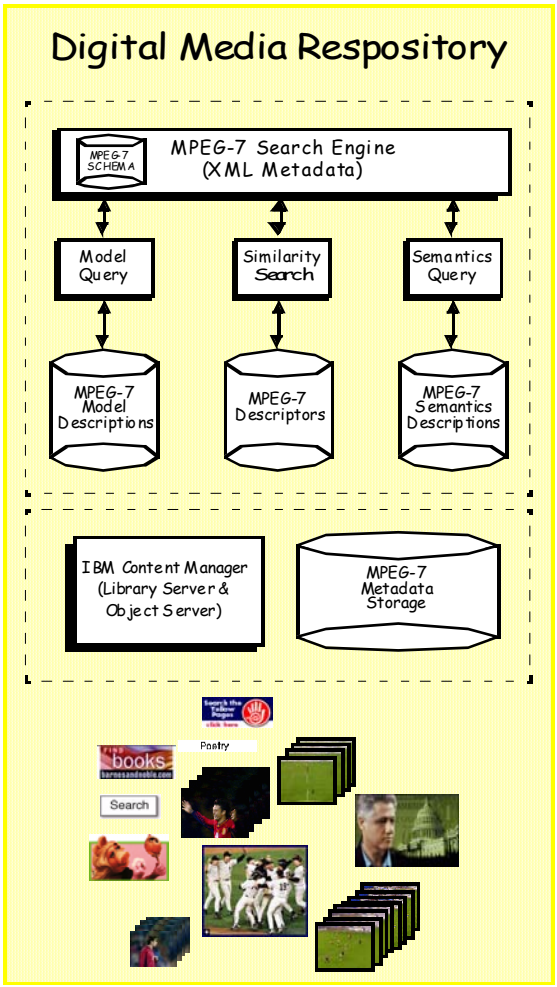
- Also known as ’Multimedia Content Description Interface’.
- An ISO/IEC standard
- Strictly speaking, MPEG-7 is *not* a data compression scheme.
- MPEG-7 is mainly a software implementation.
- MPEG-7 specifies the rules as to how to *describe* audiovisual data content whereas MPEG-1,2 and 4 make content *available*.
- MPEG-7 is not targeted at specific application. It aims to be as generic as possible for further extension.

Digital Media Value Chain (MPEG-7 Metadata Management)

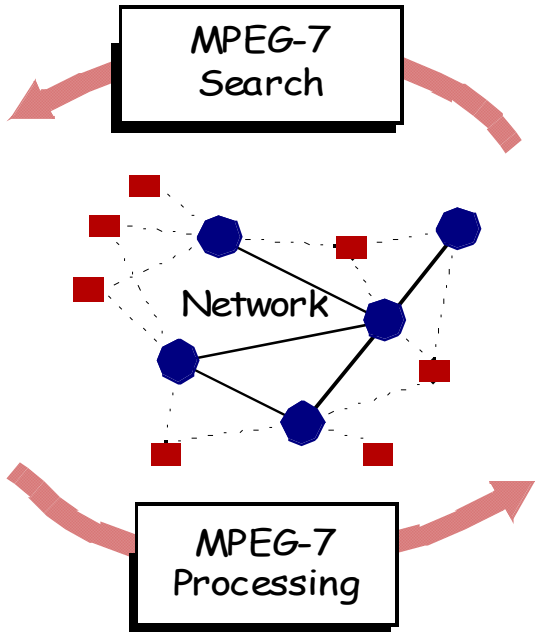


- Content value chain:
  - Life-cycle:
    - Increase content asset value through repurposing
  - Metadata layer:
    - First-class role of metadata (smart bits) throughout digital media lifecycle
    - Central role of metadata management (XML schemas, catalog records, ontologies)
    - MPEG-7 for content description
    - MPEG-21 for packaging, rights mgmt, transactions
  - Digital media metadata access functions:
    - **Create**: search, annotate, extract
    - **Manage**: index, annotate, collate
    - **Transact/Distribute**: search, package, adapt

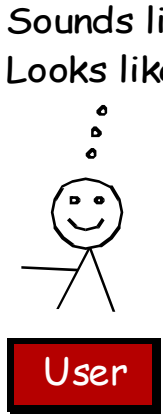
MPEG-7 Multimedia Indexing and Searching

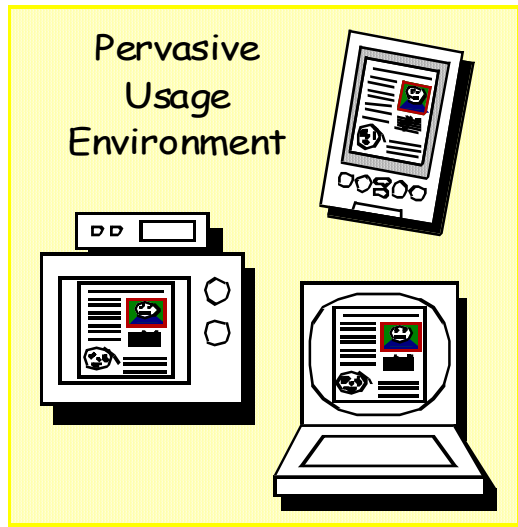


- **MPEG-7 Indexing & Searching:**
  - Semantics-based (people, places, events, objects, scenes)
  - Content-based (color, texture, motion, melody, timbre)
  - Metadata (title, author, dates)



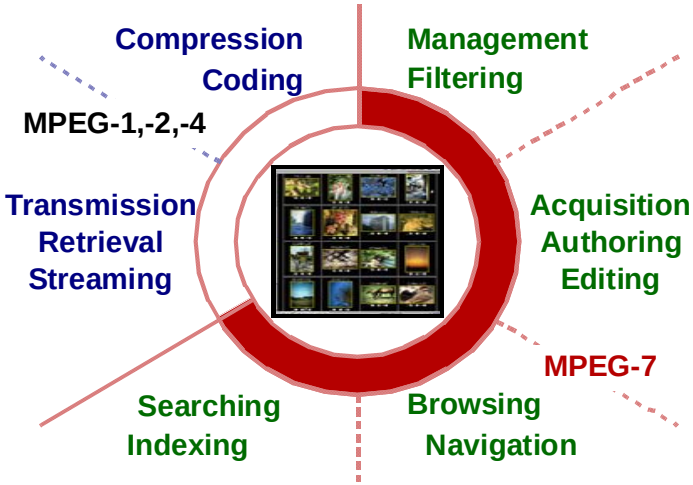
- **MPEG-7 Access & Delivery:**
  - Media personalization
  - Adaptation & summarization
  - Usage environment (user preferences, devices, context)



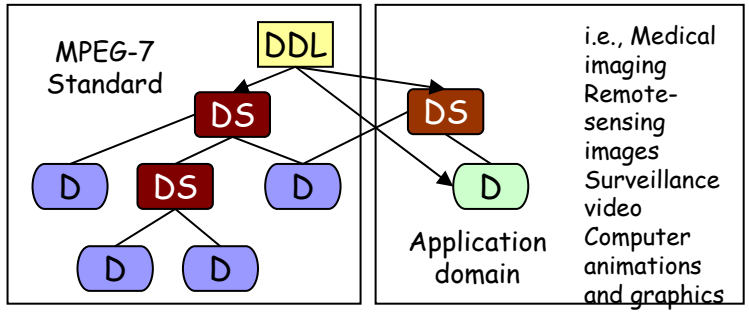


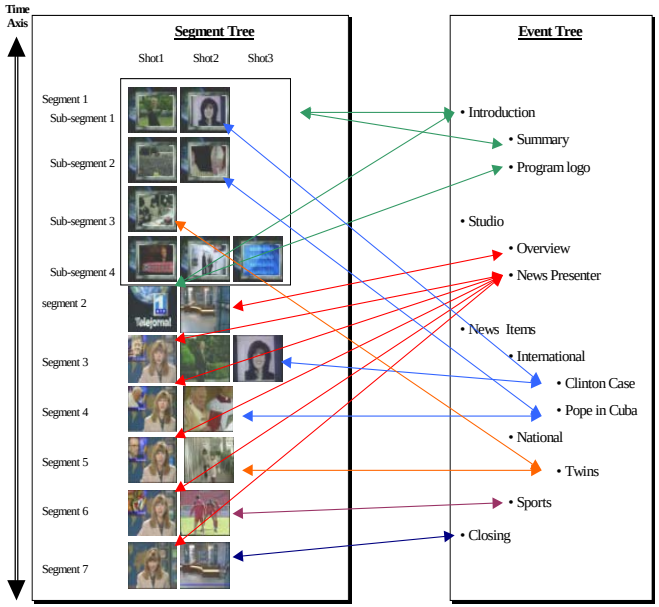
MPEG-7 Overview

(XML for Multimedia Content Description)

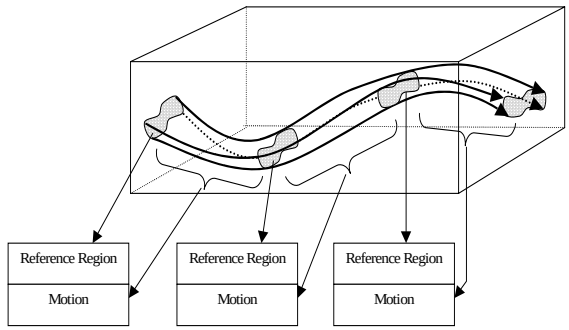


- **MPEG-7 Normative elements:**
  - Descriptors and Description Schemes
  - DDL for defining Description Schemes
  - Extensible for application domains





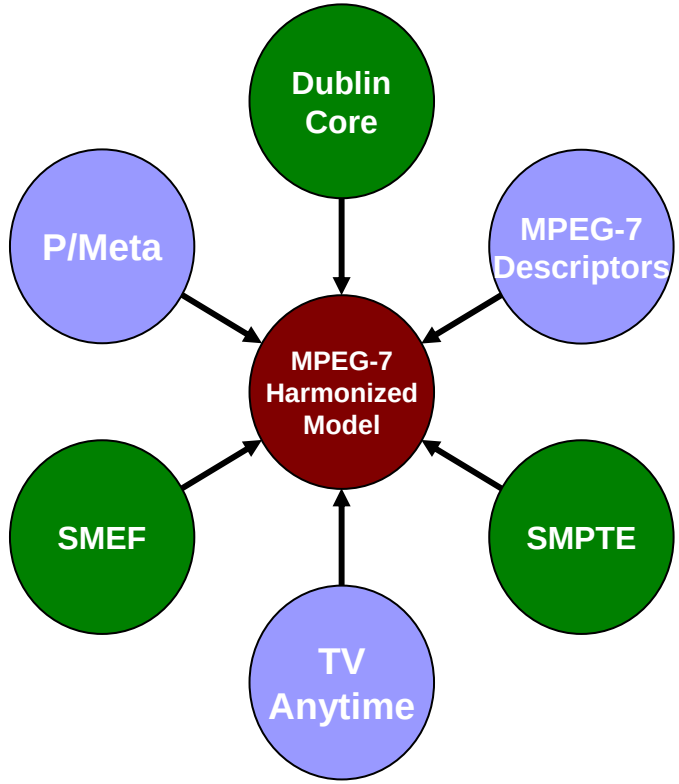
- **Rich, highly granular multimedia content description:**
  - Video segments, moving regions, shots, frames, …
  - Audio-visual features: color, texture, shape, …
  - Semantics: people, events, objects, scenes, …



Harmonization of Multimedia Metadata Standards

MPEG-7, SMEF, P/Meta, SMPTE, Dublin Core, TV-Anytime, Indecs

- MPEG-7: Moving Picture Experts Group
  - Infrastructure standard for Multimedia Metadata
  - Supports interpretation of the information's meaning
  - Supports broad range of applications
- SMEF – Standard Media Exchange Framework:
  - BBC developed data models for information involved in the Production, Development, Use, and Management of media assets
- P/Meta – EBU P/Meta Project:
  - Exchange of program content between high-level business functions of EBU members: Production, Delivery/Broadcast, & Archive
- SMPTE – Metadata dictionary & MXF:
  - Addresses Program Interchange independent of format
- Dublin Core Metadata Initiative:
  - Interoperable online metadata standards supporting broad range of purposes and business models.
- TV-Anytime – TV-Anytime Metadata:
  - Attractors/descriptors used e.g. in Electronic Program Guides (EPG), or in Web pages to describe content.
- Indecs – Indecs Metadata Framework
  - An international initiative of rights owners creating metadata standards for e-commerce.



- MPEG-7 Harmonized Model:
  - Harmonized elements with other standards and existing practices
  - Extensible framework
  - Registration authority for classification schemes, controlled terms, ontologies

MPEG-7 Constituent Components

- ISO/IEC 15938-1 MPEG-7 Systems
- ISO/IEC 15938-2 MPEG-7 DDL (Description Definition Language)
- ISO/IEC 15938-3 MPEG-7 Visual
- ISO/IEC 15938-4 MPEG-7 Audio
- ISO/IEC 15938-5 MPEG-7 MDS (Multimedia Description Schemes)
- ISO/IEC 15938-6 MPEG-7 Reference Software
- ISO/IEC 15938-7 MPEG-7 Conformance

MPEG-7 Systems

- Defines
  - the terminal architecture and the normative interfaces.
  - how descriptors and description schemes are stored, accessed and transmitted
  - tools that are needed to allow synchronization between content and descriptions

Comprehensive AV Descriptions

- Catalog
  - Title, Creator, Rights
- Semantics
  - Who, what, when, where of objects and events
- Structural features of AV content
  - Color of image, timbre of sound
- Leverage AV data representations
  - MPEG-1, -2, -4

Interoperability

- Uses XML Schema for content description
  - Over 100 XML industry Standard Groups
  - XML Repository at *www.xml.org*
- Groups with similar Objectives to MPEG-7
  - Society of Motion Picture and Television Engineers (SMPTE) [Metadata Dictionary]
  - European Broadcasting Union (EBU) [P/Meta]
  - Dublin [Core]
  - Digital Imaging Group (DIG)
  - TV-Anytime
  - Ohio Online Computer Center / Research Libraries Group (OCLC/RLG)

Similar approaches with notable divergence from MPEG-7

MPEG-7 Standardized Tools

- Enable detailed structural description
  - Descriptors
  - Description schemes
  - Language
- Different Granularity
  - Region, Image, Video Segment, Collection
- Different Areas
  - Content description, management, organization, navigation

MPEG-7 Applications

- Support and facilitate
  - Media portals
  - Content broadcasting
  - Ubiquitous multimedia
- Multimedia processing important to end user
- Multimedia processing important to providers of service and content

MPEG-7 Data Applications (1)

- Play a few notes on a keyboard and retrieve a list of musical pieces similar to the required tune, or images matching the notes in a certain way, e.g. in terms of emotions.
- Draw a few lines on a screen and find a set of images containing similar graphics, logos, ideograms,...
- Define objects, including color patches or textures and retrieve examples among which you select the interesting objects to compose your design.

MPEG-7 Data Applications (2)

- On a given set of multimedia objects, describe movements and relations between objects and so search for animations fulfilling the described temporal and spatial relations.
- Describe actions and get a list of scenarios containing such actions.
- Using an excerpt of Pavarotti's voice, obtaining a list of Pavarotti's records, video clips where Pavarotti is singing and photographic material portraying Pavarotti.

Some Application Domains with Applications

- Digital Libraries
  - Image catalog, musical dictionary, biomedical imaging
- Multimedia editing
  - Media authoring, personal electronic news service
- Cultural Services
  - History museums, art galleries
- Multimedia directory services
  - Yellow pages, tourist geographical information services
- Broadcast media selection
  - Radio channel, TV channel

The Ds of MPEG-7

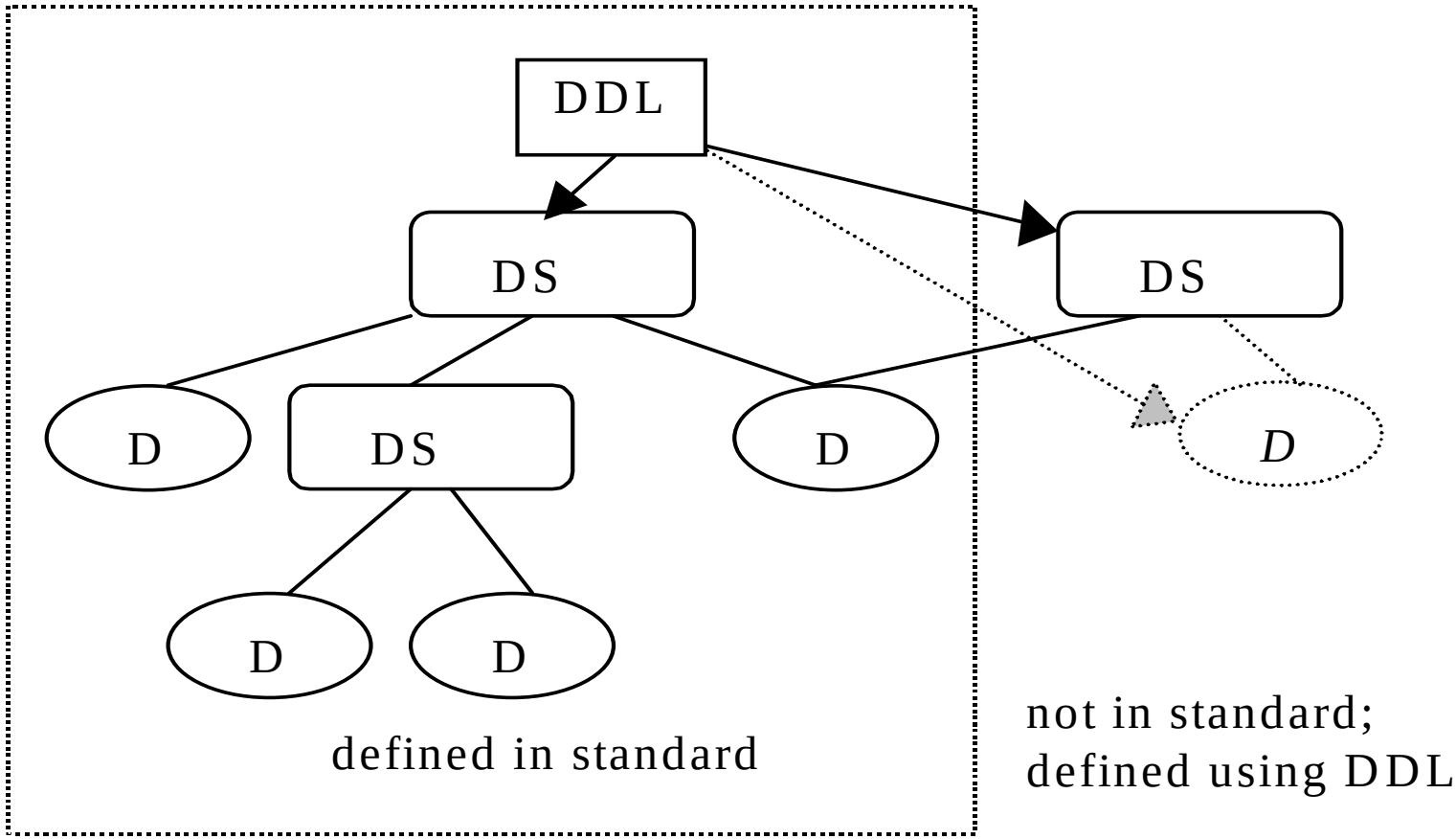
- Audio-Visual Descriptor (D)
- Description Schemes (DSs)
- Description Definition Language (DDL)

Multimedia Description Tools:

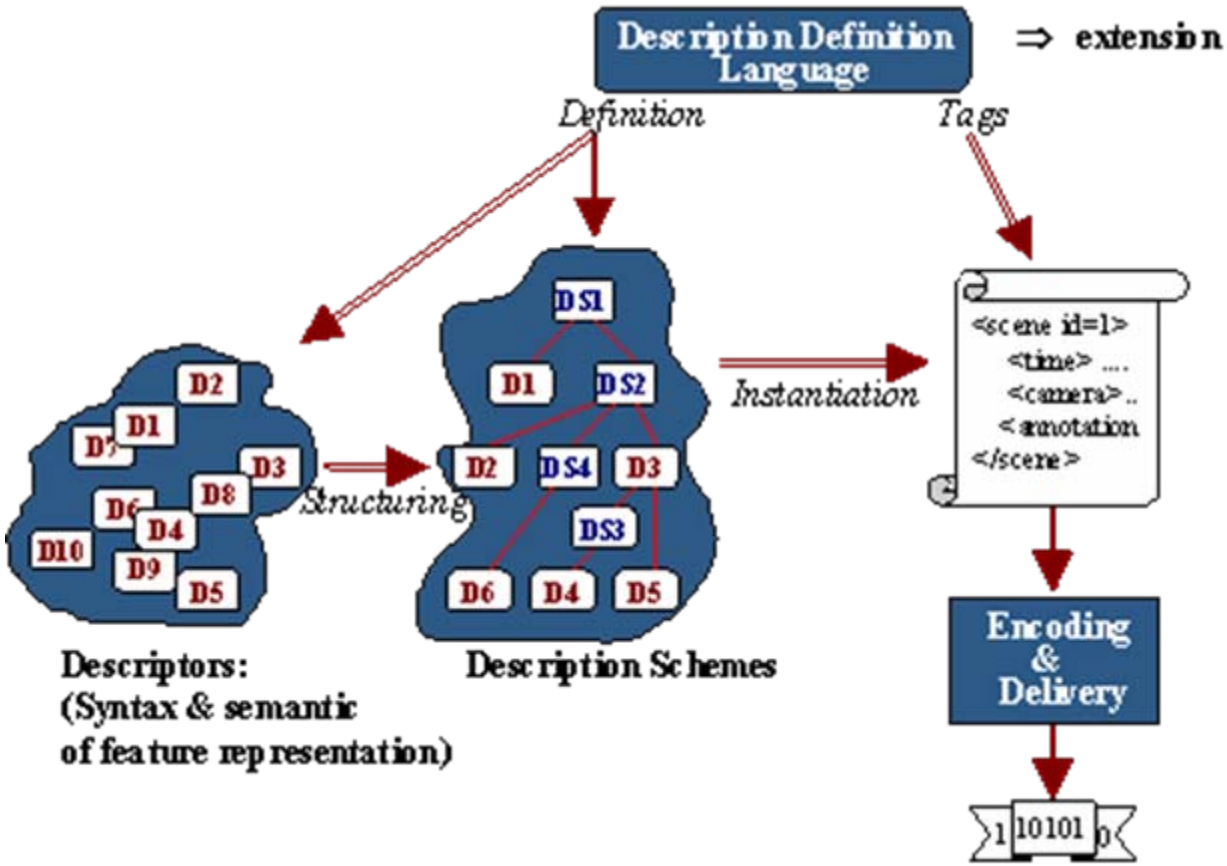
- Descriptors (D):
  - just like ‘data types’: may be atomic or composite (combination of descriptors)
  - Describes *low-level* audio or visual features such as color, motion, texture etc as well as audiovisual content such as location, time etc
  - It is possible to have several Ds for the same feature (e.g. timecode or color histo)
- Description Schemes (DS):
  - One or more Ds (and DSs)
  - Describes *high-level* audiovisual (AV) features such as regions, segments, events etc. DS not only describes relationships among D’s, but relationships among basic DS’s as well.
  - Describe and model AV content in terms of structure and semantics

- Descriptors are designed primarily to describe low-level audio or visual features
  - Automatic extraction
- Description Schemas are designed for higher level features (e.g. regions, segments, objects, etc.)
  - DSs produce more complex descriptions by integrating Ds and DSs and declaring relationships

Relation Between the Different MPEG-7 Elements

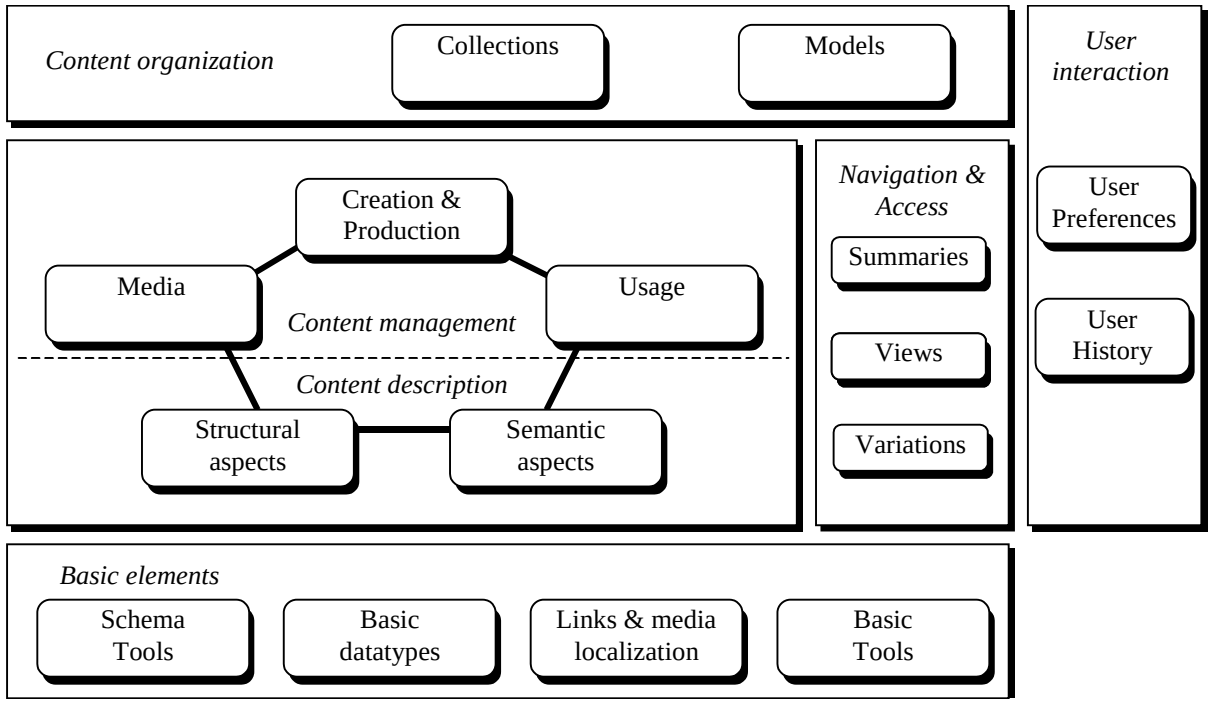


Relation Between the Different MPEG-7 Elements (another view)



Description Tools:

- Over 100 description tools have been defined. They can be divided as follows:



- The basic elements, at the lower level, deal with basic data types, mathematical structures, schema tools, linking and media localization tools, as well as basic DSs, which are elementary components of more complex DSs. The Schema tools section specifies elements for creating valid MPEG-7 schema instance documents and description fragments.

- In addition, this section specifies tools for managing and organizing the elements and datatypes of the schema. Based on this lower level, content description and management elements can be defined.

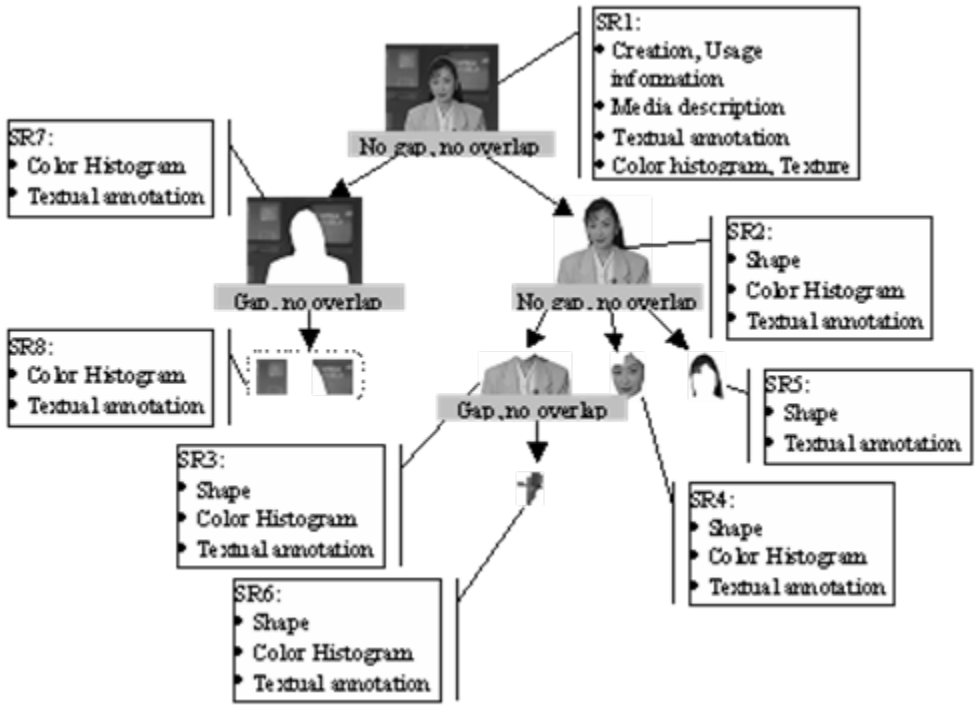
  These elements describe the content from several viewpoints. Currently five viewpoints are defined: creation and production, media, usage, structural aspects, and conceptual aspects.

  The first three elements primarily address information that's related to the management of the content (content management), whereas the last two are mainly devoted to the description of perceivable information (content description).

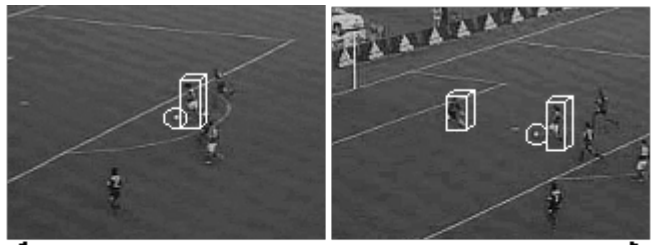
- ***Creation and Production:*** Contains meta information that describes the creation and production of the content; typical features include title, creator, classification, and purpose of the creation. Most of the time this information is author-generated since it can't be extracted from the content.
- ***Usage:*** Contains meta information that's related to the usage of the content; typical features involve rights holders, access rights, publication, and financial information. This information may be subject to change during the lifetime of the AV content.
- ***Media:*** Contains the description of the storage media; typical features include the storage format, the encoding of the AV content, and elements for the identification of the media. *Note:* Several instances of storage media for the same AV content can be described.
- ***Structural aspects:*** Contains the description of the AV content from the viewpoint of its structure. The description is structured around segments that represent physical, spatial, temporal, or spatio-temporal components of the AV content. Each segment may be described by signal-based features (color, texture, shape, motion, audio) and some elementary semantic information.
- ***Conceptual Aspects:*** Contains a description of the AV content from the viewpoint of its conceptual notions.
- The five sets of Description Tools are presented here as separate entities, however, they are interrelated and may be partially included in each other. For example, Media, Usage or Creation & Production elements can be attached to individual segments involved in the structural description of the content. Tools are also defined for *navigation and access* and there is another set of tools for *Content organization* which addresses the organization of content by classification, by the definition of collections and by modeling. Finally, the last set of tools is *User Interaction* which describes user's preferences for the consumption of multimedia content and usage history.

Example:

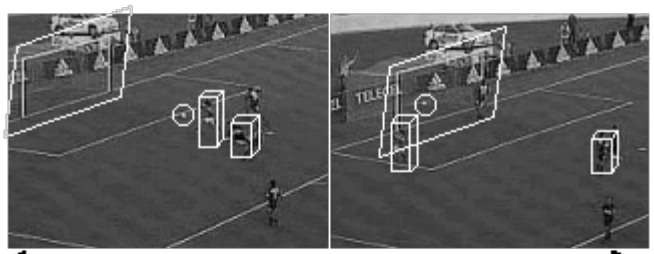
- One DS in “content description” ‘toolbox’ is the Segment DS.



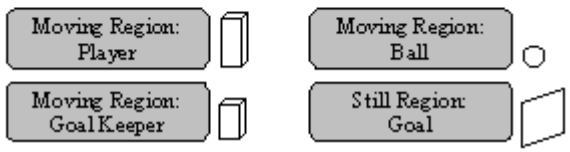
Another example: SegmentRelationship DS

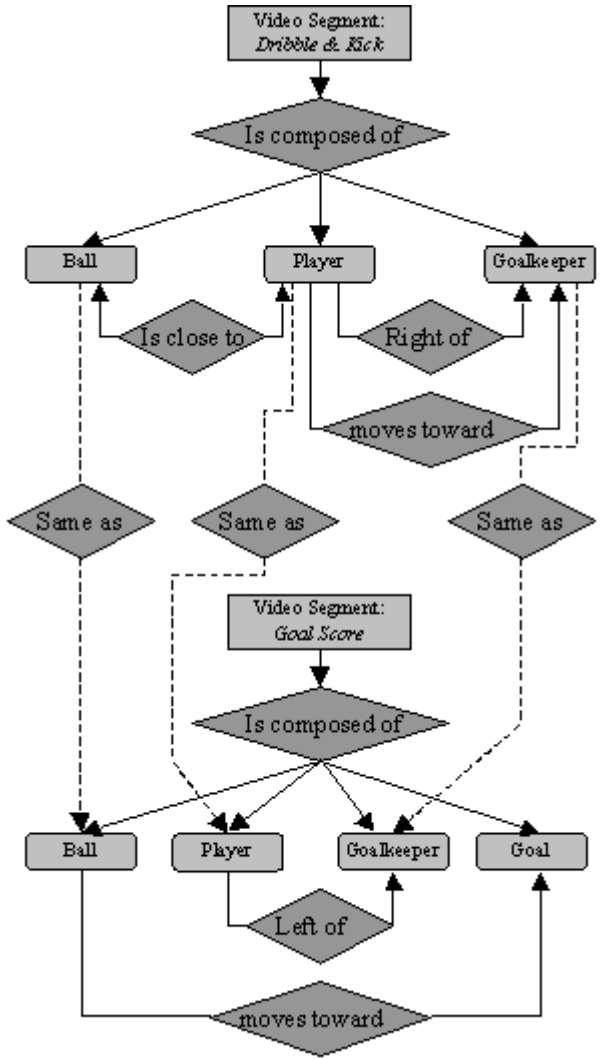


Video Segment: *Dribble & Kick*



Video Segment 2: *Goal Score*





- Video, Audio and Audio-Visual Segment Description Schemes can describe segments that are not necessarily connected in space and/or time

MPEG-7 Terminology: Data

- Audio-visual information described using MPEG-7 without regard to storage, coding, display, transmission, medium or technology
- Intended to be sufficiently broad to encompass graphics, still images, video, film, music, speech, sounds, text, …

Data Examples

- MPEG-4 stream
- Video tape
- CD containing music
- Sound or speech
- Picture printed on paper
- Interactive multimedia installation on the web

MPEG-7 Terminology: Feature

- Distinctive characteristic of data signifying something to someone
- Cannot be compared without meaningful feature representation (**descriptor**) and its instantiation (**descriptor value**)

Feature Examples

- Color of an image
- Pitch of a speech segment
- Rhythm of an audio segment
- Camera motion in a video
- Style of a video
- Title of a movie
- Actors in a movie

MPEG-7 Terminology: Descriptor (D)

- Representation of a Feature
- Defines syntax and semantics of the Feature representation
- Allows evaluation of corresponding feature by means of the Descriptor Value
- Several Descriptors may represent a single feature by addressing different relevant requirements

MPEG-7 Terminology: Descriptor Value

- Instantiation of a Descriptor for a given data set, or subset of that data set
- Descriptor Values are combined using a Description Scheme to form a Description

Descriptor Example

Color Feature

- Color histogram
- Average of frequency components
- Motion field
- Text of the title

MPEG-7 Terminology: Description Scheme

- Specifies structure and semantics of relationships between its components
- Components may be both Descriptors and Description Schemes
  - A Descriptor contains only basic data types, provided by the Description Definition Language
  - A Descriptor does not refer to another Descriptor

Description Scheme Example

- Movie, temporally structured as scenes and shots
  - Including textual descriptors at the scene level
  - Including color, motion and audio descriptors at the shot level

Description Schemes in MPEG-7

- Creation and Production
  - Title, creator, classification, purpose of creation
- Usage
  - Rights holders, access rights, publication, financial info
- Media
  - Storage format, AV content encoding, media identification
- Structural Aspects
  - Color, texture, shape, motion, audio
- Conceptual Aspects
  - AV conceptual notions
- Basic Elements
  - Data types, math structures, schema tools

MPEG-7 Terminology: Description

- Consists of a Description Scheme and the set of Descriptor Values (instantiations) that describe the Data
- The Description Scheme may not be fully instantiated, depending upon completeness of the Descriptor Values set

MPEG-7 Terminology: Description Definition Language (DDL)

- Language that enables creation of new Description Schemes and Descriptors
- Enables extension and modification of existing Description Schemes
- Expresses relations, object orientation, composition, partial instantiation

DDL Logical Components

- XML Schema structural language components
- XML Schema structural datatype components
- MPEG-7 specific extensions
  - Datatypes for matrices and arrays
  - Datatypes for time point and duration
  - Data value propagation (HeaderType)

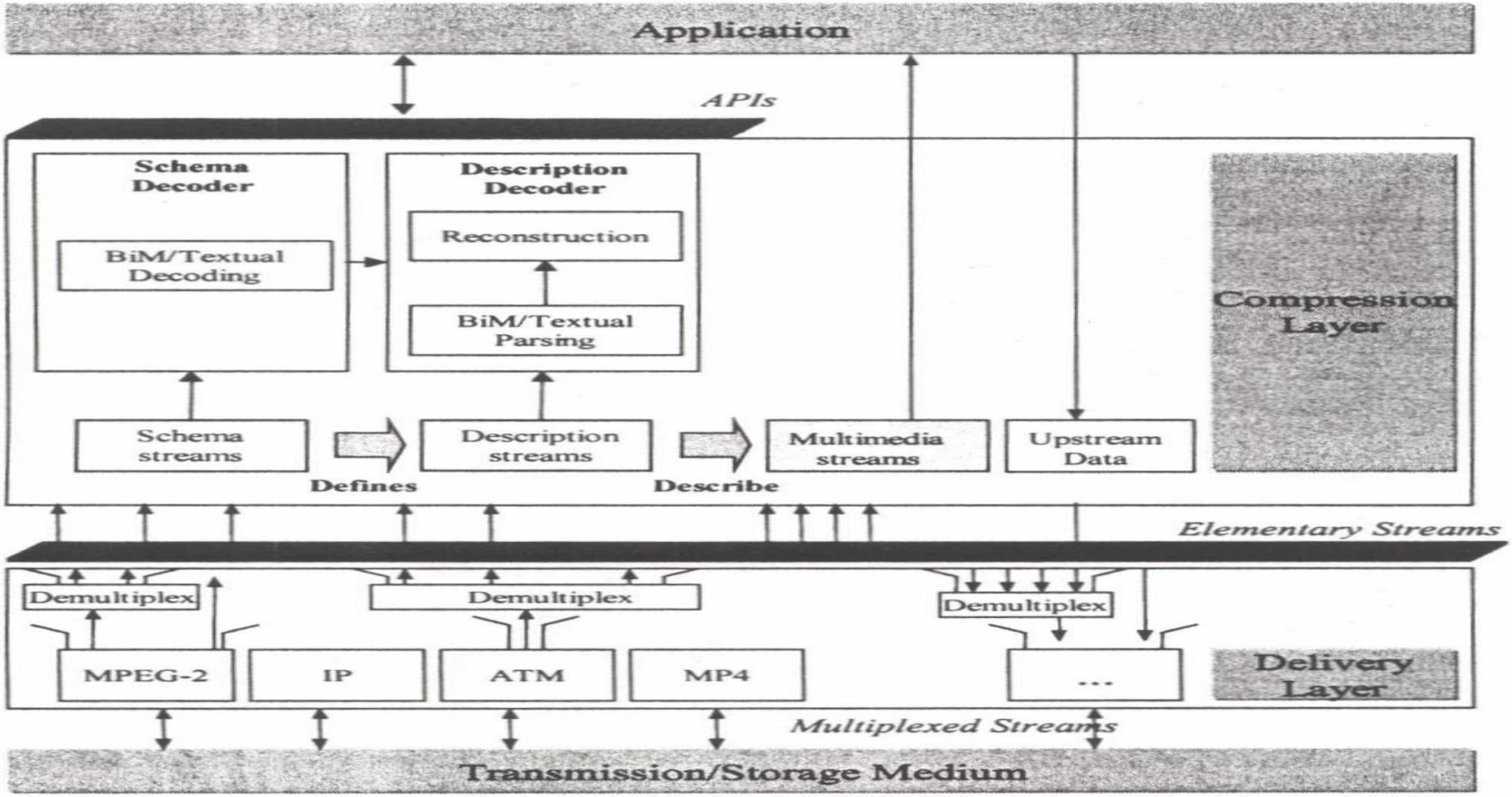
MPEG-7 Systems

- Specifies functionalities such as preparation of MPEG-7 Descriptions
  - Efficient transport/storage
  - Synchronization of content and description
  - Development of conformant decoders
- Mechanism for providing multimedia content is considered part of a complete application and lies outside the scope of the standard

MPEG-7 Terminal

- Obtains MPEG-7 data from transport
- Extracts elementary streams from delivery layer
  - Undo transport/storage specific framing/multiplexing
  - Retain synchronization timing
- Forwards elementary streams of individual access units to compression layer
- Decodes
  - Schema streams describing data structure
  - Full or partial content description streams
- Generates user requested multimedia streams
- Feeds back via delivery layer for transmission/storage

MPEG-7 Terminal



MPEG-7 DDL

With extensions, XML meets key requirements

- Datatype definition
- D and DS declaration
- Attribute declaration
- Typed reference
- Content model
- Inheritance/subclassing mechanism
- Abstract D and DS
- DS inclusion

Description Definition Language (DDL)

- “…a language that allows the creation of new Description Schemes and, possibly, Descriptors.”
- “It also allows the extension and modification of existing Description Schemes.”

MPEG-7 Requirement Documents V.13

DDL (2)

- It is based on XML Schema Language
- Consists of
  - XML Schema Structural Components
  - XML Schema Data Types
  - MPEG-7 Specific Extensions

DDL (3)

- A Simplified Example:

```
<VideoSegment id = "VS1" >
        <MediaTime>
                <MediaTimePoint> T0:0:0   </MediaTimePoint>
                <MediaDuration>   PT10M  </MediaDuration>
        </MediaTime>

        <StructuredAnnotation>
                This is an example of video segment
        </StructuredAnnotation>

        <GofGopColorHistogram HistogramTypeInfo = "Average">
                ....
        </GofGopColorHistogram>

        <SegmentDecomposition Gap = "true" Overlap = "true" DecompositionType = "temporal">
                <VideoSegment id = "VS2" > .... </VideoSegment>
                <VideoSegment id = "VS3" > .... </VideoSegment>
                <VideoSegment id = "VS4" > .... </VideoSegment>
        </SegmentDecomposition>
</VideoSegment>
```

MPEG-7 Visual

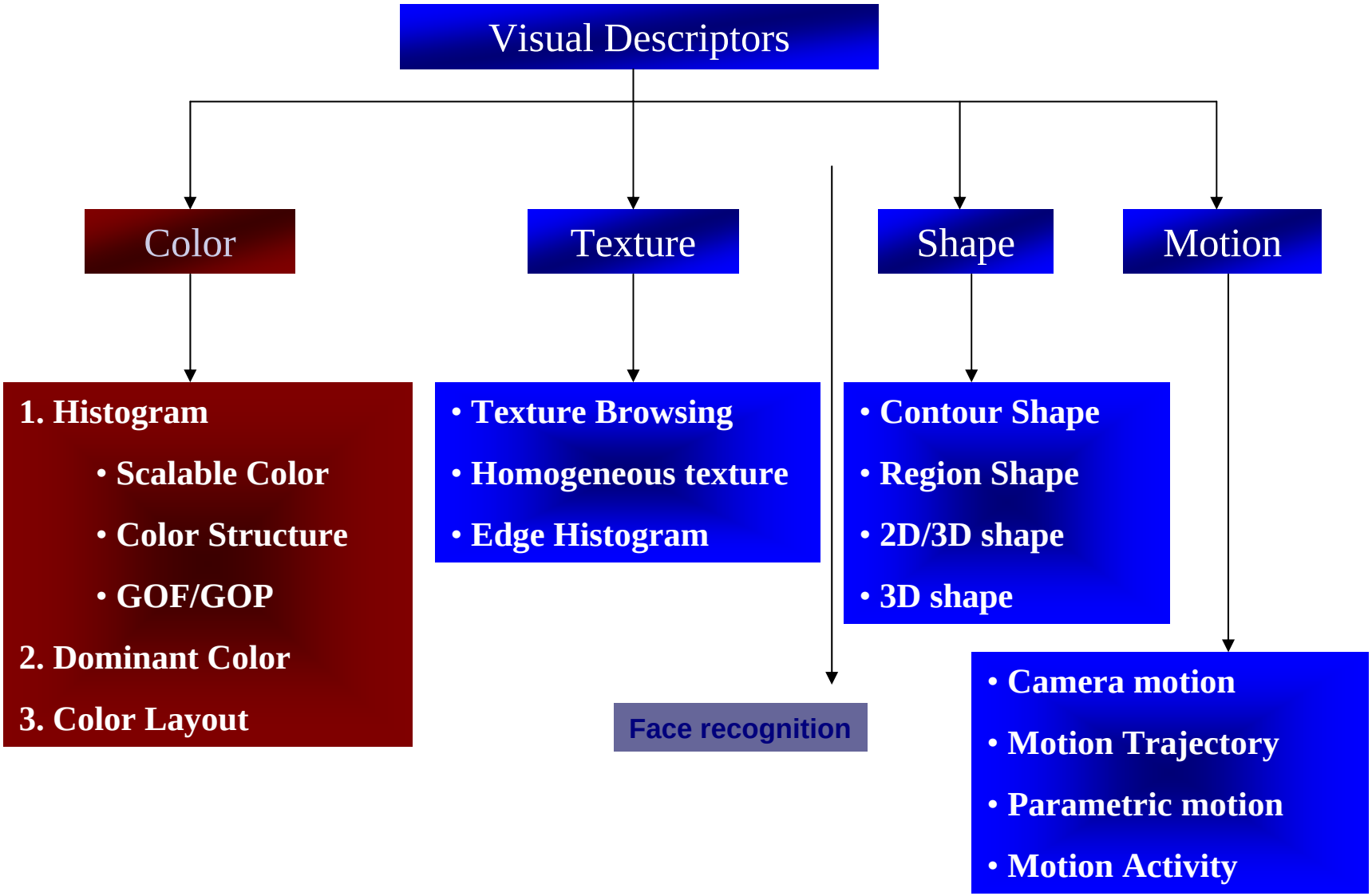
- Specifies set of standardized Ds and DSs
- Mainly address specific features
  - Color, texture, motion
- Often requires other low-level Ds or support elements
  - Structure – grid layout, spatial coordinates
  - Viewpoint – multiple view
  - Localization – region locator
  - Temporal – time series, temporal interpolation

MPEG-7 Visual Standardized Descriptors

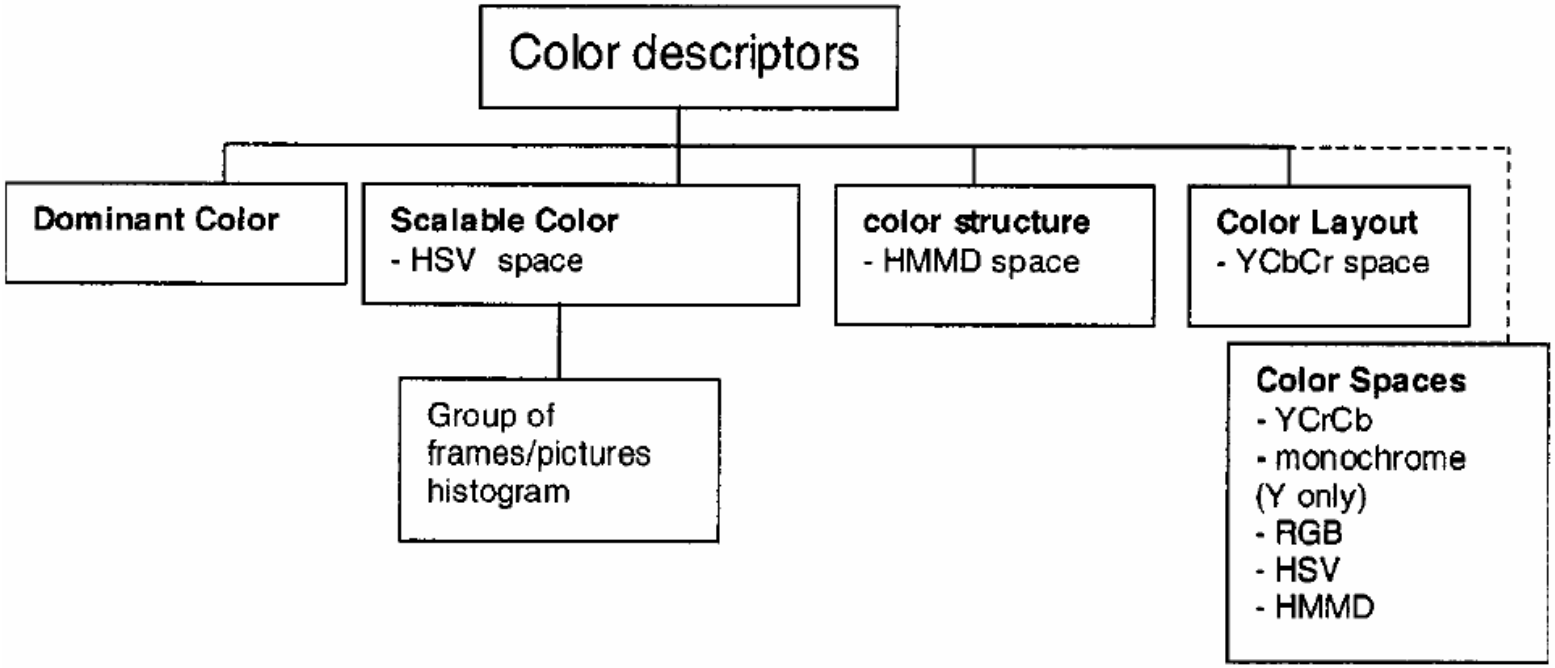
- Color
  - Color Space, Color Quantization, Dominant Color, Scalable Color Color Layout,Color Structure, Group of Picture Color
- Texture
  - Homogeneous Texture, Texture Browsing, Edge histogram
- Shape
  - Region Shape, Contour Shape, Shape 3D
- Motion
  - Camera Motion, Motion Trajectory, Parametric Motion, Motion Activity
- Face Recognition, others

Other Multimedia Description Tools

- MPEG-7 have a general set of multimedia Description Tools.
- From this general set 2 other sets are defined:
  - MPEG-7 Visual: Description tools dealing with only visual description.
  - MPEG-7 Audio: Description tools dealing with only audio description.


Visual Descriptors
Color
Texture
Shape
Motion
1. Histogram
• Scalable Color
• Color Structure
• GOF/GOP
2. Dominant Color
3. Color Layout
• Texture Browsing
• Homogeneous texture
• Edge Histogram
Face recognition
• Contour Shape
• Region Shape
• 2D/3D shape
• 3D shape
• Camera motion
• Motion Trajectory
• Parametric motion
• Motion Activity

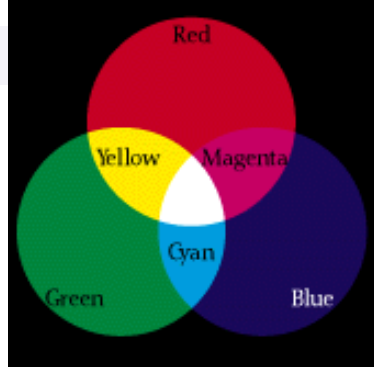
Color Descriptors



Color Spaces

- Constrained color spaces
  - Scalable Color Descriptor uses HSV
  - Color Structure Descriptor uses HMMD
- MPEG-7 color spaces:
  - Monochrome
  - RGB
  - HSV
  - YCrCb
  - HMMD

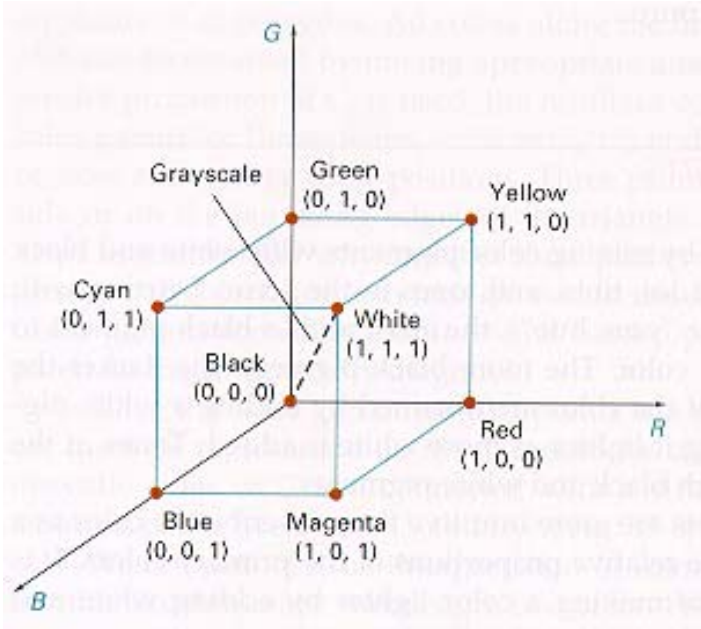
RGB Color Space

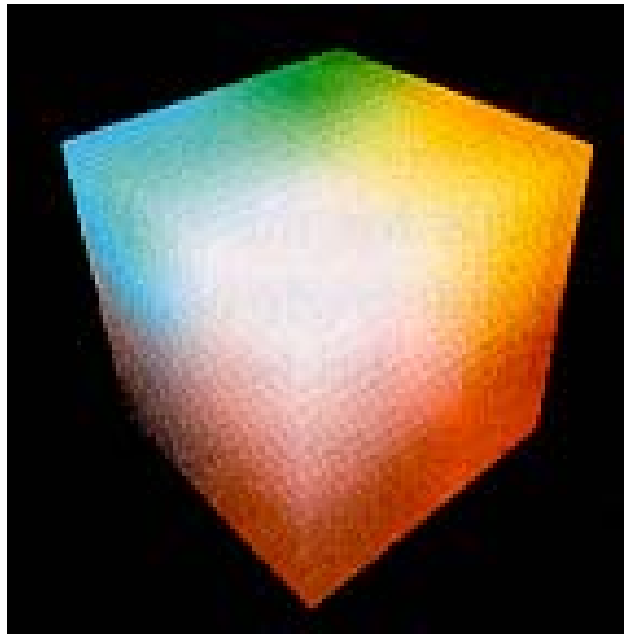


- Hardware Oriented Model:

RGB Color Space: 3 values to represent a color.

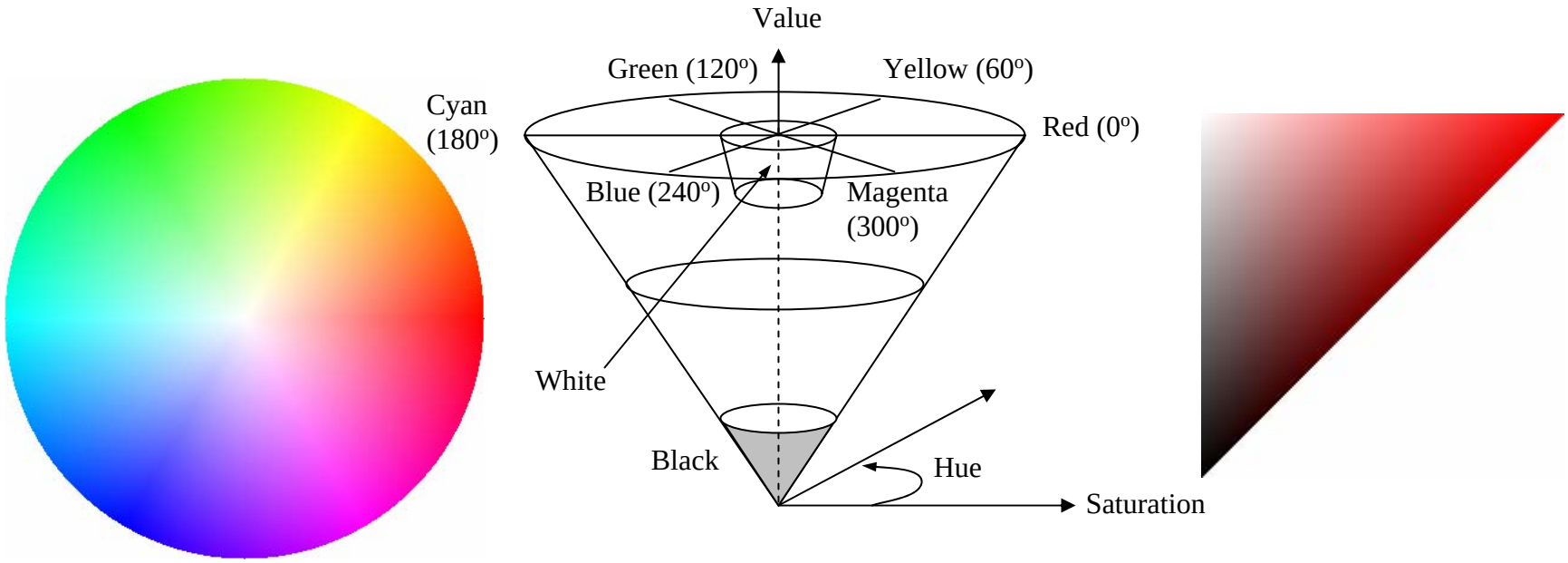
Red Green Blue





HSV Color Space

- HSV – Hue Saturation Value
  - close to human perception
  - 3 values to represent a color.



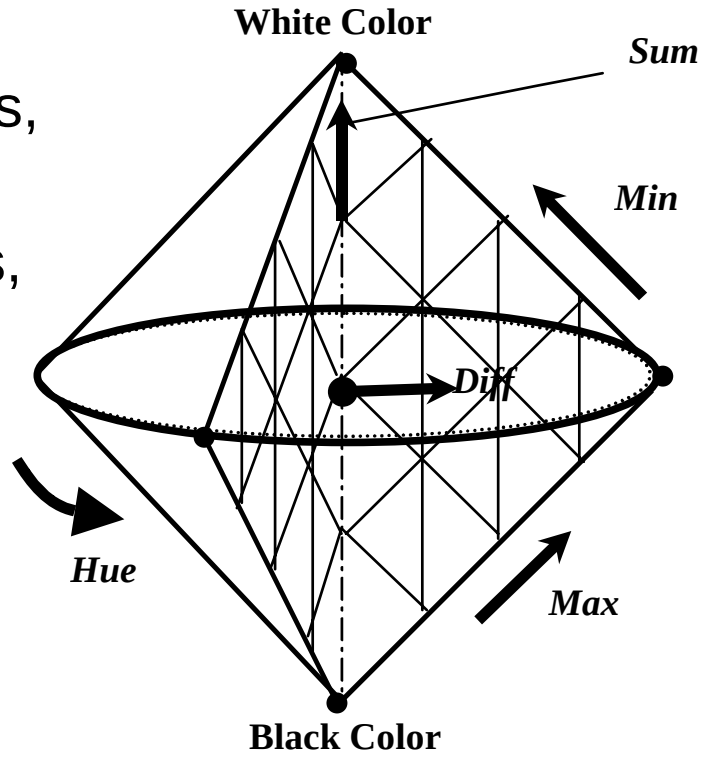
YCbCr Color Space

- Y is the Luminance and Cb and Cr are the Chrominance Values of this Color Space.
- Decouples intensity and color information
- A monochrome color representation has only the Y value.
- Used in JPEG/MPEG and video standards

HMMD Color Space

- HMMD (Hue-Max-Min-Diff) color space is closer to a perceptually uniform color space.
  - Min = min(R,G,B); Max = max(R,G,B);
  - Diff = Max – Min; Sum = (Max + Min)/2; Hue the same as in HSV

Max:indicates how much black color it has, giving a flavor of shade or blackness.
Min:indicates how much white color it has, giving a flavor of tint or whiteness.
Diff:indicates how much gray it contains and how close to the pure color, giving a flavor of tone or colorfulness.
Sum:simulates the brightness of the color.



```
void rgb2hmmd(int r, int g, int b, int[] hmmd) {

        float max = (int)Math.max(Math.max(r,g), Math.max(g,b));
        float min = (int)Math.min(Math.min(r,g), Math.min(g,b));
        float diff = (max - min);
        float sum = (float) ((max + min)/2.);

        float hue = 0;
        if (diff == 0)
                hue = 0;
        else if (r == max && (g - b) > 0)
                hue = 60*(g-b)/(max-min);
        else if (r == max && (g - b) <= 0)
                hue = 60*(g-b)/(max-min) + 360;
        else if (g == max)
                hue = (float) (60*(2.+(b-r)/(max-min)));
        else if (b == max)
                hue = (float) (60*(4.+(r-g)/(max-min)));

        hmmd[0] = (int)(hue);
        hmmd[1] = (int)(max);
        hmmd[2] = (int)(min);
        hmmd[3] = (int)(diff);
    }
```

Color Space Description in DDL

```
<complexType name="ColorSpaceType" final="#all">
   <sequence>
      <element name="ColorTransMat" minOccurs="0">
         <simpleType>
            <restriction>
               <simpleType>
                  <list itemType="mpeg7:unsigned16"/>
               </simpleType>
               <length value="9"/>
            </restriction>
         </simpleType>
      </element>
   </sequence>
   <attribute name="colorReferenceFlag" type="boolean"
use="optional" default="false"/>
   <attribute name="type" use="required">
      <simpleType>
         <restriction base="string">
            <enumeration value="RGB"/>
            <enumeration value="YCbCr"/>
            <enumeration value="HSV"/>
            <enumeration value="HMMD"/>
            <enumeration value="LinearMatrix"/>
            <enumeration value="Monochrome"/>
         </restriction>
      </simpleType>
   </attribute>
</complexType>
```

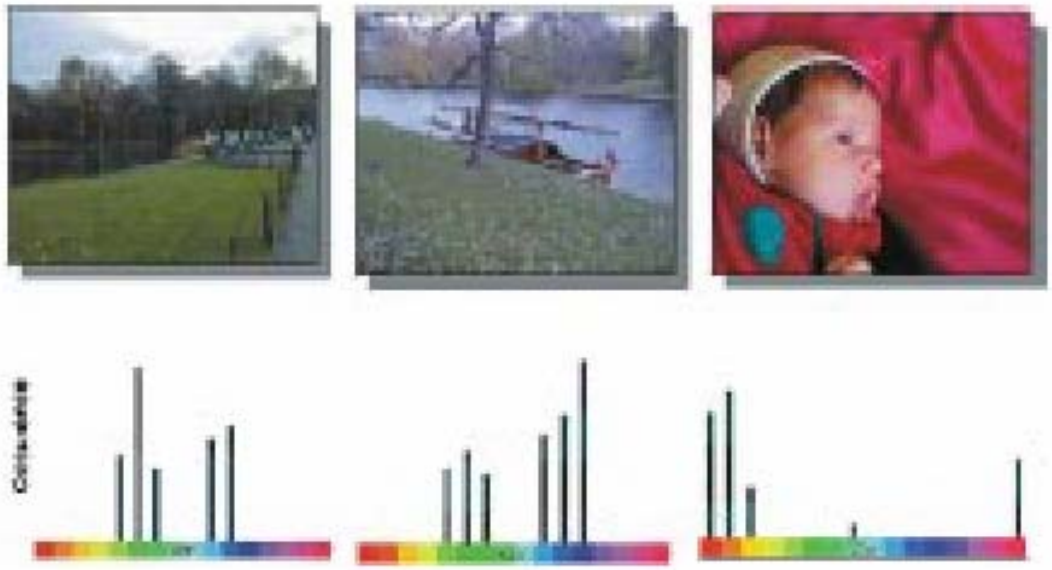
Color Quantization in DDL

This descriptor defines the uniform quantization of a color space. It provides a mapping from the floating point values of the color space to an integer representation in which each component is linearly scaled to the integer range [0,NumOfBins-1], where NumOfBins is the number of quantization levels of each color component.

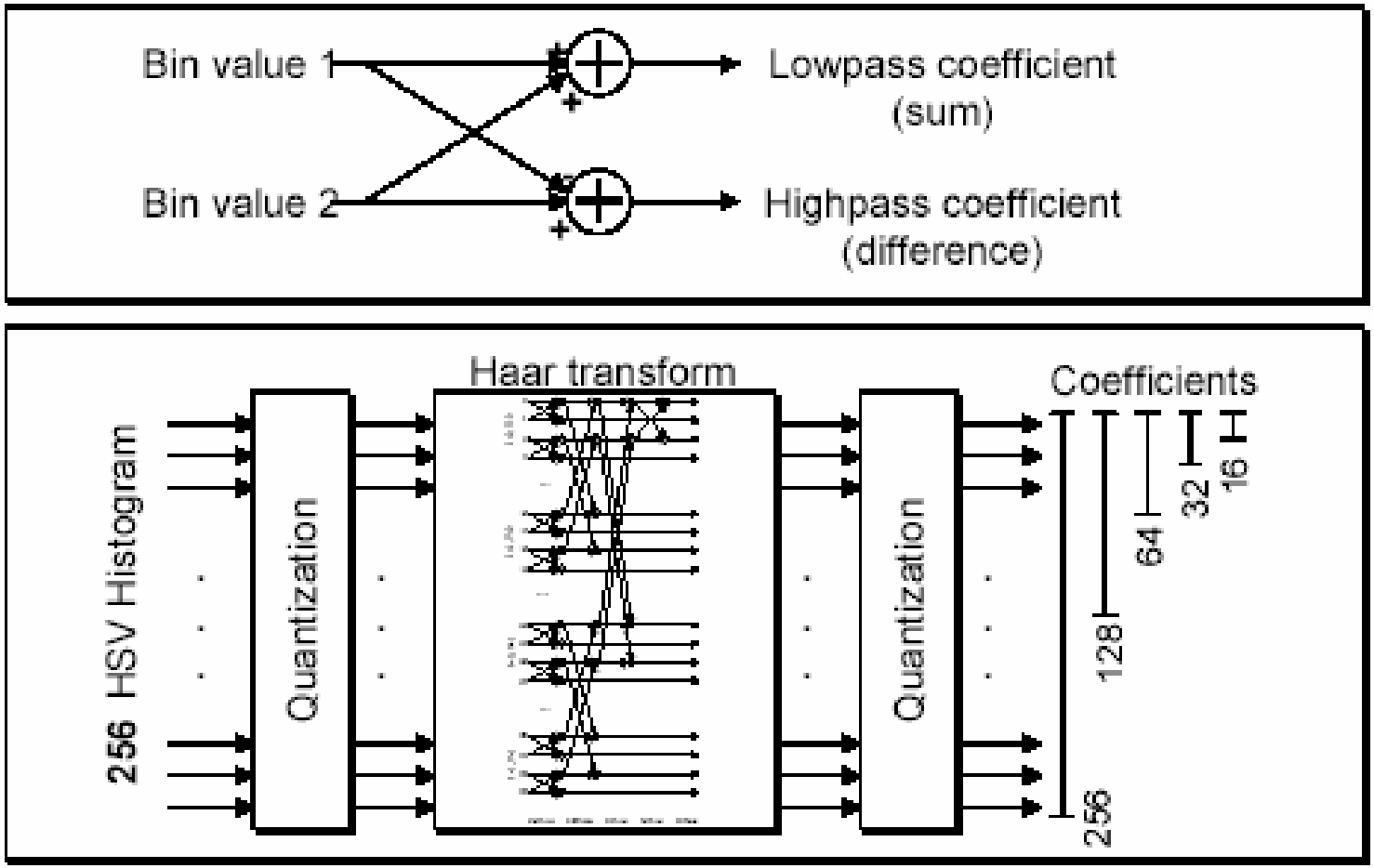
```
<complexType name="ColorQuantizationType" final="#all">
   <sequence minOccurs="1" maxOccurs="3">
      <element name="Component">
         <simpleType>
            <restriction base="string">
               <enumeration value="R"/>
               <enumeration value="G"/>
               <enumeration value="B"/>
               <enumeration value="Y"/>
               <enumeration value="Cb"/>
               <enumeration value="Cr"/>
               <enumeration value="H"/>
               <enumeration value="S"/>
               <enumeration value="V"/>
               <enumeration value="Max"/>
               <enumeration value="Min"/>
               <enumeration value="Diff"/>
               <enumeration value="Sum"/>
            </restriction>
         </simpleType>
      </element>
      <element name="NumOfBins" type="mpeg7:unsigned12"/>
   </sequence>
</complexType>
```

Scalable Color Descriptor

- A color histogram in HSV color space
- Encoded by Haar Transform



Scalable Color: Haar Transformation



Scalable Color: Haar Transformation

- Histo values tuncated to 11-bit and then mapped to non-linear 4-bit representation
  - If using 256 bins * 4 bits = 1024 bits/histo: may be not enough compact. That's why we use Haar transform.
  - The high-pass coefficients expressing differences between adjacent histogram bins usually have only small values. Exploiting this property, it is possible to truncate the high-pass coefficients to integer representation with only a low number of bits.

DDL of the Scalable Color

```
<complexType name="ScalableColorType" final="#all">
   <complexContent>
      <extension base="mpeg7:VisualDType">
         <sequence>
            <element name="Coeff" type="mpeg7:integerVector"/>
         </sequence>
         <attribute name="numOfCoeff" use="required">
            <simpleType>
               <restriction base="integer">
                  <enumeration value="16"/>
                  <enumeration value="32"/>
                  <enumeration value="64"/>
                  <enumeration value="128"/>
                  <enumeration value="256"/>
               </restriction>
            </simpleType>
         </attribute>
         <attribute name="numOfBitplanesDiscarded" use="required">
            <simpleType>
               <restriction base="integer">
                  <enumeration value="0"/>
                  <enumeration value="1"/>
                  <enumeration value="2"/>
                  <enumeration value="3"/>
                  <enumeration value="4"/>
                  <enumeration value="6"/>
                  <enumeration value="8"/>
               </restriction>
            </simpleType>
         </attribute>
      </extension>
   </complexContent>
</complexType>
```

This attribute specifies the number of bitplanes discarded in the scalable representation for each coefficient. Possible values are: 0, 1, 2, 3, 4, 6, 8.

Example: Scalable Color

```
<Mpeg7>
   <Description xsi:type="ContentEntityType">
      <MultimediaContent xsi:type="ImageType">
         <Image>
            <MediaLocator>
               <MediaUri>image.jpg</MediaUri>
            </MediaLocator>
            <TextAnnotation>
               <FreeTextAnnotation> Sunset scene </FreeTextAnnotation>
            </TextAnnotation>
            <VisualDescriptor xsi:type="ScalableColorType" numOfCoeff="16"
               numOfBitplanesDiscarded="0">
               <Coeff> 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 </Coeff>
            </VisualDescriptor>
         </Image>
      </MultimediaContent>
   </Description>
</Mpeg7>
```

Dominant Color Descriptor

- Clustering colors into a small number of representative colors
- It can be defined for each object, regions, or the whole image
- $F = \{ \{c_i, p_i, v_i\}, s\}$
  - c_i : Representative colors
  - p_i : Their percentages in the region
  - v_i : Color variances
  - s : Spatial coherency

DominantColorType

```
<complexType name="DominantColorType" final="#all">
   <complexContent>
      <extension base="mpeg7:VisualDType">
         <sequence>
            <element name="ColorSpace" type="mpeg7:ColorSpaceType"
                  minOccurs="0"/>
            <element name="ColorQuantization"
                  type="mpeg7:ColorQuantizationType" minOccurs="0"/>
            <element name="SpatialCoherency" type="mpeg7:unsigned5"/>
            <element name="Value" minOccurs="1" maxOccurs="8">
               <complexType>
                  <sequence>
                     <element name="Percentage" type="mpeg7:unsigned5"/>
                     <element name="Index">
                        <simpleType>
                           <restriction>
                              <simpleType>
                                 <list itemType="mpeg7:unsigned12"/>
                              </simpleType>
                              <length value="3"/>
                           </restriction>
                        </simpleType>
                     </element>
                     <element name="ColorVariance" minOccurs="0">
                        <simpleType>
                           <restriction>
                              <simpleType>
                                 <list itemType="mpeg7:unsigned1"/>
                              </simpleType>
                              <length value="3"/>
                           </restriction> </sequence> </complexType> </element> </sequence>
   </attribute> </extension> </complexContent> </complexType>
```

DominantColor Meaning

- Value

This element specifies an array of elements that hold percentages and values of colors in a visual item. The array elements consist of Percentage, ColorValueIndex and ColorVariance.

- Percentage

This element specifies the percentage of pixels that have the associated color value. The percentage value is uniformly quantized to 5 bits with 0 corresponding to 0 percentage and 31 corresponding to 100%. Note that the sum of the Percentage values for a given visual item does not have to be equal to 100%.

- Index

This element specifies the index of the dominant color in the selected color space as defined in ColorQuantization. The number of bits for each component is derived from the ColorQuantization element.

- ColorVariance

This element specifies an integer array containing the value of the variance of color values of pixels corresponding to the dominant color in the selected color space, i.e.

$$CV_j = \frac{1}{N} \sum_{k=0}^{N-1} \left(m_j - p_{kj} \right)^2$$

where *j* indexes the color component, *mj* is *j*-th component of the dominant color, *pkj* is *j*-th component of the *k*-th pixel value, and the summation is over *N* pixels corresponding to the dominant color under consideration.

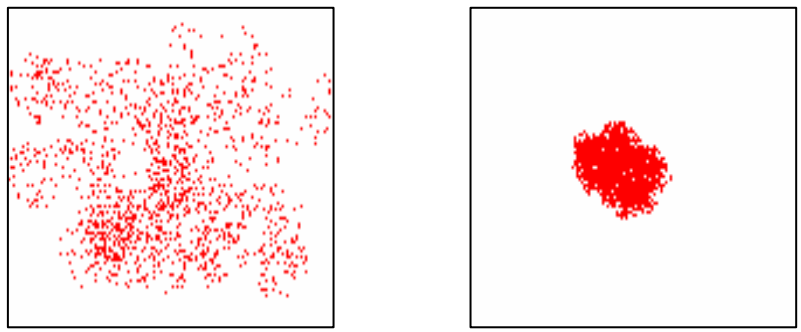
Spatial Coherency

SpatialCoherency

- This element specifies the spatial coherency of the dominant colors described by the descriptor. It is computed as a single value by the weighted sum of per-dominant-color spatial coherencies. The weight is proportional to the number of pixels corresponding to each dominant color. Spatial coherency per dominant color captures how coherent the pixels corresponding to the dominant color are and whether they appear to be a solid color in the given image region

- Spatial coherency per dominant color is computed by the normalized average connectivity (8-connectedness) for the corresponding dominant color pixels.

- Examples of low (a) and high (b) spatial coherency of color.



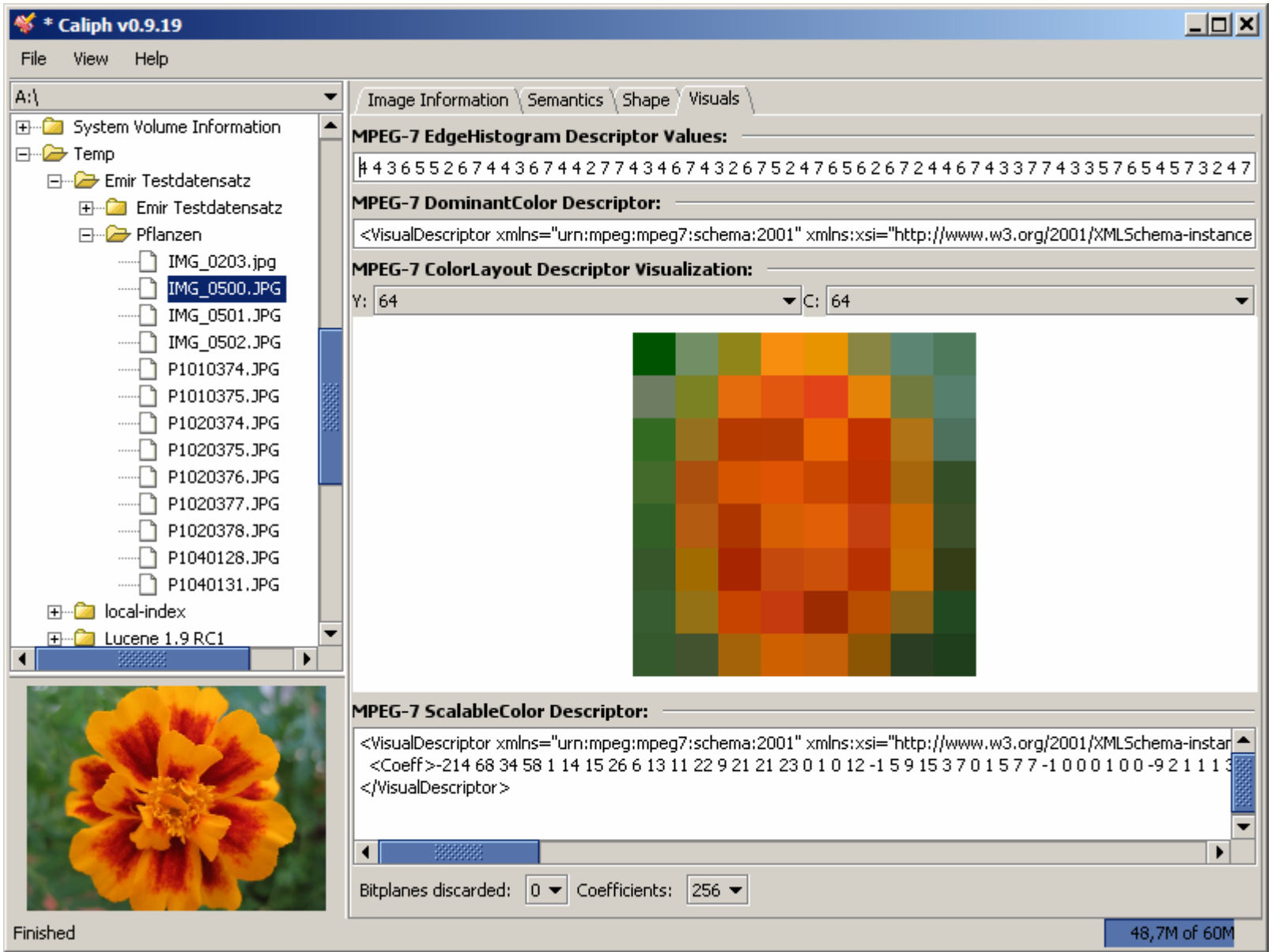
Example: a completely red image

```
<Mpeg7 xmlns="urn:mpeg:mpeg7:schema:2001"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:mpeg7="urn:mpeg:mpeg7:schema:2001"
xsi:schemaLocation="urn:mpeg:mpeg7:schema:2001 Mpeg7-2001.xsd">
    <Description xsi:type="ContentEntityType">
        <MultimediaContent xsi:type="ImageType">
            <Image>
                <MediaLocator>
                    <MediaUri> file://red.jpg</MediaUri>
                </MediaLocator>
                <TextAnnotation>
                    <FreeTextAnnotation>
                        A completely red image :-)
                    </FreeTextAnnotation>
                </TextAnnotation>
                <VisualDescriptor xsi:type="DominantColorType">
                    <SpatialCoherency>31</SpatialCoherency>
                    <Value>
                        <Percentage>31</Percentage>
                        <Index>255 0 0</Index>
                        <ColorVariance>1 0 0</ColorVariance>
                    </Value>
                </VisualDescriptor>
            </Image>
        </MultimediaContent>
    </Description>
</Mpeg7>
```

Color Layout Descriptor

- Clustering the image into 64 (8x8) blocks
- Deriving the average color of each block (or using DCD)
- Applying DCT and encoding
- Efficient for
  - Sketch-based image retrieval
  - Content Filtering using image indexing
  - Compact: even 63 bits are enough !

Example: CLD visualization in Caliph

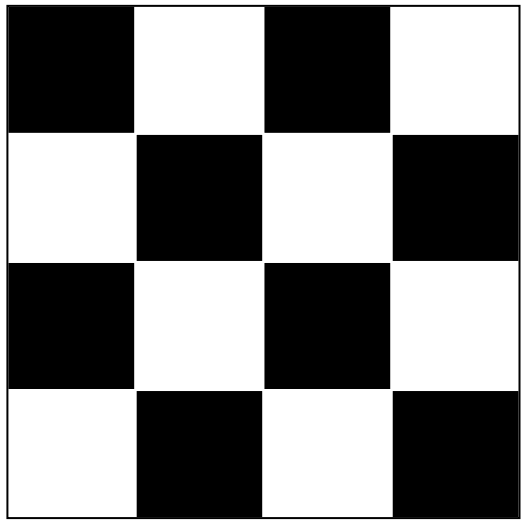


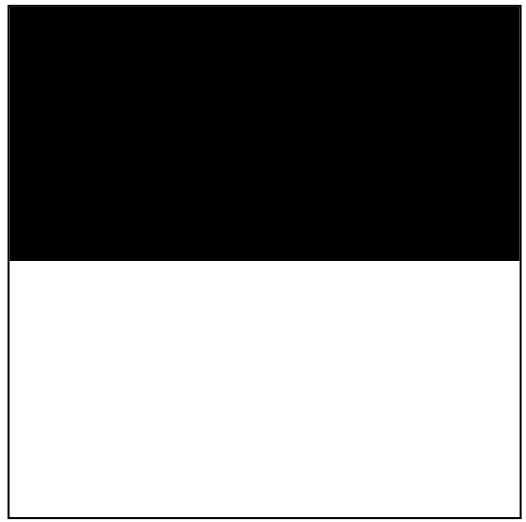
Color Structure Descriptor

- Scanning the image by an 8x8 pixel block
- Counting the number of blocks containing each color
- Generating a color histogram (HMMD)
- Main usages:
  - Still image retrieval
  - Natural images retrieval

Color Structure Descriptor (CSD)

- The CSD represents an image by both the color distribution and the local structure.



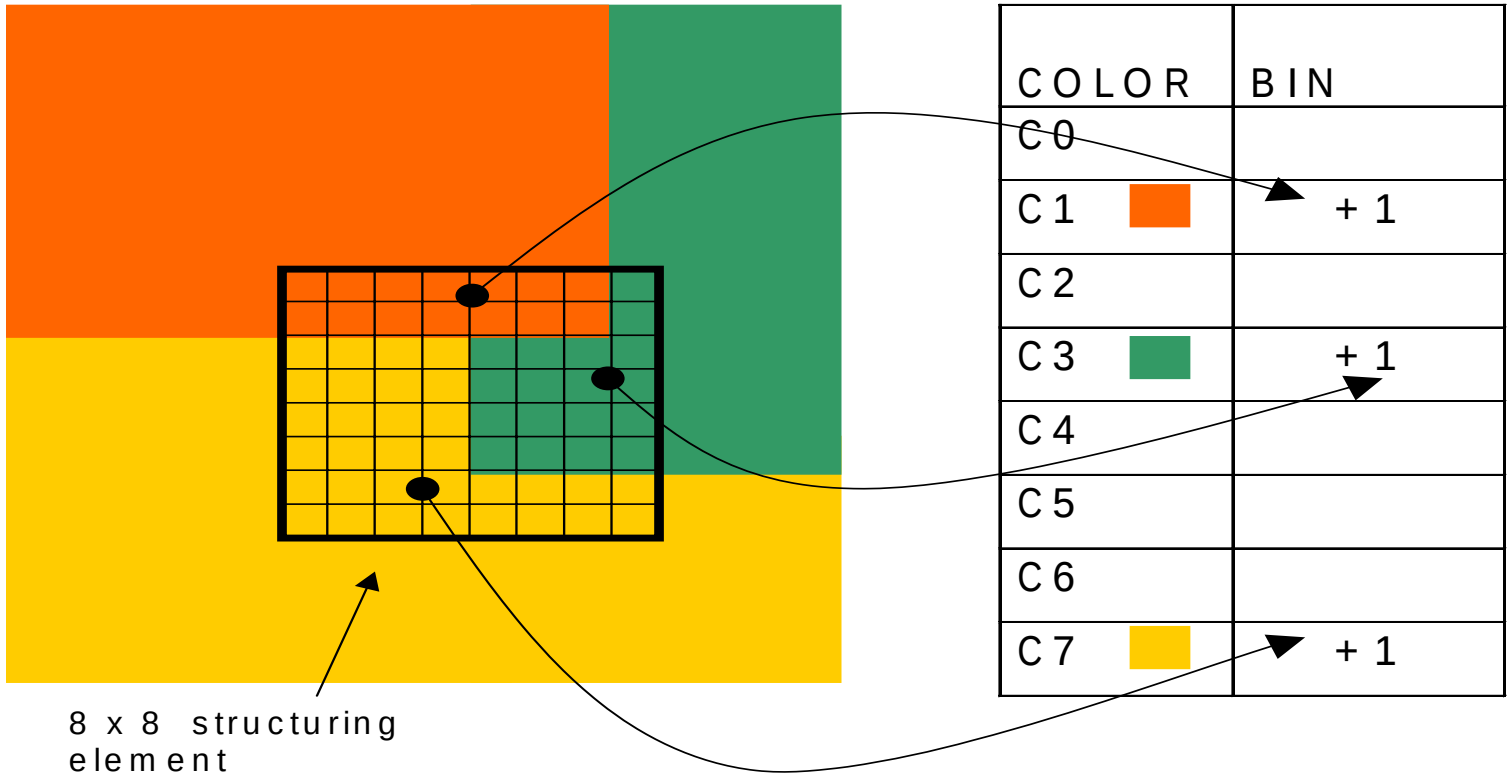


Scalable Color Descriptor may not distinguish both images.
But the CSD can do it.

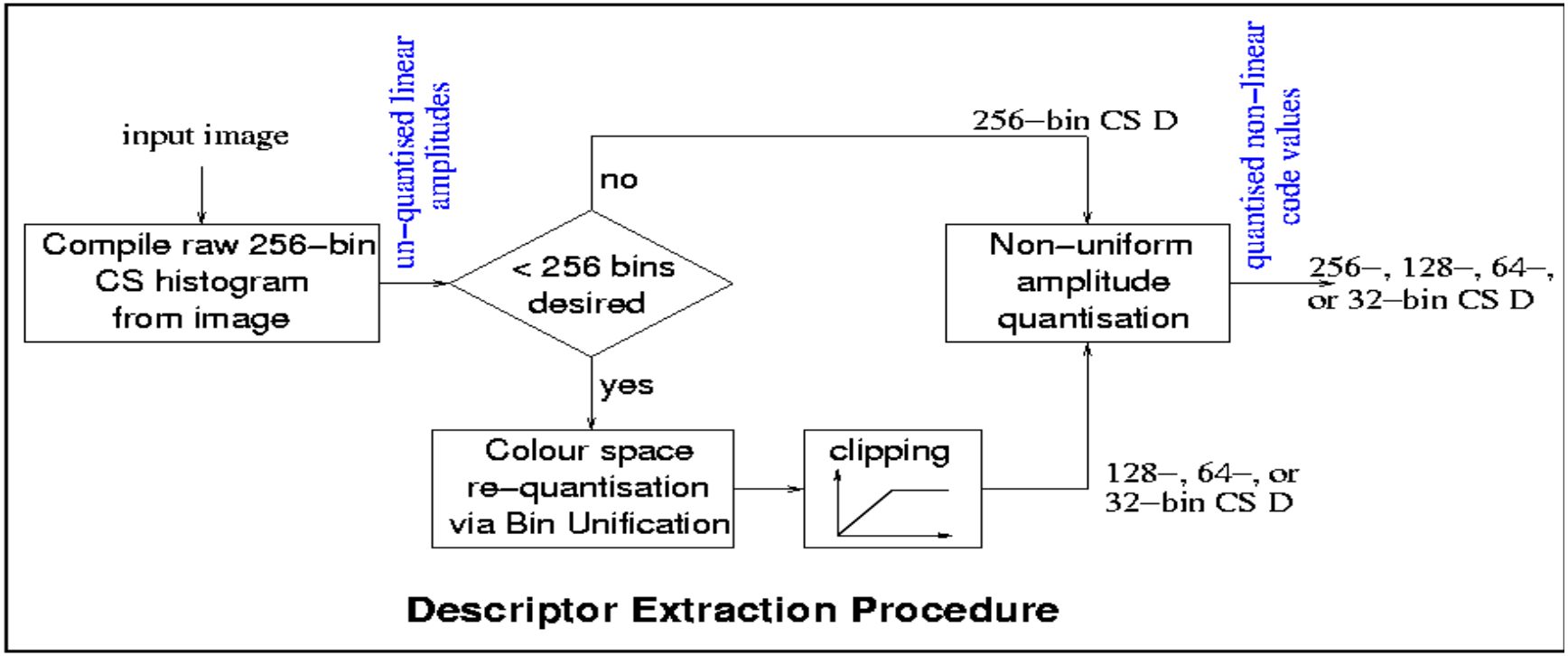
Color Structure Descriptor (CSD)

- The CSD is identical to a 1-D color histogram, but has a different semantics:
  - CSD = h(m), m $\in$ {0, 1, ..., M-1},
  - M is chosen from the set of {256, 128, 64, 32}.
  - The bin value is quantized to 8 bits.
  - The histogram values h(m) count how many colors are present in a so-called structuring element.
  - The structuring element is a 8x8 element running over the image. For any position the increase in the bins is computed if a color is present or not.
  - The color space is HMMD color space.

Color Structure Descriptor (CSD) - Example



Color Structure Descriptor (CSD) - Extraction

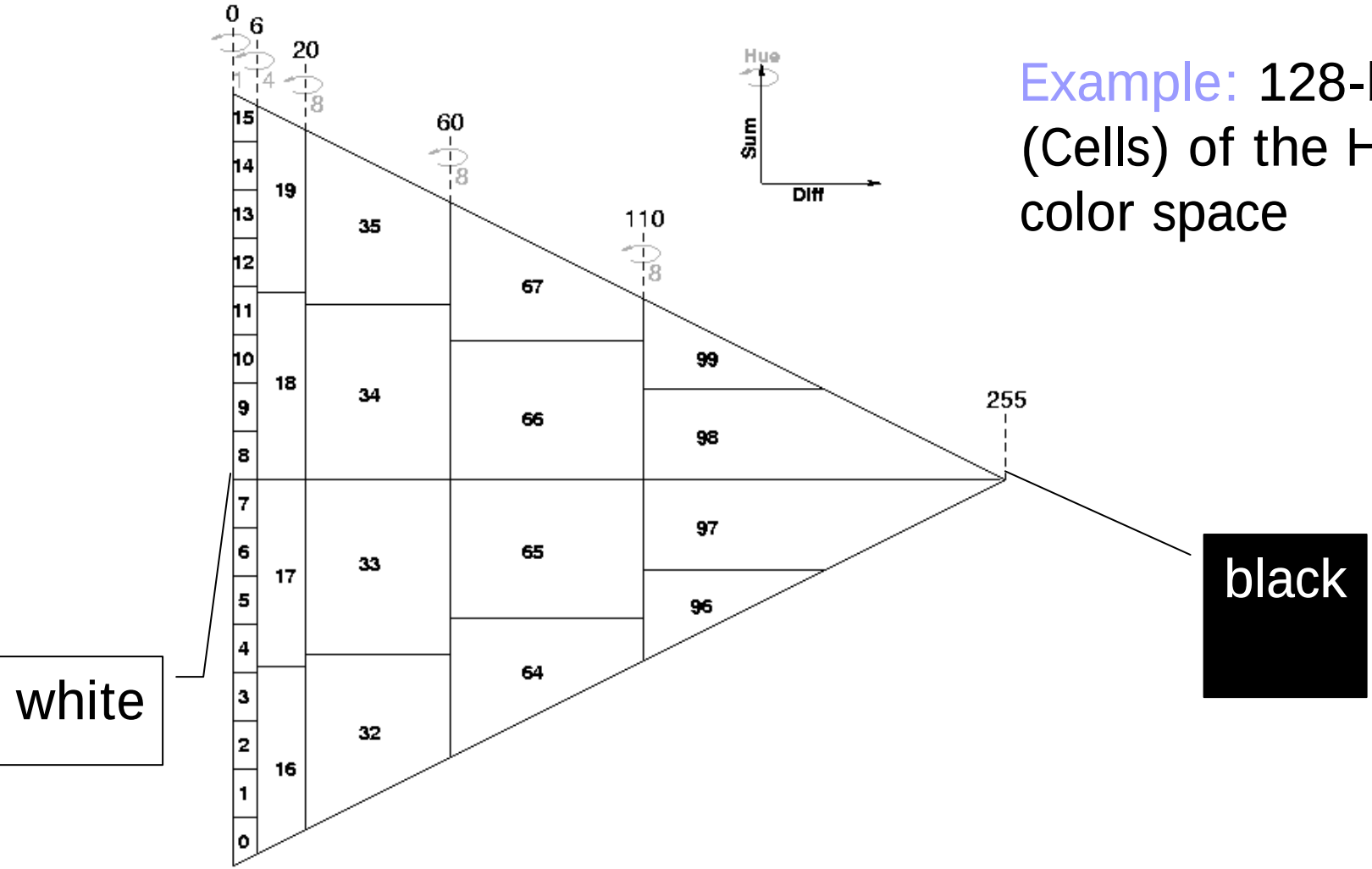


Color Structure Descriptor (CSD) - DDL

```
<complexType name="ColorStructureType" final="#all">
   <complexContent>
      <extension base="mpeg7:VisualDType">
         <sequence>
            <element name="Values">
               <simpleType>
                  <restriction>
                     <simpleType>
                        <list itemType="mpeg7:unsigned8"/>
                     </simpleType>
                     <minLength value="1"/>
                     <maxLength value="256"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="colorQuant" type="mpeg7:unsigned3" use="required"/>
      </extension>
   </complexContent>
</complexType>
```

8-bit quantized values with 1-256 histogram bins specified in colorQuant

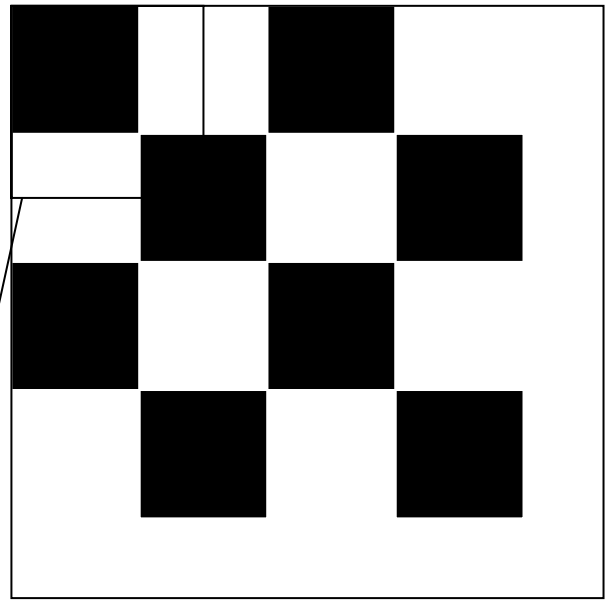
Example of Color Structure Descriptor (1)

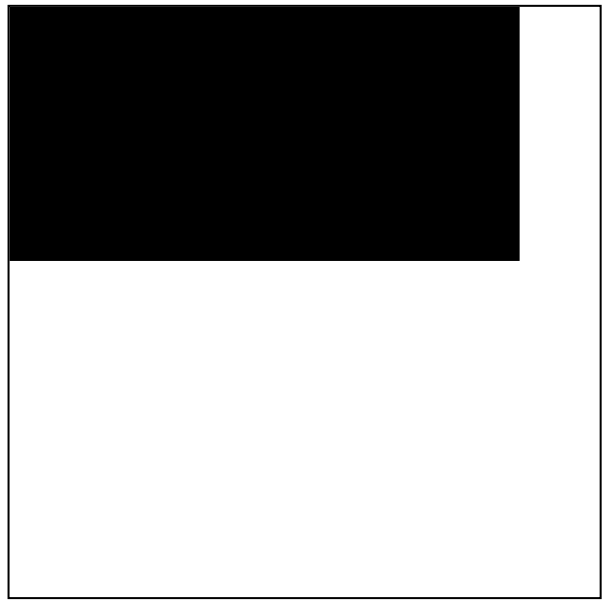


Example: 128-bins (Cells) of the HMMD color space

Example of Color Structure Descriptor (2)

- Example Images (left and right)





8x8

Example of Color Structure Descriptor (3)

- XML-file for the Left Image

```
<Mpeg7 xmlns="urn:mpeg:mpeg7:schema:2001" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:mpeg7="urn:mpeg:mpeg7:schema:2001" xsi:schemaLocation="urn:mpeg:mpeg7:schema:2001
    Mpeg7-2001.xsd">
    <Description xsi:type="ContentEntityType">
        <MultimediaContent xsi:type="ImageType">
            <Image>
                <MediaLocator>
                    <MediaUri> file://red.jpg</MediaUri>
                </MediaLocator>
                <TextAnnotation>
                    <FreeTextAnnotation>
                        The left black and white image
                    </FreeTextAnnotation>
                </TextAnnotation>
                <VisualDescriptor xsi:type="ColorStructureType" colorQuant="128">
                    <Values>  9 0 0 0 ... 0  9 </Values>
                </VisualDescriptor>
            </Image>
        </MultimediaContent>
    </Description>
</Mpeg7>
```

Example of Color Structure Descriptor (4)

- XML-file for Right Image

```
<Mpeg7 xmlns="urn:mpeg:mpeg7:schema:2001" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:mpeg7="urn:mpeg:mpeg7:schema:2001" xsi:schemaLocation="urn:mpeg:mpeg7:schema:2001
    Mpeg7-2001.xsd">
    <Description xsi:type="ContentEntityType">
        <MultimediaContent xsi:type="ImageType">
            <Image>
                <MediaLocator>
                    <MediaUri> file://red.jpg</MediaUri>
                </MediaLocator>
                <TextAnnotation>
                    <FreeTextAnnotation>
                        A black and white image
                    </FreeTextAnnotation>
                </TextAnnotation>
                <VisualDescriptor xsi:type="ColorStructureType" colorQuant="128">
                    <Values>  6 0 0 0 ... 0  7 </Values>
                </VisualDescriptor>
            </Image>
        </MultimediaContent>
    </Description>
</Mpeg7>
```

GoF/GoP Color Descriptor

- Extends Scalable Color Descriptor
- Generates the color histogram for a video segment or a group of pictures
- Calculation methods:
  - Average
  - Median
  - Intersection

Example:
Group of Frames and Group of Pictures

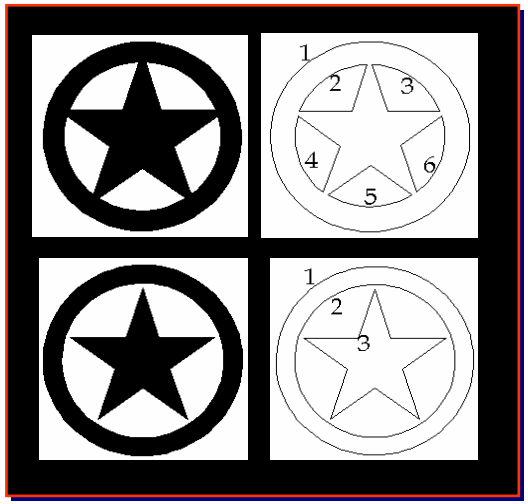
```
<Segment type="VideoSegment" id="VS1">
<MediaTime>
    <MediaTimePoint>T00H00M00S00N24F</MediaTimePoint>
    <MediaDuration>T00H00M02S13N24F</MediaDuration>
</MediaTime>
<Descriptor xsi:type="GoFGoPColorTypeMediaTime" Aggregation="Average">
    <Descriptor xsi:type="ScalableColorType" NumberOfCoefficients="4"
    NumberOfBitplanesDiscarded="0">
          <Coefficients>0 0 0 0 0 0 0 0 0 0 0 0 0 1 15 0 2 2 5 4 2 4 2 0 2 1 2 0 0 0
    1 0 2 1 4 6 9 14 1 1 0 0 1 0 0 0 0 0 0 1 3 11 126 18 3 1 0 1 0 0 0 0 0 1 3 4 11
    8 12 23 6 5 5 7 8 1 0 2 0 0 4 11 23 100 211 203 16 7 1 2 3 1 1 1 0 0 0 0 3 45
    268 180 3 0 0 0 0 0 0 0 0 0 0 0 0 6 260 124 0 0 0 0 0 0 0 0 0 2 9 10 31 53 64 89
    22 12 11 12 6 1 2 3 0 0 0 1 18 127 494 334 11 0 0 0 1 0 0 0 0 0 0 0 0 1 1465
    1003 0 0 0 0 0 0 0 0 0 0 0 0 0 2 16959 75007 0 0 0 0 0 0 0 0 0 37 15 15 80 150
    237 277 60 48 12 7 8 0 4 4 0 0 0 0 8 64 504 269 3 0 0 0 0 0 0 0 0 0 0 0 0 0 131
    85 0 0 0 0 0 0 0 0 0 0 0 0 0 0 179 91 0 0 0 0 0 0 0 0 0</Coefficients>
    </Descriptor>
</Descriptor>
<Segment xsi:type="StillRegionType" name="ID" type="IDREF" use="keyframe">
<MediaTime>
    <MediaTimePoint>T00H00M00S00N24F</MediaTimePoint>
</MediaTime>
</Segment>
</Segment>
```

Shape Descriptors

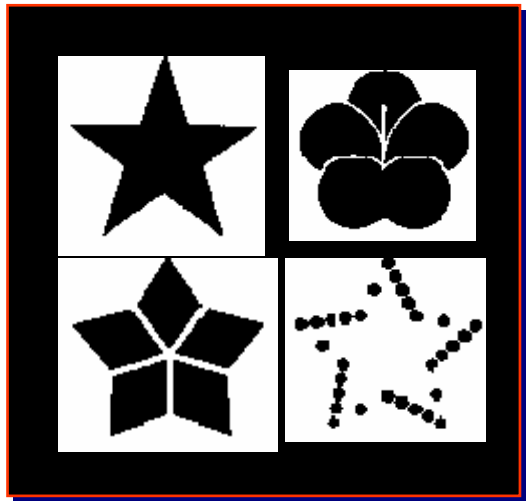
- Region-based Descriptor
- Contour-based Shape Descriptor
- 2D/3D Shape Descriptor
- 3D Shape Descriptor

Shape Descriptors

- Contour shape
- Region shape



Contour-based
shape descriptor

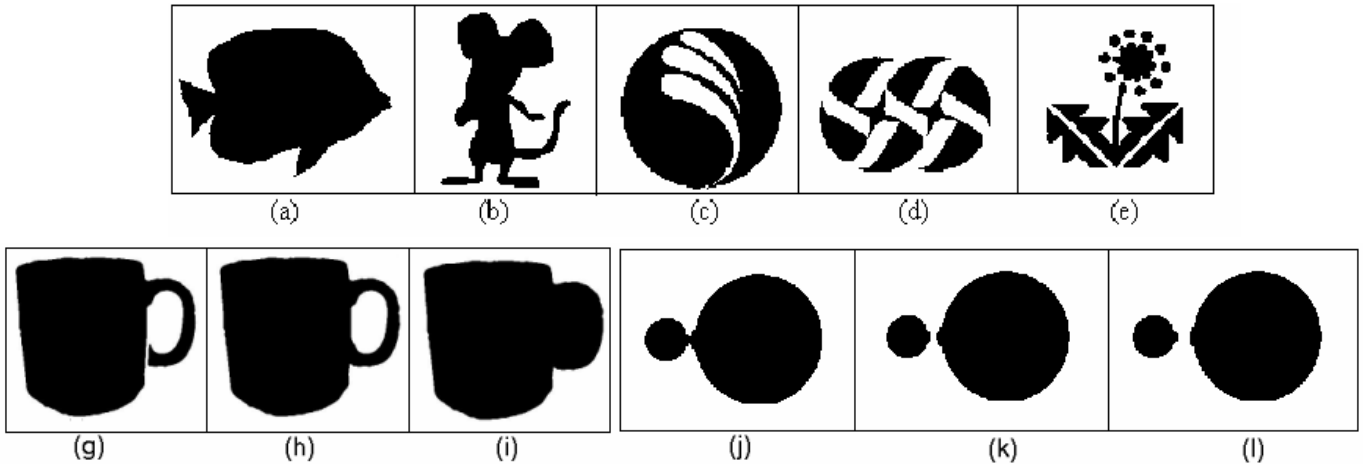


Region-based
shape descriptor

Region-based Descriptor

- Expresses pixel distribution within a 2-D object region
- Employs a complex 2D-Angular Radial Transformation (ART)
- Advantages:
  - Describes complex shapes with disconnected regions
  - Robust to segmentation noise
  - Small size
  - Fast extraction and matching

Region-based Descriptor (2)



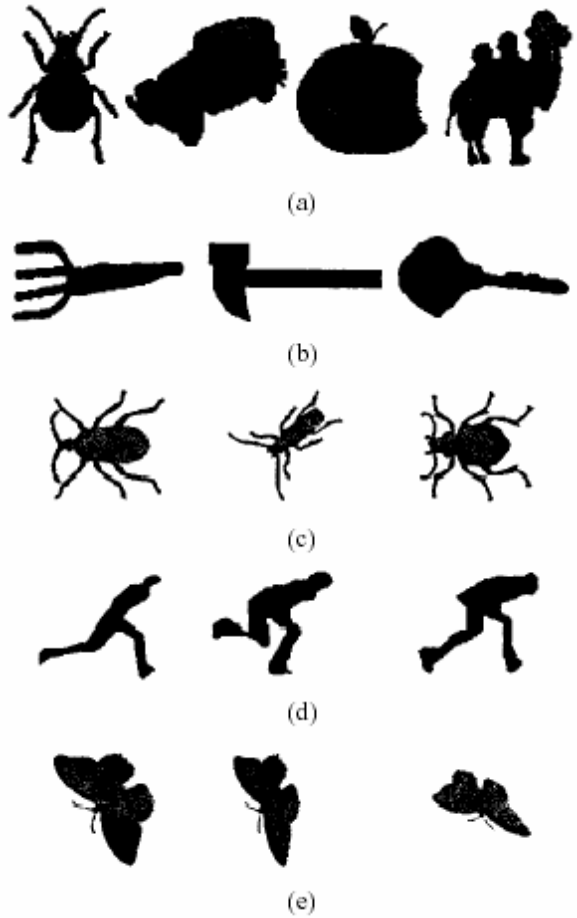
- Applicable to figures (a) – (e)
- Distinguishes (i) from (g) and (h)
- (j), (k), and (l) are similar

Contour-Based Descriptor

- It is based on Curvature Scale-Space representation
- Advantages:
  - Captures the shape very well
  - Robust to the noise, scale, and orientation
  - It is fast and compact

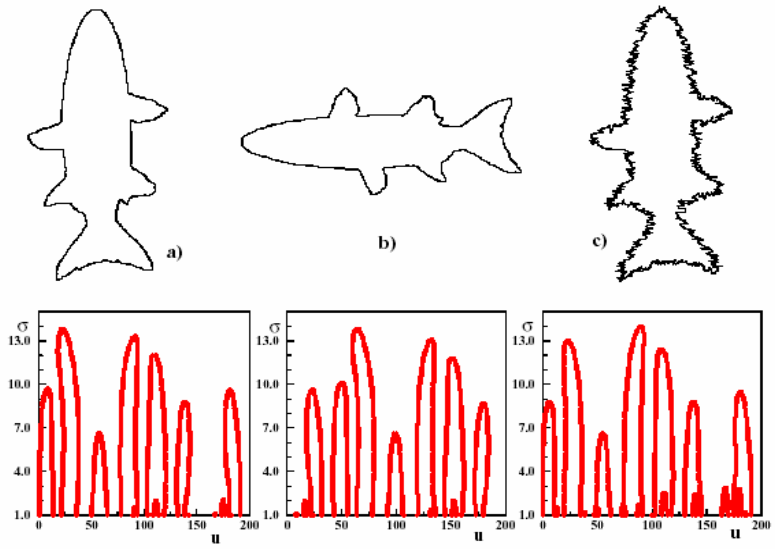
Contour-Based Descriptor (2)

- Applicable to (a)
- Distinguishes differences in (b)
- Find similarities in (c) - (e)

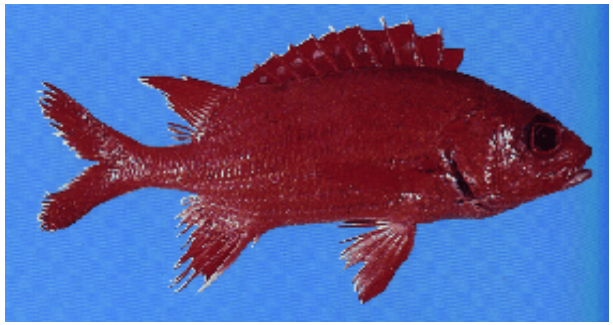


Curvature Scale-Space

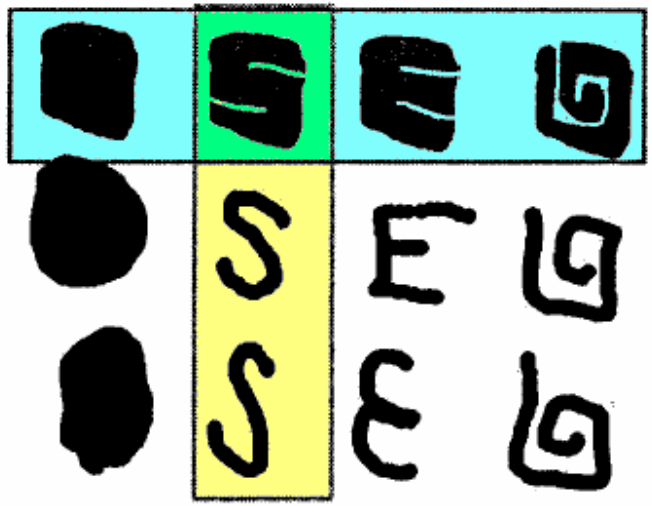
- Finds curvature zero crossing points of the shape's contour (key points)
- Reduces the number of key points step by step, by applying Gaussian smoothing
- The position of key points are expressed relative to the length of the contour curve



Curvature Scale Space (2)



Comparison



- Blue: Similar shapes by Region-Based
- Yellow: Similar shapes by Contour-Based

2D/3D Shape Descriptor

- A 3D object can be roughly described by snapshots from different angels
- Describes a 3D object by a number of 2D shape descriptors
- Similarity Matching: matching multiple pairs of 2D views

3D Shape Descriptor

- Based on Shape spectrum
- An extension of Shape Index (A local measure of 3D Shape to 3D meshes)
- Captures information about local convexity
- Computes the histogram of the shape index over the whole 3D surface

Texture Descriptors

- Texture refers to a visual pattern that has properties of homogeneity that do not result from the presence of only a single color or intensity.

- Homogenous Texture Descriptor
- Non-Homogenous Texture Descriptor (Edge Histogram)

Texture Descriptors: possible queries

- An image can be considered as a mosaic of homogeneous textures so that these texture features associated with the regions can be used to index the image data.
  - For instance, a user browsing an aerial image database may want to identify all parking lots in the image collection. A parking lot with cars parked at regular intervals is an excellent example of a homogeneous textured pattern when viewed from a distance, such as in an Air Photo.
  - Similarly, agricultural areas and vegetation patches are other examples of homogeneous textures commonly found in aerial and satellite imagery.

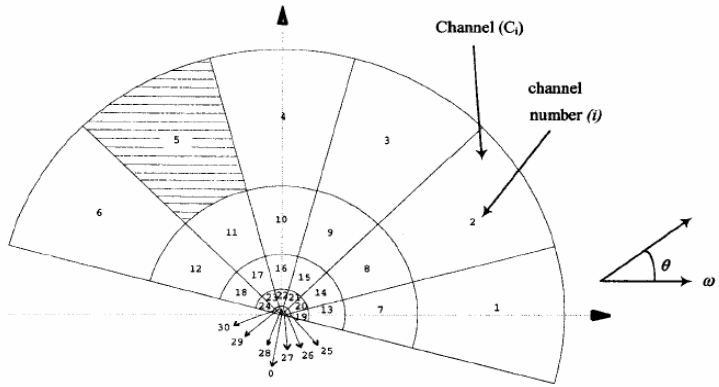
- Examples of queries that could be supported in this context could include “Retrieve all Land-Satellite images of Santa Barbara which have less than 20% cloud cover” or “Find a vegetation patch that looks like this region”.

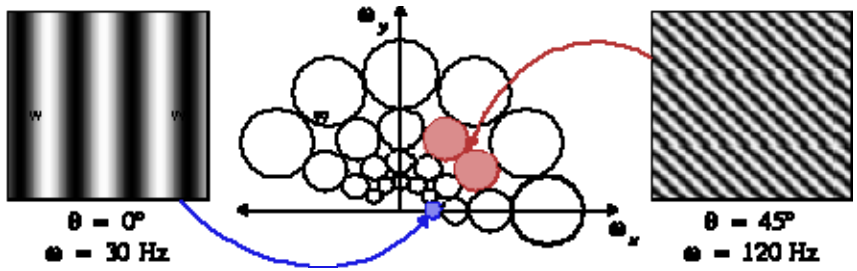
Homogenous Texture Descriptor

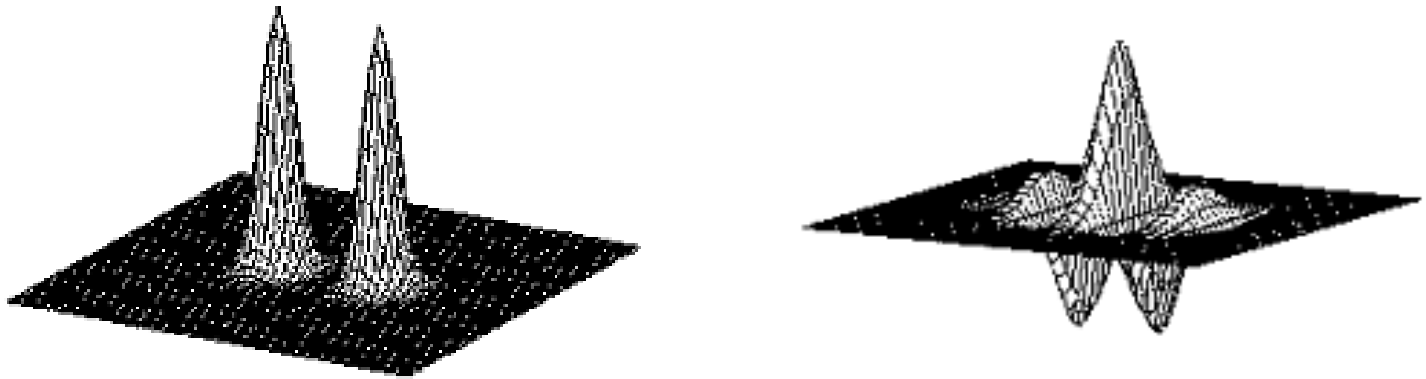
- Partitioning the frequency domain into 30 channels (modeled by a 2D-Gabor function)
  - 5x6 = 30 where 5 is the number of “scales” and 6 is the number of “directions” used in the multi-resolution decomposition using Gabor functions.
- Computing the energy and energy deviation for each channel
- Computing mean and standard variation of frequency coefficients
- $F = \{f_{DC}, f_{SD}, e_1,\ldots, e_{30}, d_1,\ldots, d_{30}\}$
- An efficient implementation:
  - Radon transform followed by Fourier transform

2D-Gabor Function

- It is a Gaussian weighted sinusoid
  - Gabor filters can be considered orientation and scale tunable edge & line detectors
- It is used to model individual channels
- Each channel filters a specific type of texture



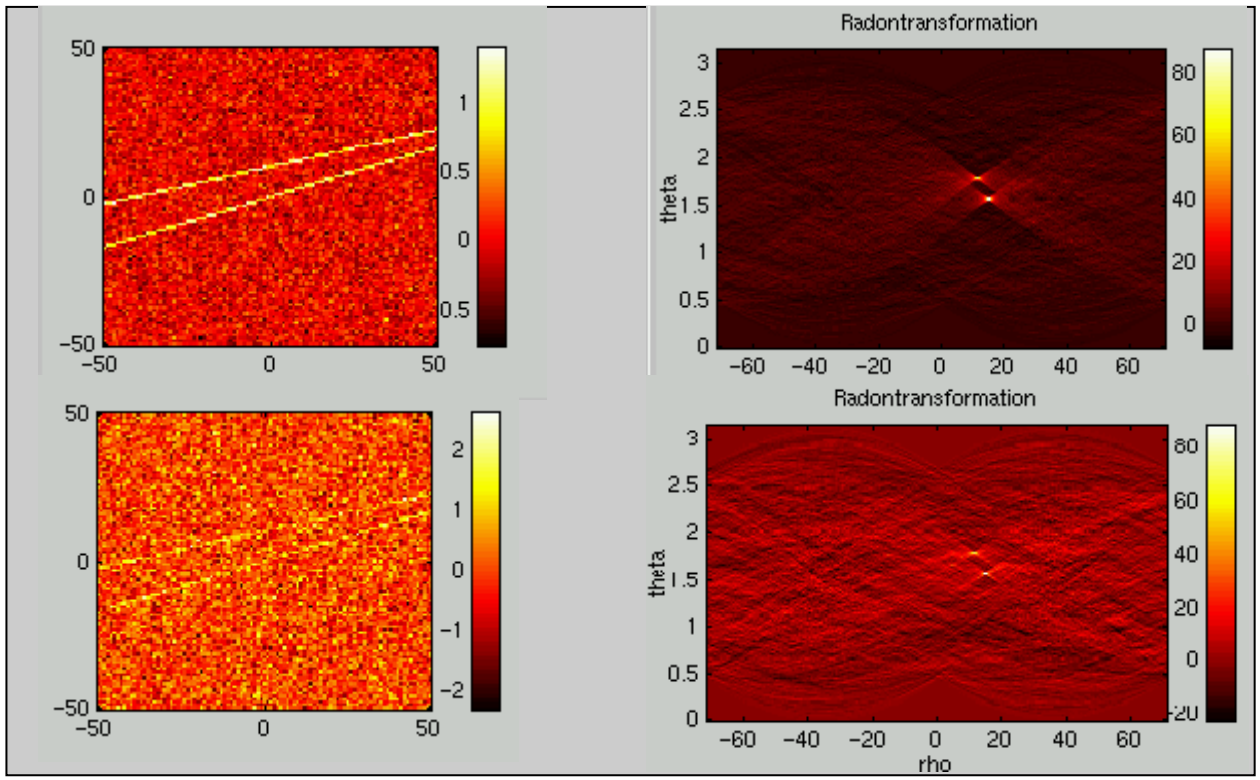




- Gabor filter: frequency (left) and spatial (right) domain

Radon Transform

- Transforms images with lines into a domain of possible line parameters
- Each line will be transformed to a peak point in the resulted image

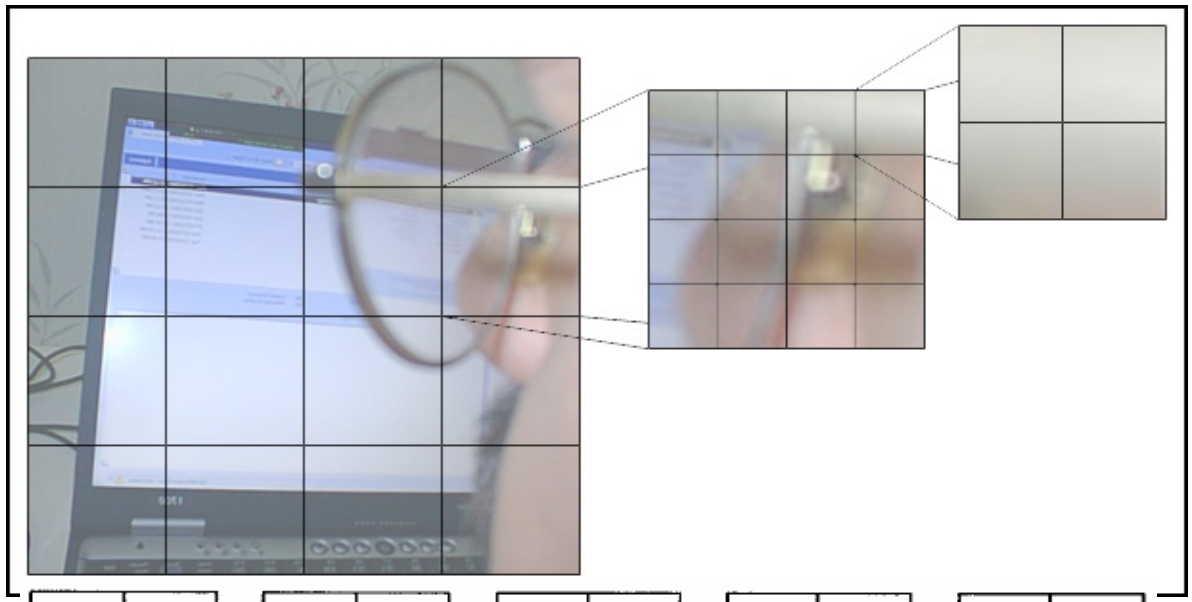


Non-Homogenous Texture Descriptor: EHD

- Represents the spatial distribution of five types of edges
  - vertical, horizontal, 45°, 135°, and non-directional
- Dividing the image into 16 (4x4) blocks
- Generating a 5-bin histogram for each block
- It is scale invariant

- A given image is first divided into 16 sub-images 4x4 and local edge histograms are computed for each sub-image.
- To compute the edge histogram, each of the 16 sub-images is further subdivided into image blocks. The size of each image block is proportional to the size of the original image and is assumed to be a multiple of two.
- Each image block is then partitioned into four 2x2 blocks of pixels, and the pixel intensities for these four divisions are computed by averaging the luminance values of the existing pixels.
- There are 16 sub-images and each has a five-bin histogram, a total of 16x5=80 bins in the histogram is achieved.

Non-Homogenous Texture Descriptor (2)



1	-1
1	-1

1	1
-1	-1

$\sqrt{2}$	0
0	$-\sqrt{2}$

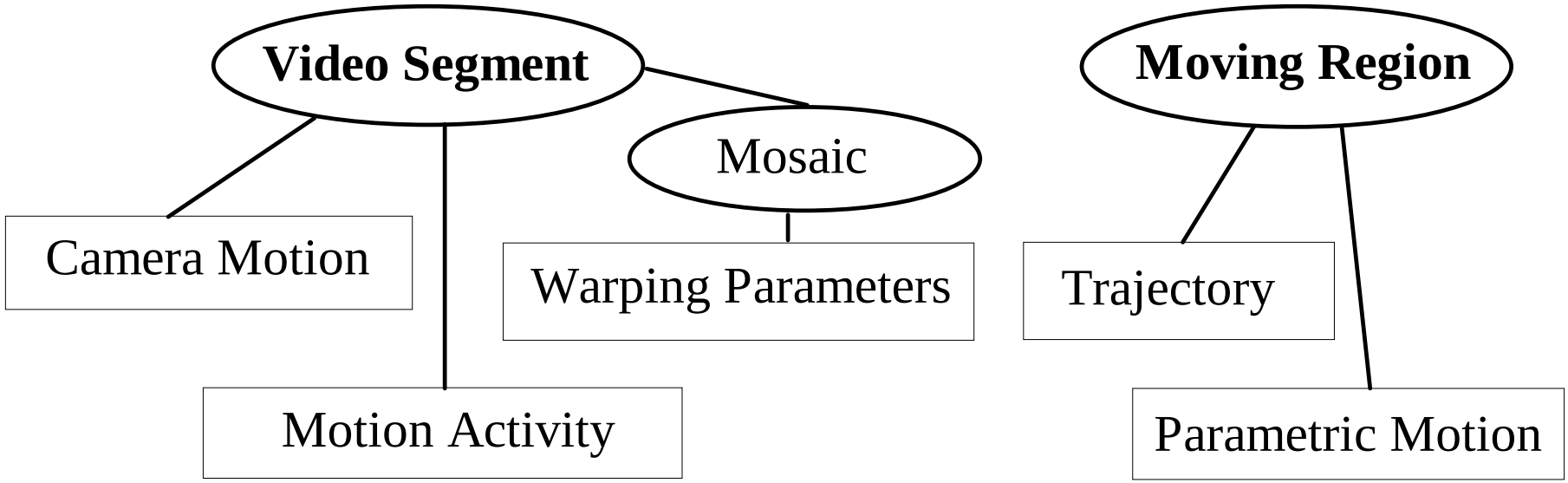
0	$\sqrt{2}$
$-\sqrt{2}$	0

2	-2
-2	2

Motion Descriptors

- Motion Activity Descriptors
- Camera Motion Descriptors
- Motion Trajectory Descriptors
- Parametric Motion Descriptors

Motion Descriptors

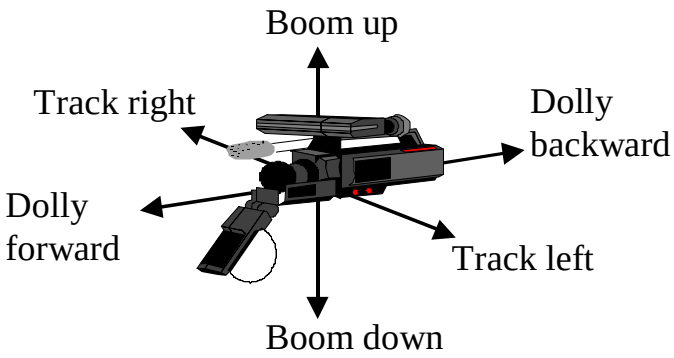


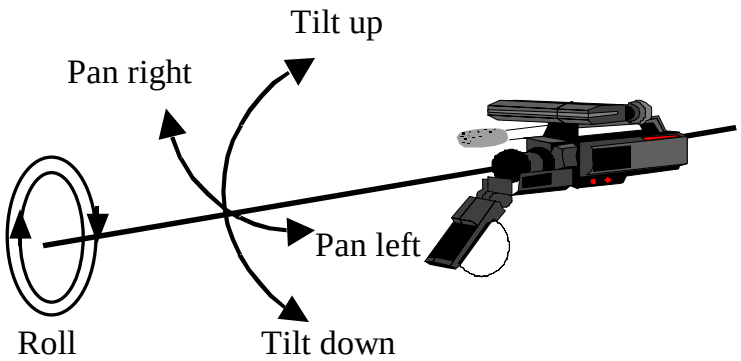
Motion Activity Descriptor

- Captures '*intensity of action*' or '*pace of action*'
- Based on standard deviation of motion vector magnitudes
- Quantized into a 3-bit integer [1, 5]

Camera Motion Descriptor

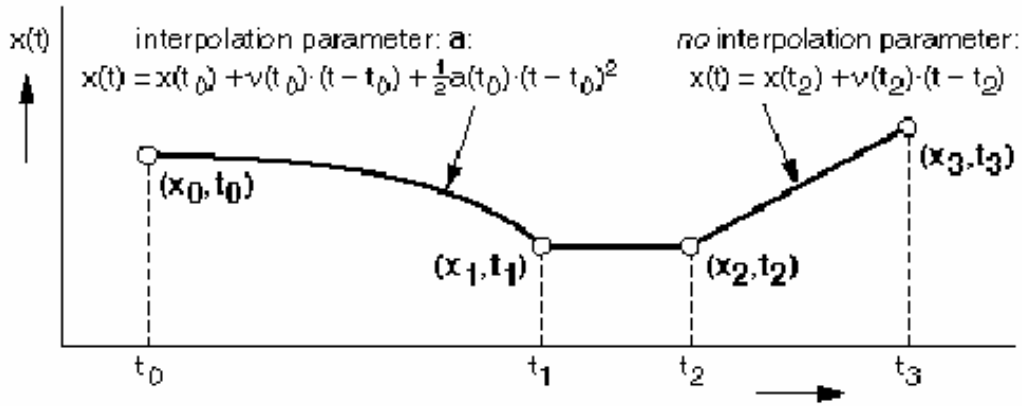
- Describes the movement of a camera or a virtual view point
- Supports 7 camera operations





Motion Trajectory

- Describes the movement of one representative point of a specific region
- A set of key-points (x, y, z, t)
- A set of interpolation functions describing the path



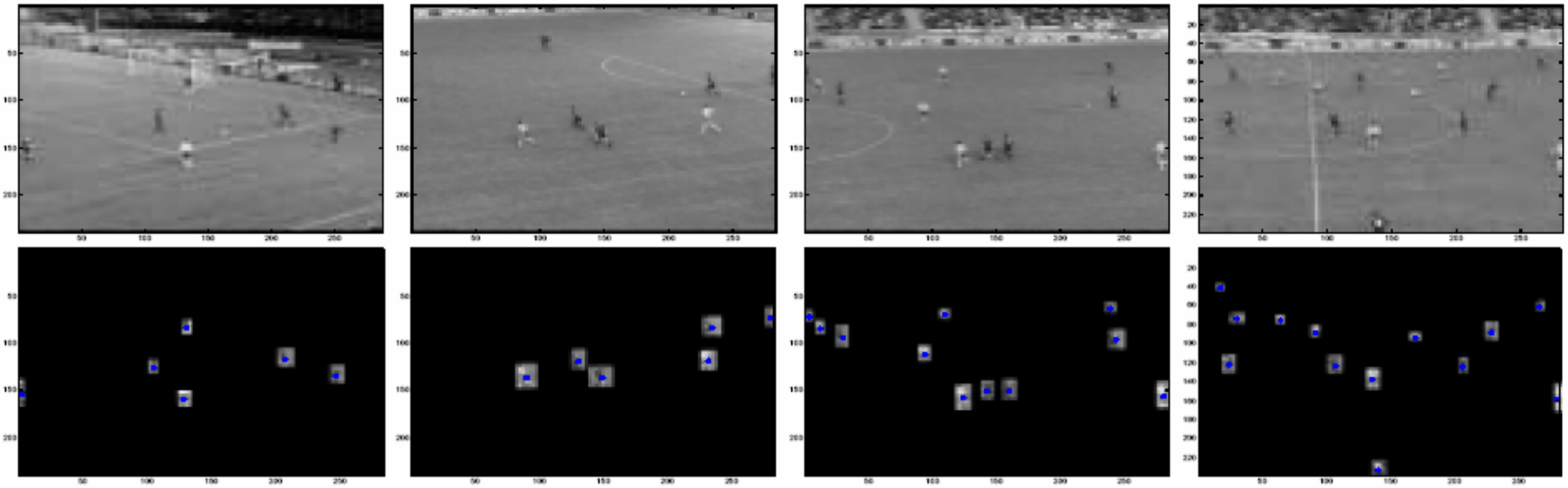
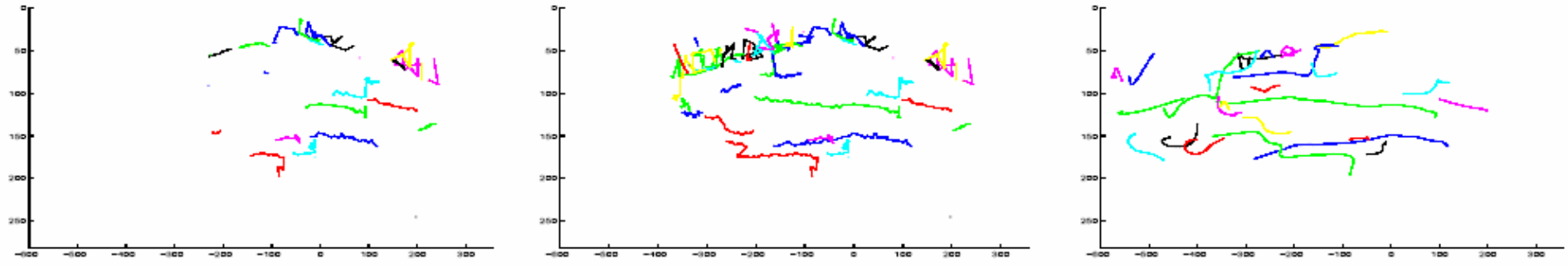


Figure 6: Motion blob detection in sample frames of the soccer sequence



(a) (b) (c)

Trajectories extracted from (a) frames 1 to 100 and (b) frames 1 to 200 in the soccer sequence (c) Final trajectory extracted after refinement from 300 frames

```
<?xml version="1.0" encoding="utf-8"?>
<!--Mpeg-7 Motion Trajectory Descriptor for "Soccer" sequence-->
<MPEG7 xmlns="http://www.mpeg7.org/2001/MPEG-7_Schema" xmlns:xsi="http://www.w3.org/2000/10/XMLSchema-instance">
   <MediaLocator>
        <MediaUri>soccer.mpg</MediaUri>
   </MediaLocator>
   <DescriptionUnit xsi:type="DescriptorCollectionType">
        <Descriptor xsi:type="MotionTrajectoryType" CameraFollows=1>
         <CoordRef ref="ID1" spatialRef="1"/>
           <Params>
                <Dimension>2</Dimension>
                <NumOfKeyPoints> 87</NumOfKeyPoints>
                <KeyTimePointList>
                        <MediaRelIncrTimePoint mediaTimeUnit="PT1N25F"
                                mediaTimeBase="../MediaLocator[1]">3</MediaRelIncrTimePoint>
                        ......
                        <MediaRelIncrTimePoint mediaTimeUnit="PT1N25F"
                                mediaTimeBase="../MediaLocator[1]">243</MediaRelIncrTimePoint>
                </KeyTimePointList>
                <KeyValueList>
                        <KeyValue><x> 91 </x><y> 127 </y></KeyValue>
                        ......
                        <KeyValue><x> -360 </x><y> 110 </y></KeyValue>
                </KeyValueList>
                <InterpolationFunc>firstOrder </InterpolationFunc>
        </Params>
       </Descriptor>

       <Descriptor xsi:type="MotionTrajectoryType" CameraFollows=1>
                    ...
       </Descriptor>
          ......
       <Descriptor xsi:type="MotionTrajectoryType" CameraFollows=1>
                    ...
       </Descriptor>
   </DescriptionUnit>
</MPEG7>
```

Parametric Motion

- Characterizes the evolution of regions over time
- Uses 2D geometric transforms
- Example:
  - Rotation/Scaling:
    - $D_x(\mathbf{x},\mathbf{y}) = a + b\mathbf{x} + c\mathbf{y}$
    - $D_y(\mathbf{x},\mathbf{y}) = d - c\mathbf{x} + b\mathbf{y}$

Motion Activity

- Need to capture “pace” or Intensity of activity
  - “High Action” chase scenes segments
  - “Low Action” talking heads segments
- Use Gross Motion Characteristics
  - avoiding object segmentation, tracking etc.

INTENSITY

- Expresses “pace” or Intensity of Action
- Uses scale of very low - low - medium - high - very high
- Extracted by suitably quantizing variance of motion vector magnitude

SPATIAL DISTRIBUTION

- Captures the size and number of moving regions in the shot on a frame by frame basis

- Enables distinction between shots with one large region in the middle (e.g.,talking heads) and shots with multiple small moving regions (e.g.,aerial soccer shots)

TEMPORAL DISTRIBUTION

- Expresses fraction of the duration of each level of activity in the total duration of the shot
- Straightforward extension of the intensity of motion activity to the temporal dimension
- A talking head, typically exclusively low activity, would have zero entries for all levels except one

DIRECTION

- Expresses dominant direction if definable as one of a set of eight equally spaced directions
- Extracted by using averages of angle (direction) of each motion vector
- Useful where there is strong directional motion

MPEG-7 Audio

- Specifies set of standardized Ds and DSs
- Addresses four classes of audio
  - Pure music, Pure speech, Pure sound effects, Arbitrary soundtracks
- May address audio features
  - Silence, Spoken content, Timbre Sound effects, Melody, etc
- Often requires other low-level Descriptor categories
  - Scalable Series – **ScalableSeries, SeriesofScalarType, etc**
  - Audio Description Framework – **AudioSampledType, AudioWaveformEnvelopeType**

MPEG-7 Audio

- Comprises 5 technologies:
  - Audio description framework (17 low-level descriptors)
  - High-Level Audio Description Tools (Ds & DSs)
    - Instrumental timbre description tools
    - Sound recognition tools
    - Spoken content description tools
    - Melody description tools (facilitate query-by-humming)

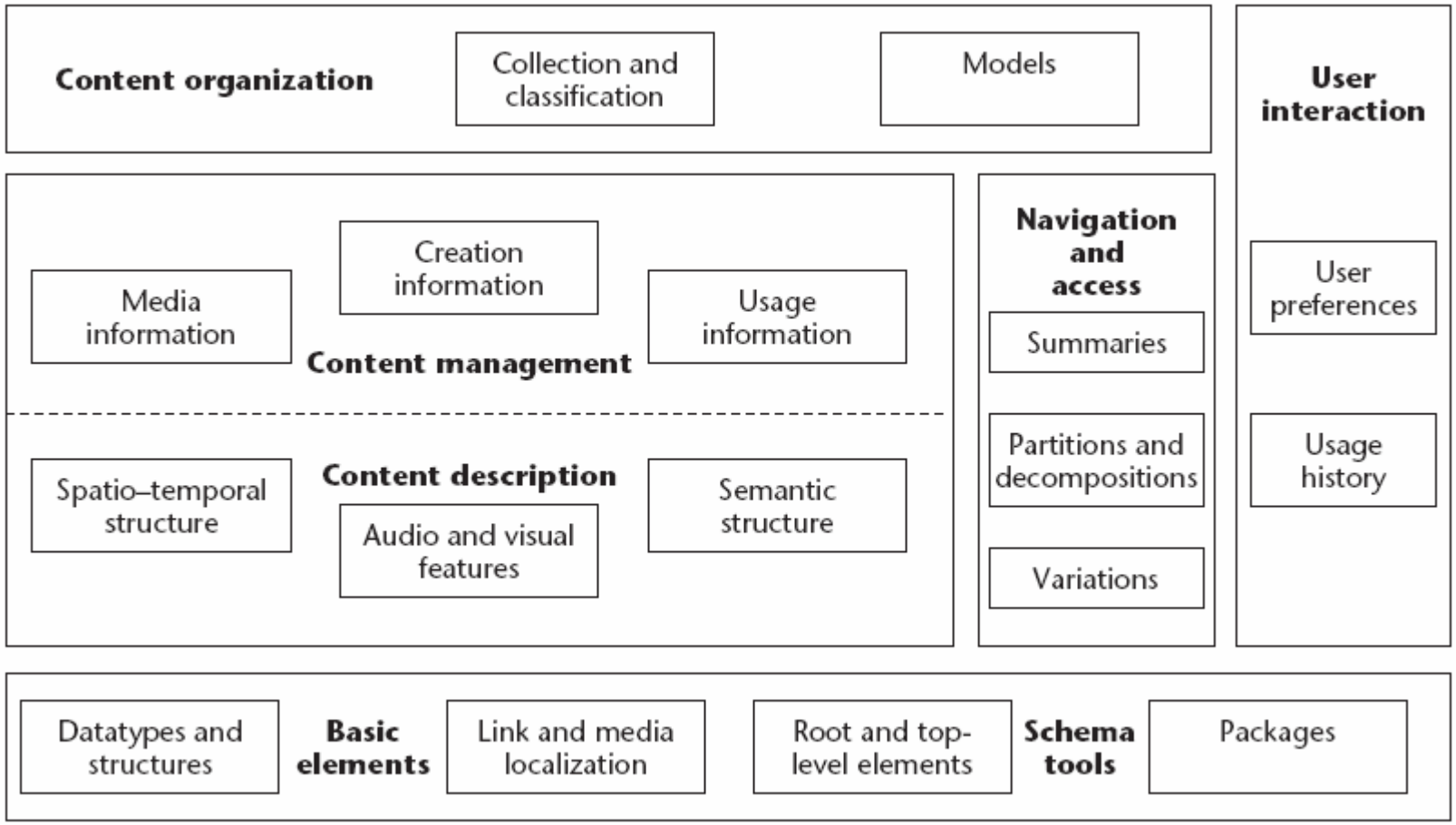
MPEG-7 Audio Standardized Descriptors

- Silence
  - **SilenceType**
- Spoken content **(from speech recognition)**
  - **SpokenContentSpeakerType**
- Timbre **(perceptual features of instrument sounds)**
  - **InstrumentTimbreType, HarmonicInstrumentTimbreType, PercussiveInstrumentTimbreType**
- Sound effects
  - **AudioSpectrumBasisType, SoundEffectFeatureType**
- Melody Contour
  - **CountourType, MeterType, BeatType**
- Description Schemes utilizing these Descriptors are also defined

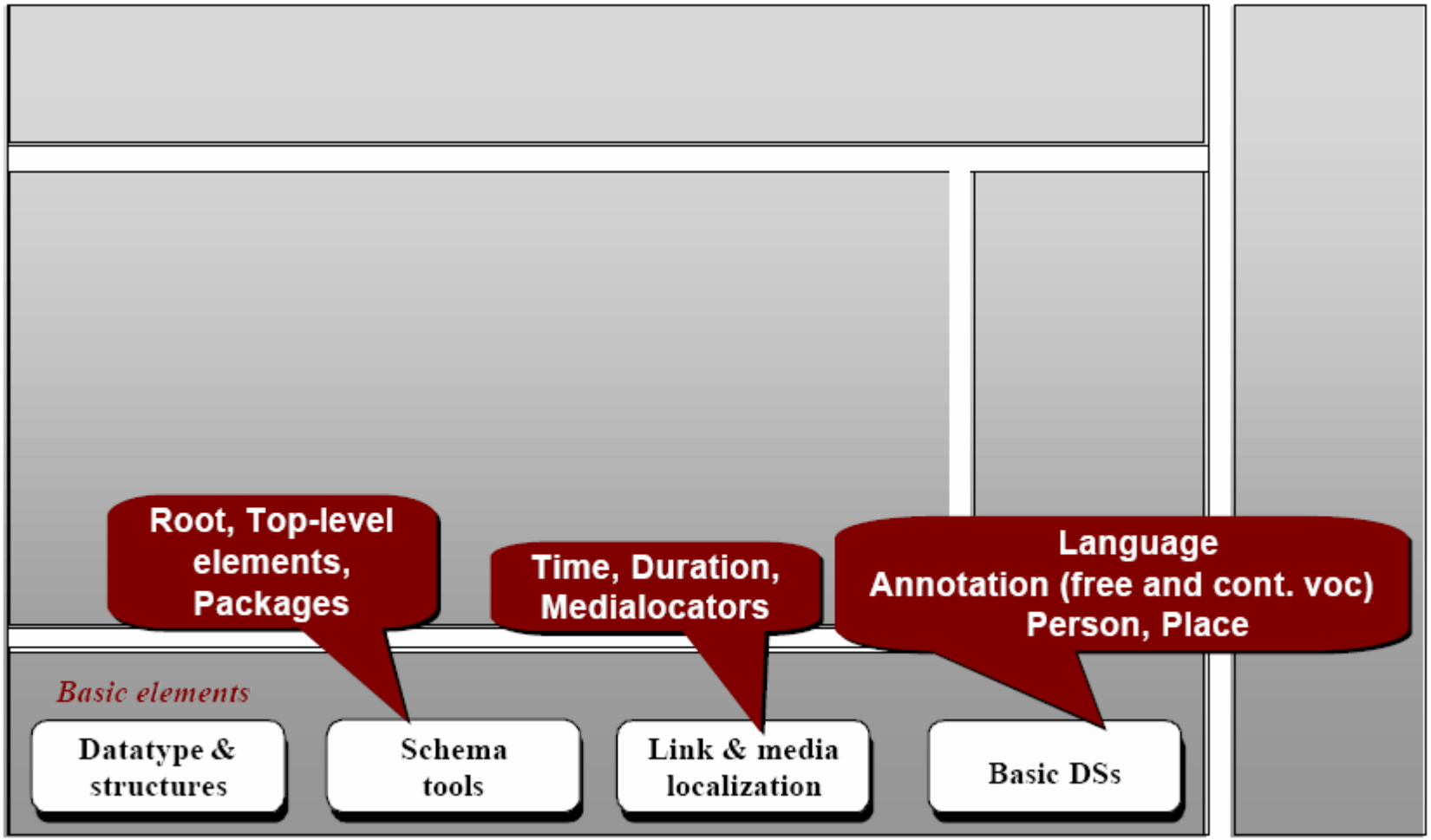
MPEG-7 Multimedia Description Schemes (MDS)

- Specifies high-level framework for generic descriptions of all kinds of multimedia
- Contrasts with specific descriptions addressed by Visual and Audio, (parts 3 and 4)
- Levels
  - Basic elements
  - Content management & Content description
    - Creation and production viewpoint
    - Media
    - Usage
    - Structural Aspects,
    - Conceptual Aspects

MDS Hierarchy: Levels and Relationships



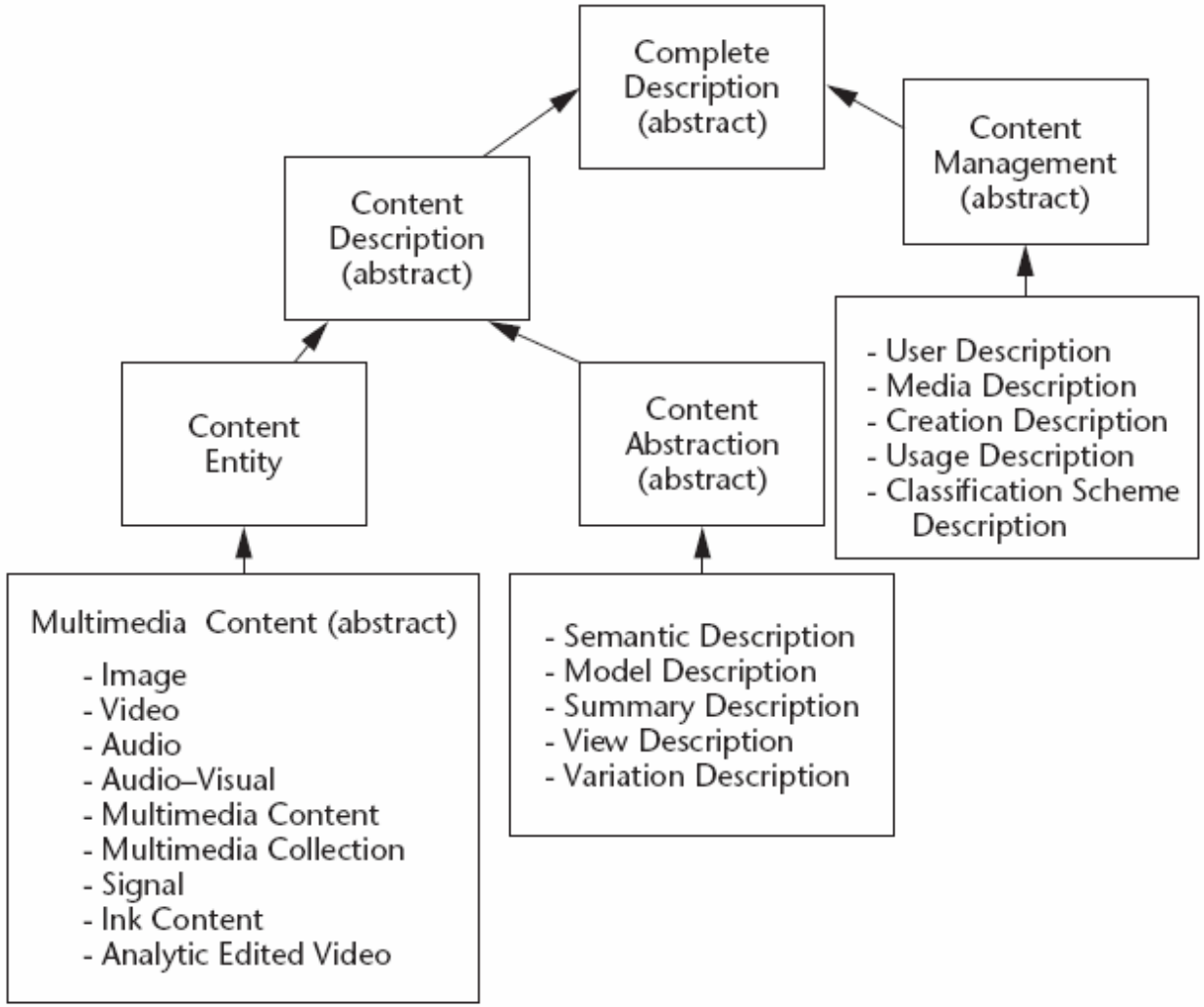
Basic elements



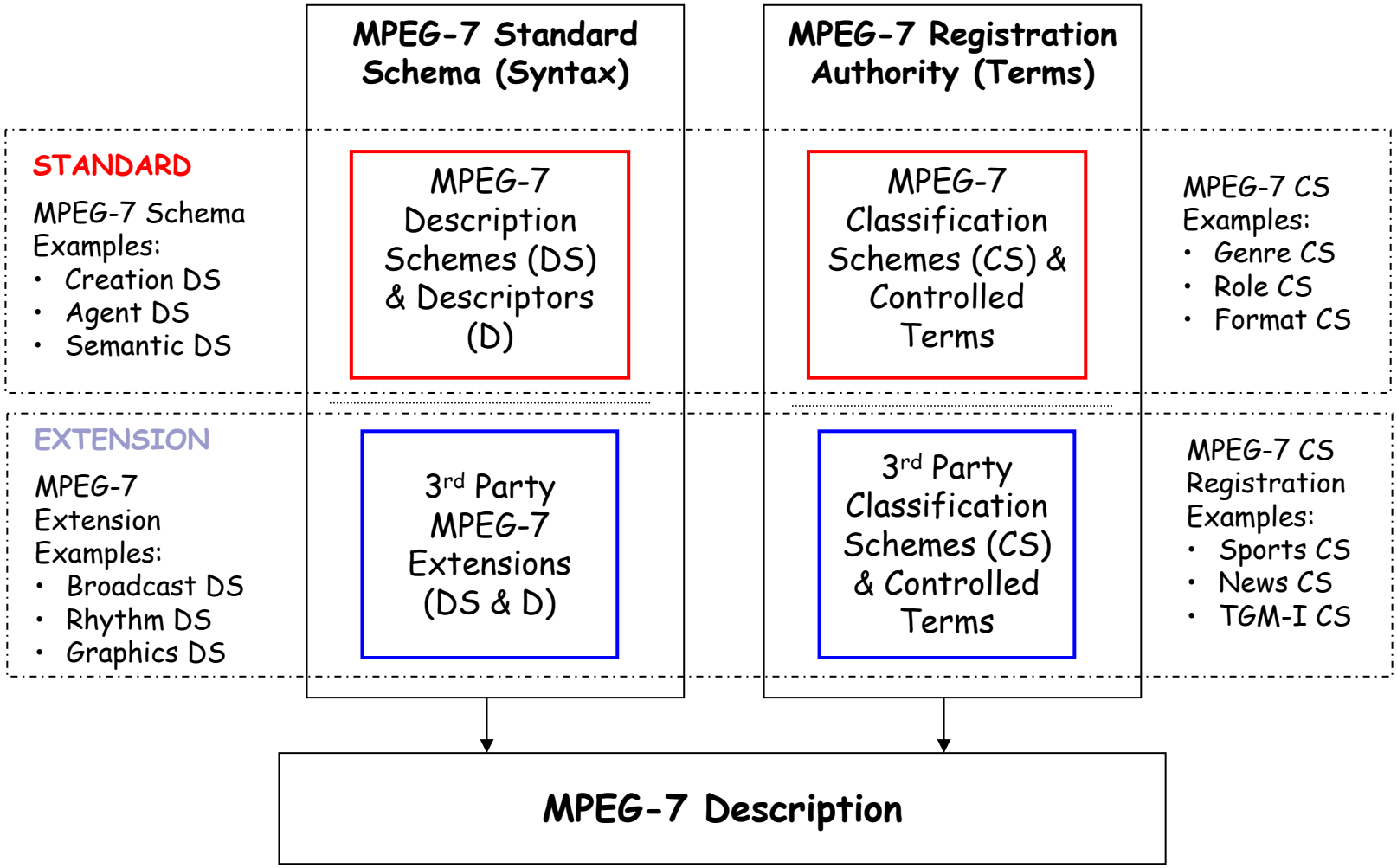
Basic elements

- Time units and temporal information (ISO 8601)
- Textual annotation
  - Free
  - Structured: Who ? What object ? What action ? Where ? When ? Why ? How ?
  - Classification schemes and controlled terms
    - Form a vocabulary for a particular application or domain

Type hierarchy for top level elements



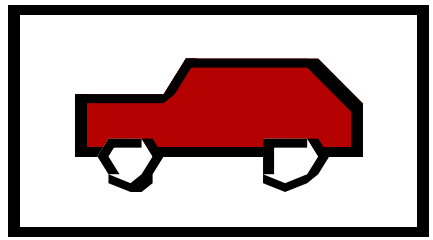
MPEG-7 Classification Schemes



MPEG-7 MDS: Free Text Annotation Example

- The following example gives an MPEG-7 description of a car that is depicted in an image:

```
<Mpeg7>
  <Description xsi:type="SemanticDescriptionType">
    <Semantics>
      <Label>
        <Name> Car </Name>
      </Label>
      <Definition>
        <FreeTextAnnotation>
          Four wheel motorized vehicle
        </FreeTextAnnotation>
      </Definition>
      <MediaOccurrence>
        <MediaLocator>
          <MediaUri> image.jpg </MediaUri>
        </MediaLocator>
      </MediaOccurrence>
    </Semantics>
  </Description>
</Mpeg7>
```



Root Element is the starting point

- The **root element** shall be used as the **topmost** element in an MPEG-7 description.
- First Choice to do
  - Complete Description: describes multimedia content using the top-level types. For example, the description of an image is a complete description.
  - Description Unit: describes an instance of a D, DS, or header. A description unit can be used to represent partial information from a complete description. For example, the description of a shape or color is a description unit.

Definition of the Root MPEG-7 Element

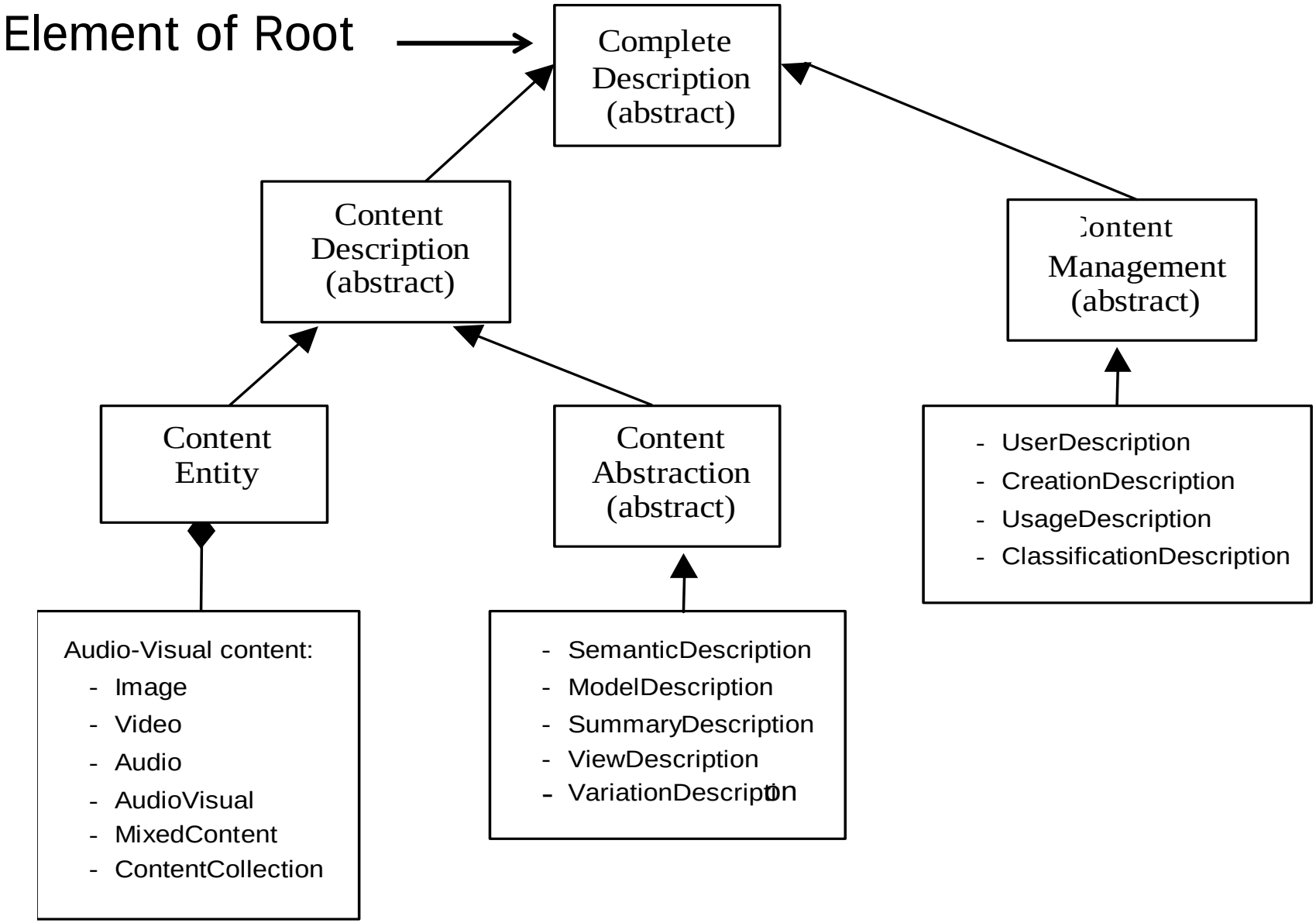
```
<!-- Definition of Mpeg7 Element -->
<element name="Mpeg7">
   <complexType>
      <complexContent>
         <extension base="mpeg7:Mpeg7Type">
            <choice>
               <element name="DescriptionUnit"
                        type="mpeg7:Mpeg7BaseType"/>
               <element name="Description"
                        type="mpeg7:CompleteDescriptionType"
                     minOccurs="1" maxOccurs="unbounded"/>
            </choice>
         </extension>
      </complexContent>
   </complexType>
</element>
```

Example of a Description Unit

Description Unit
with ambiguous meaning
What video is indexed?

```
<Mpeg7>
	<DescriptionMetadata>
		<Version>1.0</Version>
		<PrivateIdentifier>descriptionUnitExample</PrivateIdentifier>
	</DescriptionMetadata>
	<DescriptionUnit xsi:type="ScalableColorType" numOfCoeff="16"
		numOfBitplanesDiscarded="0">
		<Coeff> 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 </Coeff>
	</DescriptionUnit>
</Mpeg7>
```

The root element glues together

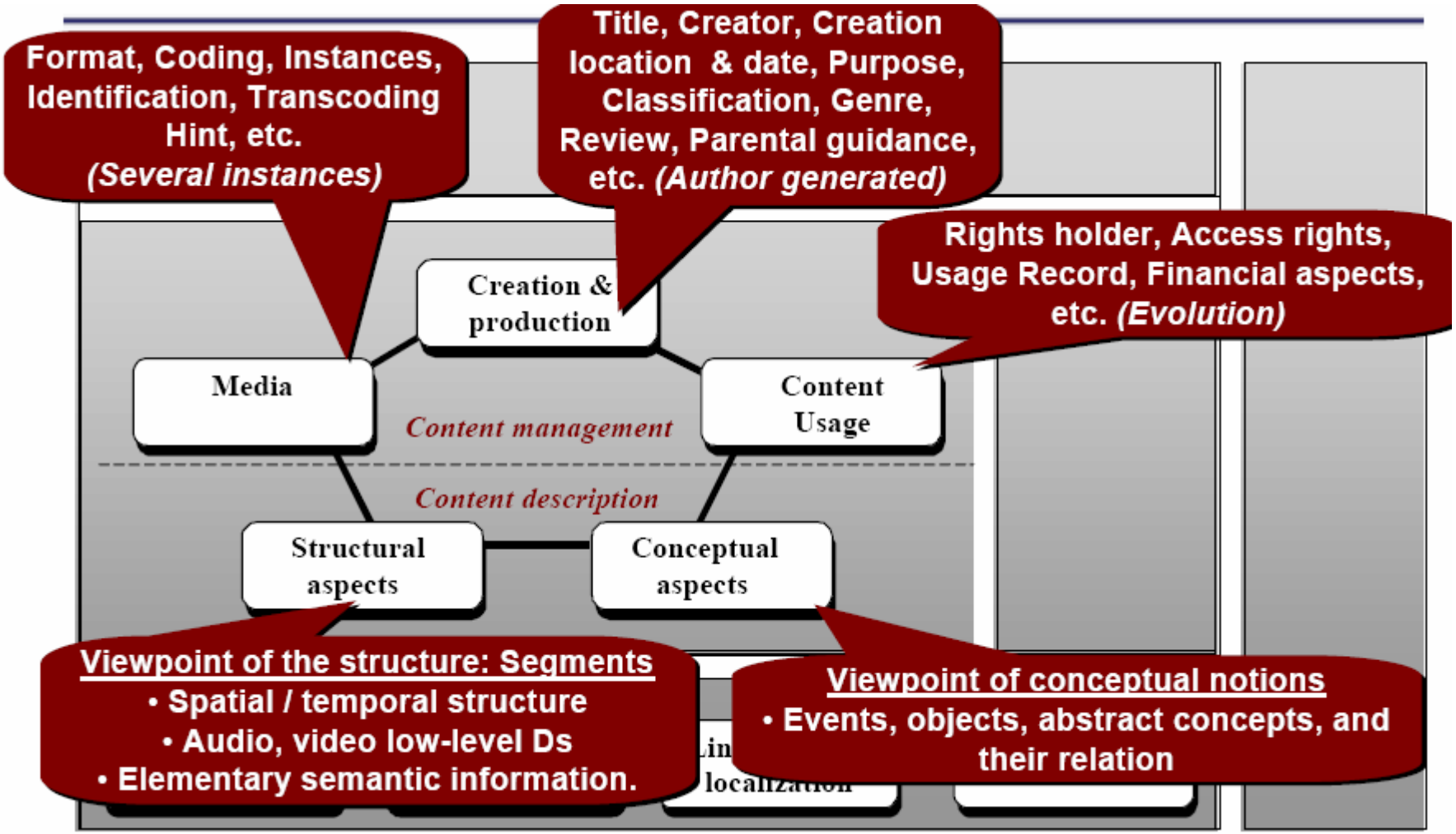


Complete Description: element of root

`CompleteDescriptionType` (abstract): top-level type for complete descriptions.

```
<!-- Definition of CompleteDescription Top-level Type -->
<complexType name="CompleteDescriptionType" abstract="true">
   <sequence>
      <element name="DescriptionMetadata"
         type="mpeg7:DescriptionMetadataType" minOccurs="0"/>
      <element name="Relationships" type="mpeg7:GraphType"
         minOccurs="0" maxOccurs="unbounded"/>
       <element name="OrderingKey" type="mpeg7:OrderingKeyType"
         minOccurs="0" maxOccurs="unbounded"/>
   </sequence>
</complexType>
```

Content management & description



ContentEntity: Giving structure

- `ContentEntityType`: top-level type for describing multimedia content entities (and their structure) such as images, videos, audio, collections, and so forth.

```
<!-- Definition of ContentEntity Top-level Type -->
<complexType name="ContentEntityType">
   <complexContent>
      <extension base="mpeg7:ContentDescriptionType">
         <sequence>
            <element name="MultimediaContent"
                  type="mpeg7:MultimediaContentType"
               minOccurs="1" maxOccurs="unbounded"/>
         </sequence>
      </extension>
   </complexContent>
</complexType>
```

MultimediaContent(Type) may be an:

```
<!-- Definition of Image Content Entity -->
<complexType name="ImageType">
   <complexContent>
      <extension base="mpeg7:MultimediaContentType">
         <sequence>
            <element name="Image" type="mpeg7:StillRegionType"/>
         </sequence>
      </extension>
   </complexContent>
</complexType>

<!-- Definition of Video Content Entity -->
<complexType name="VideoType">
   <complexContent>
      <extension base="mpeg7:MultimediaContentType">
         <sequence>
            <element name="Video" type="mpeg7:VideoSegmentType"/>
         </sequence>
      </extension>
   </complexContent>
</complexType>

<!-- Definition of Audio Content Entity -->
<complexType name="AudioType">
   <complexContent>
      <extension base="mpeg7:MultimediaContentType">
         <sequence>
            <element name="Audio" type="mpeg7:AudioSegmentType"/>
         </sequence>
      </extension>
   </complexContent>
</complexType>
```

MultimediaContent(Type) Example

```
<Mpeg7>
   <Description xsi:type="ContentEntityType">
      <MultimediaContent xsi:type="VideoType">
         <Video>
            <CreationInformation>
               <Creation>
                  <Title> Worldcup Soccer </Title>
               </Creation>
            </CreationInformation>
            <MediaTime>
               <MediaTimePoint>T00:00:00</MediaTimePoint>
               <MediaDuration>PT1M30S</MediaDuration>
            </MediaTime>
            <VisualDescriptor xsi:type="GoFGoPColorType"
                        aggregation="Average">
               <ScalableColor numOfCoeff="16"
                        numOfBitplanesDiscarded="0">
                  <Coeff> 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 </Coeff>
               </ScalableColor>
            </VisualDescriptor>
         </Video>
      </MultimediaContent>
   </Description>
</Mpeg7>
```

MultimediaContent may be mixed, decomposed

Mixed Content

Decomposition of Mixed Content

VideoSegment has not the Type Video, but VideoSegment

```
<Mpeg7>
   <Description xsi:type="ContentEntityType">
      <MultimediaContent xsi:type="MultimediaType">
         <Multimedia>
            <MediaSourceDecomposition gap="false" overlap="false">
               <Segment xsi:type="StillRegionType">
                  <TextAnnotation>
                     <FreeTextAnnotation> image </FreeTextAnnotation>
                  </TextAnnotation>
               </Segment>
               <Segment xsi:type="VideoSegmentType">
                  <TextAnnotation>
                     <FreeTextAnnotation> video </FreeTextAnnotation>
                  </TextAnnotation>
                  <MediaTime>
                     <MediaTimePoint>T00:00:00</MediaTimePoint>
                     <MediaDuration>PT0M15S</MediaDuration>
                  </MediaTime>
               </Segment>
               <Segment xsi:type="AudioSegmentType">
                  <TextAnnotation>
                     <FreeTextAnnotation> audio </FreeTextAnnotation>
                  </TextAnnotation>
               </Segment>
            </MediaSourceDecomposition>
         </Multimedia>
      </MultimediaContent>
   </Description>
</Mpeg7>
```

MultimediaContent is structure descriptive

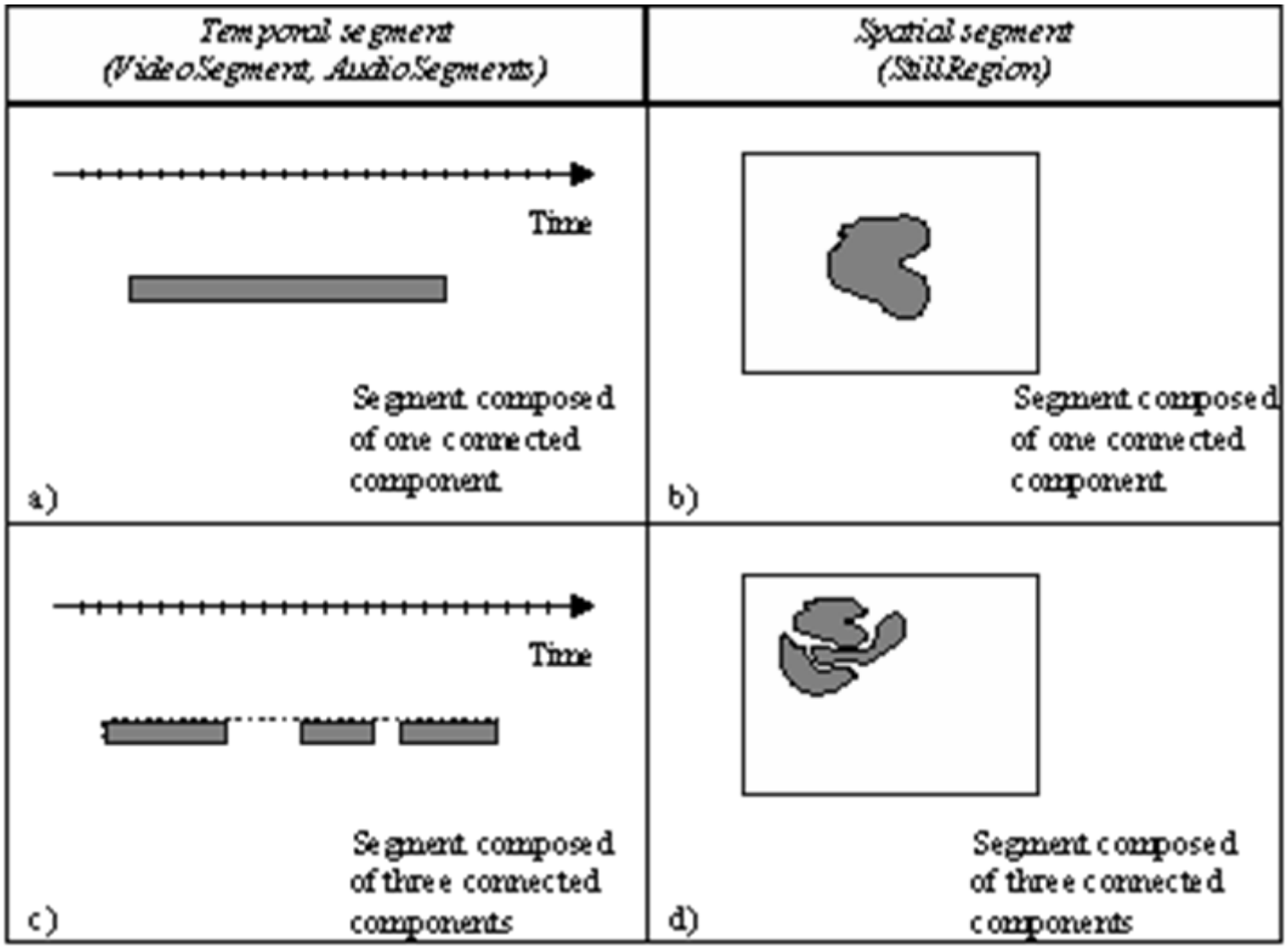
Structure description tools:

- Describes the structure of the multimedia content
- The structural features include:
  - Spatial,
  - Temporal, or
  - Spatio-temporal segments of the multimedia
  - content,
  - Audio-visual features,
  - Segmentation.
- The root element is SegmentType

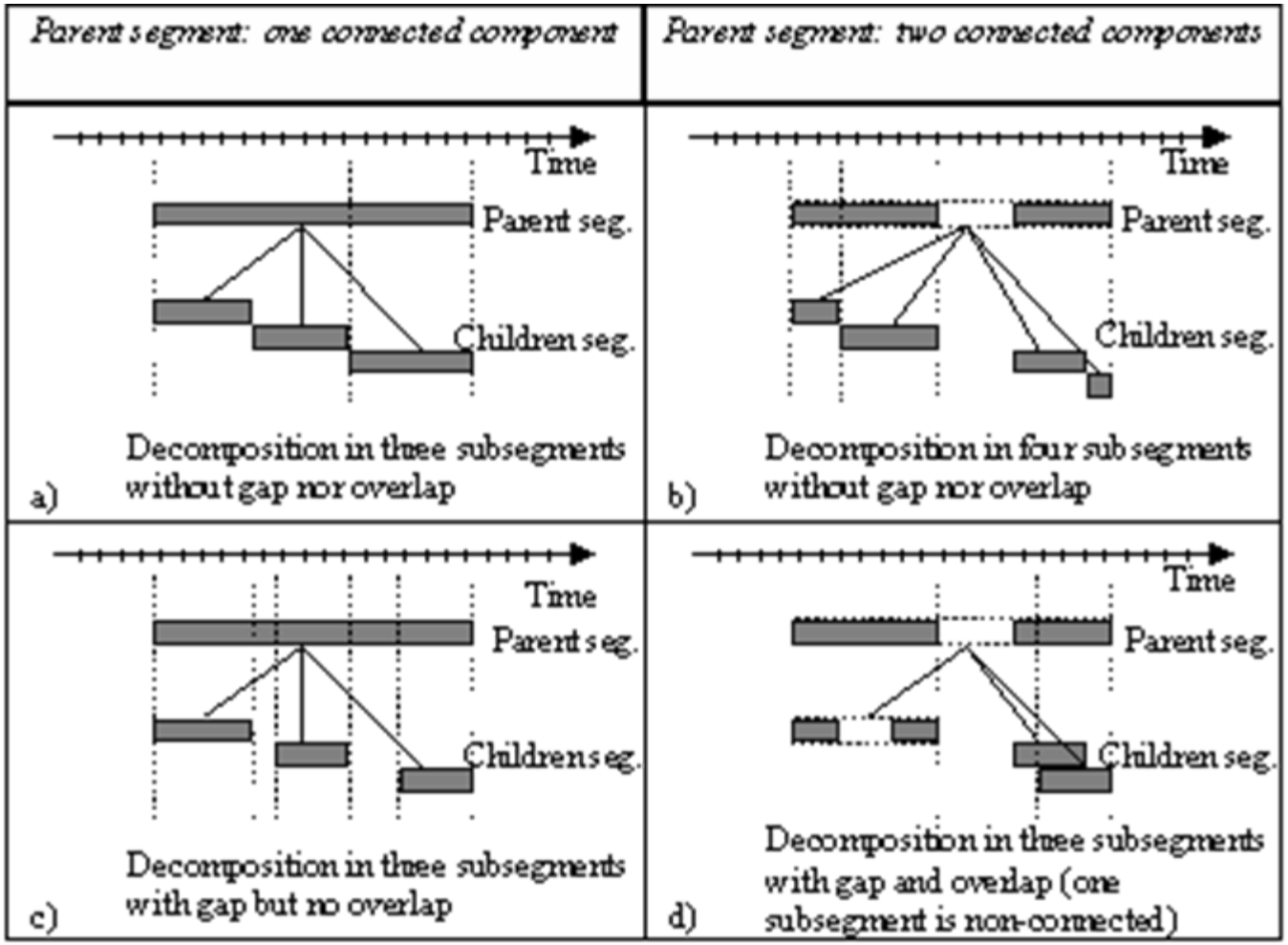
Segment DS

- Abstract type of audio, video, AV, moving regions and still regions segments
  - Has spatio-temporal properties

Feature	Video Segment	Still Region	Moving Region	Audio Segment
Time (MDS)	x		x	x
Shape (Visual)		x	x	
Color (Visual)	x	x	x	
Texture (Visual)		x		
Motion (Visual)	x		x	
Camera motion (Visual)	x			
Audio features (Audio)			x	x

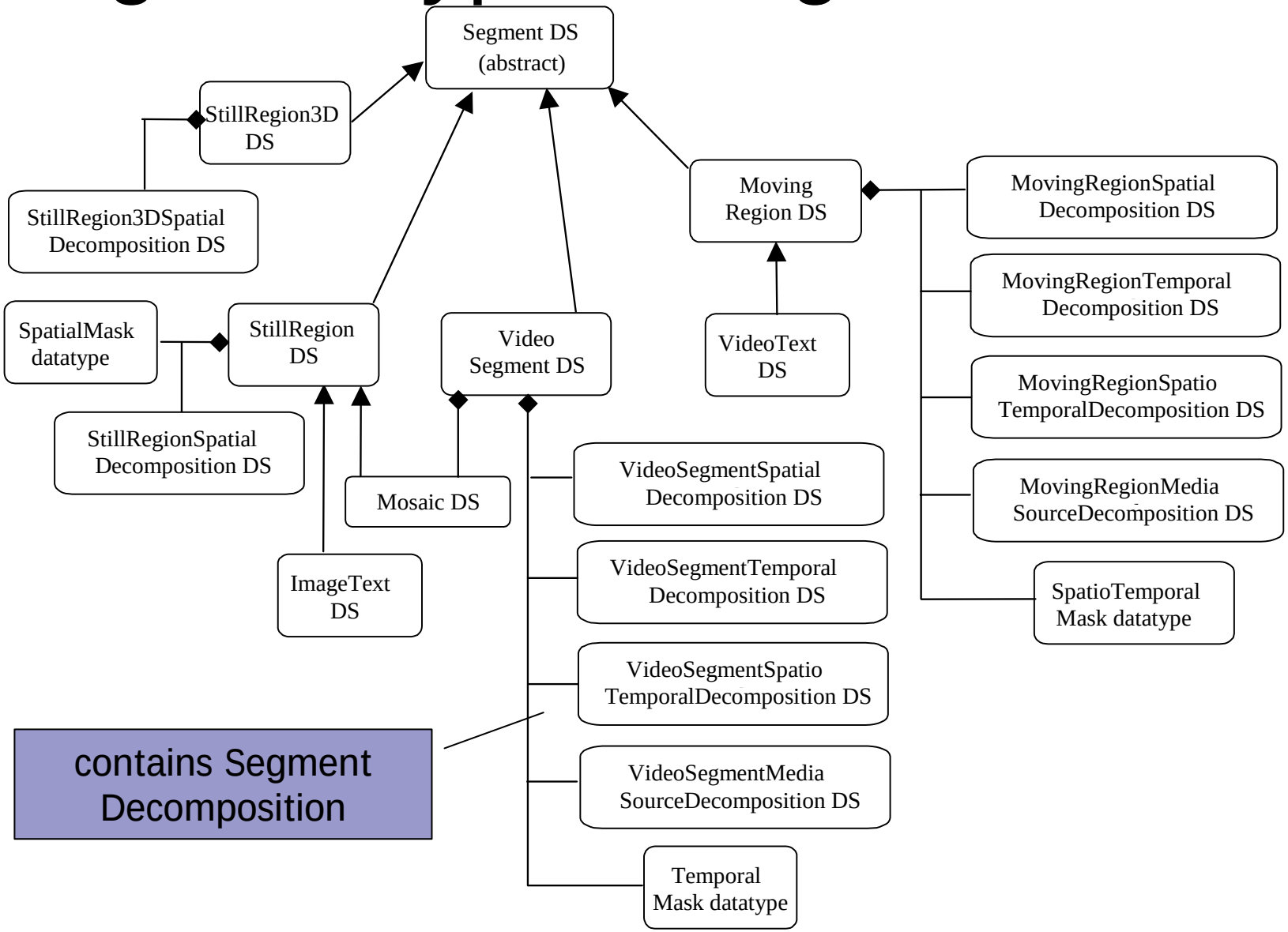


- Examples of segments: a) and b) segments composed of one single connected component; c) and d) segments composed of three connected components



- Examples of Segment Decomposition: a) and b) Segment Decompositions without gap nor overlap; c) and d) Segment Decompositions with gap or overlap

SegmentType = Segment DS



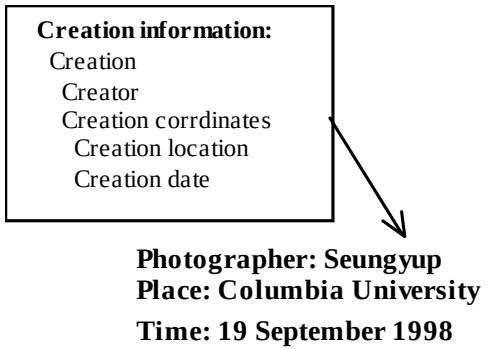
Segment Type

We shall not repeat other definitions, thus use *Ref*

Relation to Semantics (later)

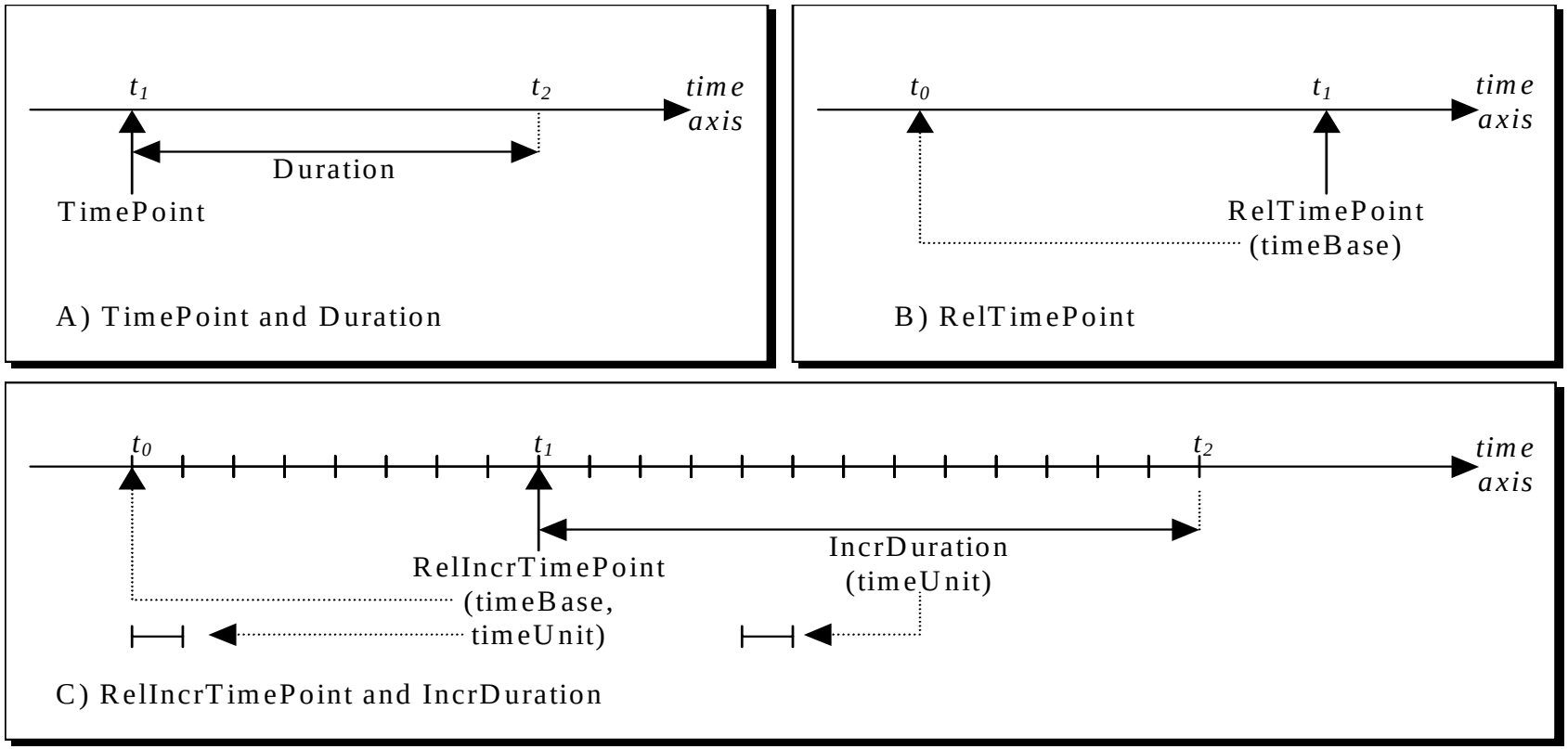
```
<complexType name="SegmentType" abstract="true">
   <complexContent>
      <extension base="mpeg7:DSType">
         <sequence>
            <choice minOccurs="0">
               <element name="MediaInformation"
type="mpeg7:MediaInformationType"/>
               <element name="MediaInformationRef"
type="mpeg7:ReferenceType"/>
               <element name="MediaLocator"
type="mpeg7:MediaLocatorType"/>
            </choice>
            …
            <choice minOccurs="0">
               <element name="CreationInformation"
                  type="mpeg7:CreationInformationType"/>
               <element name="CreationInformationRef"
type="mpeg7:ReferenceType"/>
            </choice>
            <choice minOccurs="0">
               <element name="UsageInformation"
type="mpeg7:UsageInformationType"/>
               <element name="UsageInformationRef"
type="mpeg7:ReferenceType"/>
            </choice>
            <element name="TextAnnotation" minOccurs="0"
maxOccurs="unbounded">
             …
            </element>
            <choice minOccurs="0" maxOccurs="unbounded">
               <element name="Semantic" type="mpeg7:SemanticType"/>
               <element name="SemanticRef" type="mpeg7:ReferenceType"/>
            </choice>
            …
         </sequence>
      </extension>
   </complexContent>
</complexType>
```

Segment needs Creation,Usage and other info



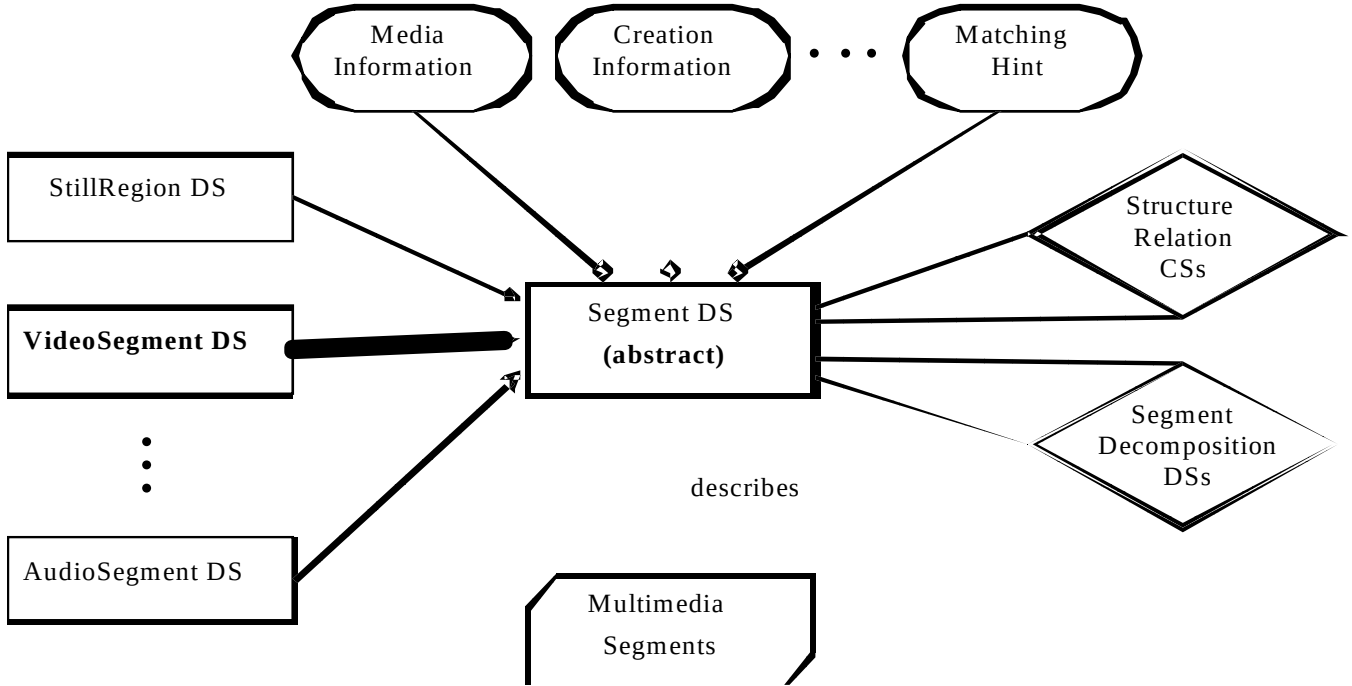

```
<CreationInformation>
 <Creation>
  <Creator>
   <Role><Name>Photographer</Name></Role>
   <Person> <Name> <GivenName>Seungyup</GivenName> </Name> </Person>
  </Creator>
  <CreationCoordinates>
   <CreationLocation> <Name xml:lang="en">Columbia University</Name>
    </CreationLocation>
   <CreationDate> <TimePoint>1998-09-19</TimePoint> </CreationDate>
  </CreationCoordinates>
 </Creation>
</CreationInformation>
<MediaInformation>
 <MediaProfile master="true">
  <MediaFormat> <Content>image</Content>
   <VisualCoding>
    <Format colorDomain="color“
    href="urn:mpeg:VisualCodingFormatCS:1">JPG</Format>
    <Frame height="480" width="704"/>
   </VisualCoding>
   </MediaFormat>
  <MediaInstance id="mastercopy">
   <MediaLocator> <MediaUri> http://www.alex&ana.jpg </MediaUri>
    </MediaLocator>
  </MediaInstance>
 </MediaProfile>
</MediaInformation>
<UsageInformation>
 <Rights>
  <RightsId organization="Columbia University“> columbia:1919:alex&ana_image
    </RightsId>
 </Rights>
</UsageInformation>
```

VideoSegment needs Timing Information

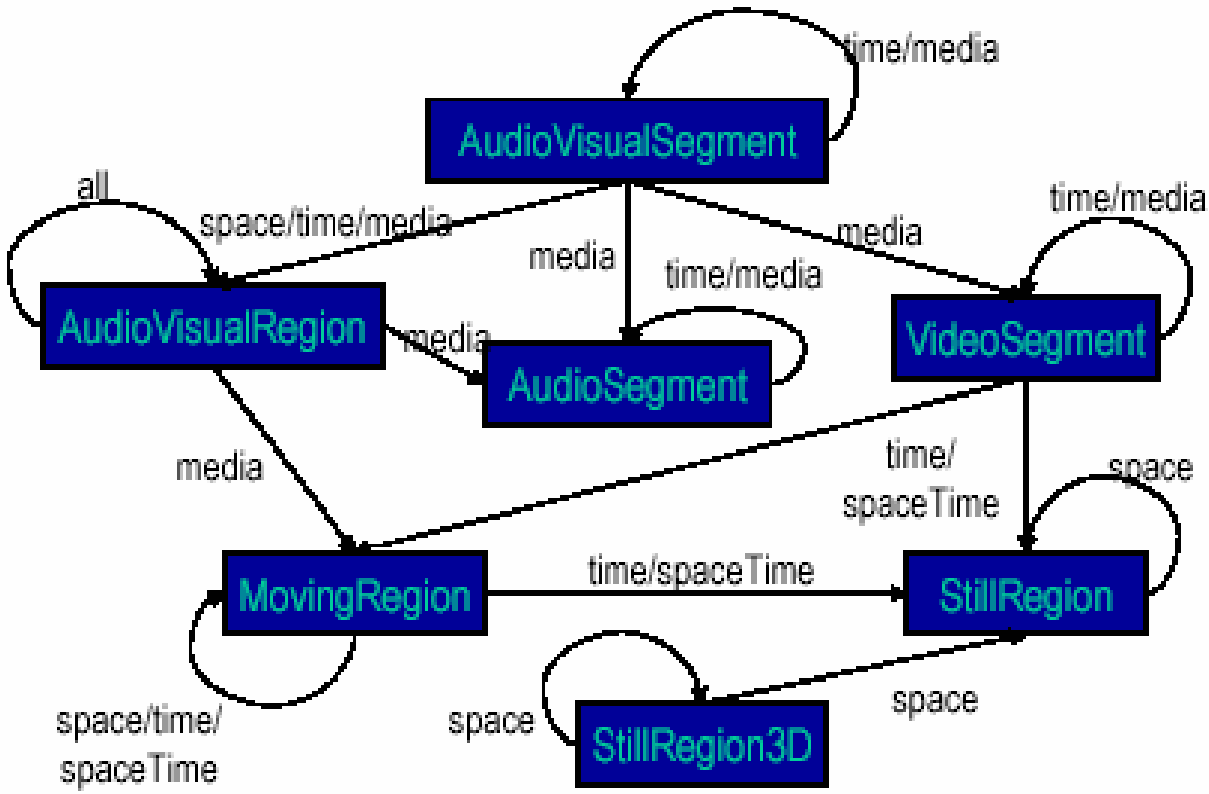


Relation to MPEG-7 Root

- Relation to MultimediaContentType and the MPEG-7 Root:
- VideoSegment, StillRegion,etc. is subtype of SegmentType.
- Videosegment is an element in VideoType.
- VideoType is a subtype MultimediaContentType which is subtype of CompleteDescriptionType, which is elment of MPEG-7 ROOT

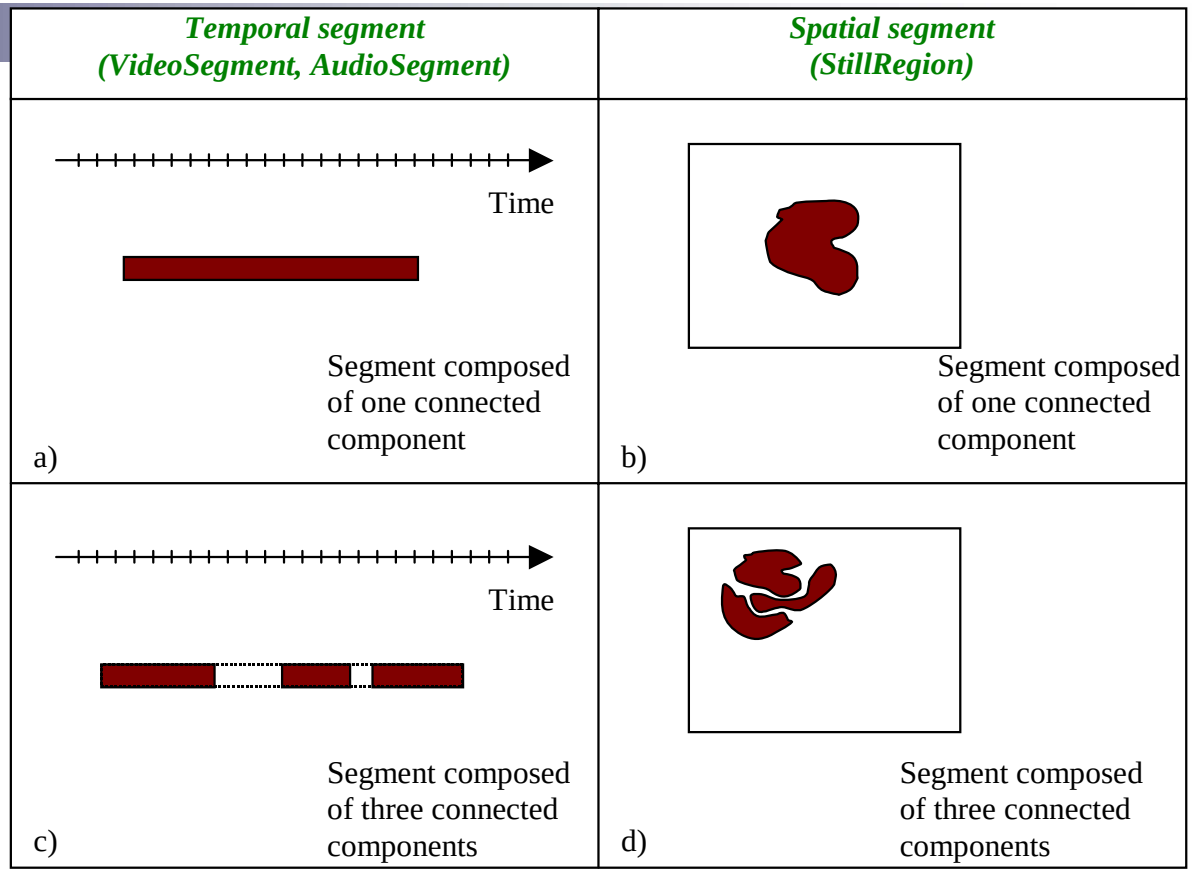


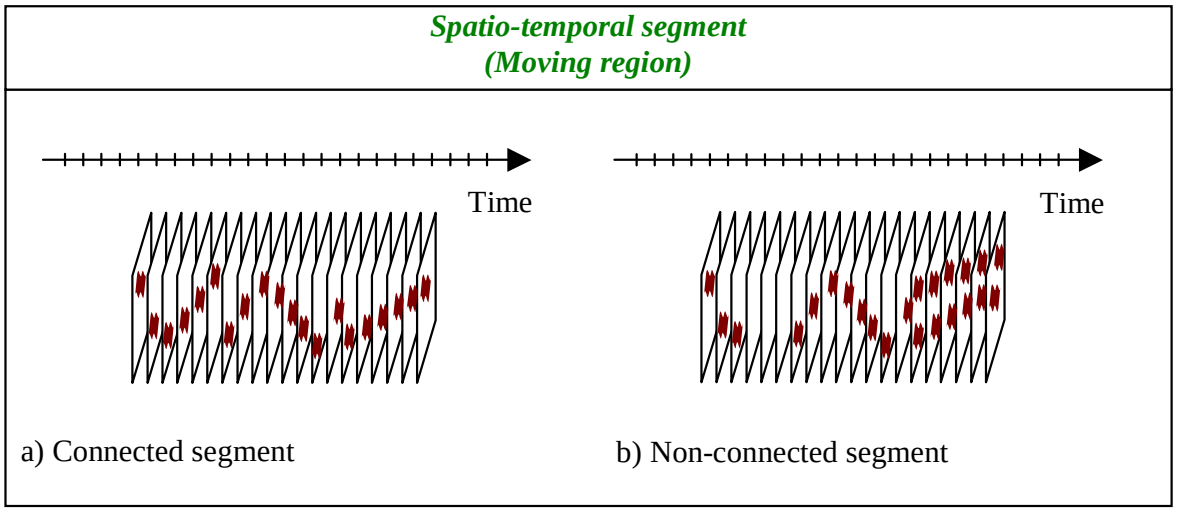
Segment Decomposition



- **Spatial** (image decomposition into regions)
- **Temporal** (video decomposition into segments)
- **Spatio-temporal** (video decomposition into moving regions)
- **Media** (video decomposition into audio and video tracks)

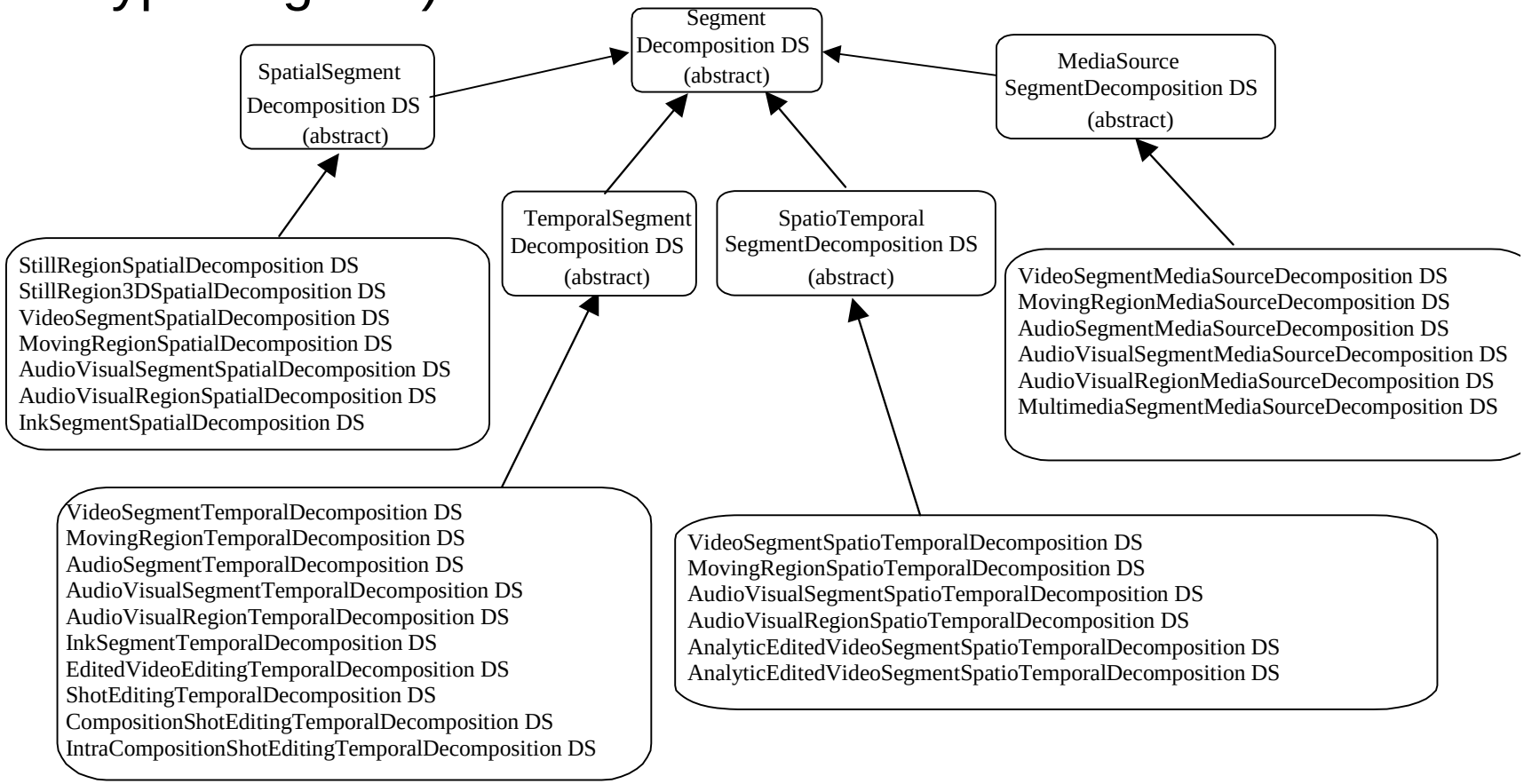
Illustration of Segment Decomposition





SegmentDecompositionType

- Relationship between the segments is translated by the Segment DecompostionType (abstract needs to be subtyped again !)

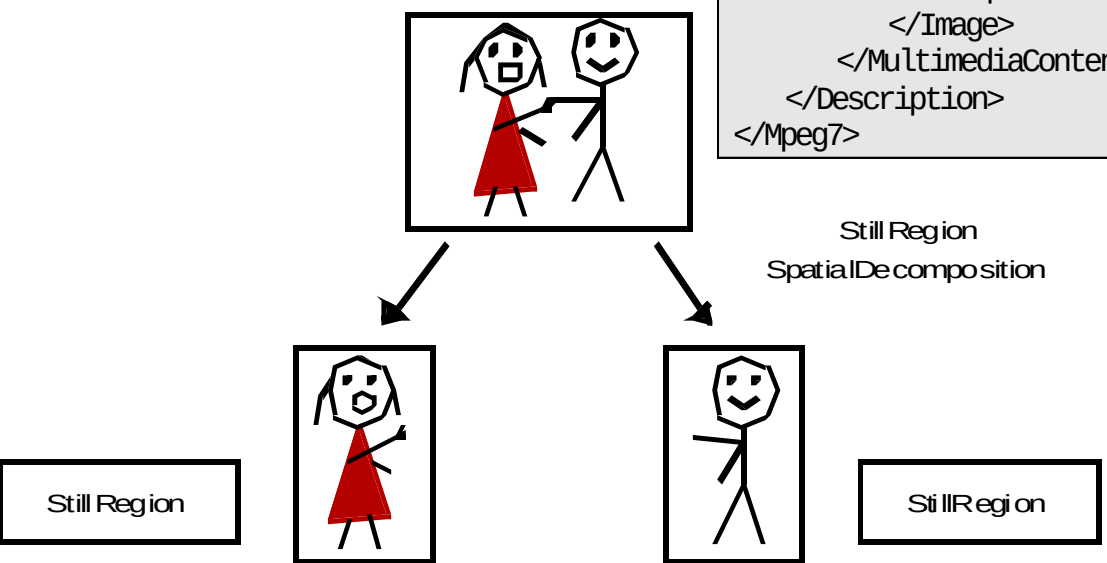


Examples for Segments and their SegmentDecompositions : Image/Video/MovingRegion

Image Descriptions (1)

- Content management metadata:
  - Media information
  - Creation & production;
  - usage
- Visual features:
  - Visual descriptors
- Structure:
  - Regions, spatial decomp.
  - spatio-temporal relations

```
<Mpeg7>
   <Description xsi:type="ContentEntityType">
      <MultimediaContent xsi:type="ImageType">
         <Image>
            <SpatialDecomposition gap="true" overlap="false">
               <StillRegion id="AlexSR">
                  <VisualDescriptor xsi:type="ScalableColorType"
numOfCoeff="16"
                     numOfBitplanesDiscarded="0">
                     <Coeff> 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 </Coeff>
                  </VisualDescriptor>
               </StillRegion>
               <StillRegion id="AnaSR">
   <VisualDescriptor xsi:type="ScalableColorType" numOfCoeff="16"
                     numOfBitplanesDiscarded="0">
                     <Coeff> 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 </Coeff>
                  </VisualDescriptor>
               </StillRegion>
            </SpatialDecomposition>
         </Image>
      </MultimediaContent>
   </Description>
</Mpeg7>
```

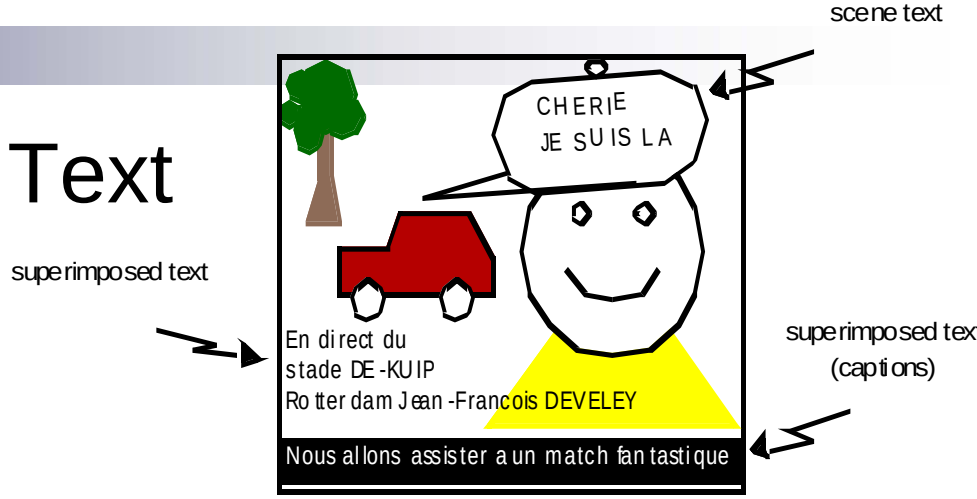


ImageText DS


```
<!-- ################################################ -->
<!-- Definition of ImageText DS                       -->
<!-- ################################################ -->
<complexType name="ImageTextType">
                              <complexContent>
    <extension base="mpeg7:StillRegionType">
       <sequence>
         <element name="Text" type="mpeg7:TextualType" minOccurs="0"/>
       </sequence>
       <attribute name="textType" use="optional">
         <simpleType>
           <restriction base="string">
             <enumeration value="superimposed"/>
             <enumeration value="scene"/>
           </restriction>
         </simpleType>
                              </attribute>
       <attribute name="fontSize" type="positiveInteger" use="optional"/>
       <attribute name="fontType" type="string" use="optional"/>
    </extension>
  </complexContent>
</complexType>
```

Image Descriptions with Text

A StillRegion may contain also Text Regions




```
<Mpeg7>
   <Description xsi:type="ContentEntityType">
      <MultimediaContent xsi:type="ImageType">
         <Image>
            <SpatialDecomposition gap="true" overlap="false">
               <StillRegion xsi:type="ImageTextType" textType="scene">
                  <SpatialLocator>
                     <!-- more elements here -->
                  </SpatialLocator>
                  <Text xml:lang="fr"> Cherie, je suis la </Text>
               </StillRegion>
               <StillRegion xsi:type="ImageTextType"
textType="superimposed"
                  fontSize="40" fontType="Arial">
                  <SpatialLocator>
                     <!-- more elements here -->
                  </SpatialLocator>
                  <Text xml:lang="fr">
               En direct du state DE-KUIP Rotterdam Jean-Francois DEVELEY
</Text>
               </StillRegion>
               <!-- more Still Regions here here -->
            </SpatialDecomposition>
         </Image>
      </MultimediaContent>
   </Description>
</Mpeg7>
```

StillRegionType

Moving Region ?

Subimage ?

Relation to Visual Part

```
<complexType name="StillRegionType">
   <complexContent>
      <extension base="mpeg7:SegmentType">
         <sequence>
            <choice minOccurs="0">
               <element name="SpatialLocator"
                  type="mpeg7:RegionLocatorType"/>
               <element name="SpatialMask" type="mpeg7:SpatialMaskType"/>
            </choice>
            <choice minOccurs="0">
               <element name="MediaTimePoint"
                  type="mpeg7:mediaTimePointType"/>
               <element name="MediaRelTimePoint"
                        type="mpeg7:MediaRelTimePointType"/>
               <element name="MediaRelIncrTimePoint"
                        type="mpeg7:MediaRelIncrTimePointType"/>
            </choice>
            <choice minOccurs="0" maxOccurs="unbounded">
               <element name="VisualDescriptor"
                       type="mpeg7:VisualDType"/>
               <element name="VisualDescriptionScheme"
                            type="mpeg7:VisualDSType"/>
               <element name="GridLayoutDescriptors"
                           type="mpeg7:GridLayoutType"/>
            </choice>
              …
            <element name="SpatialDecomposition"
                     type="mpeg7:StillRegionSpatialDecompositionType"
                     minOccurs="0" maxOccurs="unbounded"/>
         </sequence>
      </extension>
   </complexContent>
</complexType>
```

How to describe the spatial properties ?

- Use of a SpatialMask in StillRegionType (element):

```
<choice minOccurs="0">
   <element name="SpatialLocator" type="mpeg7:RegionLocatorType"/>
   <element name="SpatialMask" type="mpeg7:SpatialMaskType"/>
</choice>
<choice minOccurs="0">
   <element name="MediaTimePoint" type="mpeg7:mediaTimePointType"/>
   <element name="MediaRelTimePoint"
         type="mpeg7:MediaRelTimePointType"/>
   <element name="MediaRelIncrTimePoint"
         type="mpeg7:MediaRelIncrTimePointType"/>
</choice>
```

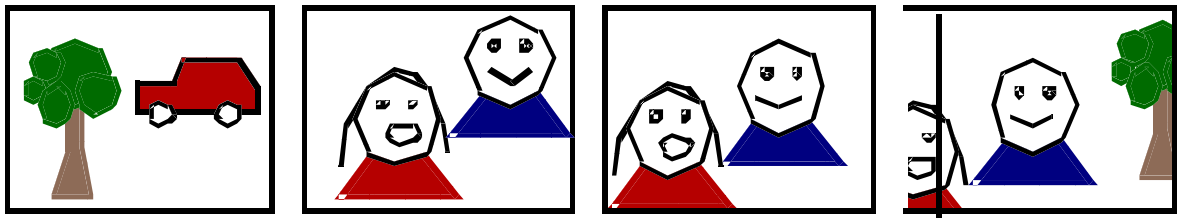
- Example for 2D:

```
<SpatialMask>
   <SubRegion>
      <Polygon>
         <Coords mpeg7:dim="2 5"> 5 25 10 20 15 15 10 10 5 15 </Coords>
      </Polygon>
   </SubRegion>
   <SubRegion>
      <Polygon>
         <Coords mpeg7:dim="2 6"> 7 24 15 24 20 27 18 25 15 22 7 22 </Coords>
      </Polygon>
   </SubRegion>
   <SubRegion>
      <Polygon>
         <Coords mpeg7:dim="2 4"> 7 30 15 30 15 25 7 25</Coords>
      </Polygon>
   </SubRegion>
</SpatialMask>
```

Video Descriptions (1): Example

Example attributes:

- CreationInformation
- MediaInformation
- UsageInformation
- Semantics

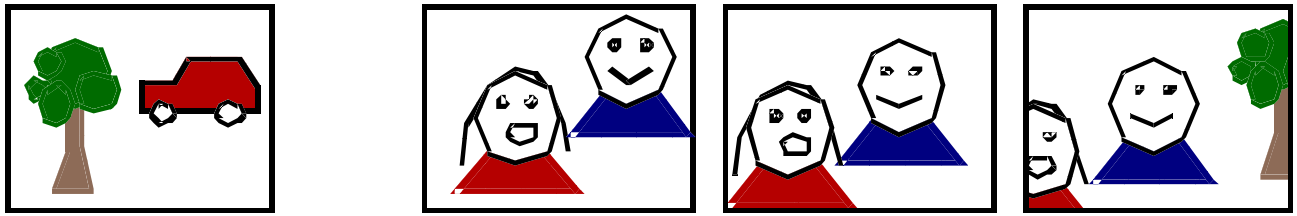


VideoSegment

VideoSegment
TemporalDecomposition

Example attributes:

- TextAnnotation
- MediaTime
- MotionActivity
- VisualDescriptor



VideoSegment

VideoSegment for the example

How to specify the time property of the Video Segment ?

```
<Mpeg7>
   <Description xsi:type="ContentEntityType">
      <MultimediaContent xsi:type="VideoType">
         <Video id="RootVS">
            <MediaTime>
               <MediaTimePoint>T00:00:00</MediaTimePoint>
               <MediaDuration>PT1M30S</MediaDuration>
            </MediaTime>
            <VisualDescriptor xsi:type="GoFGoPColorType"
aggregation="Average">
               <ScalableColor numOfCoeff="16"
numOfBitplanesDiscarded="0">
                  <Coeff> 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 </Coeff>
               </ScalableColor>
            </VisualDescriptor>
            <TemporalDecomposition gap="false" overlap="false">
               <VideoSegment id="ChaseVS">
                  <TextAnnotation>
                     <FreeTextAnnotation> Chase </FreeTextAnnotation>
                  </TextAnnotation>
                  <MediaTime>
                     <MediaTimePoint>T00:00:00</MediaTimePoint>
                     <MediaDuration>PT0M15S</MediaDuration>
                  </MediaTime>
                  <VisualDescriptor xsi:type="GoFGoPColorType"
                     aggregation="Average">
                     <ScalableColor numOfCoeff="16"
numOfBitplanesDiscarded="0">
                        <Coeff> 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 </Coeff>
                     </ScalableColor>
                  </VisualDescriptor>
               </VideoSegment>
               <VideoSegment id="CaptureVS">
                  <TextAnnotation>
                     <FreeTextAnnotation> Capture </FreeTextAnnotation>
                  </TextAnnotation>
                  <MediaTime>
                     <MediaTimePoint>T00:00:15</MediaTimePoint>
                     <MediaDuration>PT1M15S</MediaDuration>
                  </MediaTime>
                  <VisualDescriptor xsi:type="GoFGoPColorType"
                     aggregation="Average">
                     <ScalableColor numOfCoeff="16"
numOfBitplanesDiscarded="0">
                        <Coeff> 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 </Coeff>
                     </ScalableColor>
                  </VisualDescriptor>
               </VideoSegment>
            </TemporalDecomposition>
         </Video>
      </MultimediaContent>
   </Description>
</Mpeg7>
```

How to express the time property of a video ?

The following example describes the time of an event that started on the 3rd October 1987 at 14:13 in Germany and has a duration of 10 days:

```
<Time>
    <TimePoint>1987-10-03T14:13+01:00</TimePoint>
    <Duration>P10D</Duration>
</Time>
```

The following example describes the time at which a 3rd is taken picture given that one picture is taken each day:

```
<Time>
    <RelIncrTimePoint timeUnit="P1D"
timeBase="../../../Time[1]">3</RelIncrTimePoint>
</Time>
```

The following example counts in time units of one Day and refers to the start time of the event as a `timeBase` (e.g. first time point specification in the description). The period of five days after the initial event is described as follows:

```
<Time>
    <RelIncrTimePoint timeUnit="P1D"
timeBase="../../../Time[1]">0</RelIncrTimePoint>
    <IncrDuration timeUnit="P1D">5</IncrDuration>
</Time>
```

The following example describes the time of event occurring in England, two hours and 20 minutes after the initial one:

```
<Time>
    <RelTimePoint timeBase="../../../Time[1]">PT2H20M-01:00Z</RelTimePoint>
</Time>
```

MediaTimePointType is the anchor

- YYYY-MM-DDThh:mm:ss:nnnFNNN T
  - Y: Year, can be a variable number of digits,
  - M: Month, D:Day,
  - h: hour, m: minute, s: second,
  - n: number of fractions, nnn can be any number between 0 and NNN-1 (NNN and with it nnn can have an arbitrary number of digits).
  - N: number of fractions of one second which are counted by nnn. NNN can have a arbitrary number of digits and is not limited to three. Also delimiters for the time specification (T) and the number of fractions of one second are used (F).
- Example:
  - T13:20:01:235 can be expressed as T13:20:01:235F1000

MediaDurationType gives length

- A simpleType representing a duration in time using a lexical representation of days (nD), time duration and a fraction specification (TnHnMnSnN) including the specification of the number of fractions of one second (nF):(-)PnDTnHnMnSnNnF

- Example: PT16M20S means ?

The Video World is Frame-based.

- Use of the attribute MediaTimeUnit of Type MediaDurationType.
- Typical examples of 1N which is 1/30 of a second according to 30F.
- Other example: the exact sampling rate of NTSC of 29.97 Hz is also possible with: mediaTimeUnit="PT1001N30000F".
- Last example: For the representation of the exact sampling rate of 25Hz in the case of the PAL or SECAM video format the time unit shall be instatiated by mediaTimeUnit="PT1N25F".

```
<!-- MediaTime for Segment3(1-10sec) -->
<MediaTime>
      <MediaRelIncrTimePoint mediaTimeUnit="PT1N30F"

      mediaTimeBase="../../MediaLocator[1]">30</MediaRelIncrTimePoint>
      <MediaIncrDuration mediaTimeUnit="PT1N30F">270</MediaIncrDuration>
</MediaTime>
```

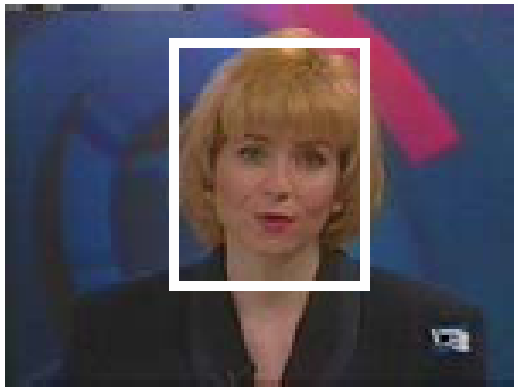
VideoSegmentType

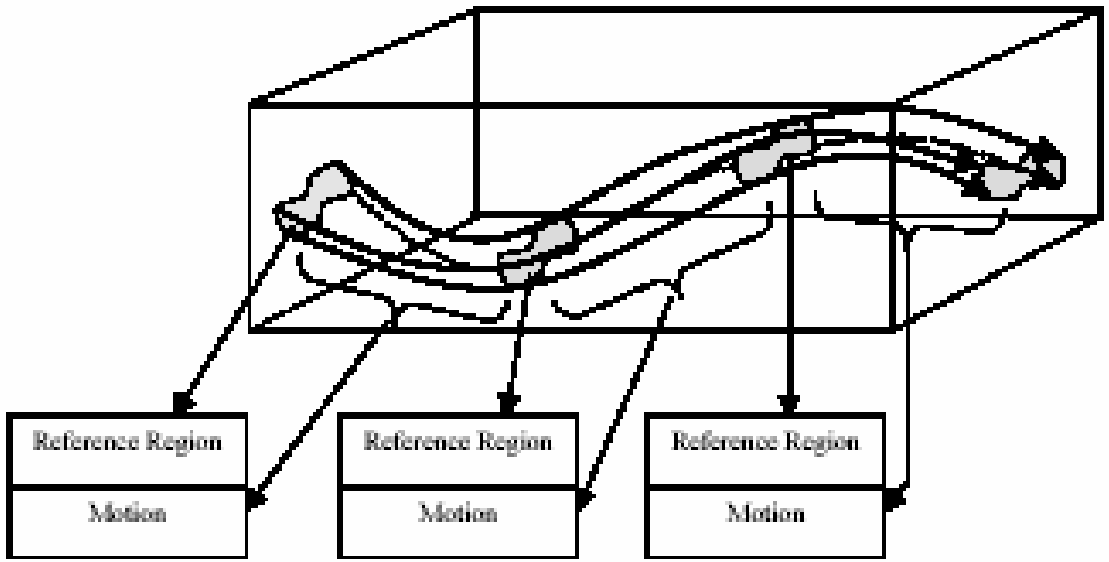
```
<complexType name="VideoSegmentType">
   <complexContent>
      <extension base="mpeg7:SegmentType">
         <sequence>
            <choice minOccurs="0">
               <element name="MediaTime" type="mpeg7:MediaTimeType"/>
               <element name="TemporalMask"
                   type="mpeg7:TemporalMaskType"/>
            </choice>
            <choice minOccurs="0" maxOccurs="unbounded">
               <element name="VisualDescriptor" type="mpeg7:VisualDType"/>
               <element name="VisualDescriptionScheme"
                   type="mpeg7:VisualDSType"/>
               <element name="VisualTimeSeriesDescriptor"
                  type="mpeg7:VisualTimeSeriesType"/>
            </choice>
            <element name="MultipleView" type="mpeg7:MultipleViewType"
                     minOccurs="0"/>
            <element name="Mosaic" type="mpeg7:MosaicType"
                     minOccurs="0" maxOccurs="unbounded"/>
            <choice minOccurs="0" maxOccurs="unbounded">
               <element name="SpatialDecomposition"
                    type="mpeg7:VideoSegmentSpatialDecompositionType"/>
               <element name="TemporalDecomposition"
                  type="mpeg7:VideoSegmentTemporalDecompositionType"/>
               <element name="SpatioTemporalDecomposition"
                  type="mpeg7:VideoSegmentSpatioTemporalDecompositionType"/>
               <element name="MediaSourceDecomposition"
                  type="mpeg7:VideoSegmentMediaSourceDecompositionType"/>
            </choice>
         </sequence>
      </extension>
   </complexContent>
</complexType>
```

Moving Region Description

Moving Regions:

- Region Feature description (shape, color, texture)
- Spatio-Temporal Locator
- Regions Motion Trajectory and Tracking





Moving Region Description

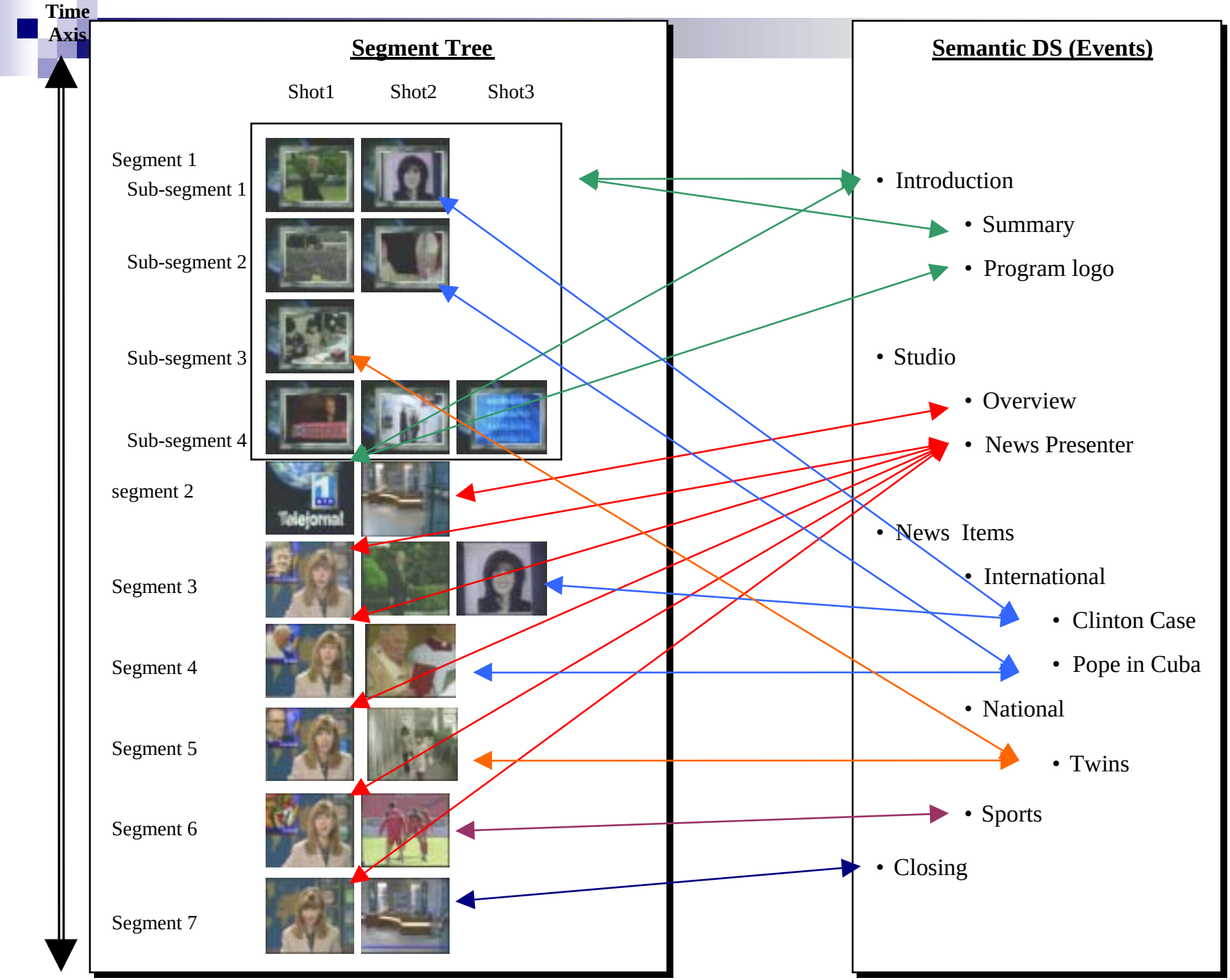
```
<Mpeg7>
   <Description xsi:type="ContentEntityType">
      <MultimediaContent xsi:type="VideoType">
         <Video id="VS1">
            <SpatioTemporalDecomposition>
               <MovingRegion id="WomanMR">
                  <TextAnnotation>
                     <FreeTextAnnotation> Woman (moving
                        region)</FreeTextAnnotation>
                  </TextAnnotation>
                  <TemporalDecomposition gap="true" overlap="false">
                     <StillRegion id="WomanKeySR">
                        <MediaLocator>
                           <MediaUri>image.jpg</MediaUri>
                        </MediaLocator>
                        <TextAnnotation>
                           <FreeTextAnnotation> Woman (still region)
                           </FreeTextAnnotation>
                        </TextAnnotation>
                        <VisualDescriptor xsi:type="ScalableColorType"
                           numOfCoeff="16" numOfBitplanesDiscarded="0">
                           <Coeff> 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6
                                 </Coeff>
                        </VisualDescriptor>
                     </StillRegion>
                  </TemporalDecomposition>
               </MovingRegion>
            </SpatioTemporalDecomposition>
         </Video>
      </MultimediaContent>
   </Description>
</Mpeg7>
```

Segment to Semantic

- Through the Semantic Relation in SegmentType

```
<choice minOccurs="0" maxOccurs="unbounded">
      <element name="Semantic" type="mpeg7:SemanticType"/>
      <element name="SemanticRef" type="mpeg7:ReferenceType"/>
</choice>
```

- Remember all other Multimedia Types are subtypes of the SegmentType
- The Semantic may be included directly or through the reference mechanism.

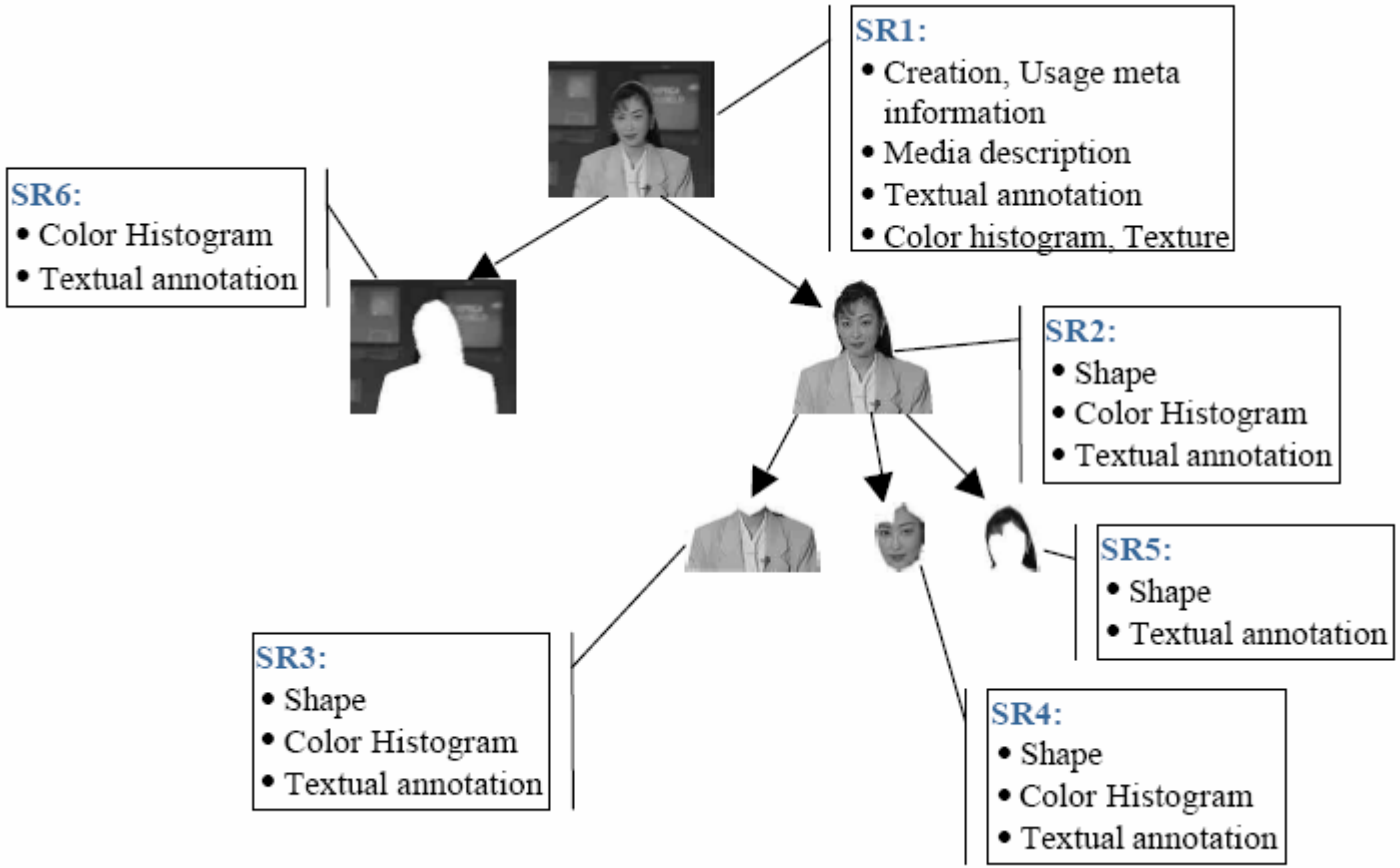

Time Axis
Segment Tree
Shot1
Shot2
Shot3
Segment 1
Sub-segment 1
Sub-segment 2
Sub-segment 3
Sub-segment 4
segment 2
Segment 3
Segment 4
Segment 5
Segment 6
Segment 7
Semantic DS (Events)
• Introduction
• Summary
• Program logo
• Studio
• Overview
• News Presenter
• News Items
• International
• Clinton Case
• Pope in Cuba
• National
• Twins
• Sports
• Closing

Segment to Semantic and much more …

```
<?xml version="1.0" encoding="iso-8859-1" ?>
- <Mpeg7 xmlns="urn:mpeg:mpeg7:schema:2001" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:mpeg7="urn:mpeg:mpeg7:schema:2001" xsi:schemaLocation="urn:mpeg:mpeg7:schema:2001
      Mpeg7-2001.xsd">
- <Description xsi:type="ContentEntityType">
- <MultimediaContent xsi:type="VideoType">
- <Video>
- <MediaInformation>
- <MediaProfile>
-<MediaFormat>
- <Content href="MPEG7ContentCS">
<Name>audiovisual</Name>
</Content>
- <FileFormat href="urn:mpeg:mpeg7:cs:FileFormatCS:2001:3">
<Name xml:lang="en">mpeg</Name>
</FileFormat>
<FileSize>60000000</FileSize>
<BitRate>128000</BitRate>
</MediaFormat>
- <MediaInstance>
```

Take care of the right namespaces

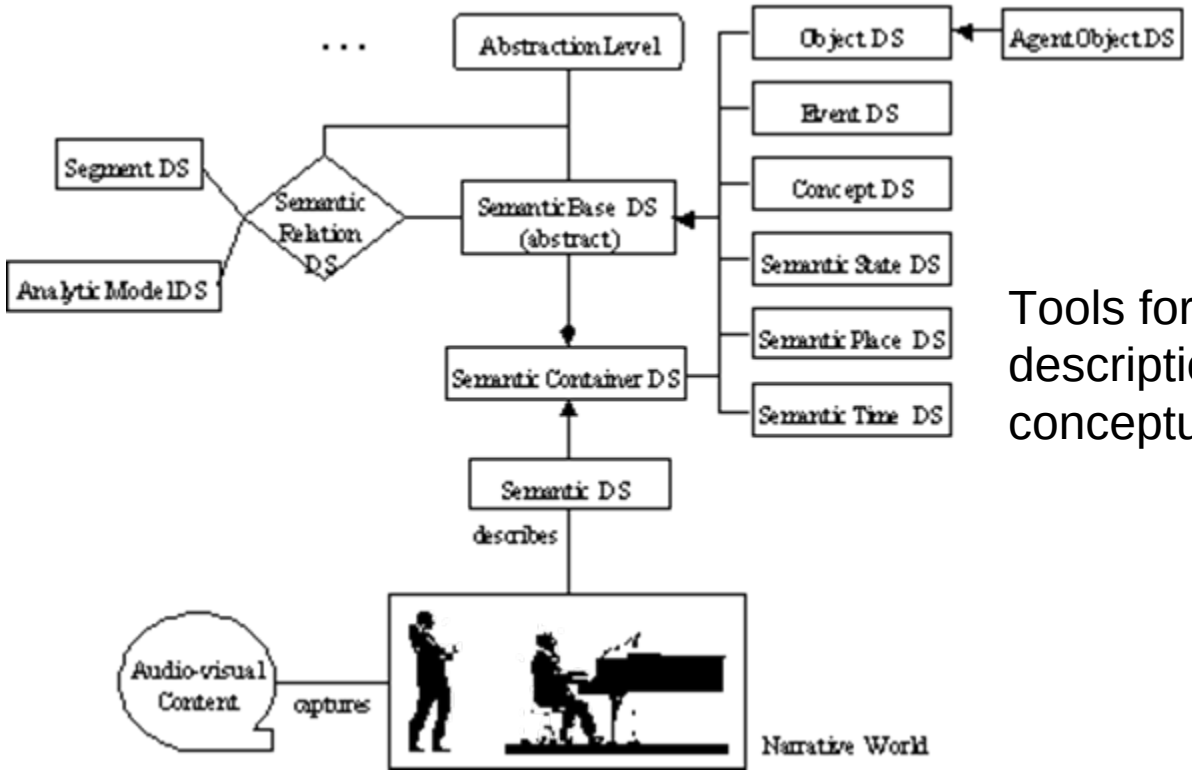
What's this ???
It is a CS-Classification Schema



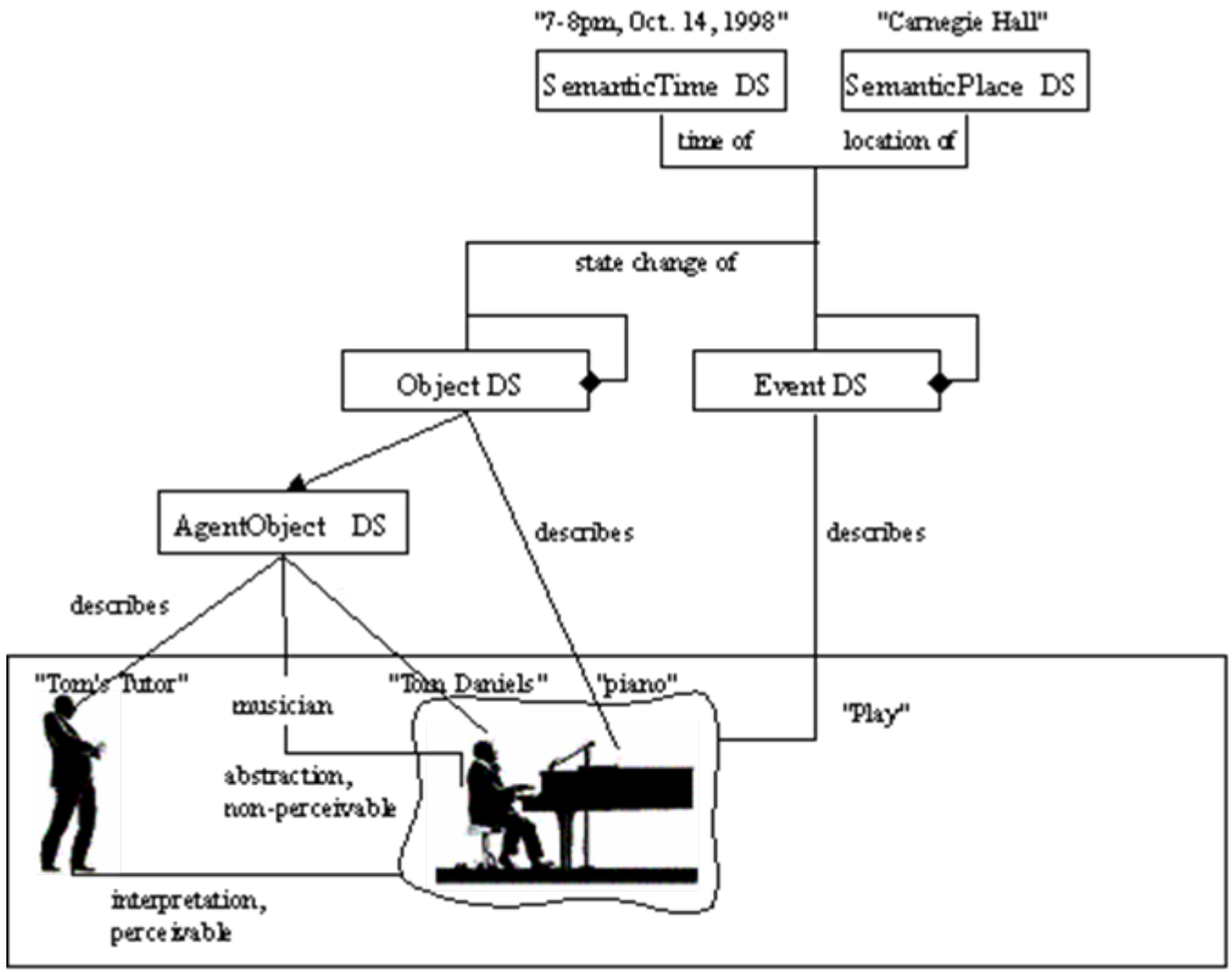
- Children segments inherit parent value (e.g. creation, usage information and media info)
- If a tree (like the above one) is not suitable for description use SegmentRelation DS
  - E.g. graph used for soccer match

Conceptual aspects

- Emphasis is on events, objects, places, time in *narrative worlds*: Semantic DS



Tools for the description of conceptual aspects



Narrative World

- Semantic DSs also allow the description of abstractions. Abstraction refers to the process of taking a description from a specific instance of audio-visual content and generalizing it to a set of multiple instances of audio-visual content or to a set of specific descriptions.
- Two types of abstraction, called media abstraction and standard abstraction, are considered.
- A media abstraction is a description that has been separated from a specific instance of audio-visual content, and can describe all instances of audio-visual content that are sufficiently similar
  - a news event, which can be applied to the description of multiple programs, that may have been broadcasted on different channels.
- A standard abstraction is the generalization of a media abstraction to describe a general class of semantic entities or descriptions.
  - In general, the standard abstraction is obtained by replacing the specific objects, events or other semantic entities by classes. For instance, if "Tom playing piano" is replaced by "a man playing piano“
  - Typically, a standard abstraction is intended for reuse, or to be used by reference in a description.

Semantics

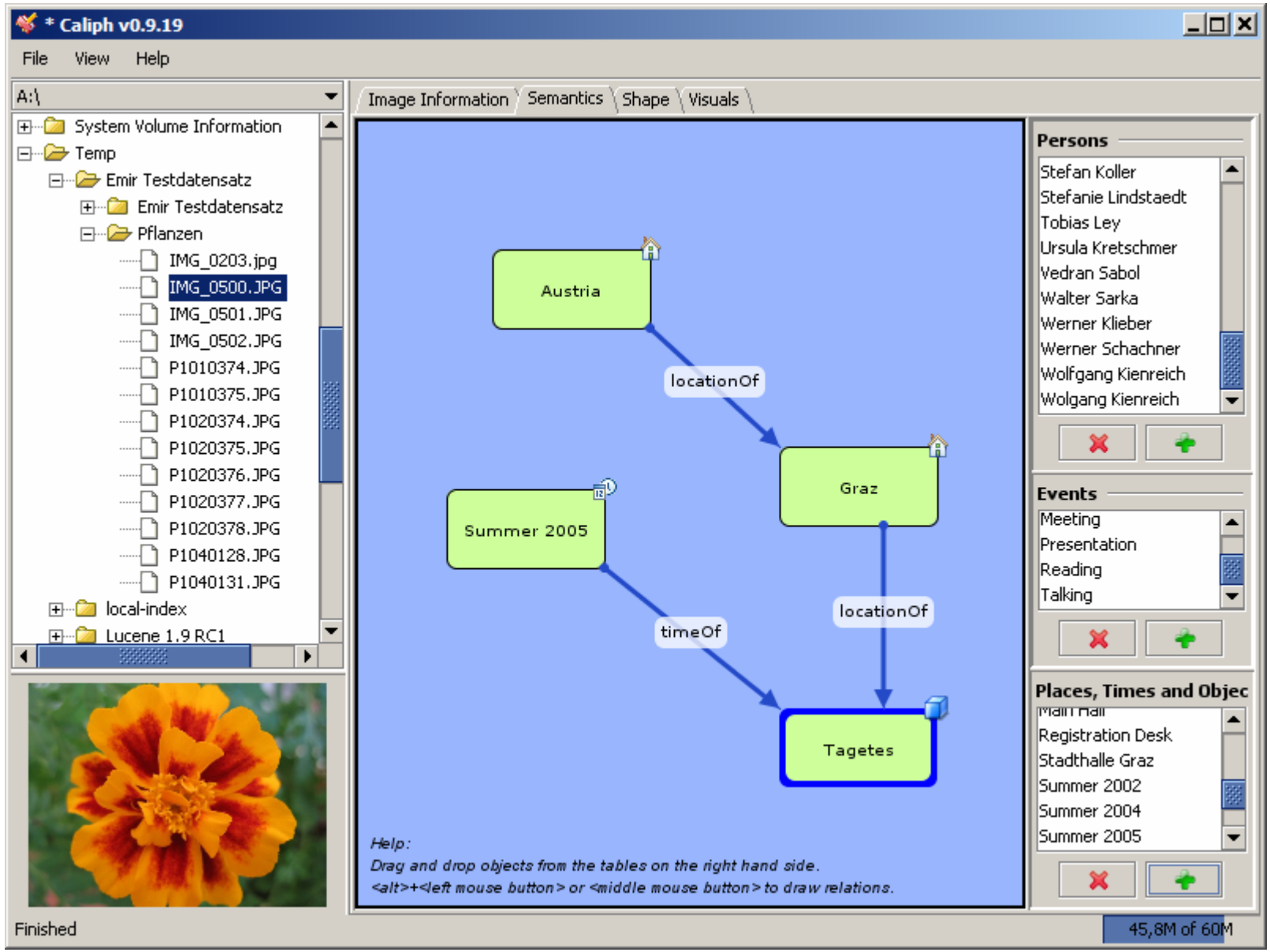
```
<Semantic>
- <Label> <Name>Shake hands</Name> </Label>
- <SemanticBase xsi:type="AgentObjectType" id="AOa">
- <Label href="urn:example:acs">
<Name>Person A</Name> </Label>
</SemanticBase>
- <SemanticBase xsi:type="AgentObjectType" id="AOb">
- <Label href="urn:example:acs">
<Name>Person B</Name> </Label>
</SemanticBase>
- <SemanticBase xsi:type="EventType" id="EV1">
- <Label href="urn:example:acs">
<Name>Shake hands</Name> </Label>
- <Definition>
<FreeTextAnnotation>Clasping of right hands by two people.</FreeTextAnnotation>
</Definition>
<Relation type="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:agent" target="#AOa" />
<Relation type="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:accompanier" target="#AOb" />
</SemanticBase>
</Semantic>
</Video>
</MultimediaContent>
</Description>
</Mpeg7>
```

Semantic Relations

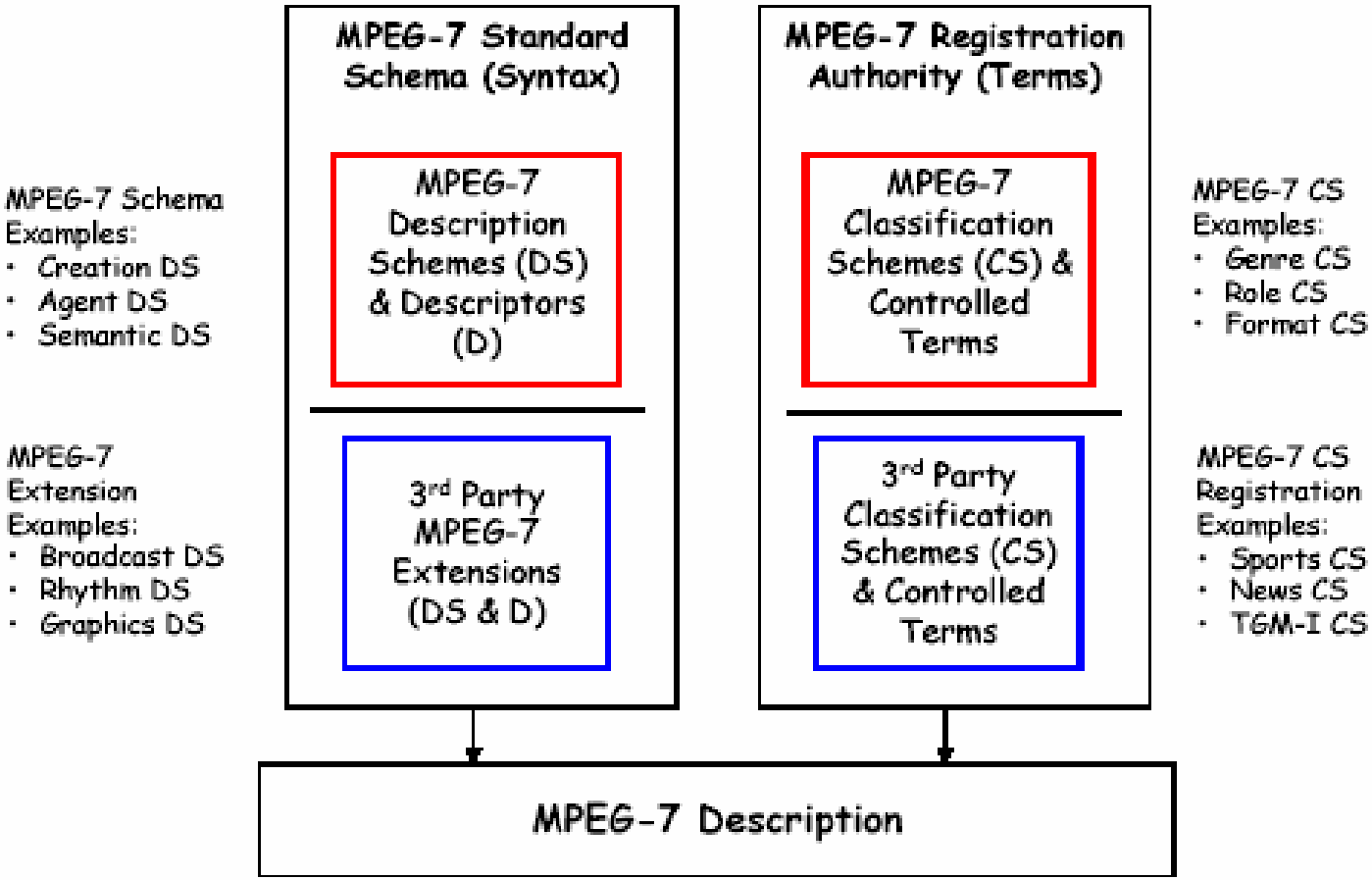
Semantics as a Reference

```
<Description xsi:type="SemanticDescriptionType">
   <Semantics id=”mysemantics”>
      <Label>
         <Name> Shake hands </Name>
      </Label>
      <SemanticBase xsi:type="AgentObjectType" id="AOa">
         <Label href="urn:example:acs">
            <Name> Person A </Name>
         </Label>
      </SemanticBase>
      <SemanticBase xsi:type="AgentObjectType" id="AOb">
         <Label href="urn:example:acs">
            <Name> Person B </Name>
         </Label>
      </SemanticBase>
      <SemanticBase xsi:type="EventType" id="EV1">
         <Label href="urn:example:acs">
            <Name> Shake hands </Name>
         </Label>
         <Definition>
            <FreeTextAnnotation>
               Clasping of right hands by two people.
            </FreeTextAnnotation>
         </Definition>
         <Relation
type="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:agent"
            target="#AOa"/>
         <Relation
type="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:accompanier"
            target="#AOb"/>
      </SemanticBase>
   </Semantics>
</Description>
```

```
<Segment xsi:type="VideoSegmentType">
   <TextAnnotation>
      <FreeTextAnnotation> video </FreeTextAnnotation>
   </TextAnnotation>
   <MediaTime>
      <MediaTimePoint>T00:00:00</MediaTimePoint>
         <MediaDuration>PT0M15S</MediaDuration>
      </MediaTime>
    <SegmentRef idref=”mysemantics”>
</Segment>
```


* Caliph v0.9.19
File
View
Help
A:\
System Volume Information
Temp
Emir Testdatensatz
Emir Testdatensatz
Pflanzen
IMG_0203.jpg
IMG_0500.JPG
IMG_0501.JPG
IMG_0502.JPG
P1010374.JPG
P1010375.JPG
P1020374.JPG
P1020375.JPG
P1020376.JPG
P1020377.JPG
P1020378.JPG
P1040128.JPG
P1040131.JPG
local-index
Lucene 1.9 RC1
Image Information
Semantics
Shape
Visuals
Austria
locationOf
Graz
Summer 2005
timeOf
locationOf
Tagetes
Help:
Drag and drop objects from the tables on the right hand side.
<alt>+<left mouse button> or <middle mouse button> to draw relations.
Persons
Stefan Koller
Stefanie Lindstaedt
Tobias Ley
Ursula Kretschmer
Vedran Sabol
Walter Sarka
Werner Klieber
Werner Schachner
Wolfgang Kienreich
Wolgang Kienreich
Events
Meeting
Presentation
Reading
Talking
Places, Times and Objec
Registration Desk
Stadthalle Graz
Summer 2002
Summer 2004
Summer 2005
Finished
45,8M of 60M

Classification Schemes (CS)



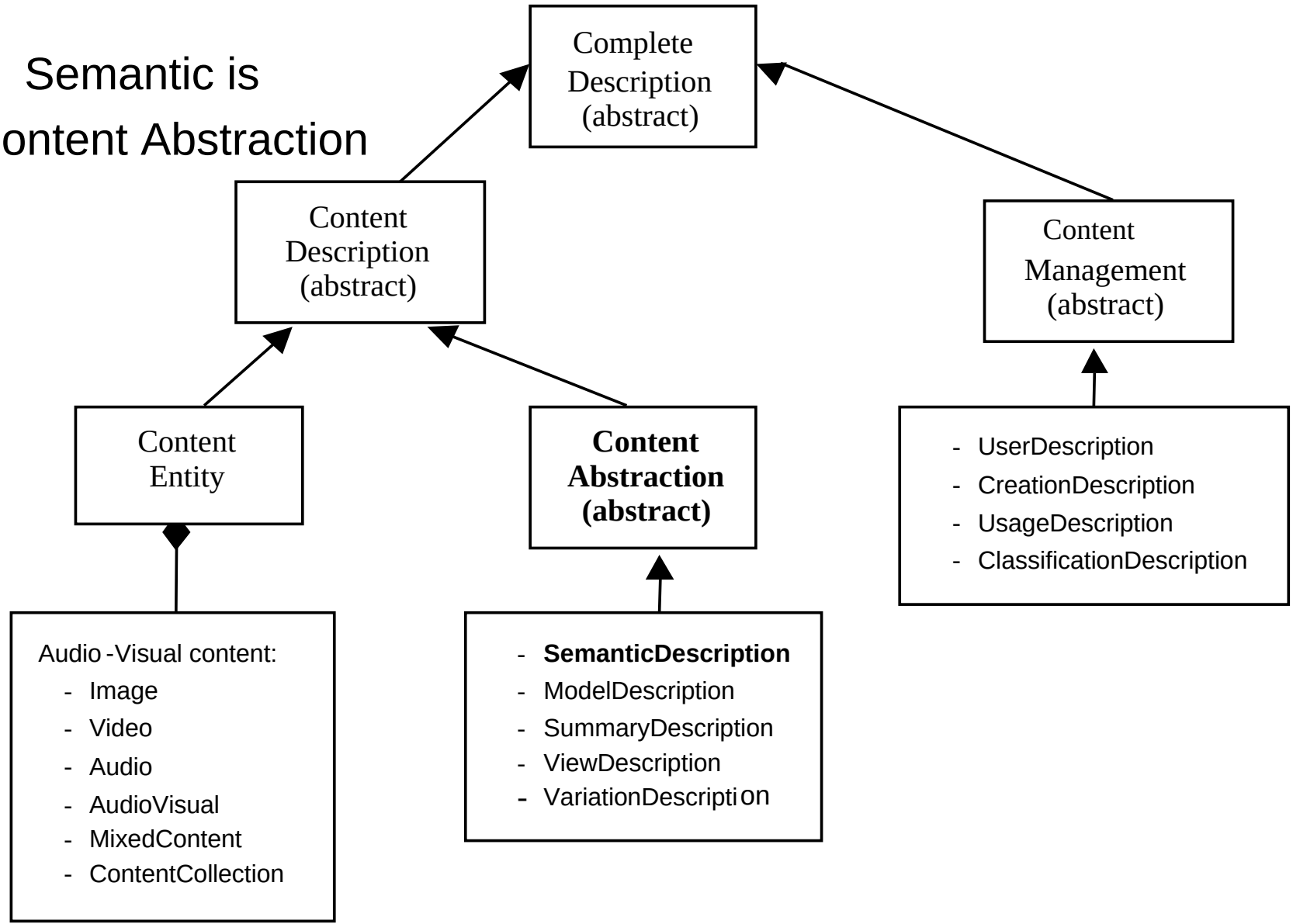
CS- Example FileFormatCS

```
<ClassificationScheme uri="urn:mpeg:mpeg7:cs:FileFormatCS:2001"
    domain="//MediaInformation/MediaProfile/MediaFormat/FileFormat">
    <Term termID="1">
        <Name xml:lang="en">jpeg</Name>
        <Name xml:lang="en">jpg</Name>
        <Name xml:lang="en">jfif</Name>
        <Definition xml:lang="en">JPEG file format</Definition>
    </Term>
    <Term termID="2">
        <Name xml:lang="en">JPEG 2000</Name>
        <Definition xml:lang="en">JPEG 2000 file format</Definition>
    </Term>
    <Term termID="3">
        <Name xml:lang="en">mpeg</Name>
        <Name xml:lang="en">mpg</Name>
        <Definition xml:lang="en">MPEG file format</Definition>
    </Term>
    <Term termID="4">
        <Name xml:lang="en">mp3</Name>
        <Definition xml:lang="en">MP3 audio format</Definition>
    </Term>
    <Term termID="5">
        <Name xml:lang="en">mp4</Name>
        <Definition xml:lang="en">MPEG-4 file format</Definition>
    </Term>
 … other terms …
</ClassificationScheme>
```

CS for parental rating, genre, intentions, quality,etc.
New CS must be registered by the MPEG-7 org.

Let's turn to the semantics

- Semantic is Content Abstraction



ContentAbstractionType

- ContentAbstractionType (abstract): top-level type for describing abstractions of multimedia content:
- SemanticDescriptionType: top-level type for describing semantics of multimedia content.
- ModelDescriptionType: top-level type for describing models of multimedia content.
- SummaryDescriptionType: top-level type for describing summaries of multimedia content.
- ViewDescriptionType: top-level type for describing views and view decompositions of audio-visual signals.
- VariationDescriptionType: top-level type for describing variations of multimedia content.

ContentAbstractionType and Semantic

```
<!-- Definition of ContentAbstraction Top-level Type -->
<complexType name="ContentAbstractionType" abstract="true">
   <complexContent>
      <extension base="mpeg7:ContentDescriptionType"/>
   </complexContent>
</complexType>

<!-- Definition of SemanticDescription Top-level Type -->
<complexType name="SemanticDescriptionType">
   <complexContent>
      <extension base="mpeg7:ContentAbstractionType">
         <choice>
            <element name="Semantics" type="mpeg7:SemanticType" minOccurs="1"
               maxOccurs="unbounded"/>
            <element name="ConceptCollection" type="mpeg7:ConceptCollectionType"
               minOccurs="1" maxOccurs="unbounded"/>
         </choice>
      </extension>
   </complexContent>
</complexType>
```

Empty !!

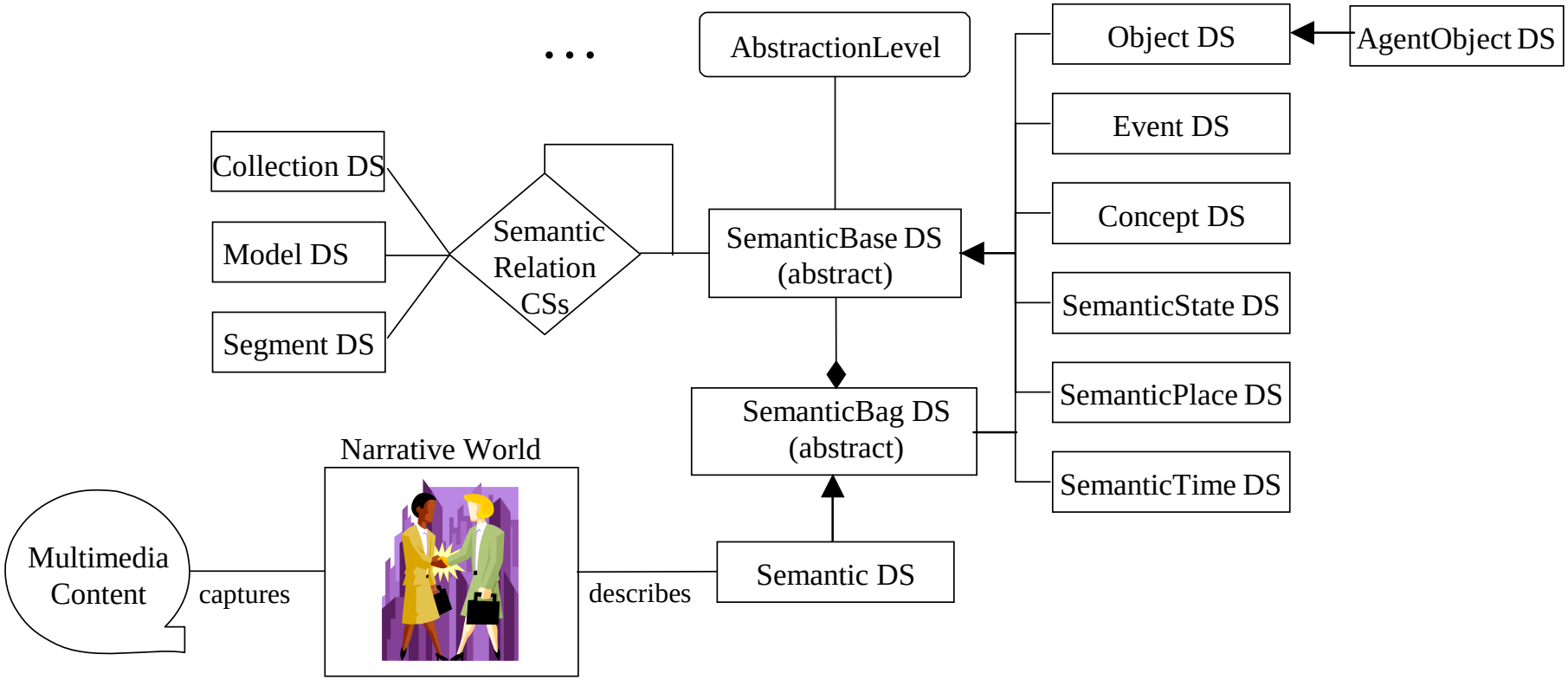
SemanticsType (1)

- That’s all ?

```
<!-- Definition of Semantic DS -->
<complexType name="SemanticType">
      <complexContent>
            <extension base="mpeg7:SemanticBagType"/>
      </complexContent>
</complexType>
```

SemanticsType (2)

That's it :



```
<!-- ######################################## -->
<!--  Definition of Semantic DS (12.3.4)  -->
<!-- ######################################## -->
<!-- Definition of Semantic DS -->
<complexType name="SemanticType">
	<complexContent>
		<extension base="mpeg7:SemanticBagType"/>
	</complexContent>
</complexType>

<!-- ########################################## -->
<!--  Definition of SemanticBag DS (12.3.3)  -->
<!-- ########################################## -->
<!-- Definition of SemanticBag DS -->
<complexType name="SemanticBagType" abstract="true">
	<complexContent>
		<extension base="mpeg7:SemanticBaseType">
			<sequence>
				<choice minOccurs="0" maxOccurs="unbounded">
					<element name="SemanticBase"
						type="mpeg7:SemanticBaseType"/>
					<element name="SemanticBaseRef"
						type="mpeg7:ReferenceType"/>
				</choice>
				<element name="Graph" type="mpeg7:GraphType" minOccurs="0"
						maxOccurs="unbounded"/>
			</sequence>
		</extension>
	</complexContent>
</complexType>
```

```
<!-- ######################################## -->
<!--  Definition of Object DS (12.3.5)  -->
<!-- ######################################## -->
<!-- Definition of Object DS -->
<complexType name="ObjectType">
	<complexContent>
		<extension base="mpeg7:SemanticBaseType">
			<choice minOccurs="0" maxOccurs="unbounded">
				<element name="Object" type="mpeg7:ObjectType"/>
				<element name="ObjectRef" type="mpeg7:ReferenceType"/>
			</choice>
		</extension>
	</complexContent>
</complexType>
<!-- ######################################## -->
<!--  Definition of AgentObject DS (12.3.6)  -->
<!-- ######################################## -->
<!-- Definition of AgentObject DS -->
<complexType name="AgentObjectType">
	<complexContent>
		<extension base="mpeg7:ObjectType">
			<choice minOccurs="0">
				<element name="Agent" type="mpeg7:AgentType"/>
				<element name="AgentRef" type="mpeg7:ReferenceType"/>
			</choice>
		</extension>
	</complexContent>
</complexType>
```

Example: Structure and Semantic Tools

- Structure descriptions:
  - Segment entities: video, audio, multimedia segments, …
  - Attributes: AV descriptors, spatio-temporal localization, …
  - Relationships: temporal, spatial, …
- Semantic descriptions:
  - Semantic entities: objects, events, concepts, states, places, times
  - Attributes: label, definition, abstraction level, media occurrences
  - Relationships: object-event, concept-semantic entity, …
  - Multiple abstraction levels: media and formal
  - Encoding of rules based on alphabet and rule graphs in the works

MPEG-7 Semantic Description

Content Management

Creation information:
Creation
Creator
Creation corrdinates
Creation location
Creation date

Photographer: Seungyup
Place: Columbia University
Time: 19 September 1998

Media information:
Media profile
Media format
Media instance

704x480 pixels
True color RGB
http://www.alex&ana.jpg

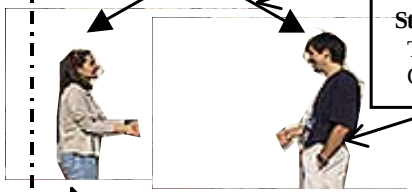
Usage unformation:
Rights

Columbia University,
All rights reserved

Content Structure

Still region SR1:
Creation information
Text annotation

Spatial segment decompos ition:
No overlap, gap



Still region SR2:
Text annotation
Color structure

Still region SR3:
Text annotation
Matching hint
Color structure

Directional spatial segment relation:
left

Content Semantics

Segment-semantic base relation:
hasMediaSymbolOf

Concept C1:
Label
Property
Property

Concept-semantic base relation:
hasPropertyOf

Comradeship

Object-event relation:
hasAccompanierOf

Shake hands

Object-event relation:
hasAgentOf

Alex

Ana

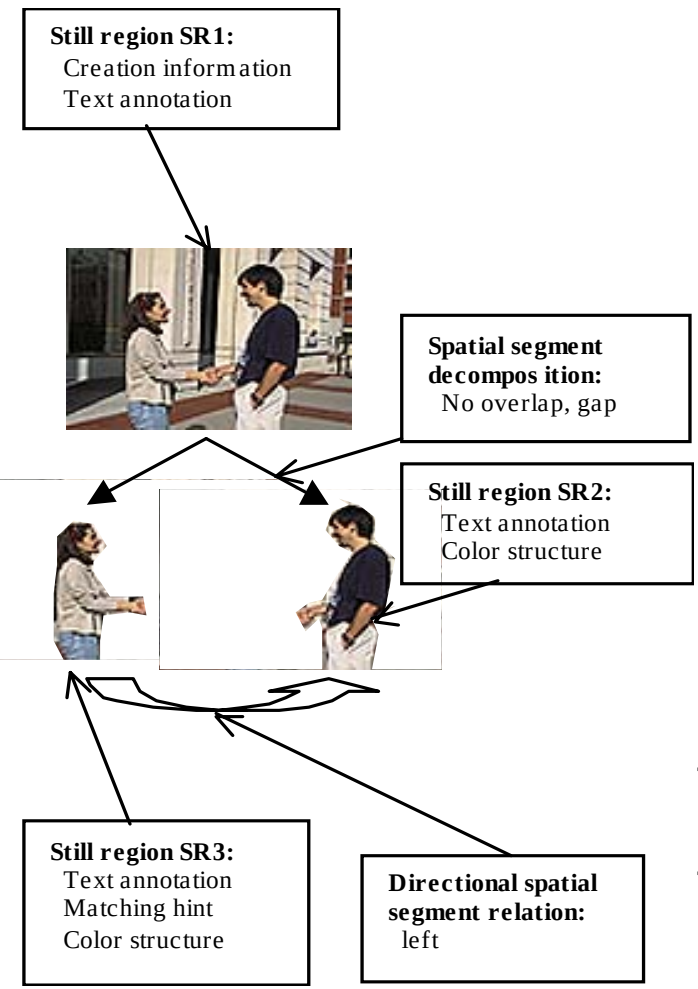
Event EV1:
Label
Semantic time
Semantic place

Agent object AO2:
Label
Person

Segment-semantic base relation:
hasMediaPerceptionOf

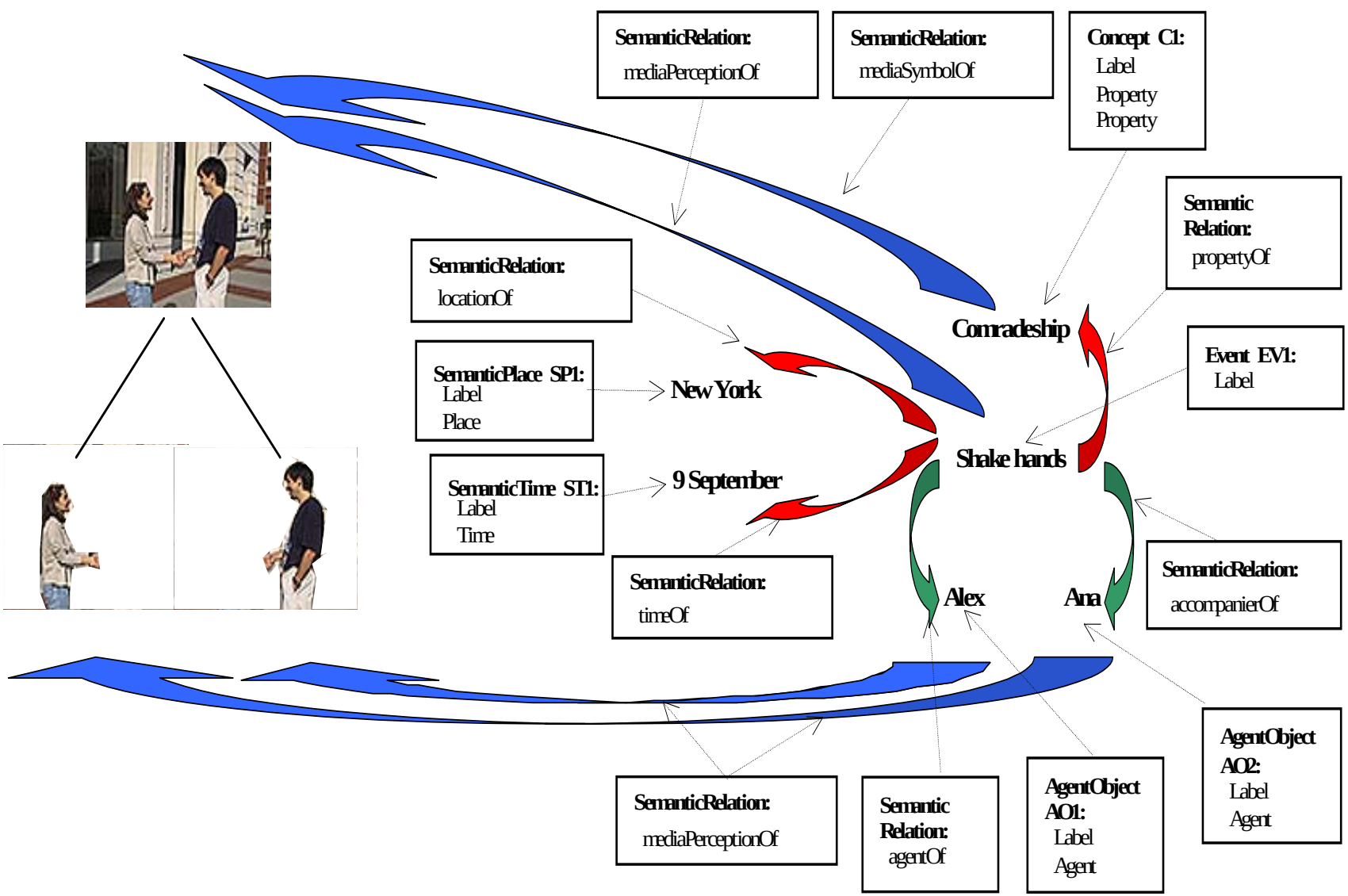
Agent object AO1:
Label
Person

MPEG-7: Content Structure Description

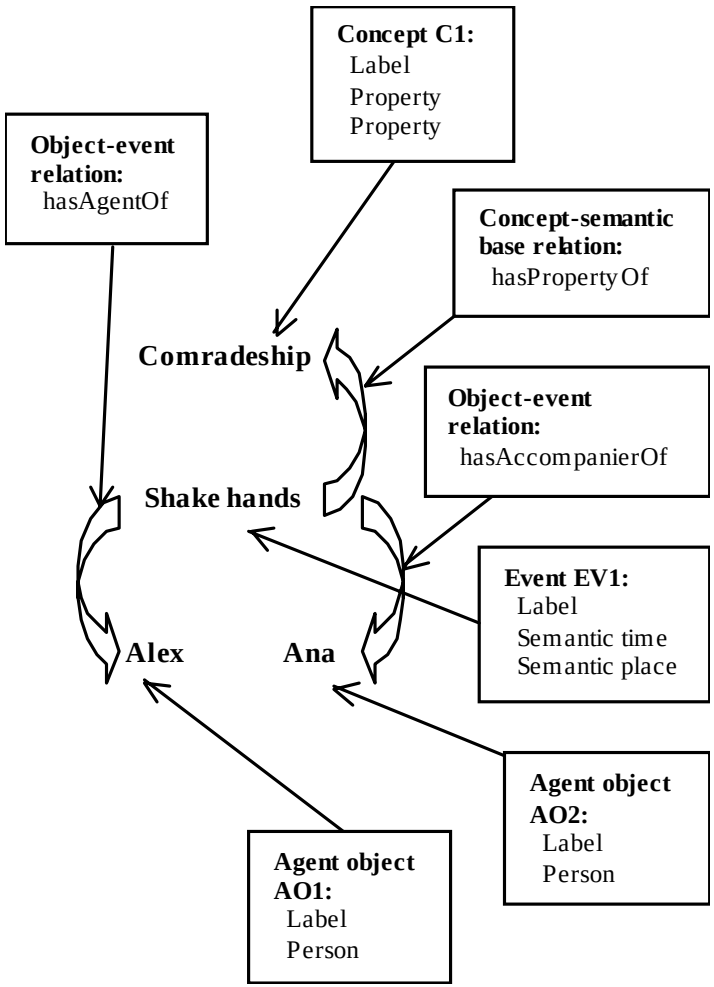



```
<StillRegion id="SR1">
  <TextAnnotation>
    <FreeTextAnnotation>
      Alex shakes hands with Ana
    </FreeTextAnnotation>
  </TextAnnotation>
  <SpatialDecomposition overlap="false" gap="true">

    <StillRegion id="SR2">
      <TextAnnotation> <FreeTextAnnotation> Alex </FreeTextAnnotation>
        </TextAnnotation>
      <VisualDescriptor xsi:type="ColorStructureType"> ... </VisualDescriptor>
    </StillRegion>

    <StillRegion id="SR3">
      <TextAnnotation> <FreeTextAnnotation> Ana </FreeTextAnnotation>
        </TextAnnotation>
      <MatchingHint>
        <Hint value="0.455" xpath=”../../VisualDescriptor"/>
      </MatchingHint>
          <Relation xsi:type="DirectionalSpatialSegmentRelationType“ name="left“
                    target="#SR2"/>
      <VisualDescriptor xsi:type="ColorStructureType"> ... </VisualDescriptor>
</StillRegion>

  </SpatialDecomposition>
</StillRegion>
```

Example (1) – Available Semantic Types

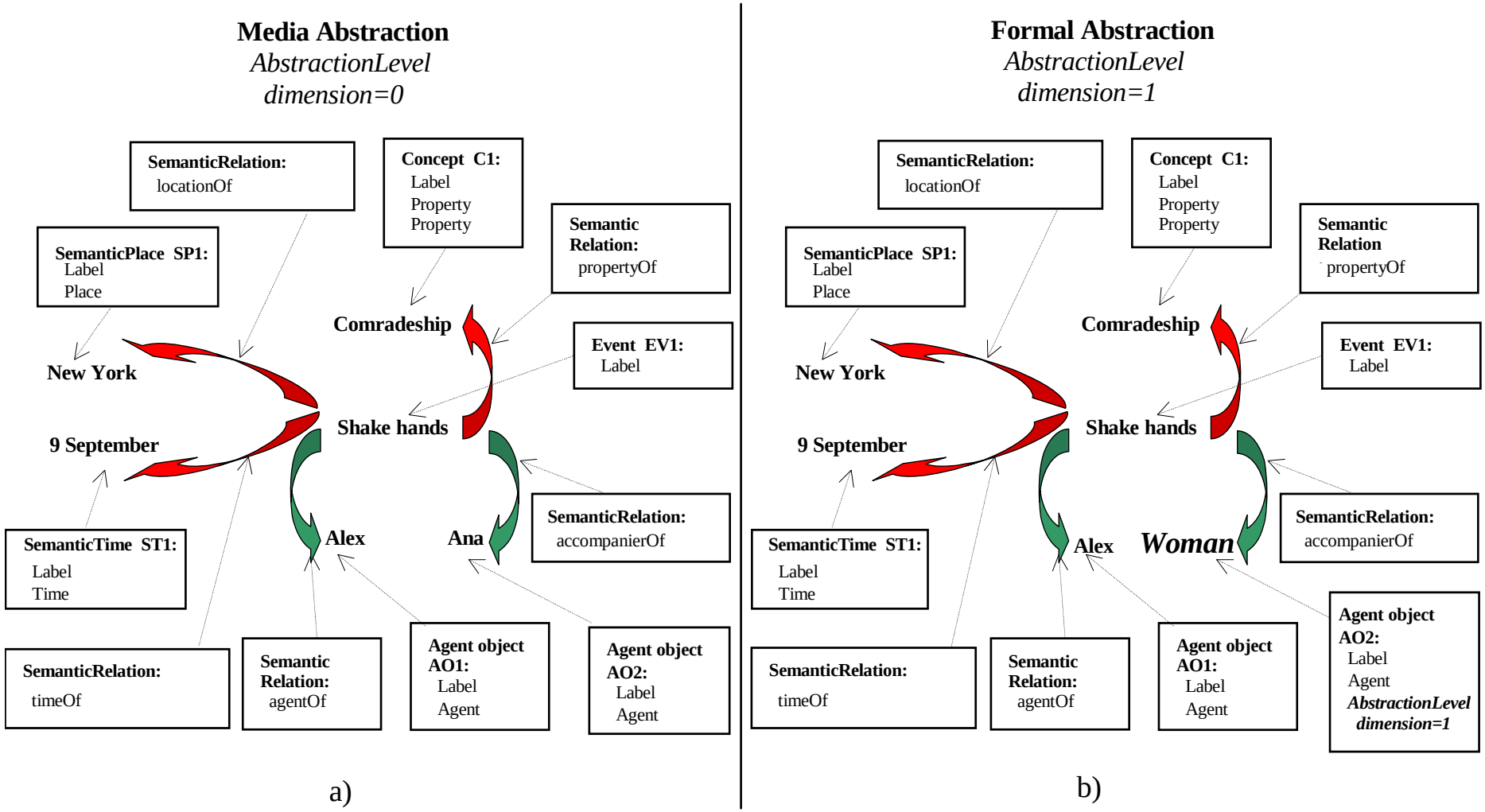


MPEG-7: Content Semantics Description




```
<Semantic>
 <Label><Name>Alex shakes hands with Ana </Name></Label>
 <SemanticBase xsi:type="EventType" id="EV1">
  <Label><Name>Shake hands</Name></Label>
  <Relation xsi:type="ObjectEventRelationType“
     name="hasAgentOf“
                target="#AO1"/>
  <Relation xsi:type="ObjectEventRelationType“
     name="hasAccompanierOf“
                 target="#AO2"/>
  <Relation xsi:type="ConceptSemanticBaseRelationType“
                name="hasPropertyOf" target="#C1"/>
  <SemanticPlace> <Label><Name>Columbia
     University</Name></Label>
  </SemanticPlace>
  <SemanticTime> <Label><Name>September 9,
     1998</Name></Label>
  </SemanticTime>
 </SemanticBase>
 <SemanticBase xsi:type="AgentObjectType" id="AO1">
  <Label><Name>Alex</Name></Label>
  <Agent xsi:type="PersonType">
     <Name><GivenName>Alex</GivenName></Name> </Agent>
 </SemanticBase>
 <SemanticBase xsi:type="AgentObjectType" id="AO2">
  <Label><Name>Ana</Name></Label>
  <Agent xsi:type="PersonType">
     <Name><GivenName>Ana</GivenName></Name> </Agent>
 </SemanticBase>
 <SemanticBase xsi:type="ConceptType" id="C1">
  <Label><Name>Comradeship</Name></Label>
  <Property>Associate</Property> <Property>Friend</Property>
 </SemanticBase>
</Semantic>
```

Example (2) – Abstraction



Object

How to describe a table !!!

```
<Mpeg7>
   <Description xsi:type="SemanticDescriptionType">
      <Semantics>
         <Label>
            <Name> Description of table </Name>
         </Label>
         <SemanticBase xsi:type="ObjectType">
            <Label>
               <Name> Table </Name>
            </Label>
            <Definition>
               <FreeTextAnnotation>
                  Piece of furniture consisting of a smooth flat slab
                  fixed on legs
               </FreeTextAnnotation>
            </Definition>
            <MediaOccurrence>
               <MediaLocator>
                  <MediaUri>image.jpg</MediaUri>
               </MediaLocator>
               <VisualDescriptor xsi:type="ScalableColorType"
numOfCoeff="16"
                  numOfBitplanesDiscarded="0">
                  <Coeff> 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 </Coeff>
               </VisualDescriptor>
            </MediaOccurrence>
            <Object id="tabletop-object">
               <Label>
                  <Name> Table top </Name>
               </Label>
               <MediaOccurrence>
                  <MediaLocator>
                     <MediaUri>image.jpg</MediaUri>
                  </MediaLocator>
                  <Mask xsi:type="SpatialMaskType">
                     <SubRegion>
                        <Polygon>
                           <Coords mpeg7:dim="2 5"> 5 25 10 20 15 15 10
10 5 15
                           </Coords>
                        </Polygon>
                     </SubRegion>
                  </Mask>
               </MediaOccurrence>
            </Object>
…
```

AgentObject

```
<Mpeg7>
    <Description xsi:type="SemanticDescriptionType">
        <Semantics>
            <Label>
                <Name> Description of person </Name>
            </Label>
            <SemanticBase xsi:type="AgentObjectType">
                <Label>
                    <Name> Student </Name>
                </Label>
                <Definition>
                    <FreeTextAnnotation> Student named Alexander Green
                    </FreeTextAnnotation>
                </Definition>
                <Agent xsi:type="PersonType">
                    <Name>
                        <GivenName abbrev="Alex"> Alexander </GivenName>
                        <FamilyName> Green </FamilyName>
                    </Name>
                </Agent>
            </SemanticBase>
        </Semantics>
    </Description>
</Mpeg7>
```

An AgentObject is a Person, maybe your dog and cat, too.

```
<!-- ######################################## -->
<!--  Definition of Event DS (12.3.7)  -->
<!-- ######################################## -->
<!-- Definition of Event DS -->
<complexType name="EventType">
      <complexContent>
            <extension base="mpeg7:SemanticBaseType">
                  <sequence>
                        <choice minOccurs="0" maxOccurs="unbounded">
                              <element name="Event" type="mpeg7:EventType"/>
                              <element name="EventRef" type="mpeg7:ReferenceType"/>
                        </choice>
                        <element name="SemanticPlace" type="mpeg7:SemanticPlaceType“
                              minOccurs="0" maxOccurs="unbounded"/>
                        <element name="SemanticTime" type="mpeg7:SemanticTimeType“
                              minOccurs="0" maxOccurs="unbounded"/>
                  </sequence>
            </extension>
      </complexContent>
</complexType>
```

Event

- Let's

Skake Hands

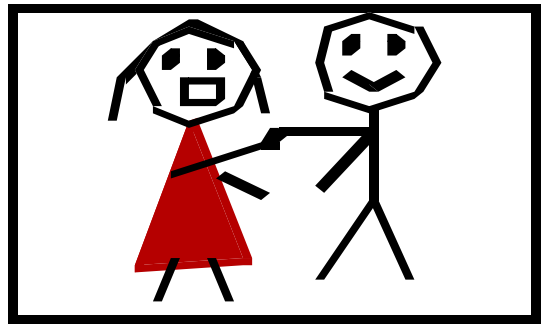
Semantic Relations

```
<Mpeg7>
   <Description xsi:type="SemanticDescriptionType">
      <Semantics>
         <Label>
            <Name> Shake hands </Name>
         </Label>
         <SemanticBase xsi:type="AgentObjectType" id="AOa">
            <Label href="urn:example:acs">
               <Name> Person A </Name>
            </Label>
         </SemanticBase>
         <SemanticBase xsi:type="AgentObjectType" id="AOb">
            <Label href="urn:example:acs">
               <Name> Person B </Name>
            </Label>
         </SemanticBase>
         <SemanticBase xsi:type="EventType" id="EV1">
            <Label href="urn:example:acs">
               <Name> Shake hands </Name>
            </Label>
            <Definition>
               <FreeTextAnnotation>
                  Clasping of right hands by two people.
               </FreeTextAnnotation>
            </Definition>
            <Relation
               type="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:agent"
               target="#AOa"/>
            <Relation
              type="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:accompanier"
               target="#AOb"/>
         </SemanticBase>
      </Semantics>
   </Description>
</Mpeg7>
```

MPEG-7 MDS: Event DS Example

- The following example gives an MPEG-7 description of the event of handshake between people:

```
<Mpeg7>
  <Description xsi:type="SemanticDescriptionType">
    <Semantics>
      <Label>
        <Name> Shake hands </Name>
      </Label>
      <SemanticBase xsi:type="AgentObjectType" id="A">
        <Label href="urn:example:acs">
         <Name> Person A </Name>
        </Label>
      </SemanticBase>
      <SemanticBase xsi:type="AgentObjectType" id="B">
        <Label href="urn:example:acs">
          <Name> Person B </Name>
        </Label>
      </SemanticBase>
      <SemanticBase xsi:type="EventType">
        <Label><Name> Handshake </Name></Label>
        <Definition>
          <FreeTextAnnotation> Clasping of right hands by two people </FreeTextAnnotation>
        </Definition>
        <Relation type="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:agent" target="#A"/>
        <Relation type="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:accompanier“ target="#B"/>
      </SemanticBase>
    </Semantics>
  </Description>
</Mpeg7>
```



Semantic Relations are CS

- The **SemanticRelation CS** defines semantic relations. The semantic relations describe semantic information that relates entities such as the relationship of events in a narrative, or relationship of an object to multimedia content that depicts the object.
- In the following specification of the SemanticRelation CS, symbol A refers to the source of the relation and symbol B refers to the target of the relation.

```
<ClassificationScheme uri="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001">
   <Header xsi:type="DescriptionMetadataType">
      <Comment>
            <FreeTextAnnotation xml:lang="en">
                  Semantic relations
            </FreeTextAnnotation>
      </Comment>
   </Header>
      <Term termID="key">
            <Definition>
                  A key B if and only if B is a key for accessing A.
            </Definition>
      </Term>
      <Term termID="keyFor">
            <Definition> Inverse relation of key </Definition>
      </Term>
      <Term termID="annotates">
            <Definition>
                  A annotates B if and only if A is an annotation or
description of B.
            </Definition>
      </Term>
```

Concept

Freedom !

```
<Mpeg7>
      <Description xsi:type="SemanticDescriptionType">
            <Semantics>
                  <Label>
                        <Name> Concept of freedom </Name>
                  </Label>
                  <SemanticBase xsi:type="ConceptType" id="freedom">
                        <Label>
                              <Name> Freedom </Name>
                        </Label>
                        <Property>
                              <Name> Open </Name>
                        </Property>
                        <Property>
                              <Name> Outspoken </Name>
                        </Property>
                        <Property>
                              <Name> Frank </Name>
                        </Property>
                        <MediaOccurrence type="symbol">
                              <MediaLocator>
                                    <MediaUri> liberty.gif
                                    </MediaUri>
                              </MediaLocator>
                        </MediaOccurrence>
                  </SemanticBase>
            </Semantics>
      </Description>
</Mpeg7>
```

Semantic State

Describe a sunset

```
<Mpeg7>
   <Description xsi:type="SemanticDescriptionType">
      <Semantics>
         <Label>
            <Name> Sunset </Name>
         </Label>
         <SemanticBase xsi:type="EventType" id="sunset-event">
            <Label>
               <Name> Sunset </Name>
            </Label>
         </SemanticBase>
         <SemanticBase xsi:type="SemanticStateType" id="sunset-state">
            <Label>
               <Name> State of sunset </Name>
            </Label>
            <AttributeValuePair>
               <Attribute>
                  <Name> Blue </Name>
               </Attribute>
               <IntegerValue>90</IntegerValue>
            </AttributeValuePair>
            <AttributeValuePair>
               <Attribute>
                  <Name> Pink </Name>
               </Attribute>
               <IntegerValue>9</IntegerValue>
            </AttributeValuePair>
            <AttributeValuePair>
               <Attribute>
                  <Name> Orange </Name>
               </Attribute>
               <IntegerValue>50</IntegerValue>
            </AttributeValuePair>
         </SemanticBase>
         <Graph>
            <!--  Relate event sunset-event and state sunset-state  -->
            <Relation
type="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:state"
               source="#sunset-event" target="#sunset-state"/>
         </Graph>
      </Semantics>
   </Description>
</Mpeg7>
```

SemanticPlace

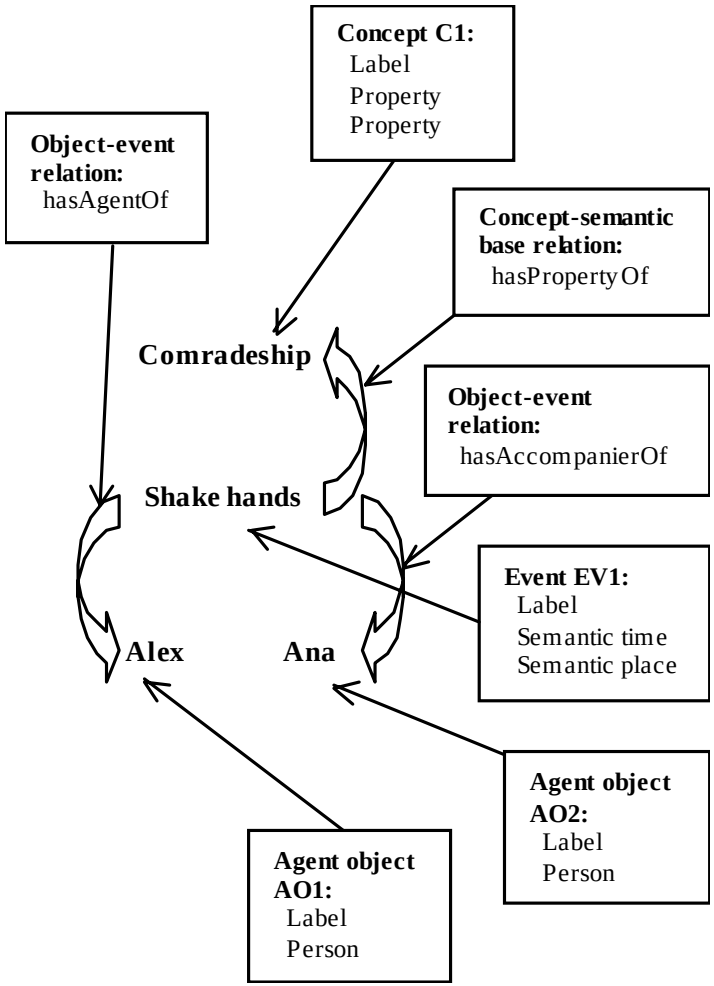
```
<Mpeg7>
   <Description xsi:type="SemanticDescriptionType">
      <Semantics>
         <Label>
            <Name> Semantic place for Columbia University </Name>
         </Label>
         <SemanticBase xsi:type="SemanticPlaceType" id="columbia-
                                          location">
            <Label>
               <Name> Columbia University </Name>
            </Label>
            <Place>
               <Name xml:lang="en">
                  Columbia University in the City of New York </Name>
               <Region> us </Region>
               <PostalAddress>
                  <AddressLine>  600 West 116th Street, New York,
                              NY</AddressLine>
                  <PostingIdentifier> U-10027 </PostingIdentifier>
               </PostalAddress>
            </Place>
         </SemanticBase>
      </Semantics>
   </Description>
</Mpeg7>
```

SemanticTime

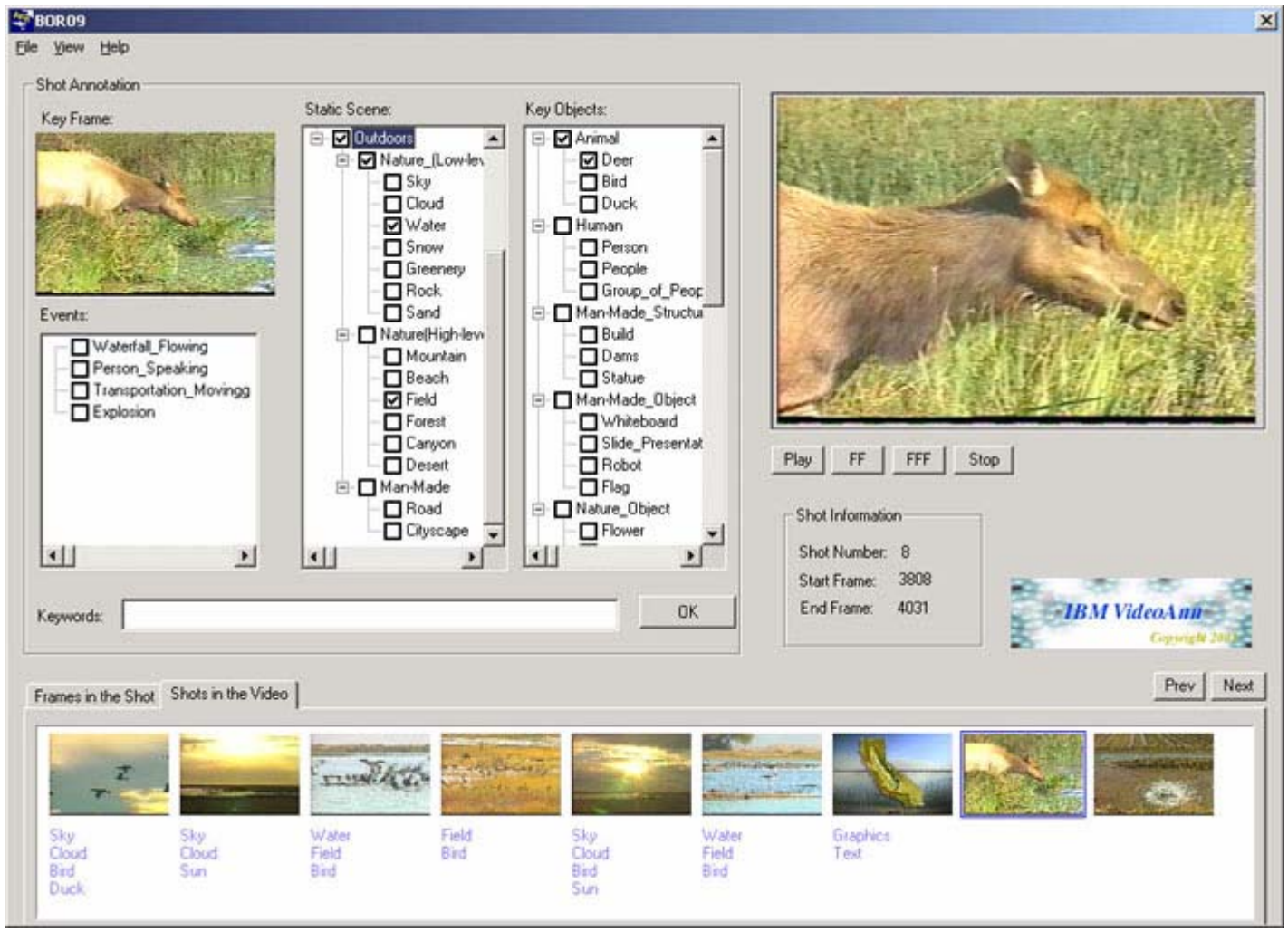
```
<Mpeg7>
    <Description xsi:type="SemanticDescriptionType">
        <Semantics>
            <Label>
                <Name> Description of last year </Name>
            </Label>
            <SemanticBase xsi:type="SemanticTimeType">
                <Label>
                    <Name> Last year </Name>
                </Label>
                <SemanticTimeInterval>
                    <TimePoint origin="now">
                        <Displacement measurementType="length" unit="year"
                              value="1"/>
                        <Direction measurementType="direction" unit="direction"
                            value="before"/>
                    </TimePoint>
                </SemanticTimeInterval>
            </SemanticBase>
        </Semantics>
    </Description>
</Mpeg7>
```

Last year ...

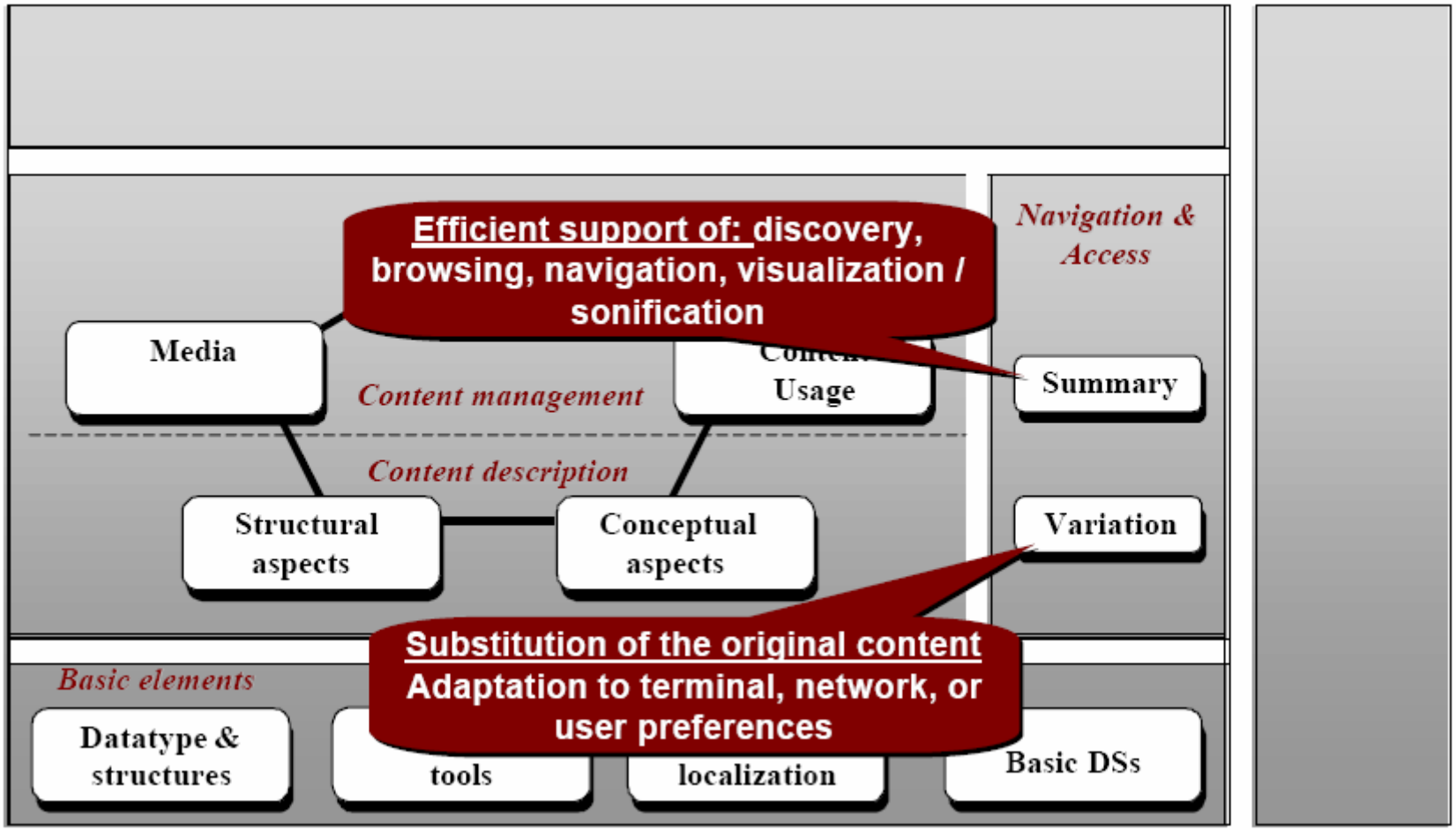
Partial Complete Semantics Description



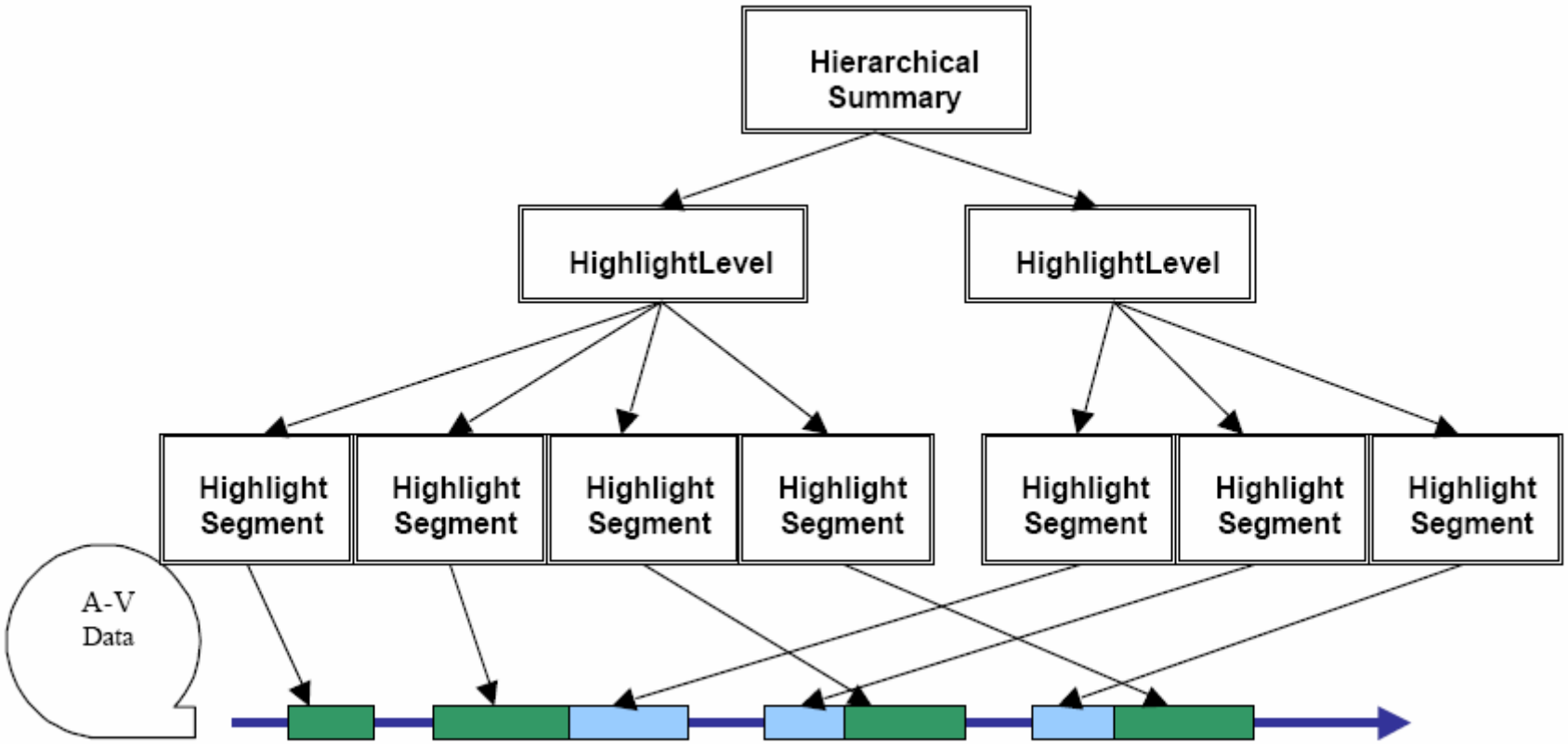

```
<Semantic>
 <Label><Name>Alex shakes hands with Ana </Name></Label>
 <SemanticBase xsi:type="EventType" id="EV1">
  <Label><Name>Shake hands</Name></Label>
  <Relation xsi:type="ObjectEventRelationType“
     name="hasAgentOf“
                target="#AO1"/>
  <Relation xsi:type="ObjectEventRelationType“
     name="hasAccompanierOf“
                 target="#AO2"/>
  <Relation xsi:type="ConceptSemanticBaseRelationType“
                name="hasPropertyOf" target="#C1"/>
  <SemanticPlace> <Label><Name>Columbia
     University</Name></Label>
  </SemanticPlace>
  <SemanticTime> <Label><Name>September 9,
     1998</Name></Label>
  </SemanticTime>
 </SemanticBase>
 <SemanticBase xsi:type="AgentObjectType" id="AO1">
  <Label><Name>Alex</Name></Label>
  <Agent xsi:type="PersonType">
     <Name><GivenName>Alex</GivenName></Name> </Agent>
 </SemanticBase>
 <SemanticBase xsi:type="AgentObjectType" id="AO2">
  <Label><Name>Ana</Name></Label>
  <Agent xsi:type="PersonType">
     <Name><GivenName>Ana</GivenName></Name> </Agent>
 </SemanticBase>
 <SemanticBase xsi:type="ConceptType" id="C1">
  <Label><Name>Comradeship</Name></Label>
  <Property>Associate</Property> <Property>Friend</Property>
 </SemanticBase>
</Semantic>
```


BOR09
File View Help
Shot Annotation
Key Frame:
Static Scene:
Key Objects:
Outdoors
Nature_(Low-lev
Sky
Cloud
Water
Snow
Greenery
Rock
Sand
Nature(High-lev
Mountain
Beach
Field
Forest
Canyon
Desert
Man-Made
Road
Cityscape
Animal
Deer
Bird
Duck
Human
Person
People
Group_of_Peop
Man-Made_Structu
Build
Dams
Statue
Man-Made_Object
Whiteboard
Slide_Presentat
Robot
Flag
Nature_Object
Flower
Events:
Waterfall_Flowing
Person_Speaking
Transportation_Movingg
Explosion
Keywords:
OK
Play
FF
FFF
Stop
Shot Information
Shot Number: 8
Start Frame: 3808
End Frame: 4031
IBM VideoAnn
Prev
Next
Frames in the Shot
Shots in the Video
Sky Cloud Bird Duck
Sky Cloud Sun
Water Field Bird
Field Bird
Sky Cloud Bird Sun
Water Field Bird
Graphics Text

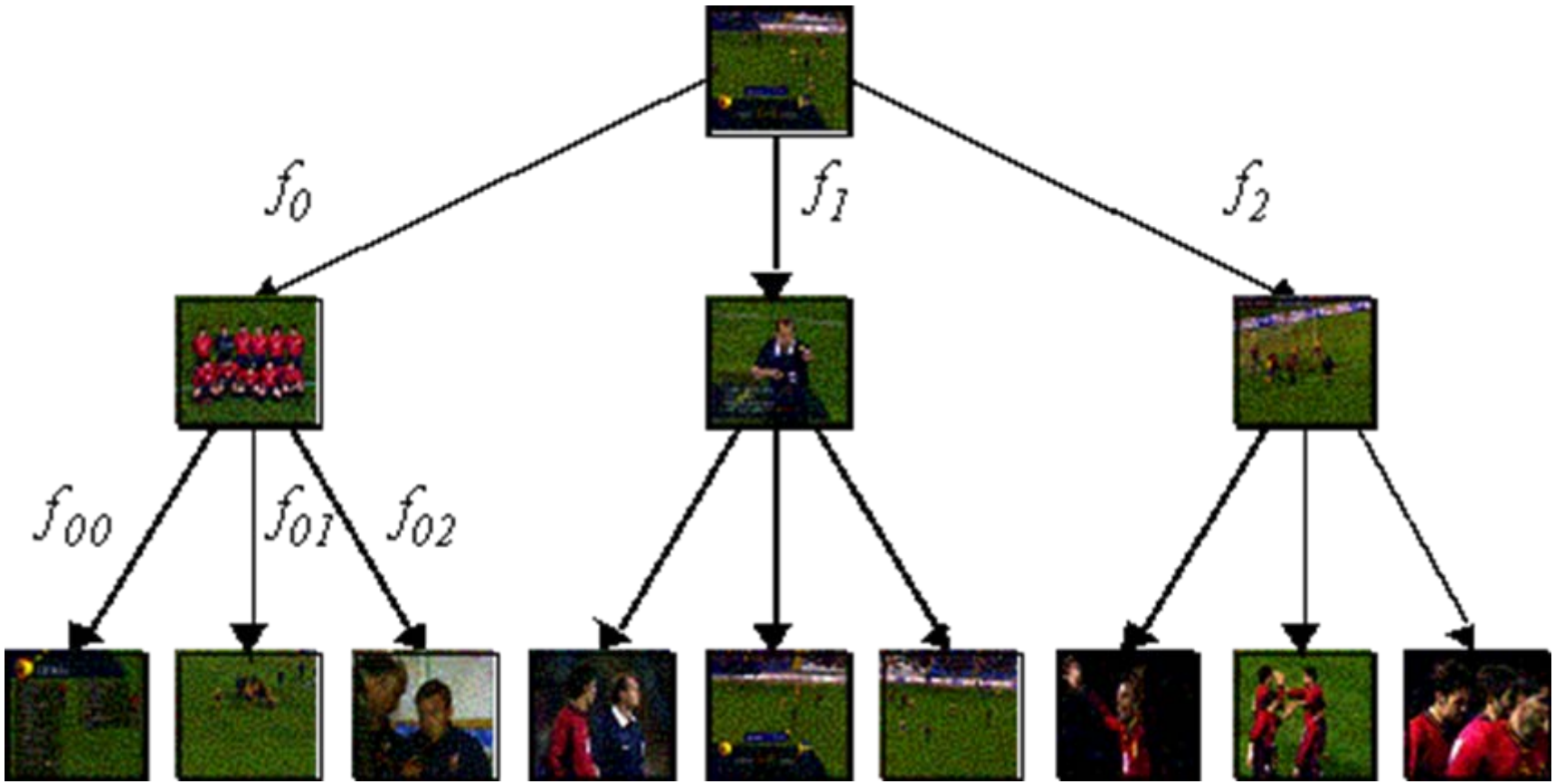
Navigation and access



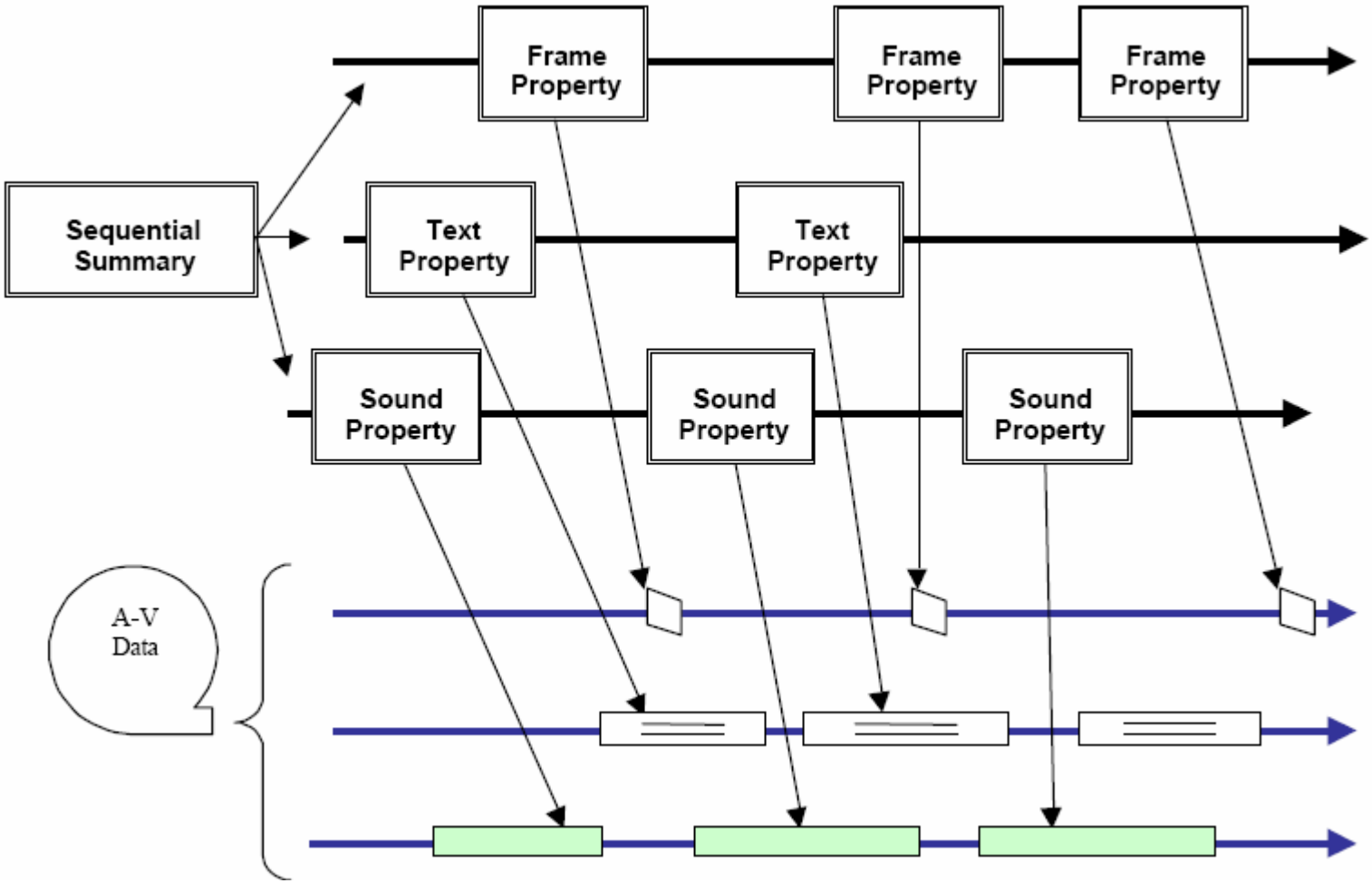
Navigation: hierarchical summary



Navigation: hierarchical summary example

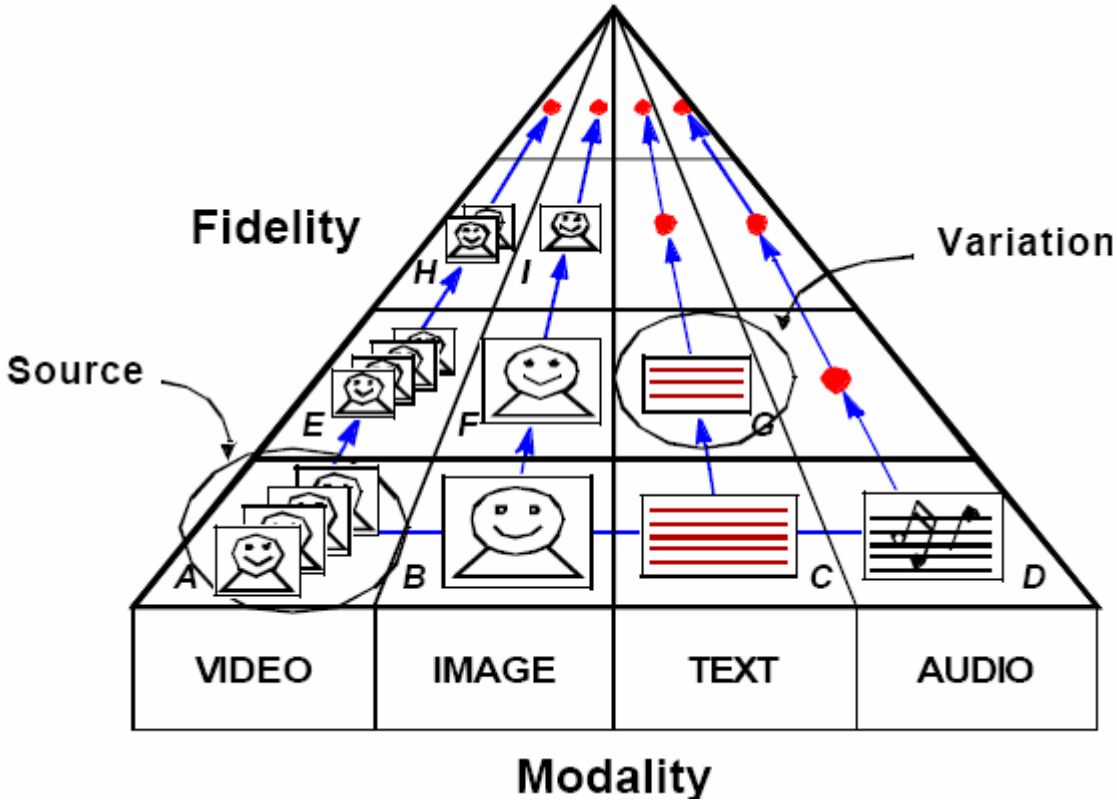


Navigation: sequential summary



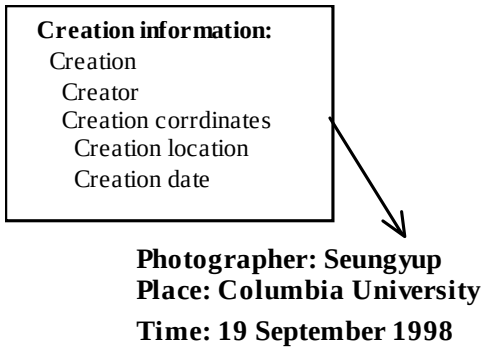
Variation

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



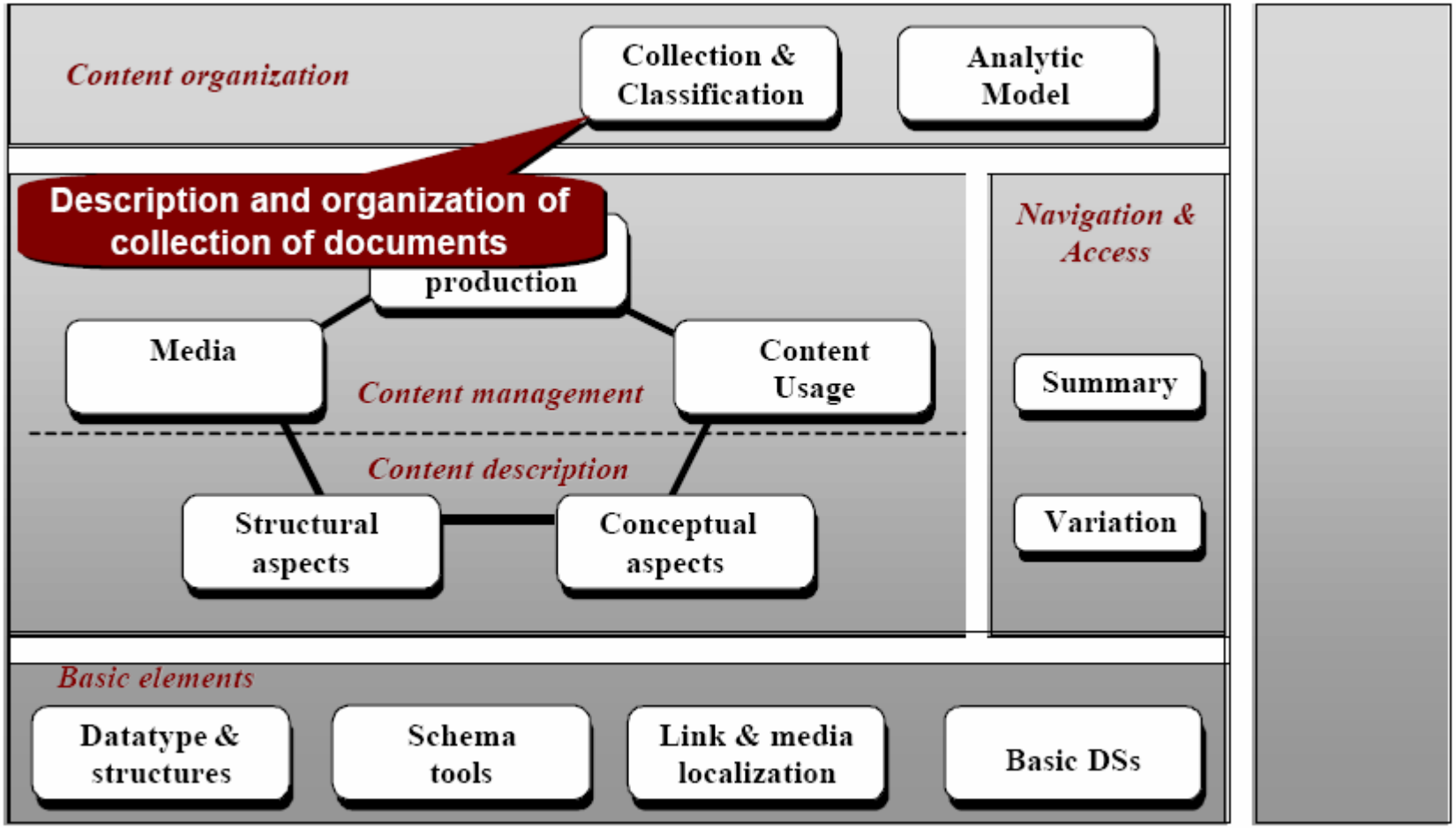
- The Variation DS describes variations of the AV content, such as compressed or low-resolution versions, summaries, different languages, and different modalities, such as audio, video, image, text, and so forth.
  - One of the targeted functionalities of the Variation DS is to allow a server or proxy to select the most suitable variation of the AV content for delivery according to the capabilities of terminal devices, network conditions, or user preferences. T
- he Variations DS describes the different alternative variations. The variations may refer to newly authored AV content, or correspond to AV content derived from another source.
- A variation fidelity value gives the quality of the variation compared to the original.
- The variation type attribute indicates the type of variation, such as summary, abstract, extract, modality translation, language translation, color reduction, spatial reduction, rate reduction, compression, and so forth.

Content Management

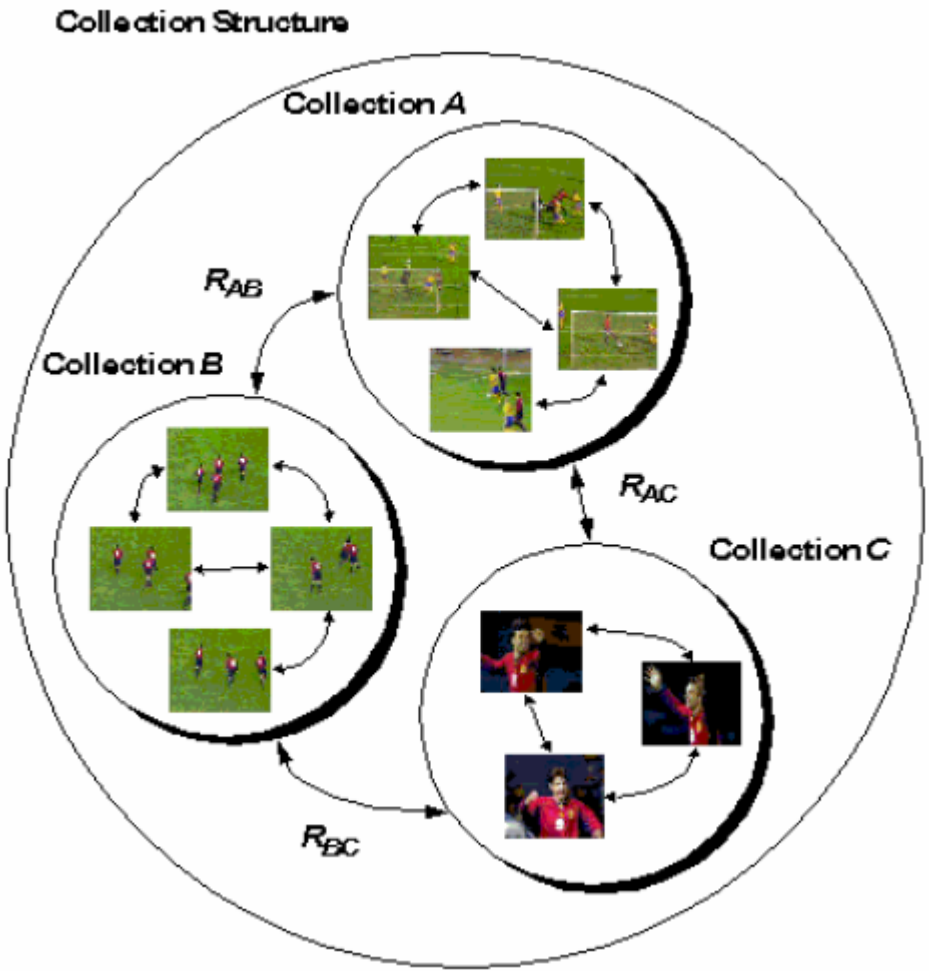



```
<CreationInformation>
  <Creation>
    <Creator>
      <Role><Name>Photographer</Name></Role>
      <Person> <Name> <GivenName>Seungyup</GivenName> </Name> </Person>
    </Creator>
    <CreationCoordinates>
      <CreationLocation> <Name xml:lang="en">Columbia University</Name>
        </CreationLocation>
      <CreationDate> <TimePoint>1998-09-19</TimePoint> </CreationDate>
    </CreationCoordinates>
  </Creation>
</CreationInformation>
<MediaInformation>
  <MediaProfile master="true">
    <MediaFormat> <Content>image</Content>
      <VisualCoding>
        <Format colorDomain="color“
        href="urn:mpeg:VisualCodingFormatCS:1">JPG</Format>
        <Frame height="480" width="704"/>
      </VisualCoding>
      </MediaFormat>
    <MediaInstance id="mastercopy">
      <MediaLocator> <MediaUri> http://www.alex&ana.jpg </MediaUri>
        </MediaLocator>
    </MediaInstance>
  </MediaProfile>
</MediaInformation>
<UsageInformation>
  <Rights>
    <RightsId organization="Columbia University“> columbia:1919:alex&ana_image
      </RightsId>
  </Rights>
</UsageInformation>
```

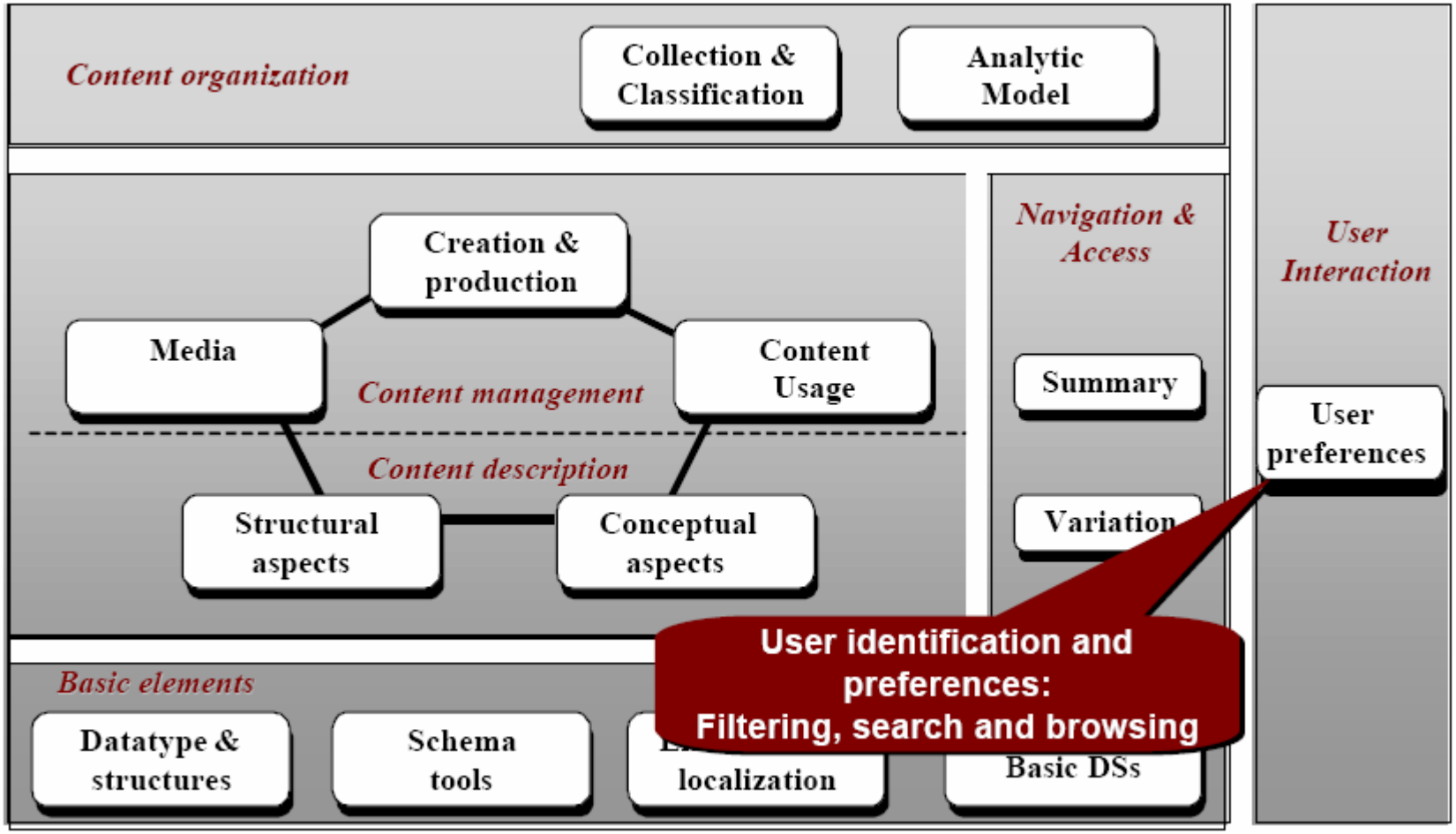
Content organization



Collection

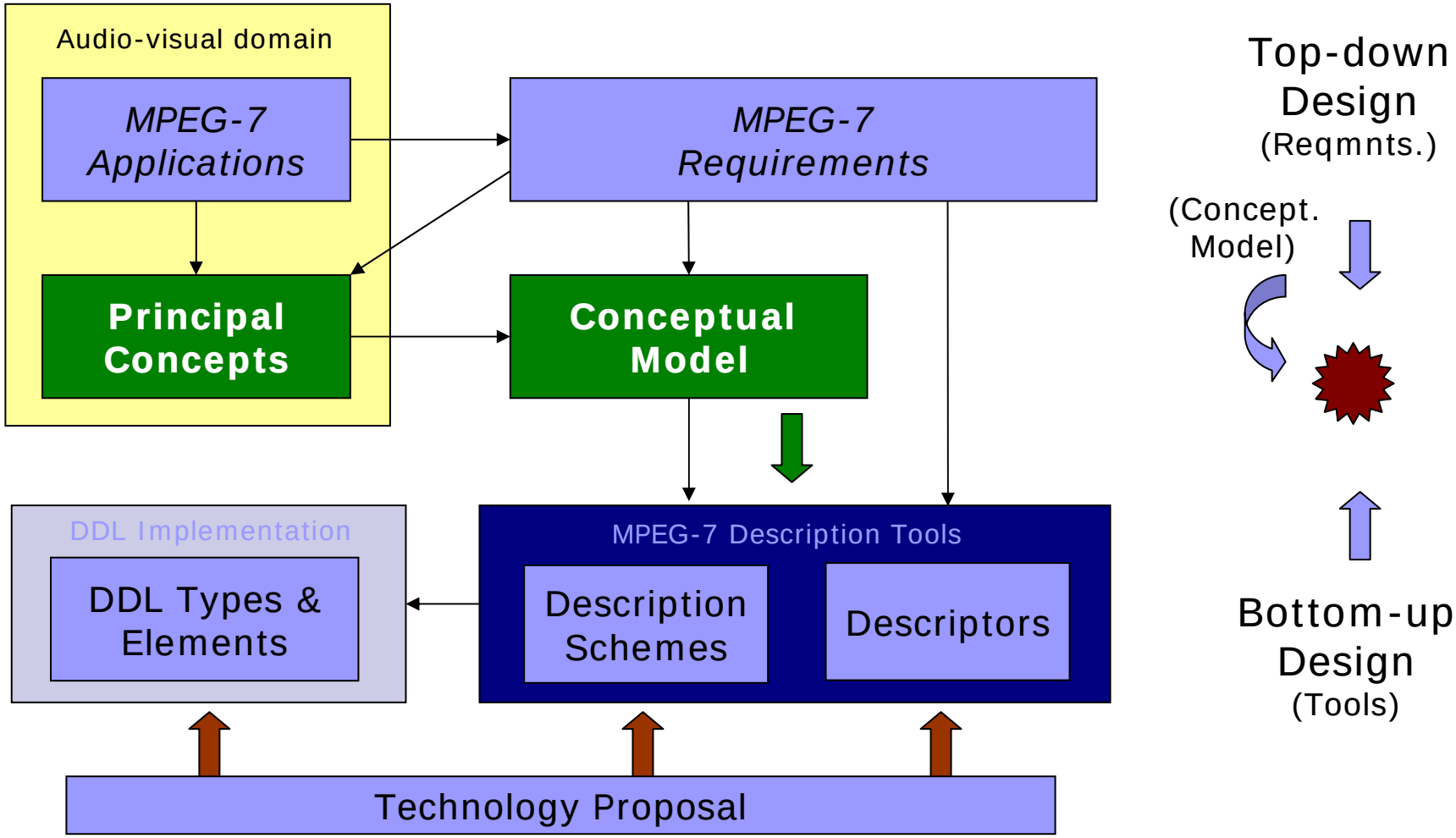

Collection Structure
Collection A
R_{AB}
Collection B
R_{AC}
Collection C
R_{BC}

User interaction



MPEG-7 Conceptual Model

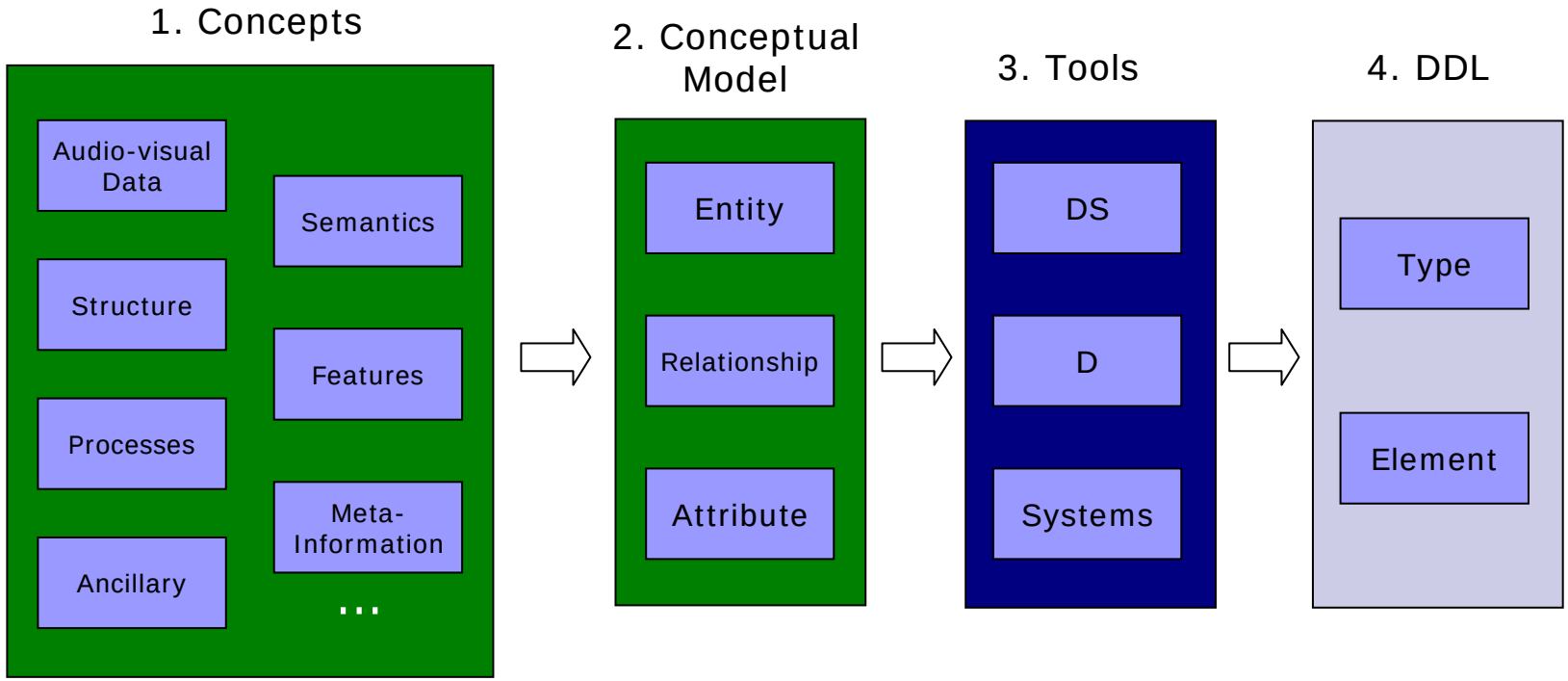
Role of Conceptual Model in MPEG-7

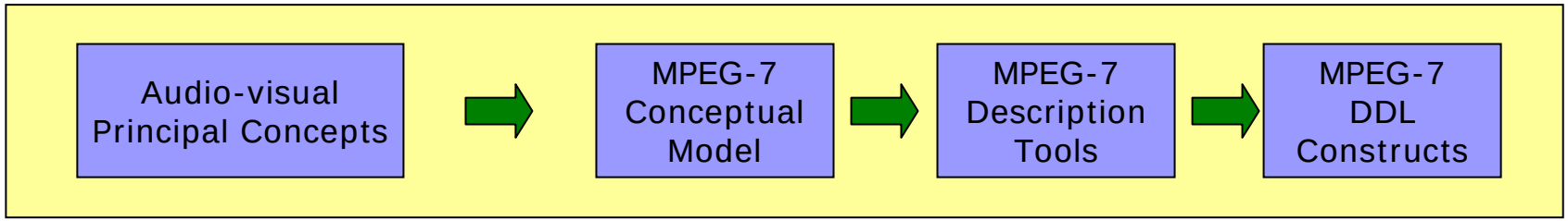


MPEG-7 Conceptual Model

1. Defines concepts relevant to MPEG-7 (*Principal Concepts List*)
2. Provides definitions of *Description Scheme* and *Descriptor* in terms of conceptual modeling constructs (*Entity*, *Relationship*, *Attribute*)
3. Tracks MPEG-7 development by checking concepts against tools
4. Distinguishes MPEG-7 tools (*DS* and *D*) for DDL specification (*types*, *elements*)
5. Ongoing: allows mapping from user concepts (user tools) to MPEG-7 Description tools (*DS* and *D*)

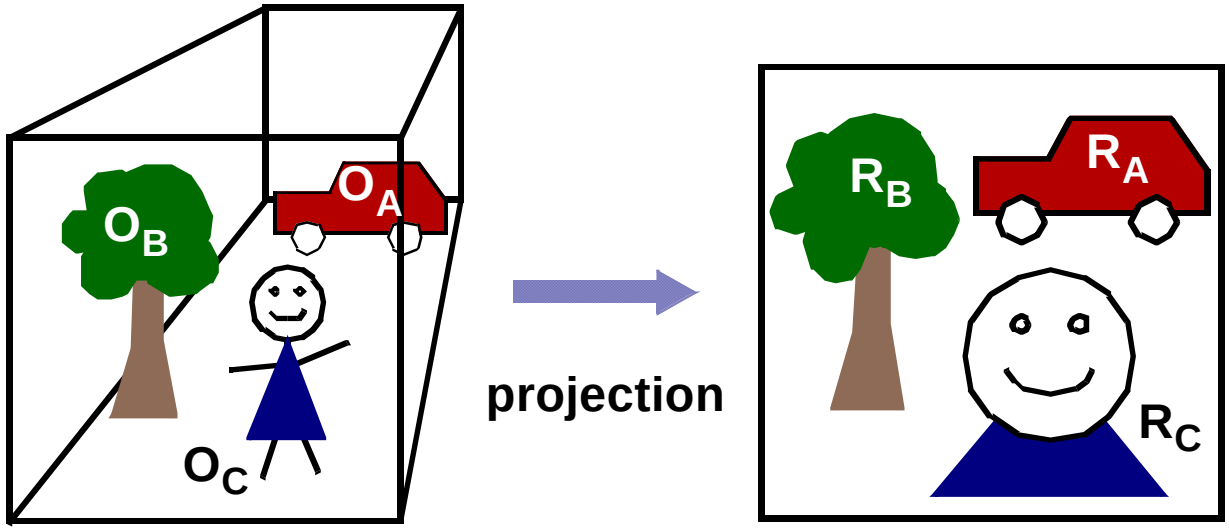
Concept Mapping





Step 1:
Identification of Audio-Visual Concepts

Formation of Multimedia Data from Real World Scenes



Real world scene
(*objects, events, ...*)

Multimedia data
(*regions, segments, ...*)

▪ **Scene Description**: ***Semantics***	▪ **Content Description**: ***Audio-visual data, Structure, Features*** ▪ **Content Management**: ***Processes, Ancillary, Meta-information***

Audio-Visual Principal Concepts

- **Scene Description:**
  - **Semantics**: Action, Character, Conversation, Dialog, Episode, Event, Face, Object, Point of view, Pose, Scene, Script, Story, …
- **Content Description:**
  - **Audio-visual data**: Audio, Edition, Film, Graphic, Image, Mix, Mosaic, Music, Program, Promotion, Rush, Sonification, Sound effect, Speech, Stream, Summary, Symbolic audio, Trademark image, Variation, Version, Video, View, …
  - **Structure**: Animation, Audio spectrum, Background, Composition Effect, Connectivity, Cut, Duration, Edit, Embedded text, Frame, Image type, Internal Transition Effect, Key frame, Linguistic structure, Locator, Region, Segment, Sequence, Shot, Spatial geometry, Spatial relationship, Synchronization, Transition Effect, …
  - **Features**: Audio features, Color, Deformation, Melody, Motion, Noisiness, Shape, Silence, Sketch, Texture, Timbre, Visual features, Volume, …
- **Content Management:**
  - **Processes**: Acquisition, Coding, Creation, Delivery, Editing, Manipulation, Presentation, Production, Publication, Segmentation, Storage, Streaming, Transcoding, …
  - **Ancillary**: Bandwidth, Camera, Client, Instrument, Medium, Modality, Person, Recorder, Terminal, Usage, Usage history, User, User preference, …
  - **Meta Information**: Annotation, Archive, Author, Classification, Context, Contract, Copyright, Date, Financial, Format, IPMP, Language, Locality, Market, Organization, Owner, Place, Quality, Rating, Rights, Technical staff , Text, Title, Translation, Unique identifier, …

Multimedia Concept Properties

Property	Description
Principal Concept	Concept name
Definition	Definition in words
Conceptual Model Construct	{Attribute, Entity, Function, Relationship, Type}
Domain	{Audio, Generic, Video}
Type	{Ancillary, Audio-visual data, Description, Feature, Language, Meta, Model, Process, Semantics, Structure, Syntax/ Semantics}
MPEG-7 Development	{MDS XM, MDS CE, MDS WD, Visual XM, Visual CE, Visual WD, Audio XM, Audio CE, Audio WD, Systems WD, DDL WD}
MPEG-7 Construct (D or DS)	{D, DS}
MPEG-7 DDL Construct	{Element, Attribute, Type}
Description	Description of development as D or DS or Systems
Related principal concepts	Related principal concepts
Related secondary concepts and terms	Related secondary concepts and terms
MPEG-7 Development	{MDS XM, MDS CE, MDS WD, Visual XM, Visual CE, Visual WD, Audio XM, Audio CE, Audio WD, Systems WD, DDL WD}

MPEG-7 Principal Concepts List

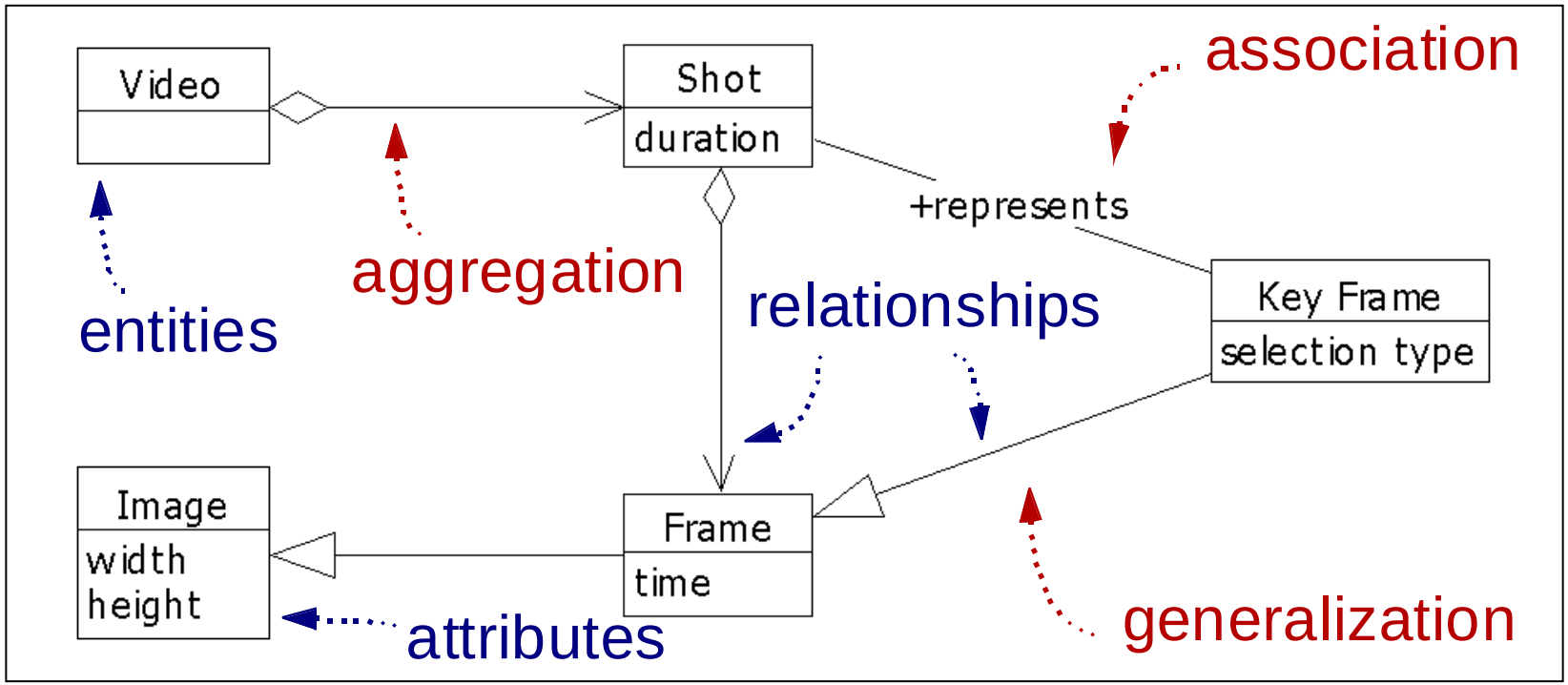
Principal Concept	Definition	Model Construct	Domain	Type	Source	MPEG-7 Development	MPEG-7 Construct (D or DS)	MPEG-7 DDL Construct	Description	Related principal concepts	Related concepts and terms
Abstraction level	The particular level of detail in which data is represented	Attribute	Generic	Audio-visual data	MPEG-7 Requirements	MDS XM	DS	Element	**Multiresolution Pyramid DS (XM):** specifies a hierarchy of views of data. **Summarization DS (WD):** is used to specify a set of summaries to enable rapid browsing, navigation, visualization and sonification of AV content. Each summary is an audio-visual abstract of the content.	Hierarchy, scalability	
Acquisition	The process of acquiring audio or visual data from a source	Function	Generic	Process	MPEG-7 Applications	MDS WD	DS	Element	**CreationMaterial (WD):** describes the devices and instruments used for the creation of the content (e.g., types of device, lens, films, instruments, settings, etc.).	Instrument, camera, editing	Shooting, recording, source, filming, take
Action	A semantically identifiable behavior of an object or group of objects, e.g., soccer player kicking ball	Attribute	Visual	Semantics	MPEG-7 Description Schemes (V0.8)	MDS CE	DS	Element	**Annotation DS (WD):** contains the description tools (Ds and DSs) intended for a simple and structured description of persons, objects, events, etc; **Semantic Relationships (XM)**: predicative semantic attributes refer to actions (events) or states among two or more elements. Examples of action relations are “To throw” and “To hit”.	Motion, object, event, animation	Interaction, action unit, interaction unit
Aggregation	Grouping of items such as objects, regions or audio-visual data	Relationship	Generic	Audio-visual data	MPEG-7 Description Schemes (V0.8)	MDS WD	DS	Element	**Cluster DS (WD):** describes the arbitrary grouping of audio-visual data items or syntactic elements.	Cluster	Set, collection, examples

Step 2:
Building Conceptual Model

Conceptual Modeling Constructs

- **Entity** – a principal object in the audio-visual domain
- **Relationship** – an association among one or more entities as follows:
  - **Generalization** - specifies an “is-A” relationship that partitions a class of entities into mutually exclusive subclasses
  - **Aggregation** - specifies a “has-A” assembly-component relationship of entities
  - **Association** - relates two or more entities which do not exhibit existence dependency
- **Attribute -** descriptive information about an entity or relationship which can be used for identification or description

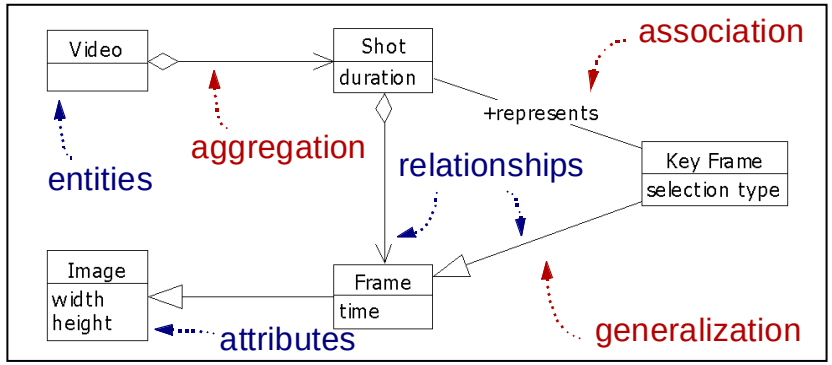
Example Audio-Visual Conceptual Model



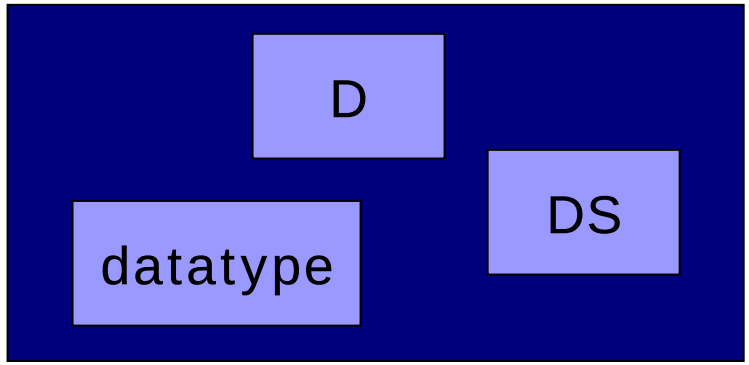
Step 3:
Mapping CM to MPEG-7 Description Tools

Mapping to MPEG-7 Description Tools (DS, D)

Conceptual Model



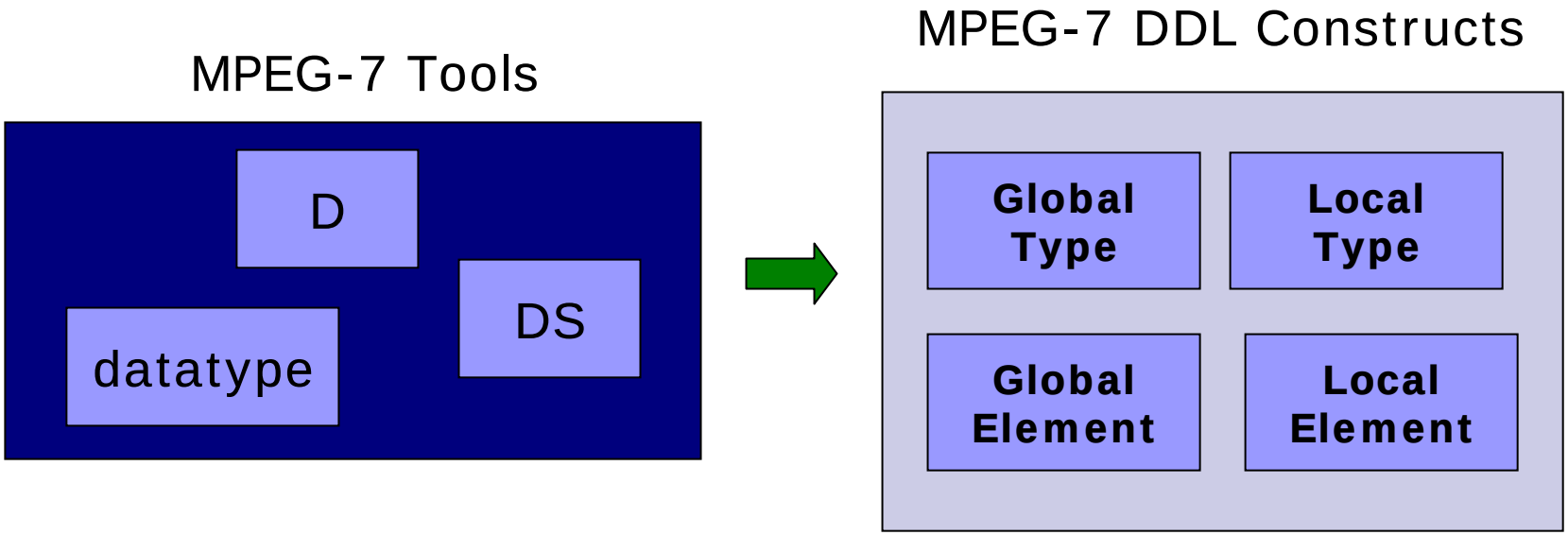
MPEG-7 Tools



Conceptual Model	Audio-visual domain	
	Yes	No
Entity	DS	None
Relationship	DS	None
Attribute	D	Datatype

Step 4:
Mapping Tools to MPEG-7 DDL Constructs

DDL Implementation

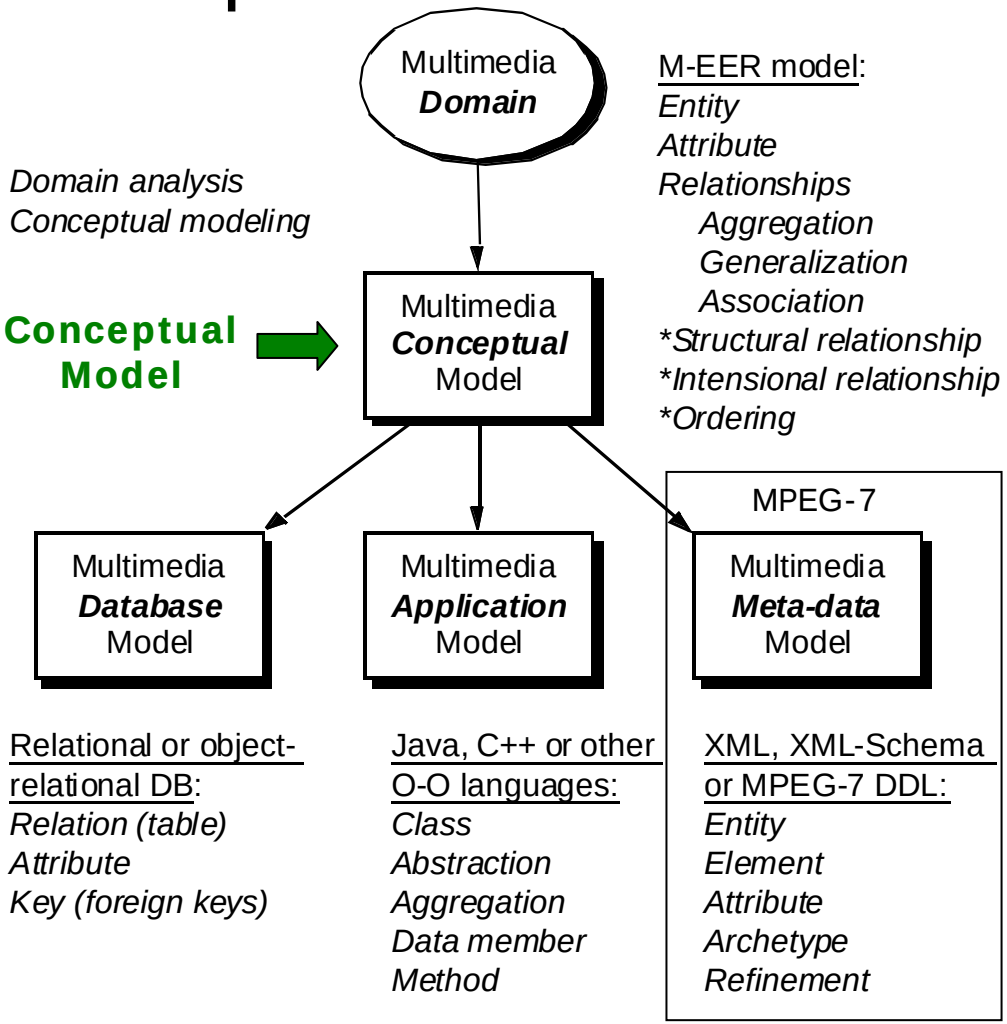


MPEG-7 Tool	MPEG-7 DDL
DS (← Entity)	Global Type (ID), Local Element (REF)
DS (← Relationship)	Global Type (ID), Local Element
D (← Attribute)	Global Type, Local Element
datatype	Global Type, Local Type, Local Element

Conceptual Model: Ongoing Uses

- Mapping for MPEG-7 database design (i.e., relational), software design (i.e., Java / C++ classes)
- User organization of MPEG-7 Tools (I.e., Packages) – user-oriented concepts and description tasks
  - **User Concept**: → Vacation (scene)
  - **Scene Concepts**: → Place, Time, Event, Persons, Objects
  - **Audio-Visual Concepts**: → Image, Video, Audio
  - **MPEG-7 Tools**: → Event DS, PersonObject DS, Agent DS, Image DS, StillRegion DS, Segment Decomposition DS, ColorLayout D, Homogeneous Texture D, Shape D, …

Conceptual Modeling for Multimedia Content Description



- **Modeling**: objective of creating single unified model (conceptual, logical, implementation)
- **Concepts**: multimedia domain analysis
- **Transformation**: of conceptual model into the logical models for
  - Database design
  - Software classes for application implementation
  - Meta-data language for interoperable annotation

MPEG-7 Reference Software

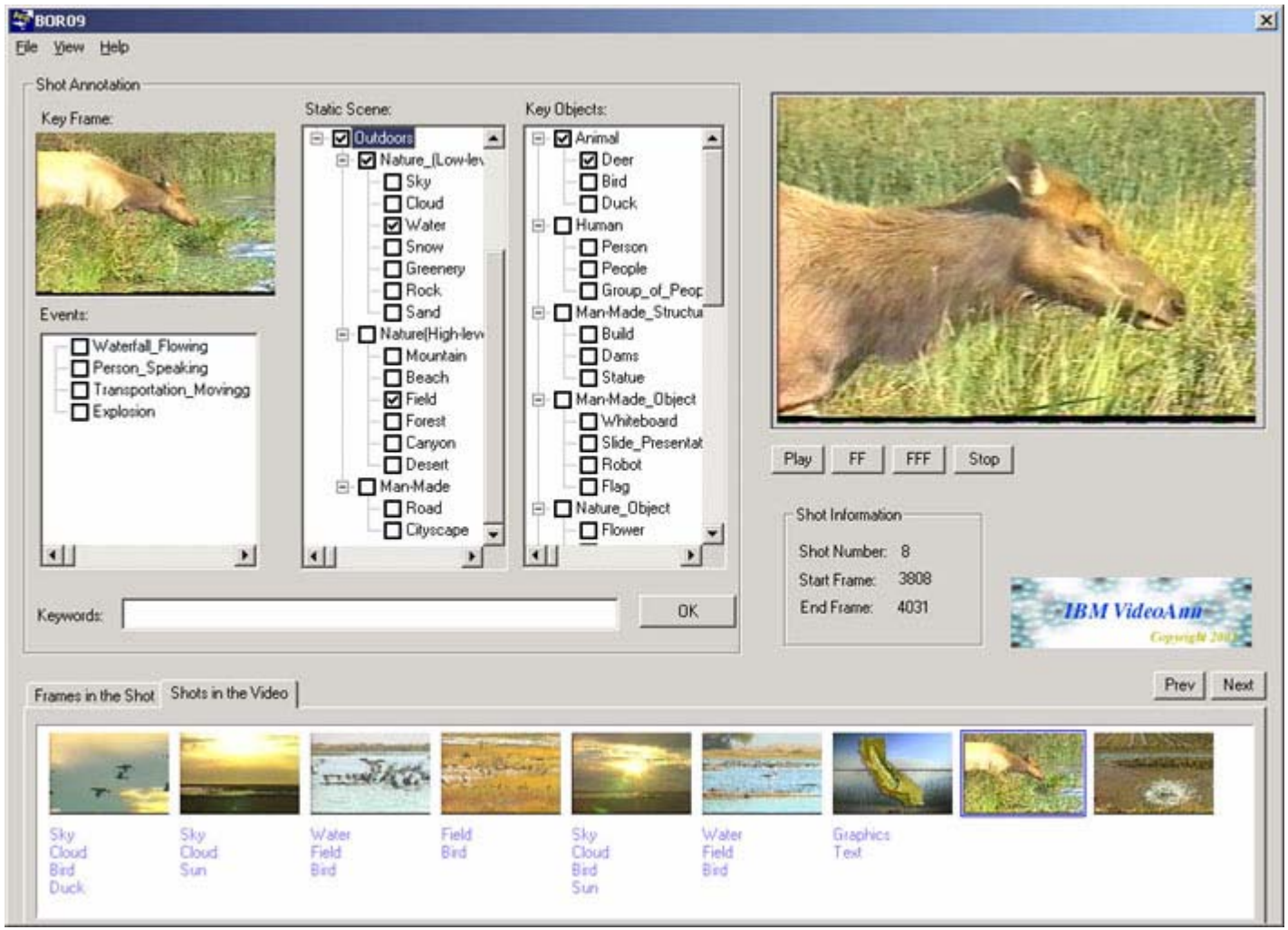
- Reference implementation of relevant MPEG-7 Standard
  - Experimentation software (XM)
- Creation of of D and DSs bitstreams with normative syntax rather than tool performance
- Four categories of components
  - DDL parses and DDL validation parser
  - Visual Descriptors
  - Audio Descriptors
  - Multimedia Description Schemes (MDS)

MPEG-7 Conformance

Guidelines and procedures for testing implementations for conformance

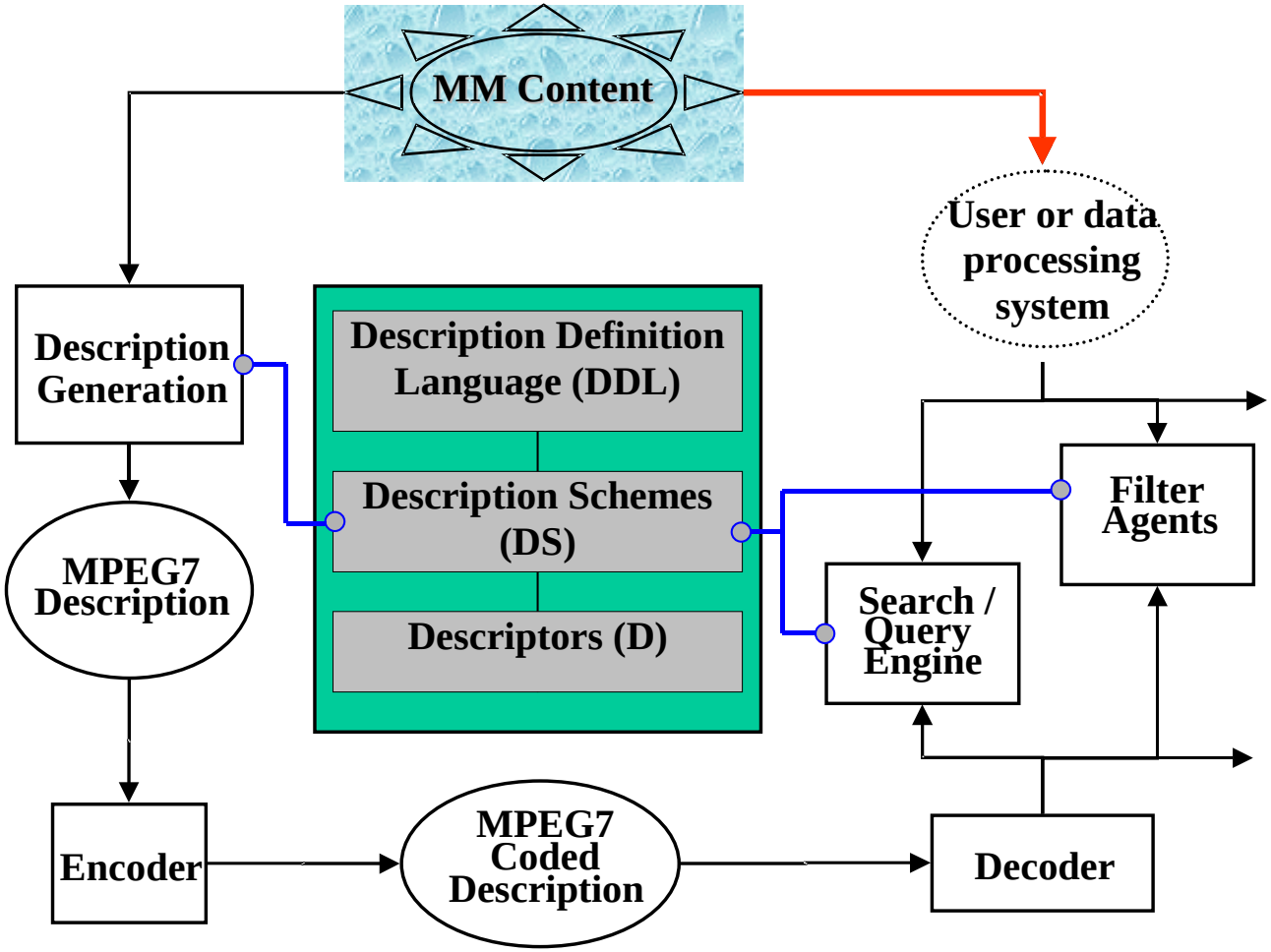
MPEG-7 applications

IBM VideoAnnex

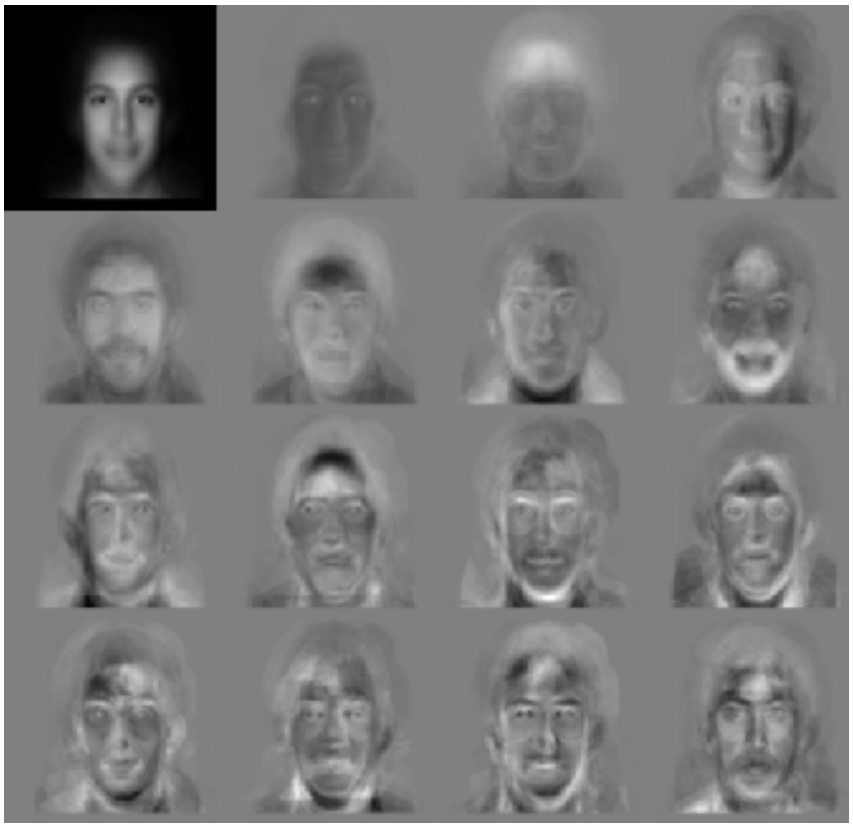


Possible MPEG-7 Applications

Abstract Representation

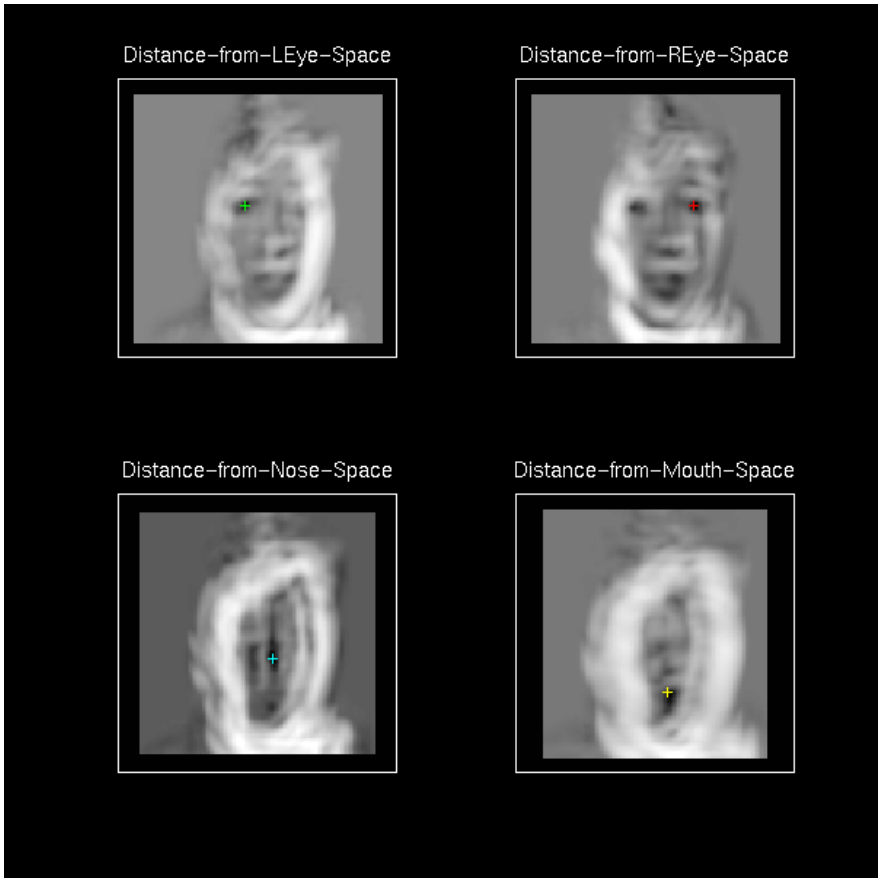


Standard Eigenfaces



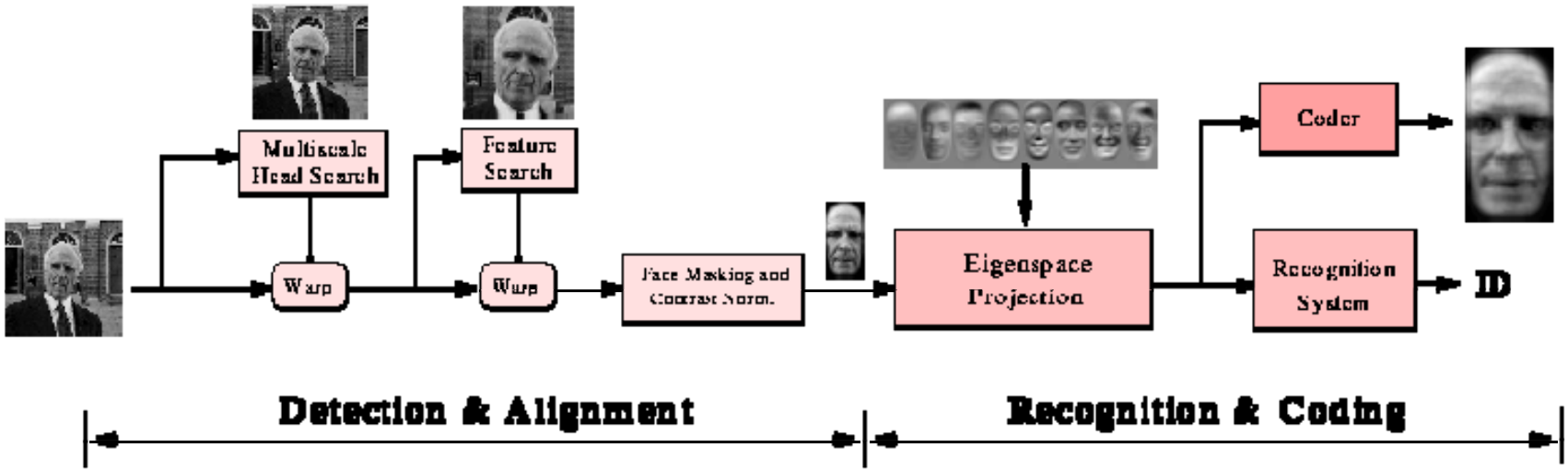
The eigenfaces for this database were approximated using a principal components analysis on a representative sample of 128 faces. Recognition and matching was subsequently performed using the first 20 eigenvectors. In addition, each image was then annotated (by hand) as to sex, race, approximate age, facial expression, and other salient features. Almost every person has at least two images in the database; several people have many images with varying expressions, headwear, facial hair, etc.

http://whitechapel.media.mit.edu/vismod/demos/facerec/basic.html


Distance-from-LEye-Space
Distance-from-REye-Space
Distance-from-Nose-Space
Distance-from-Mouth-Space

Face Recognition

http://whitechapel.media.mit.edu/vismod/demos/facerec/system.html

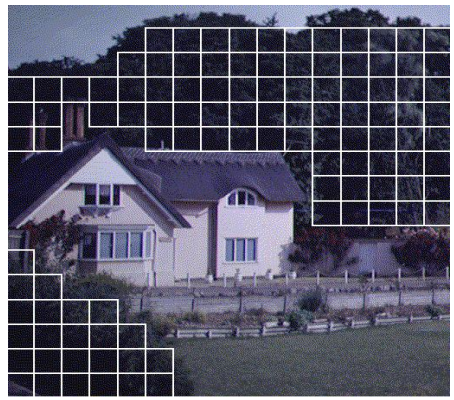


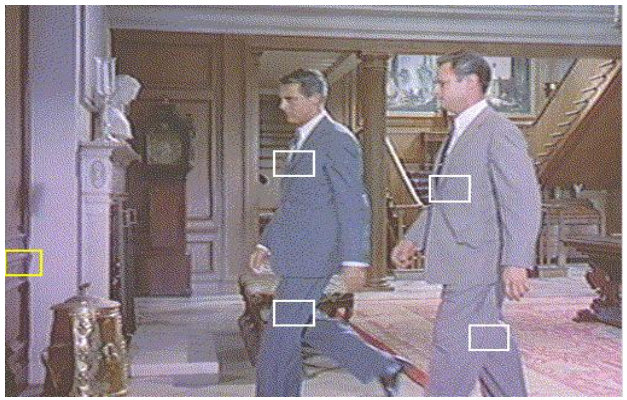
The system diagram above shows a fully automatic system for detection, recognition and model-based coding of faces for potential applications such as video telephony, database image compression, and automatic face recognition. The system consists of a two-stage object detection and alignment stage, a contrast normalization stage, and a Karhunen-Loeve (eigenspace) based feature extraction stage whose output is used for both recognition and coding. This leads to a compact representation of the face that can be used for both recognition as well as image compression.
Good-quality facial images are automatically generated using approximately 100-bytes worth of encoded data. The system has been successfully tested on a database of nearly 2000 facial photographs from the ARPA FERET database with a detection rate of 97%. Recognition rates as high as 99% have been obtained on a subset of the FERET database consisting of 2 frontal views of 155 individuals.

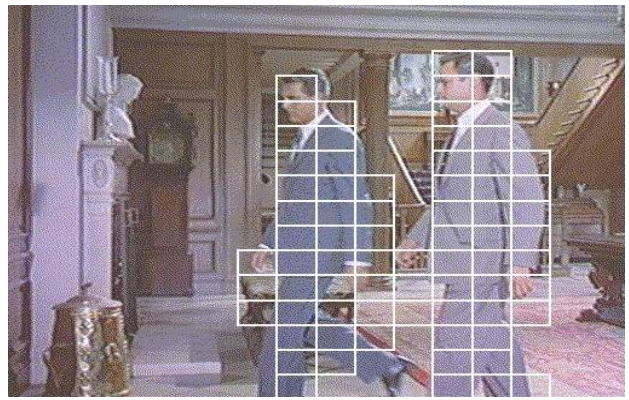
Photobook

http://wasi.www.media.mit.edu/people/tpminka/photobook/

- Tool for performing queries on image databases based on image content.
- Works by comparing features associated with images, not the images themselves.
- Features are parameter values of particular models fitted to each image.

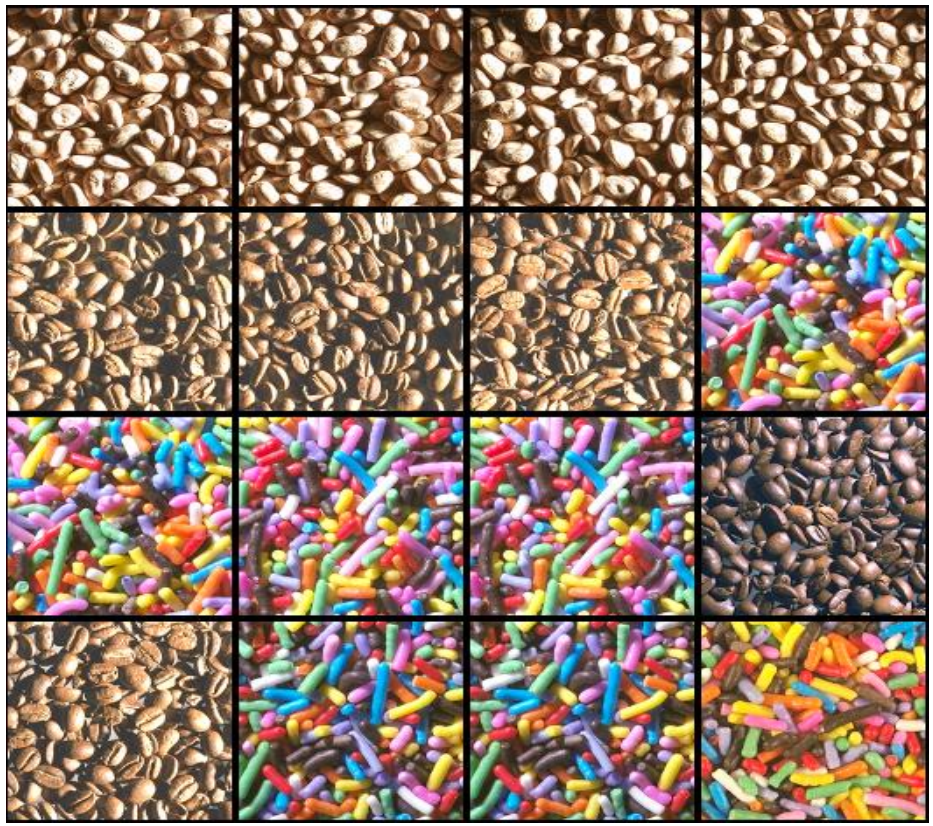


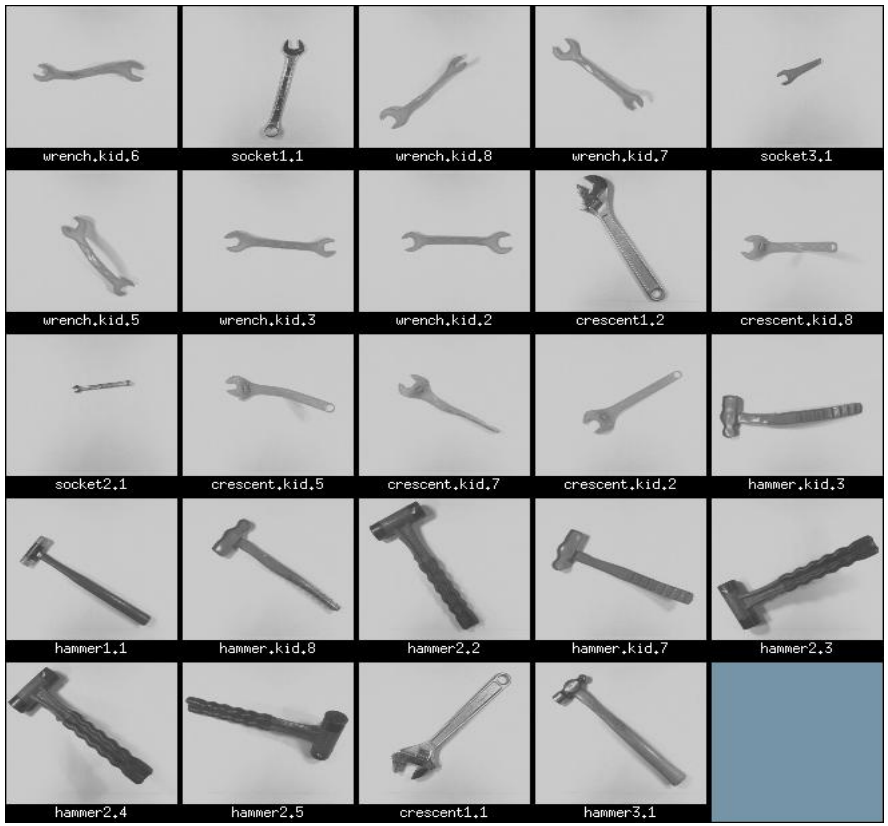




http://whitechapel.media.mit.edu/people/tpminka/photobook/foureyes/seg.html

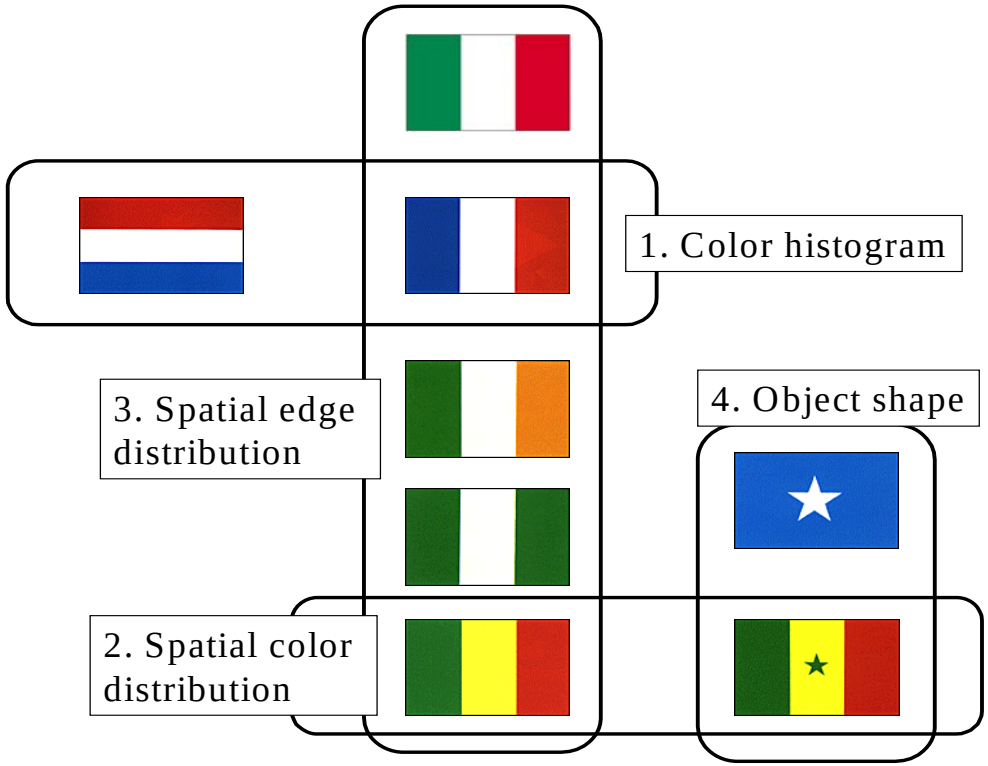
Texture Modeling



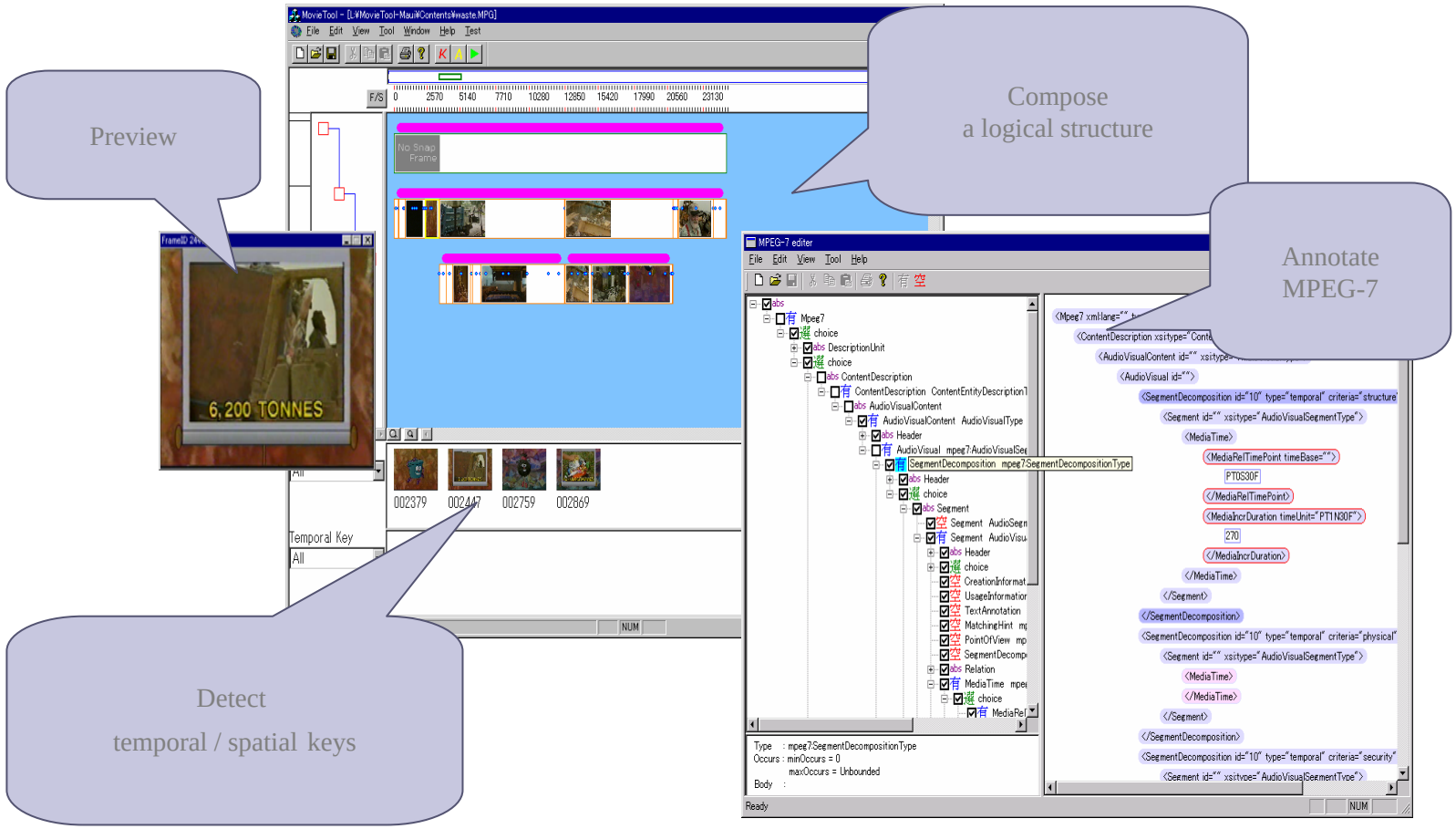


This is an example of a Photobook search based on shape. The query image is in the upper left; the images in a small tools database are displayed in raster scan order of similarity

Content Retrieval using Image as the Query



Movie Tool


Preview
Compose
a logical structure
Annotate
MPEG-7
Detect
temporal / spatial keys
MovieTool - [L:¥MovieTool-Maui¥Contents¥waste.MPG]
File Edit View Tool Window Help Test
6,200 TONNES
002379
002447
002759
002869
Temporal Key
All
MPEG-7 editer
File Edit View Tool Help
Ready
NUM

References and credits

References

- ICCE 2001 MPEG-7 Tutorial Session, 6/17/2001, Smith, Manjunath, Day
- IEEE Transactions on Circuit and Systems for Video Technology, Vol. 11, No. 6, Special Issue on MPEG-7
- IEEE Multimedia April-June 2002, July-September 2002
- Chiariglione, Leonardo: 'MPEG: achievements and current work'
- Chiariglione, Leonardo: 'The MPEG generation, new information-age elements'
  - http://www.chiariglione.org/mpeg/
- http://www.mpegif.org/resources.php
- http://mpeg.tilab.com
- http://www.darmstadt.gmd.de/mobile/MPEG7/

References

- T. Sikora, “The MPEG-7 Visual Standard for Content Description – An Overview”, IEEE Trans. Circuits Syst. Video Technol., vol. 11, pp. 696-702, June 2001
- S.-F. Chang, T.Sikora, and A. Puri, “Overview of MPEG-7 Standard”, IEEE Trans. Circuits Syst. Video Technol., vol. 11, pp. 688-695, June 2001
- J. M. Martinez, "Overview of the MPEG-7 Standard", ISO/IEC JTC1/SC29/WG1, 2001
- B.S. Manjunath, J.-R. Ohm, V.V. Vasudevan, and A. Yamada, “MPEG-7 Color and Texture Descriptors”, IEEE Trans. Circuits Syst. Video Technol., vol. 11, pp. 703-715, June 2001

References

- M. Bober, “MPEG-7 Visual Shape Descriptors”, IEEE Trans. Circuits Syst. Video Technol., vol. 11, pp. 716-719, June 2001
- A. Divakaran, “An Overview of MPEG-7 Motion Descriptors and Their Applications”, 9th Int. Conf. on Computer Analysis of Images and Patterns , CAIP 2001 Warsaw, Poland, 2001, Lecture Notes in Computer Science vol.2124, pp. 29-40
- J. Hunter, "An overview of the MPEG-7 description definition language (DDL)", IEEE Trans. Circuits Syst. Video Technol., vol. 11, pp. 765-772, June 2001
- P.Salembier and J.R. Smith, “MPEG-7 Multimedia Description Schemes”, IEEE Trans. Circuits Syst. Video Technol., vol. 11, pp. 748-759, June 2001

References

- F. Mokhtarian, S. Abbasi, and J. Kittler, “Robust and Efficient Shape Indexing through Curvature Scale Space”, Proc. International Workshop on Image DataBases and MultiMedia Search, pp. 35-42, Amsterdam, The Netherlands, 1996
- CSS Demo, http://www.ee.surrey.ac.uk/Research/VSSP/imagedb/demo.html
- Gabor Function, http://disney.ctr.columbia.edu/jrsthesis/node43.html
- Radon Transform, http://eivind.imm.dtu.dk/staff/ptoft/Radon/Radon.html

Credits

- Deryck Yeung
- Prof. Ze-Nian Li
- John R. Smith
- Dwight Borses
- Philippe Salembier
- Prof. Harald Kosch