浙江大学计算机学院
数字媒体与网络技术

Digital Asset Management
数字媒体资源管理

任课老师：张宏鑫
2014-11-17

教材

- 《数字媒体资源管理导论》
  - 浙江大学出版社（正在撰写中）
  - 在线提供重要章节的PDF
- 《数字媒体资产管理系统》
  - 中国传媒大学出版社
  - 安德烈斯.毛特 皮特.托马斯（德）
  - 宋培义 严威 译

Digital Asset Management
数字媒体资源管理

1. Introduction

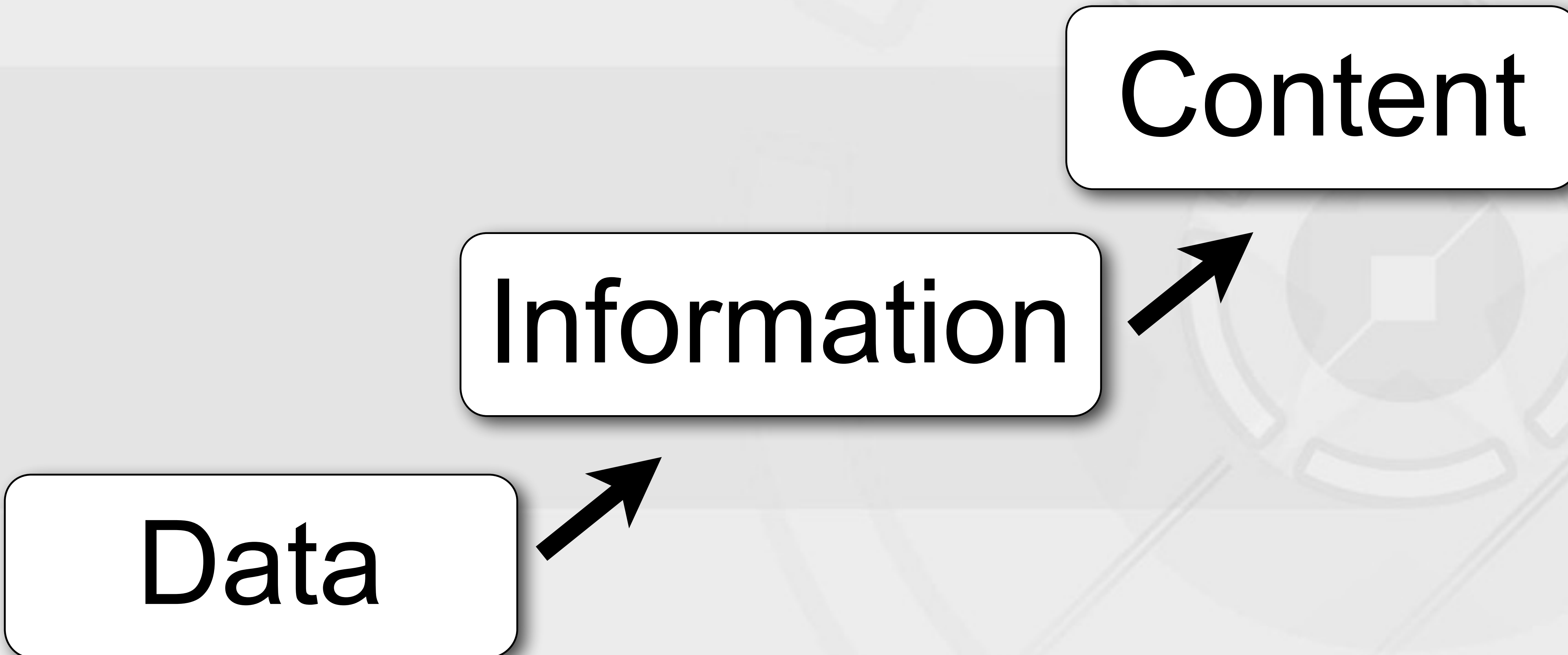
Outline

- Content management
- Industrial Analysis
- Case Study

1.1. Content management

Content ?

Content



Content Management

- Information
  - creation, representation and exchanging
- Information media (信息载体，石刻，竹简，羊皮纸，雕塑，建筑 ...)
  - collection, organization and storage
- 古老的行业

Content Management

- Examples:
  - Ancient years： Literature in Libraries and Archives (档案馆)
  - From19th century: Continuous Media (连续媒体), movie, audio ...
  - After 1980's: Digital Media (数字媒体), digitalized ~

Content Management

- process, store and transfer (data) content
- Key: non-linear creation

- Media industry:

  fusion between traditional company (news paper, broad casting, entertainment) and modern company (google, sina, apple, facebook)

- Non-Media industry:

  data and documents in big companies, education units, research units, museums

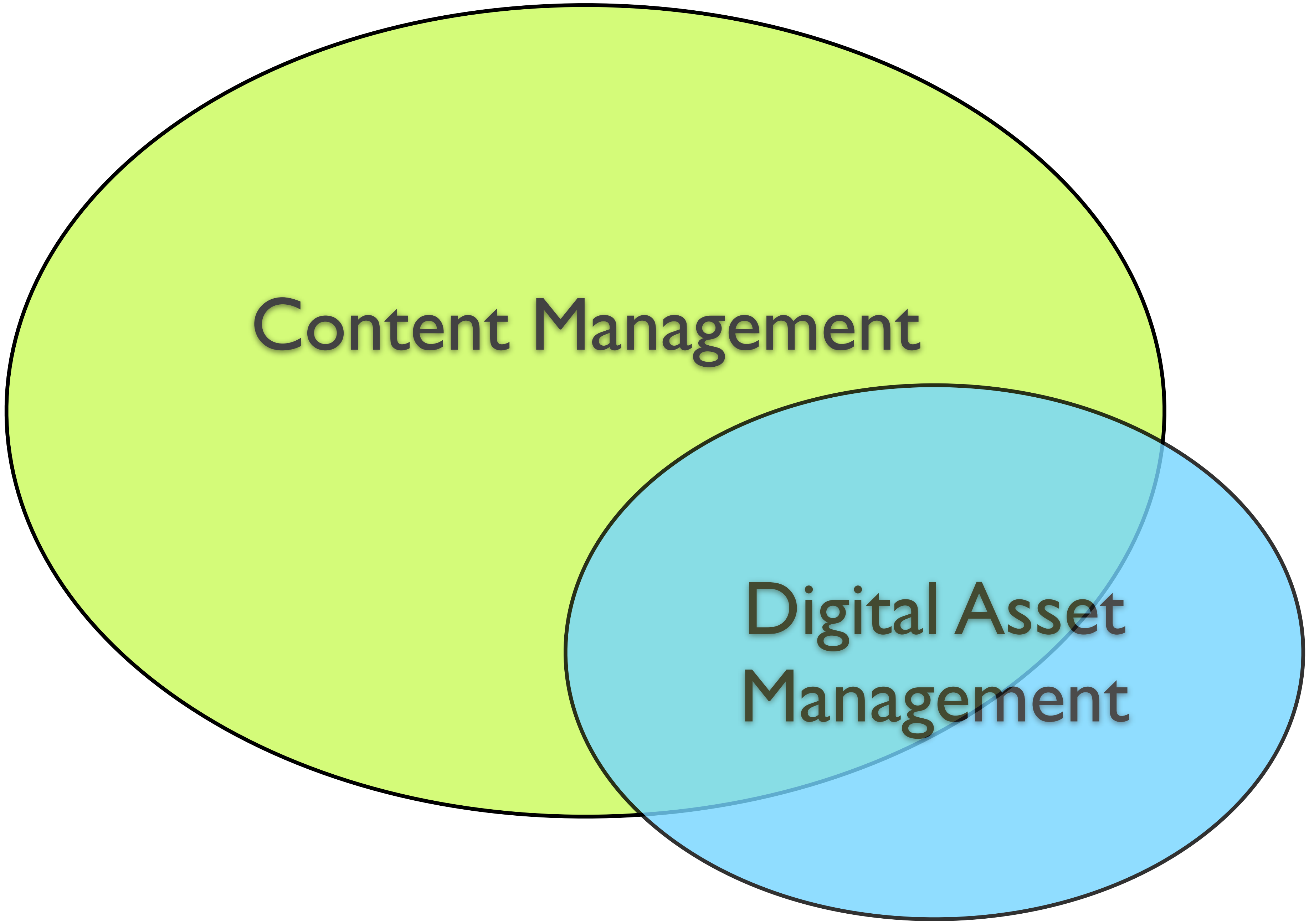
信息的银行?

- 保险柜?
- 交易平台?
- 。。。?

Typical CMS

- Website of a research unit
- Personal blog
- Wiki

Digital Asset Management



What is Digital Asset Management?

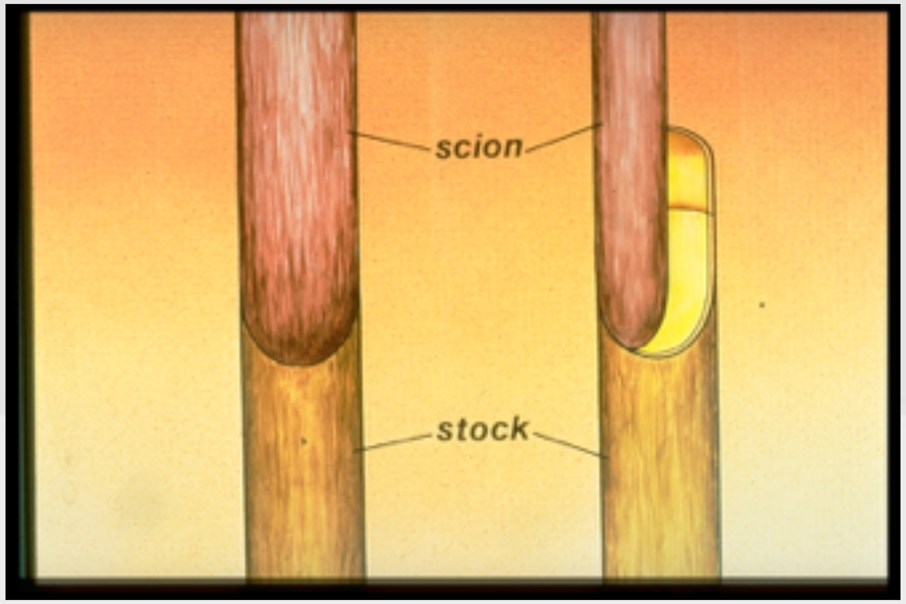
- Tools for organizing, storing and retrieving content in digital format
  - downloading, renaming, backing up, rating, grouping, archiving, optimizing, maintaining, thinning, and exporting ...
- Includes:
  - text, video, images, movies, sound, and 3D content

Content Management and DAM

- **“It’s just another binary file type”** is a superficial response
  - But so is, **“It’s just managing brand assets”**
  - Digital Asset Management involves
    - Much higher **storage volumes**
    - More complex **ownership** and **usage rights**
    - More **complex content** (layers)
  - However, an organization needs a **unified content management/digital asset management strategy** to avoid unnecessary costs in hardware, licensing, software development and support

Examples of Digital Media (Asset)

Illustrations



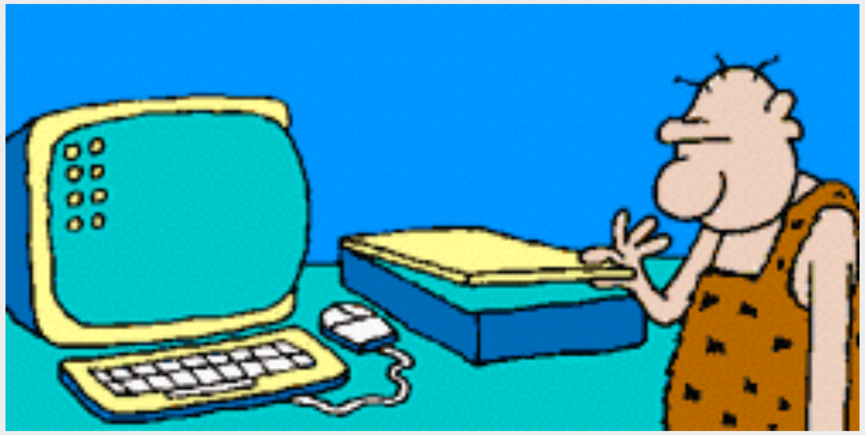
Photographs

More Digital Media (Asset)

Sound

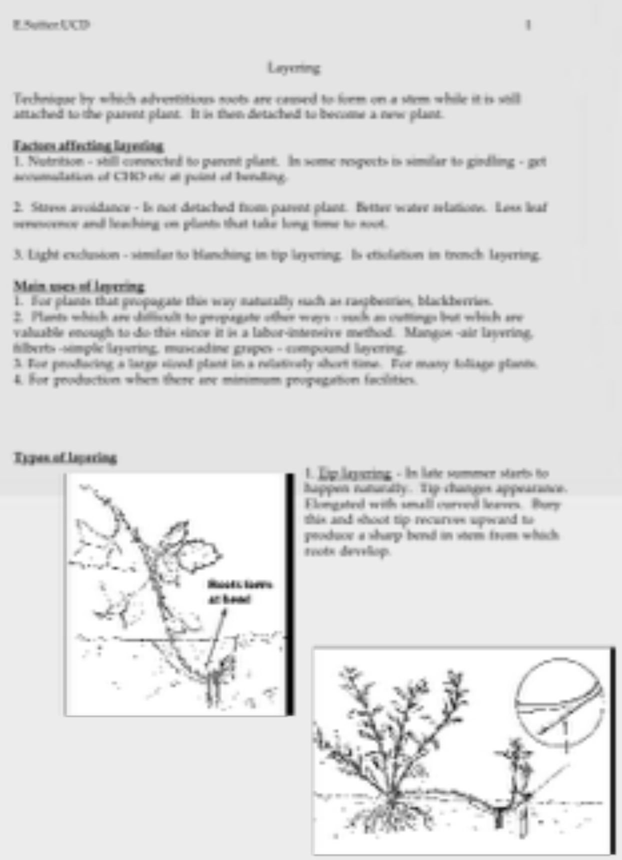
Movies

Animation



Document

Hypermedia document



E.Sutter/UCD 1

Layering

Technique by which adventitious roots are caused to form on a stem while it is still attached to the parent plant. It is then detached to become a new plant.

Factors affecting layering
1. Nutrition - still connected to parent plant. In some respects is similar to girdling - get accumulation of CHO etc at point of bending.

2. Stress avoidance - Is not detached from parent plant. Better water relations. Less leaf senescence and leaching on plants that take long time to root.

3. Light exclusion - similar to blanching in tip layering. Is etiolation in trench layering.

Main uses of layering
1. For plants that propagate this way naturally such as raspberries, blackberries.
2. Plants which are difficult to propagate other ways - such as cuttings but which are valuable enough to do this since it is a labor-intensive method. Mangos -air layering, filberts -simple layering, muscadine grapes - compound layering.
3. For producing a large sized plant in a relatively short time. For many foliage plants.
4. For production when there are minimum propagation facilities.

Types of layering

1. Tip layering - In late summer starts to happen naturally. Tip changes appearance. Elongated with small curved leaves. Bury this and shoot tip recurves upward to produce a sharp bend in stem from which roots develop.

Text

Nutrient Media

Nutrient Media

Nutrient media for plant tissue culture are designed to enable explants to grow in a totally artificial environment. In order to enable plants to grow in vitro, scientists have devised nutrient media that provide the nutrients usually available in soil. In addition to mineral elements which make up the macro- and micronutrients present in fertilizers, nutrient media also contain organic compounds such as vitamins, plant growth regulators, and a carbon source.

Mineral elements

One of the most successful media, devised by Murashige and Skoog (Murashige and Skoog, 1962) was formulated by analyzing the inorganic components in tobacco plants and then adding them to media, in amounts similar to those found in the plants. Not only did they find that the ions themselves were important, but the form in which the ions were supplied were critical as well.

Macronutrients

Macroelements consist of N, K, P, Ca, Mg, and S.

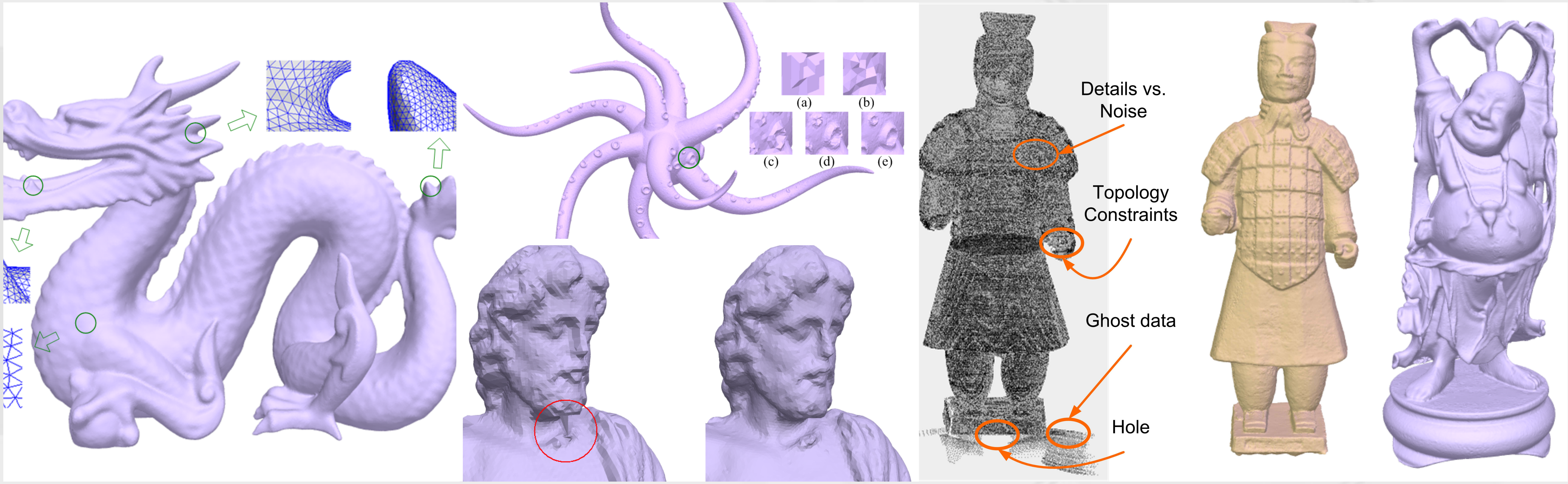
Nitrogen (N) - Nitrogen is required for general growth and is essential to plant life. Most inorganic nitrogen is converted to amino acids and then to proteins. The two most widely used forms of inorganic nitrogen used in plant nutrient media are the nitrate ion (NO3- oxidized) and the ammonium ion (NH4+ reduced) which are added as inorganic salts. Nitrate is usually added at concentrations between 25 and 40 μM and ammonium between 2 and 20 μM. In poorly buffered media, use of both forms helps maintain pH. Many plants appear to grow best if given both forms, although the reason for this is not known. In devising media, both the total amount of nitrogen as well as the relative amounts of NO3- and NH4+ are important. When the ammonium ion is used alone it may be toxic.. Inorganic nitrogen generally ranges from 25-60 mM in nutrient media. Nitrogen may also be added in an organic form as amino acids, hydrolysates (such as casein hydrolysate) and organic acids.

The organic forms of nitrogen such as amino acids are often useful when added to media that do not contain ammonium. One advantage of using organic nitrogen is that it is already reduced, the form in which most nitrogen exists in the plant and thus may be taken up more readily than inorganic nitrogen. Organic forms of nitrogen cannot, however, totally replace inorganic forms. One danger of using amino acids is that here TOO MUCH can be added in which case feedback inhibition can occur. Biochemically the cells sense that there is a great deal of a specific amino acid and consequently change the metabolic pathways to stop the natural production of the amino acid. This results in the production (or backing up) or intermediate compounds which in turn may disrupt normal metabolism.

The form of nitrogen is often critical depending on the kind of culture. There is a difference in the oxidized and reduced forms. The two main forms of nitrogen used are ammonium NH_4^+ and nitrate NO_3^-. The form of nitrogen affects the pH. When both forms of N are used there is a rapid uptake of ammonium (the more readily available form since it is reduced) which results in a decrease in pH to about 4. 4. At lower pH the uptake of nitrate is preferred and thus the pH rises. Nitrate is used in addition to ammonium because the ammonium ion in excess is usually toxic. Also pH would be much more difficult to control with just ammonium.

Plant Biology 155: Plant Tissue Culture 23

3D content



3D printer / scanner ...

Kinect...

Content

- Essence (素材) + Metadata （元数据）

- Intellectual Property Rights (IPR, 知识产权)
- Digital Right Management (DRM, 数字版权保护)

Why Do We Need DAM?

- Average creative person looks for a media file 83 times per week
- Fails to find it 35% of the time
- DAM reduces failure to 5%

Digital assets are not simple bits.

What Can DAM Do for You ?

- Catalog large numbers of formats
- Create a visual category using thumbnails
- Add keywords, data fields
- All fields can be searched
- Select images for an electronic gallery - specific lecture topics
- Share over the internet

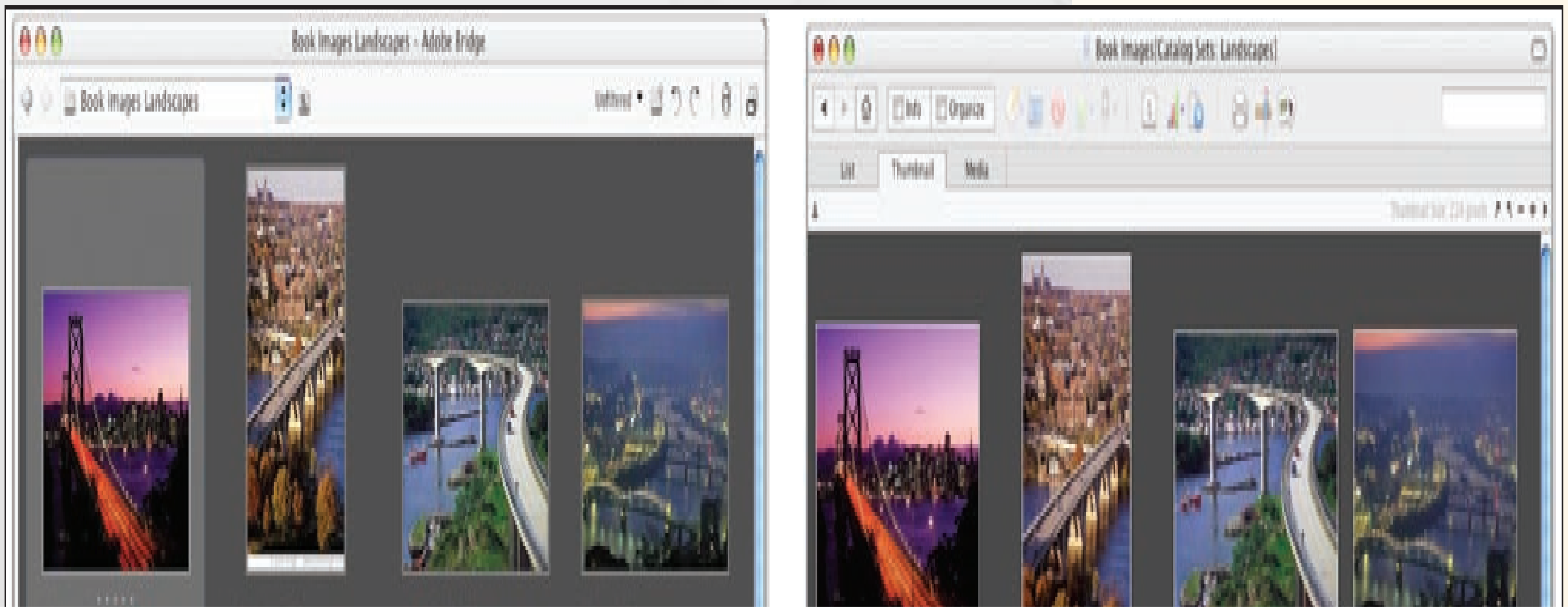
DAM Example: Picasa

Photo Management: Client Software + Web Service

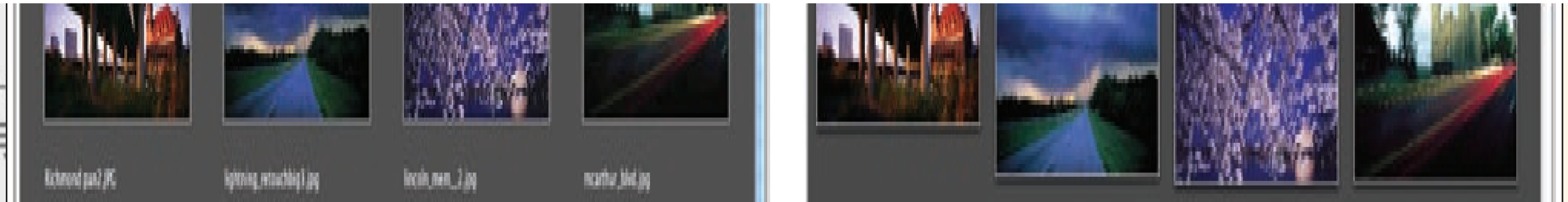
Rules of sound DAM

- Systematize
- Don’t rely on your memory
- Be comprehensive
- Build for the future
- Do it once...
- But don’t overdo it

Browsers v.s. cataloging



- DAM faster
- allows user to have virtual sets.
- knows where stuff is supposed to be.
- allows faster backup of important sorting work.
- allows you to work with offline images.



Browsers v.s. cataloging

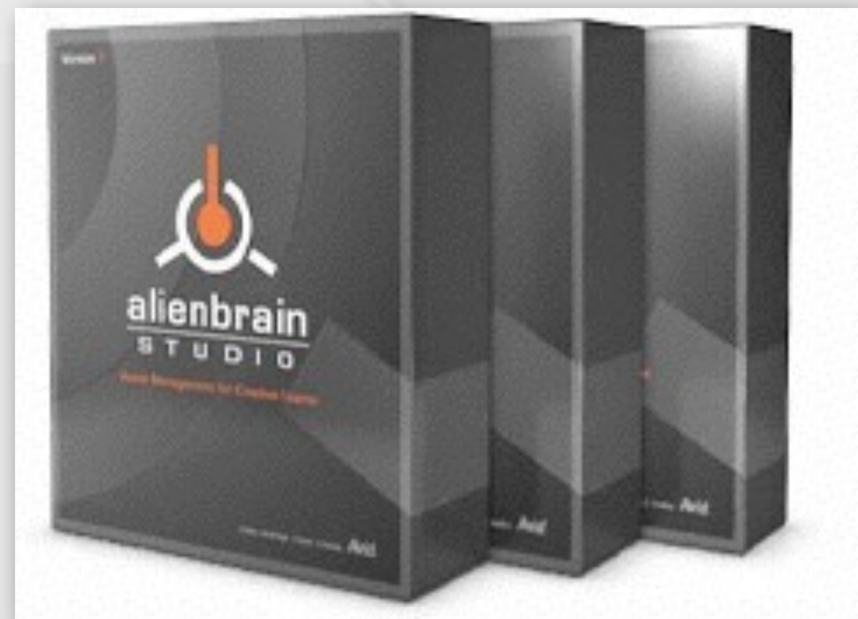
- Browsers:
  - Photoshop Bridge
- Cataloging software
  - Google Picasa
  - ACDSee

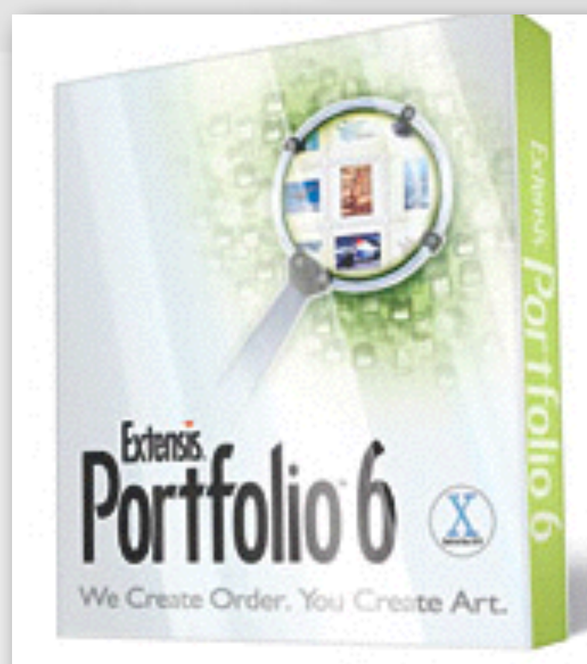
Solutions

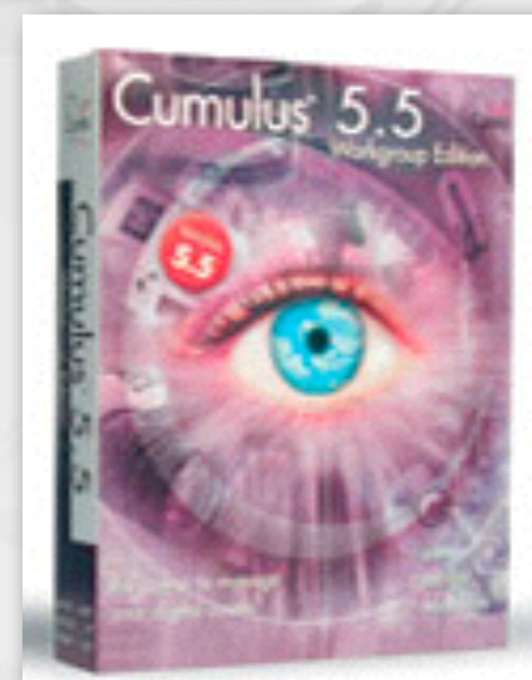
From most extensive and expensive to least financially damaging

- Enterprise solutions
  - $35,000 + (can be in millions)
- Middle tier - interdepartmental
  - $3,000 - $5,000 +
- Desktop level
  - $100-500 + (depending on server requirements)
- Future, SaaS (cloud) solution
  - free or very low price

Desktop Solutions







Desktop Solutions

- Avid Technology - Alienbrain
- Extensis - Portfolio
- Canto - Cumulus

Each of these programs is easy to use. Demonstration copies are available on the web at
www.alienbrain.com (Alienbrain)
www.extensis.com (Portfolio)
www.canto.com (Canto)

1.2. Industrial Analysis

DAM: Past and Present

- Digital Asset Management initially established Niche Markets, including
  - Publishing, Media and Entertainment
  - Broadcasting – Media Asset Management
  - etc.

DAM: Past and Present

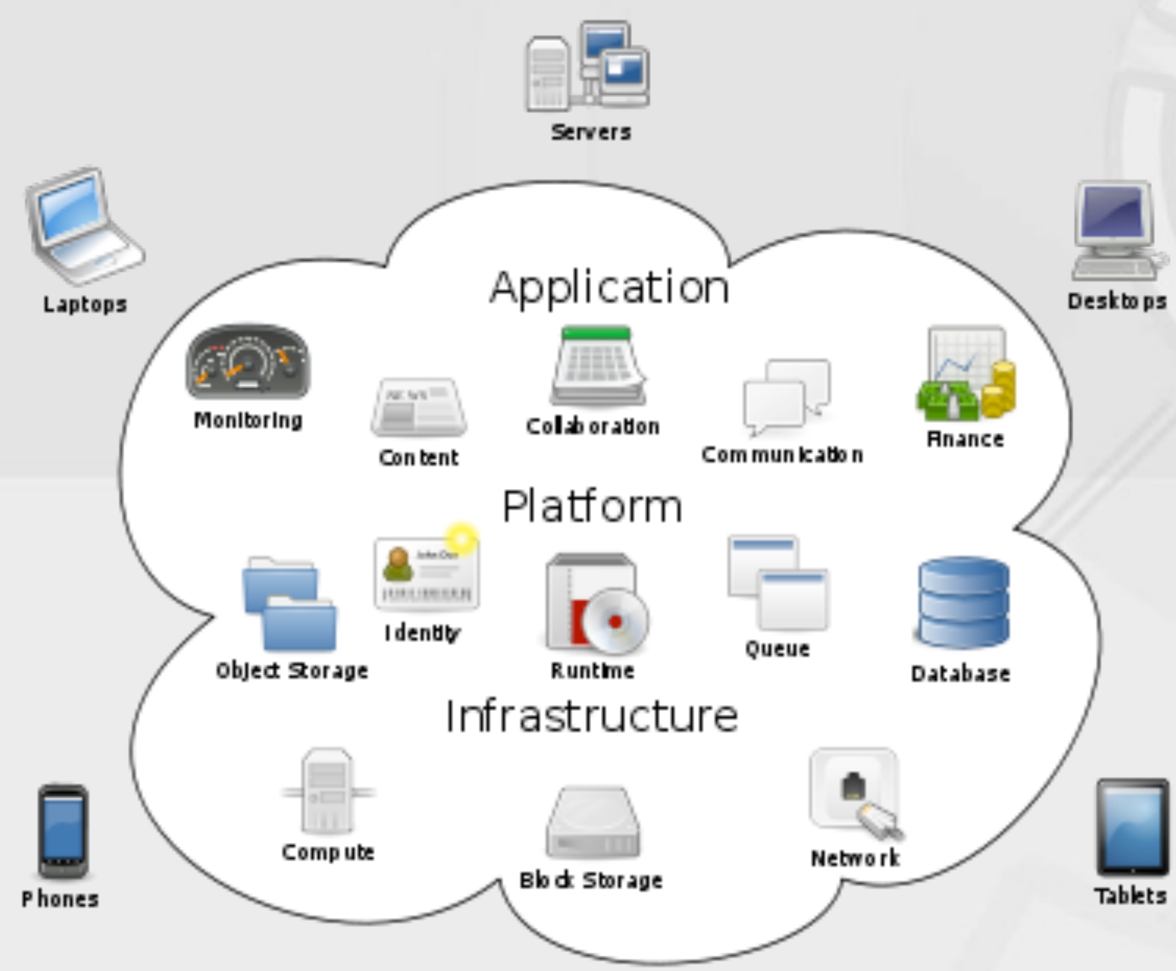
- Now on the Verge of Going Mainstream
  - Integration into
    - Enterprise Content Management and
    - Document Management Strategies
  - Cross Industry
    - Financial Services, Pharmaceuticals, Consumer Packaged Goods, etc.
  - Mainstream Vendors

DAM: Past and Present

- 淘宝电子书 http://ebook.taobao.com/

DAM: Past and Present

DAM system is moving to



Cloud Computing

浙江大学计算机学院
数字媒体与网络技术

浙江大学计算机学院
数字媒体与网络技术

Digital Asset Management – Case Study

Case Study #1: Music Publishing

- Apple (iTunes)
- Leading music publishing firm
  - own millions song copyrights and supports 100 countries and territories
- Client needed a means to further maximize and manage the value of the song copyrights that it owns through promotion, licensing and royalty processing
- Client decided to turn all their internal processes and data outward, making them available to business partners and associates everywhere, at all time

Case Study #1: Music Publishing

- Key technical aspect was integration of numerous IT systems including several territorial:
  - databases, search, application server/portal
  - not just simply a packaged DAM system deployment

- Outcome was the world's largest digital rights management (DRM) system

Case Study #1: Music Publishing

- Apple's iTunes (data 2011)
  - > 8,500,000,000 music sale
  - > 84,000,000 iPad
  - > 13,000,000 iPhone
  - > 350,000,000 iPod
  - > 400,000,000 iOS devices
  - > 435,000,000 iTunes users

Case Study #2: Cable Television

- Leading cable television network: multiple premium channels/multiple multiplex channels
- Client needed more effective means to provide affiliates access to digital assets: marketing materials, programming information, ads, etc.
- Client also needed ability to request print materials and to order services (e-commerce transactional back-end integration)
- Client required a single 3rd party system integrator that could:
  - Span technologies: Digital Asset Management, Content Management, Application Server, Portal
  - Span core competencies: Creative Design, Back-end Integration, etc.
  - Take over where a previous 3rd party systems integrator left off

Case Study #2: Cable Television

- Google TV: Android based
- Apple TV? IOS based ...

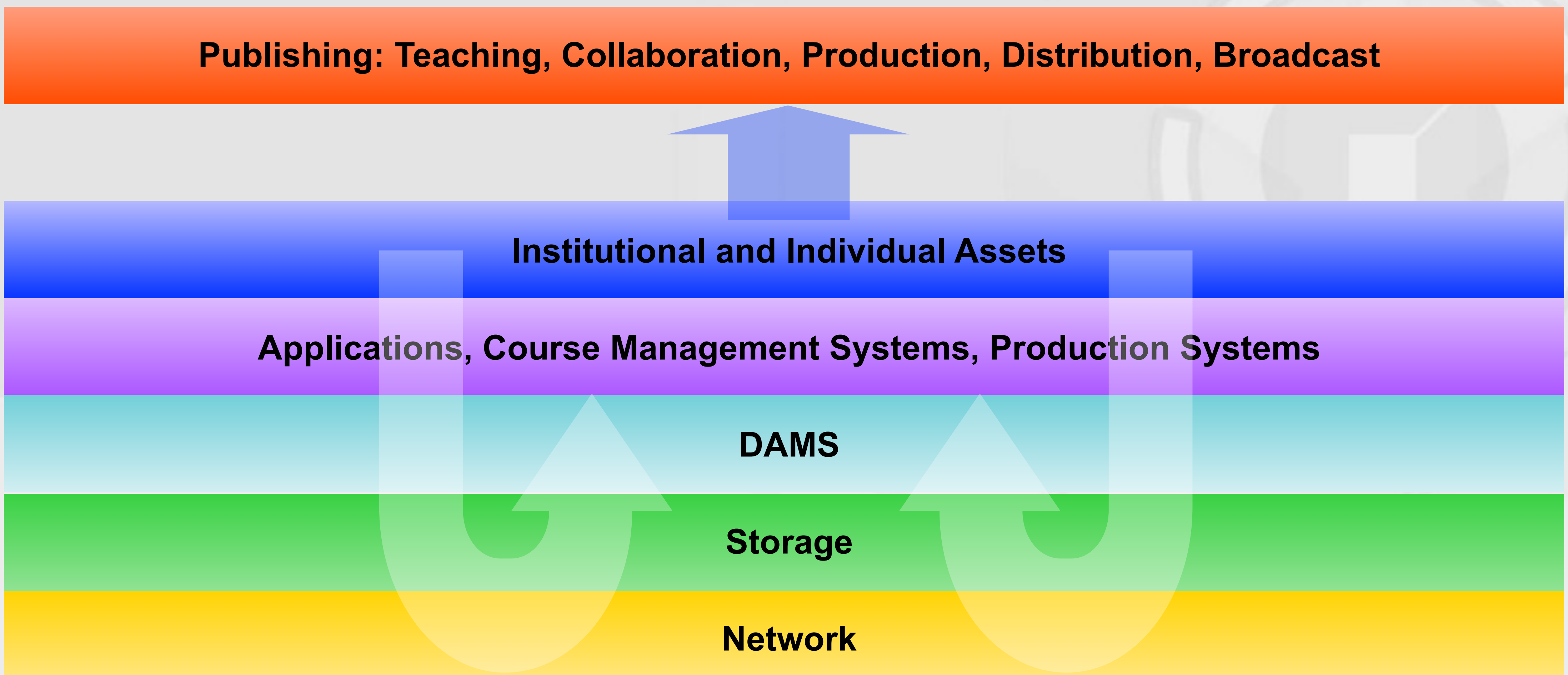
Case Study #3: A Digital Asset Management System at University of Michigan?

- Create a robust infrastructure to ingest, manage, store and publish digital rich-media (富媒体) assets and their associated metadata.
- Streamline the "workflow" required to create new works with digital rich-media assets.
- Build an environment where assets are easily searched, shared, edited and repurposed in the academic model.
- Provide a campus-wide platform for future application of rights declaration techniques (or other IP tools) to existing assets.

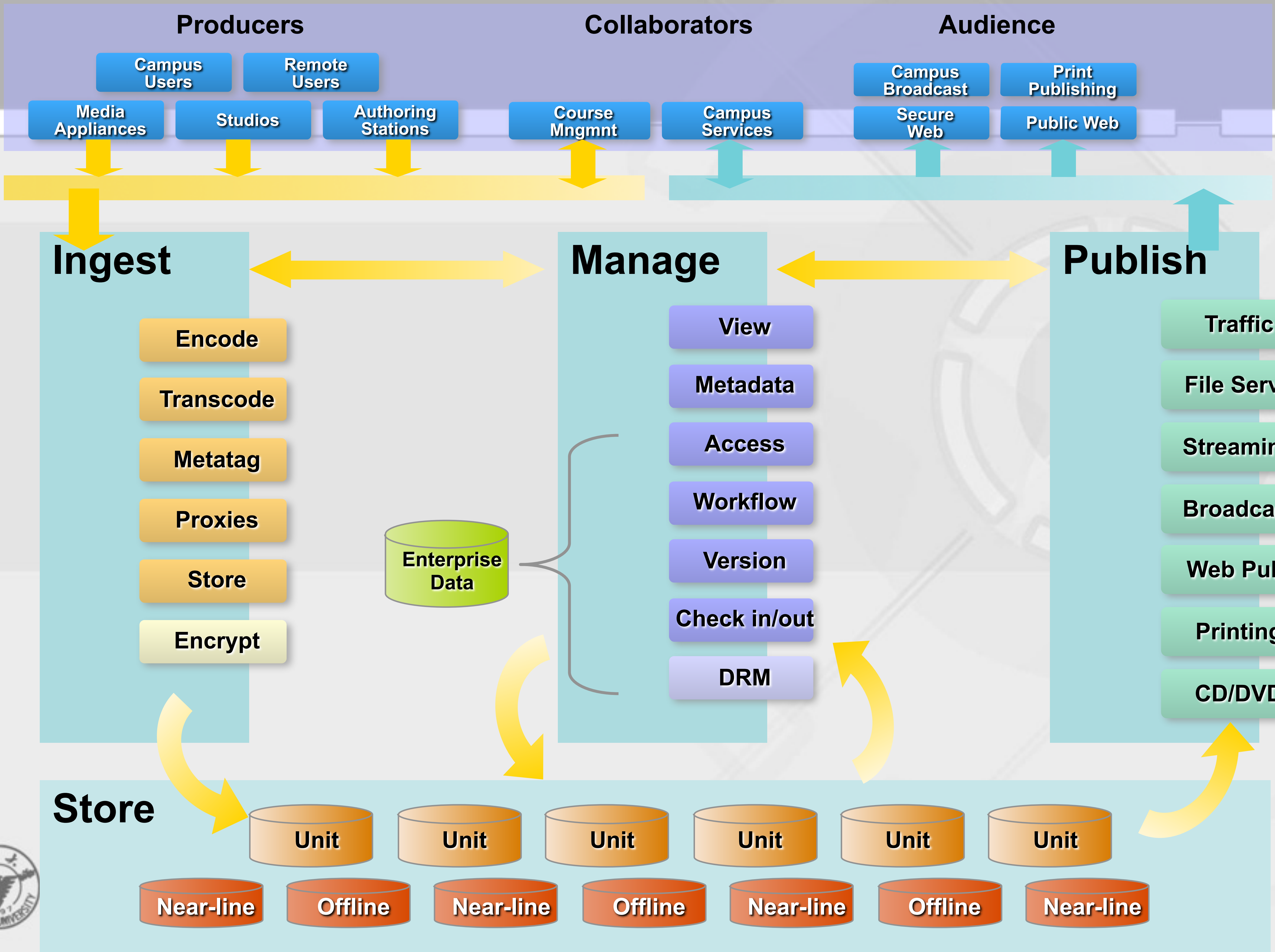
Orientation of DAMS at the UM

- Infrastructure level
- Tuned for rich media (time-based)
  - video
  - audio
  - 3D VR modeling and animation
- Capability for non time-based data (text, numerical data, still images)
- Metadata collection and management: automated or semi-automated
- Campus-wide availability
- Not primarily a content management tool nor production tool
- Coordinate with planned campus storage management practice
- Distributed management (authorization, roles, access lists)
- Integrated with centralized campus data services
- Plan for digital rights-declaration/management services

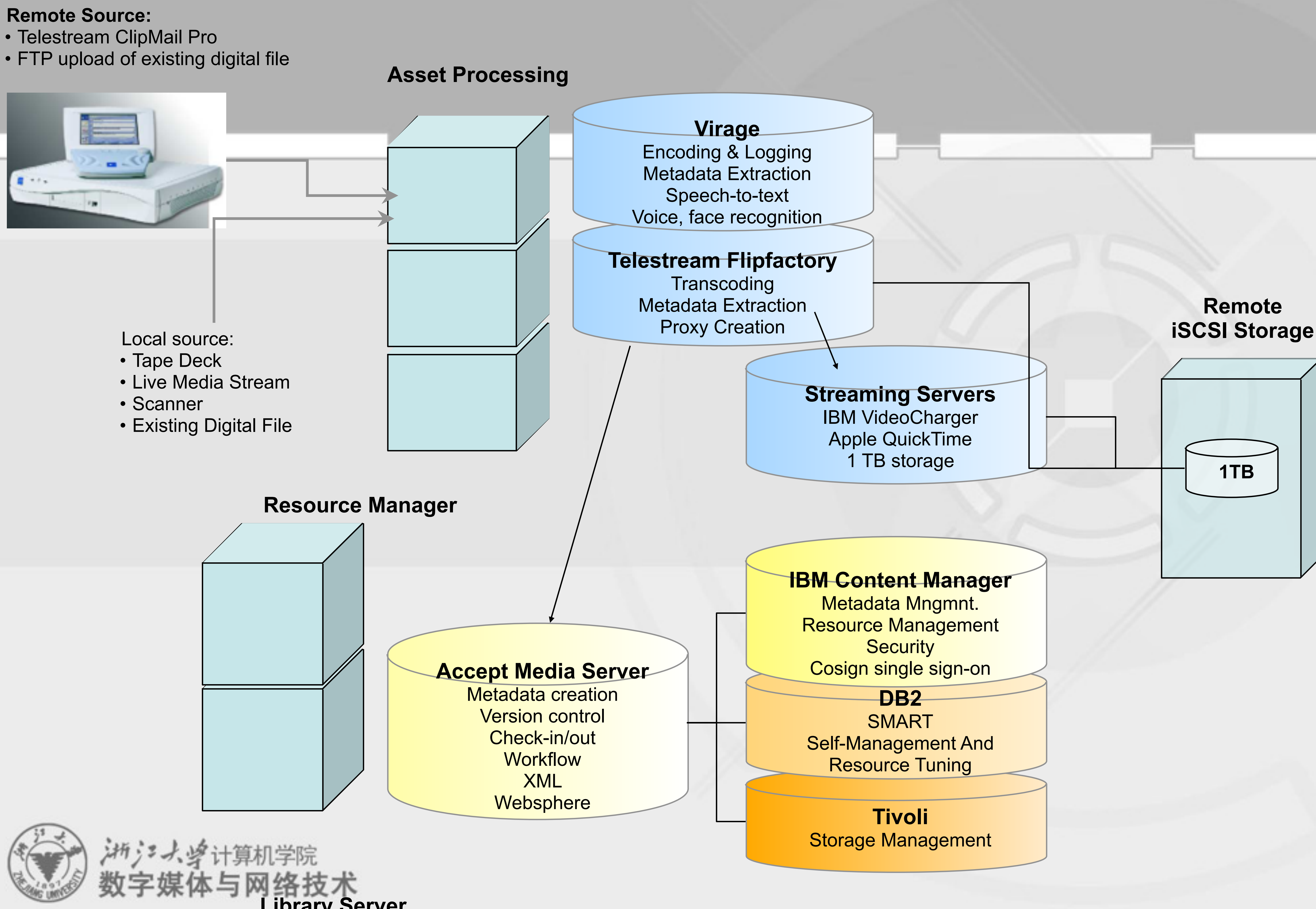
What is the place of DAMS in the campus infrastructure?



DAMS Component Services



DAMS Living Lab Configuration

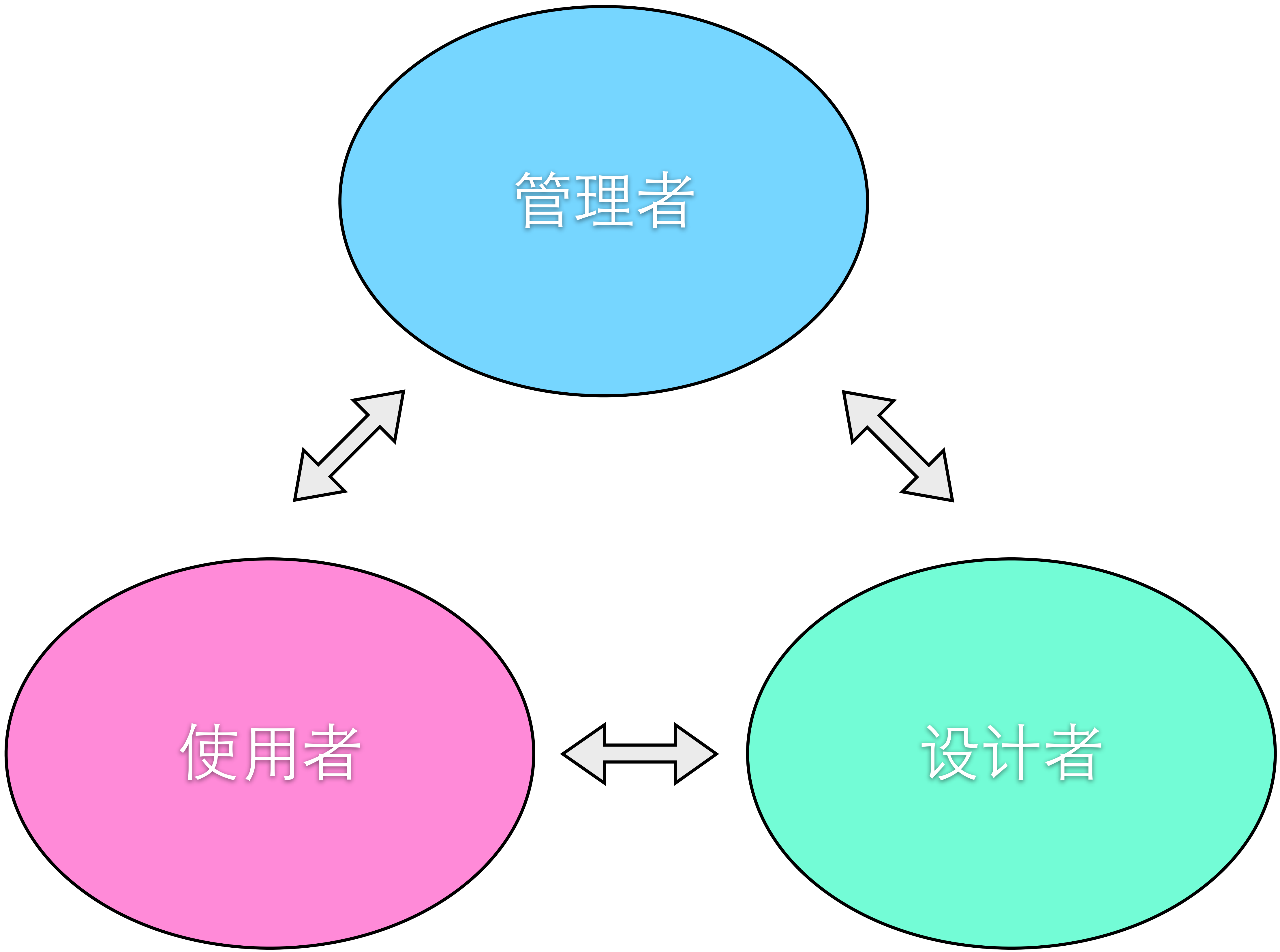


浙江大学计算机学院
数字媒体与网络技术

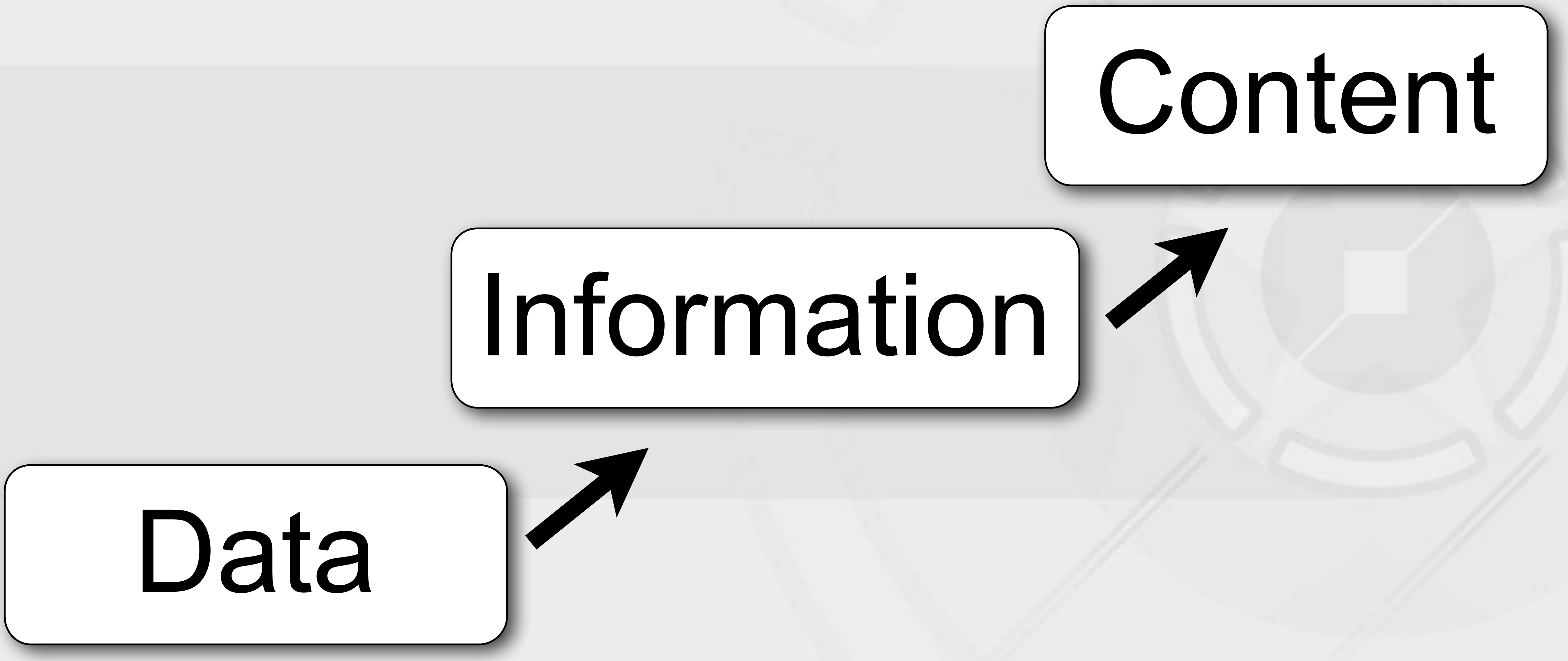
Extreme case ...

- Iron Mountain (铁山)：世界上最安全的数据中心
  - http://digi.tech.qq.com/a/20100819/000388.htm

数字媒体资源管理系统



Content



What is Digital Asset Management?

- Tools for organizing, storing and retrieving content in digital format
  - downloading, renaming, backing up, rating, grouping, archiving, optimizing, maintaining, thinning, and exporting ...

- Includes:
  - text, video, images, movies, sound, and 3D content

Content

content = essence + metadata
内容 = 素材 + 元数据

Digital media data types

Table. File format used in Macromedia Director

File import					**File export**		**Native**
Image	Palette	Sound	Video	Animation	Image	Video	
BMP	PAL	AIFF	AVI	DIR	BMP	AVI	DIR
GIF	ACT	AU	MOV	FLA		MOV	DXR
JPG		MP3		FLC			EXE
PICT		WAV		FLI			
PNG				GIF			
PNT				PPT			
PSD							
TGA							
TIFF							
WMF							

Digital Asset Management
数字媒体资源管理

2. Introduction to Digital Media Format

Key points

- To grasp features of different types of digital media

- To understand principles of coding different types of digital media

2.1 Image format and coding methods

图像编码

Common image formats

- General types:
  - –GIF
  - –JPEG
  - –PNG
  - –TIFF
  - –TGA
- Raw data:
  - –RAW
  - –DNG

- Platform spec.:
  - ➡BMP (Win)
  - ➡PAINT&PICT (Mac)
  - ➡PPM (X-Win)
- Vector data:
  - ➡WMF (Win)
  - ➡PS and PDF

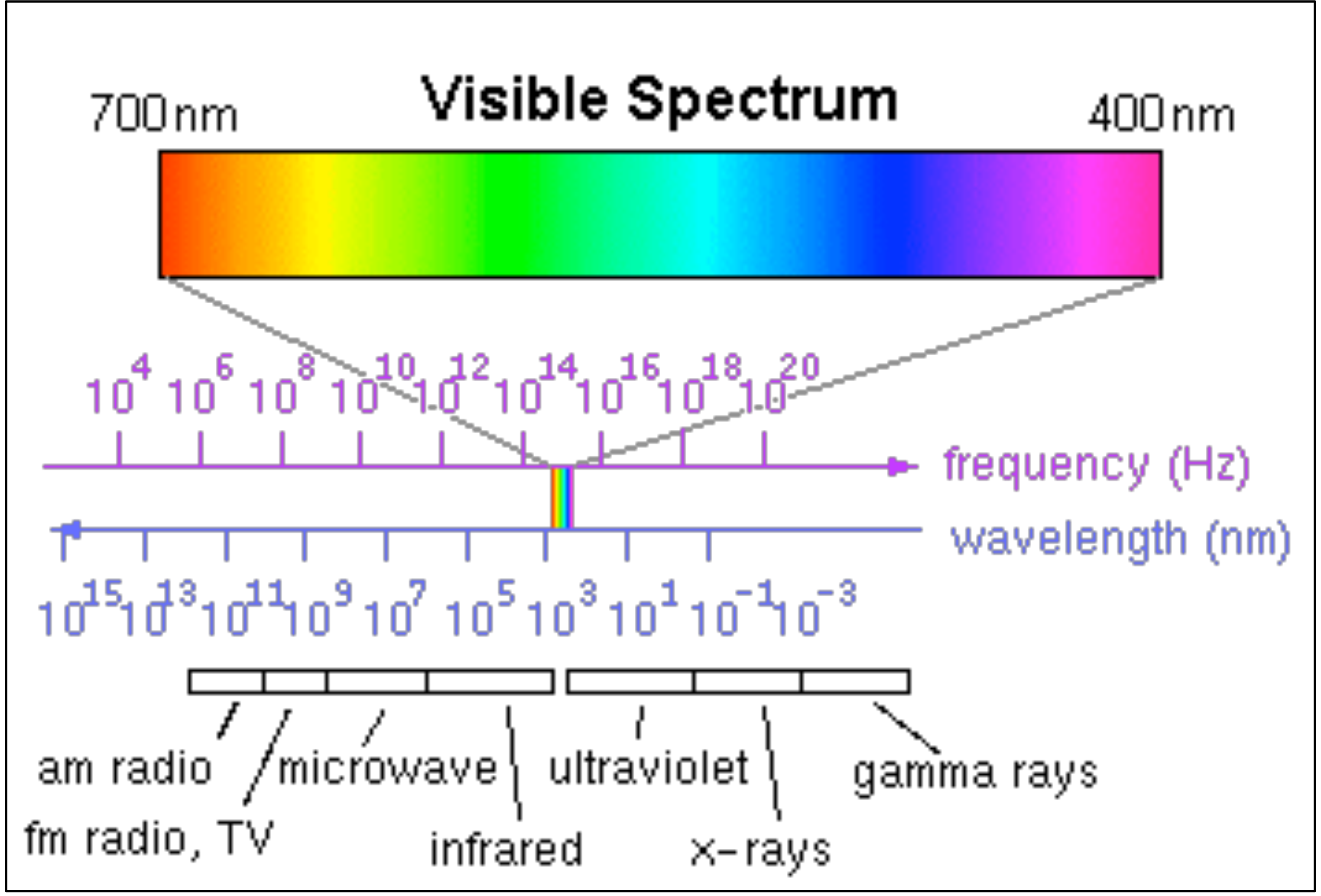
Common image formats

- Key points of storage
  - –Color space
  - –Coding methods
  - –Byte order： hardware dependent
    - MSB/LSB (most/least significant byte)

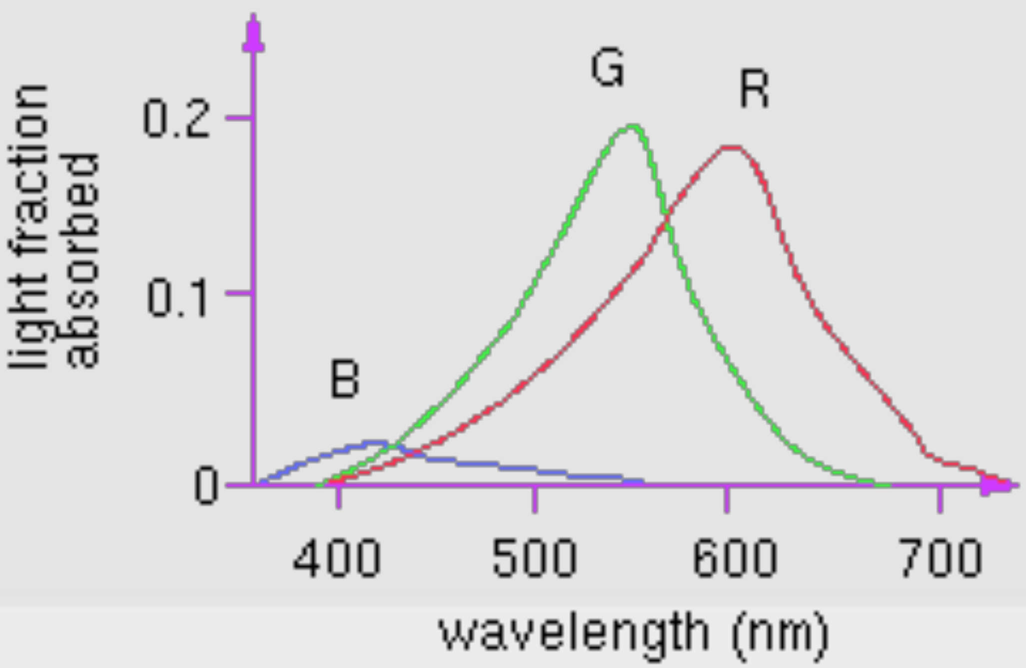
2.1.1 Color spaces

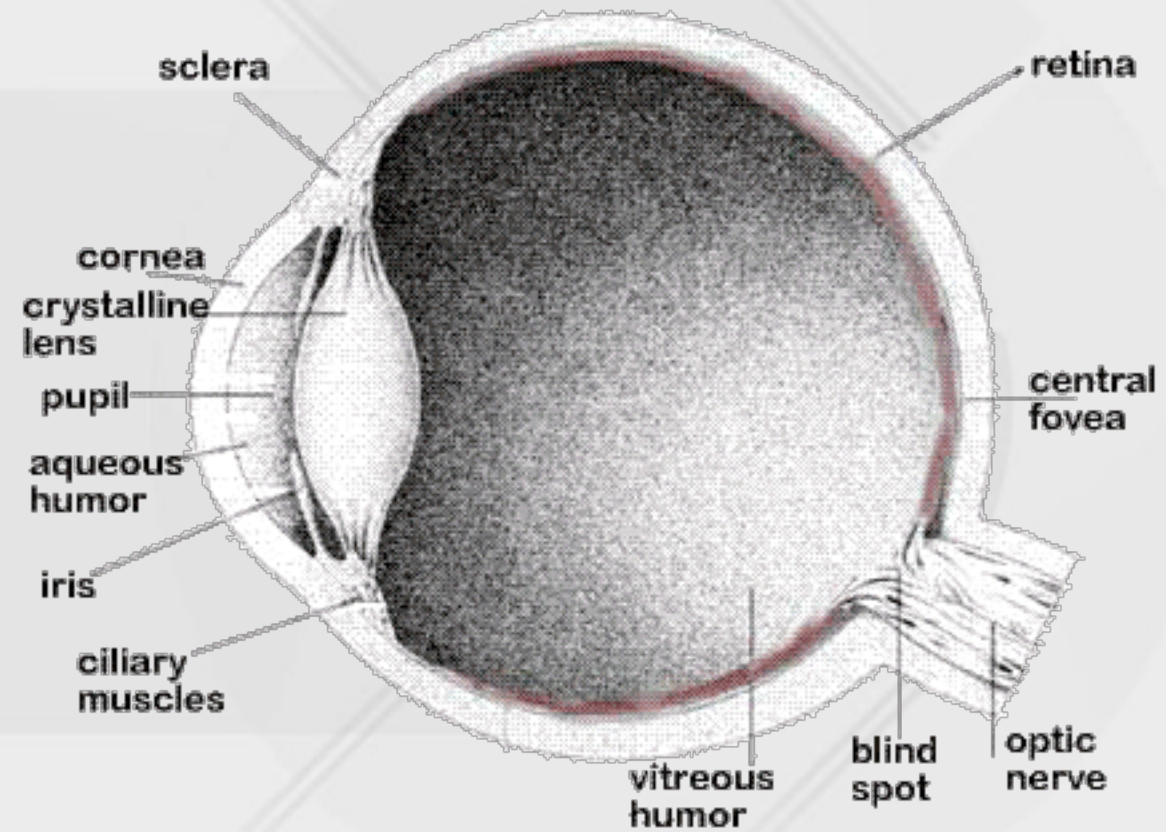
Color systems and color models

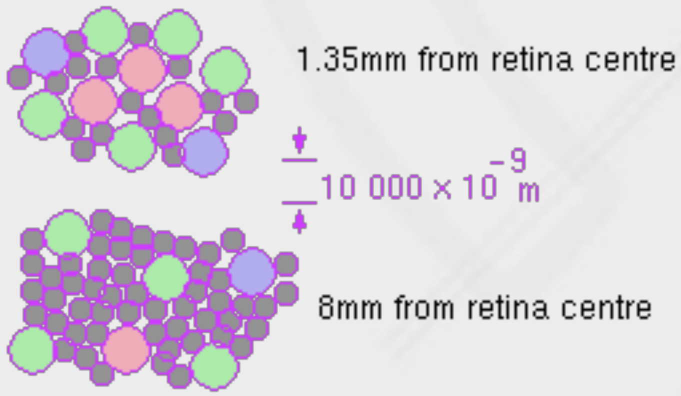
- Common systems
  - –RGB
  - –XYZ
  - –Lab
  - –YUV
  - –HSV



Color Model


light fraction absorbed
0.2
0.1
0
G
R
B
400
500
600
700
wavelength (nm)


sclera
retina
cornea
crystalline lens
pupil
aqueous humor
iris
ciliary muscles
central fovea
vitreous humor
blind spot
optic nerve


1.35mm from retina centre
10 000 × 10⁻⁹ m
8mm from retina centre

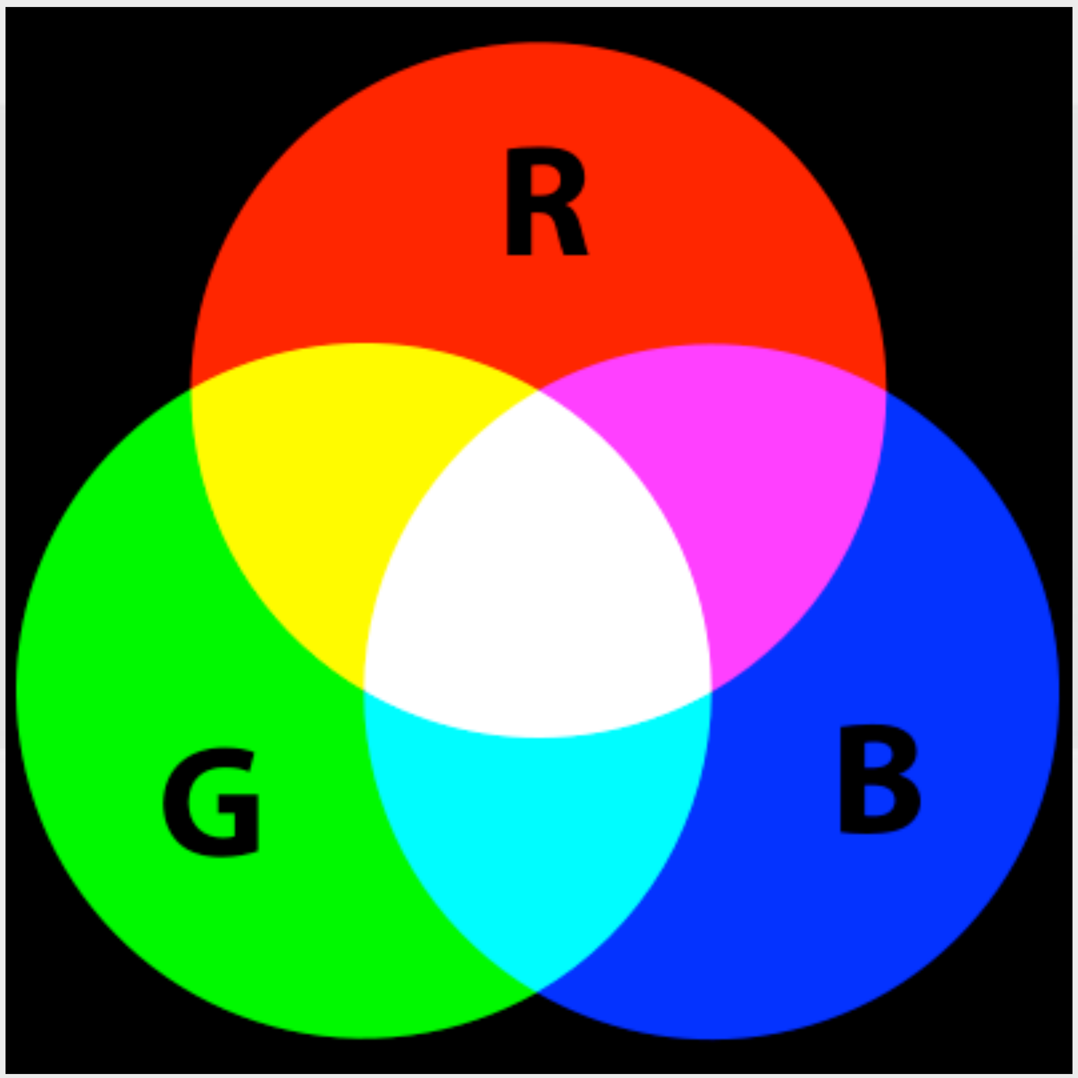
Color perception

- Three types of cones:

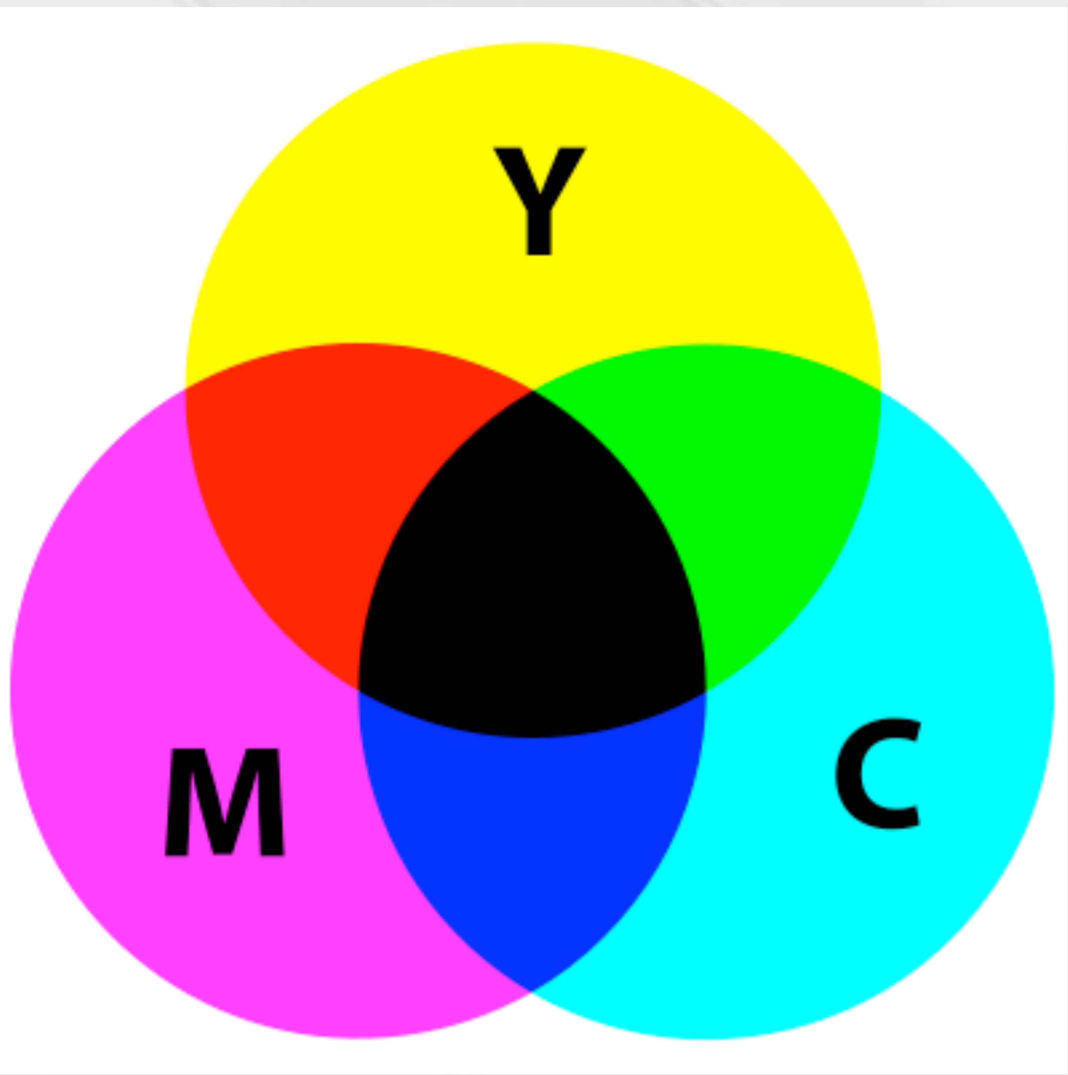
S	M	L	
Blue	Green	Red	roughly approximate
430nm	560nm	610nm	peak sensitivities

- Colorblindness results from a deficiency of one cone type.

RGB & CMYK



Additive color mixing



Subtractive color mixing

CMYK => RGB

$$t_{CMYK} = C, M, \Upsilon, K$$

RGB => CMYK?

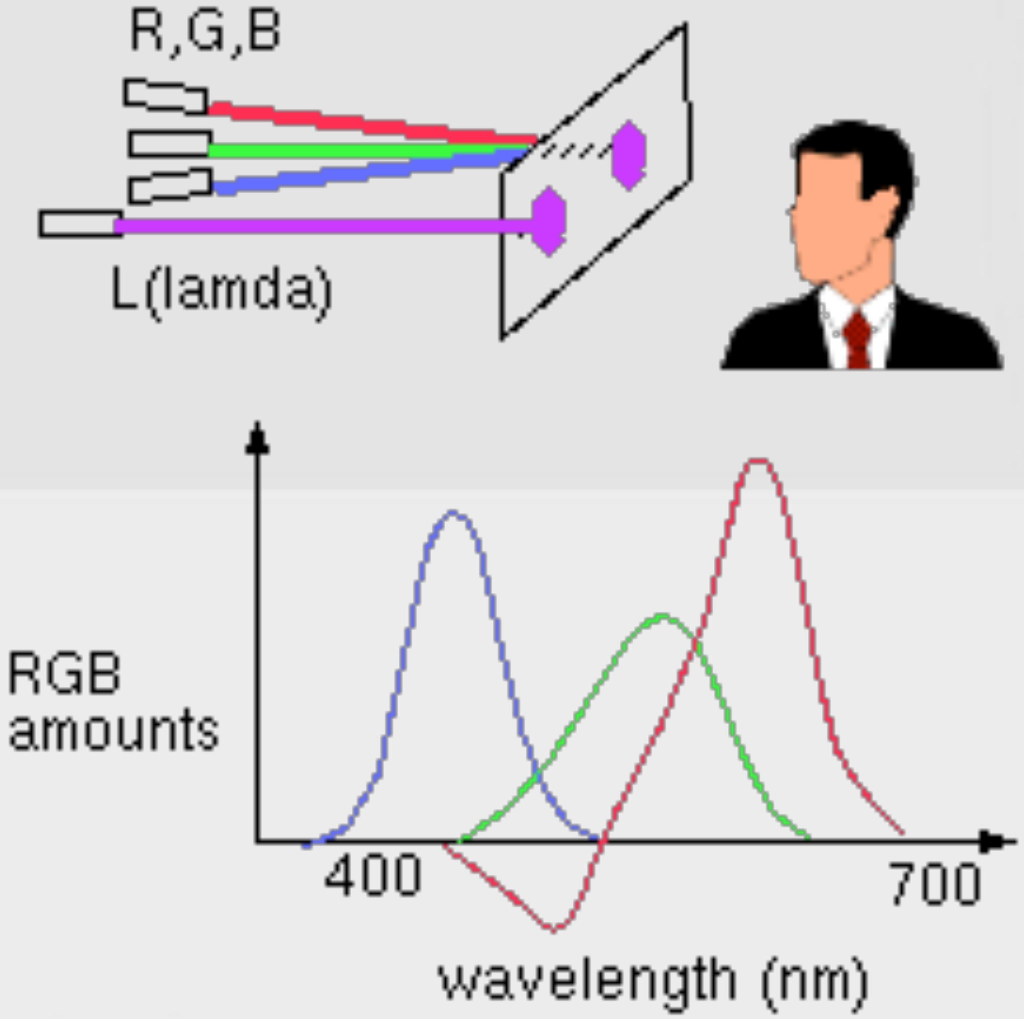
$$t_{CM\Upsilon} = C', M', \Upsilon'$$
$$= C(1-K)+K, M(1-K)+K, \Upsilon(1-K)+K$$

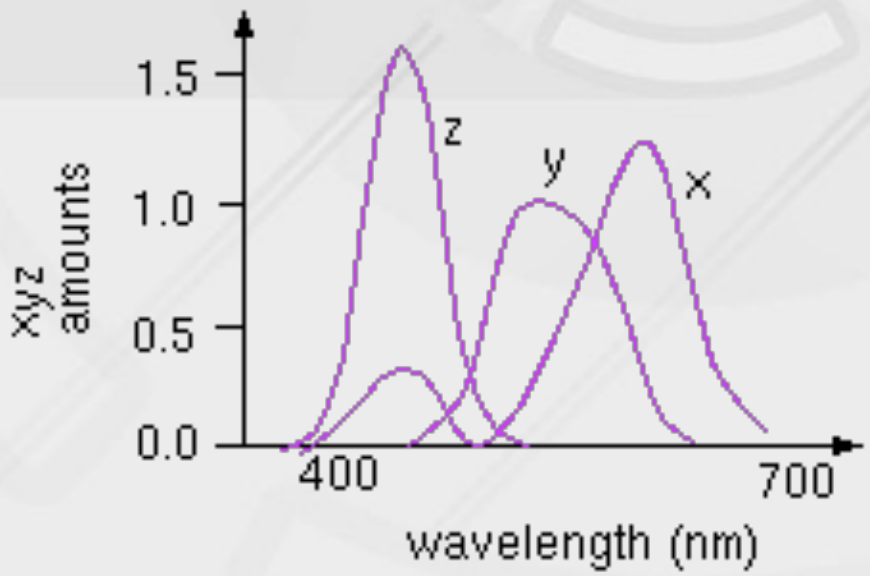
$$t_{RGB} = R, G, B$$
$$= 1-C', 1-M', 1-\Upsilon'$$

$$t_{RGB} = \{1-(C(1-K)+K), 1-(M(1-K)+K), 1-(\Upsilon(1-K)+K)\}$$
$$= \{1-C(1-K)-K, 1-M(1-K)-K, 1-\Upsilon(1-K)-K\}$$

CIE XYZ space

- CIE: Commission Internationale d'Eclairage"





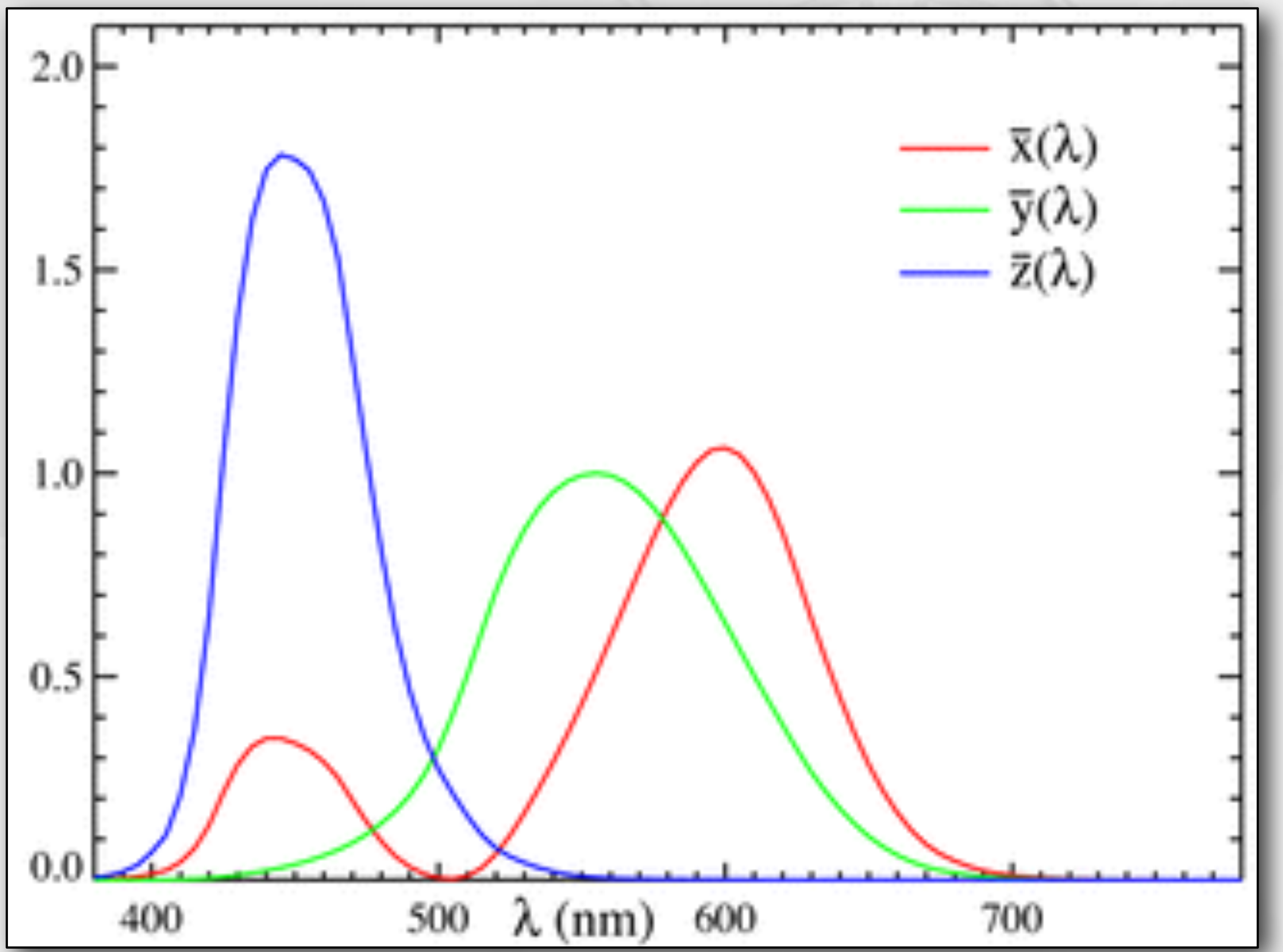
CIE XYZ space

- Color matching function

$$X = \int_0^\infty I(\lambda)\,\overline{x}(\lambda)\,d\lambda$$

$$Y = \int_0^\infty I(\lambda)\,\overline{y}(\lambda)\,d\lambda$$

$$Z = \int_0^\infty I(\lambda)\,\overline{z}(\lambda)\,d\lambda$$

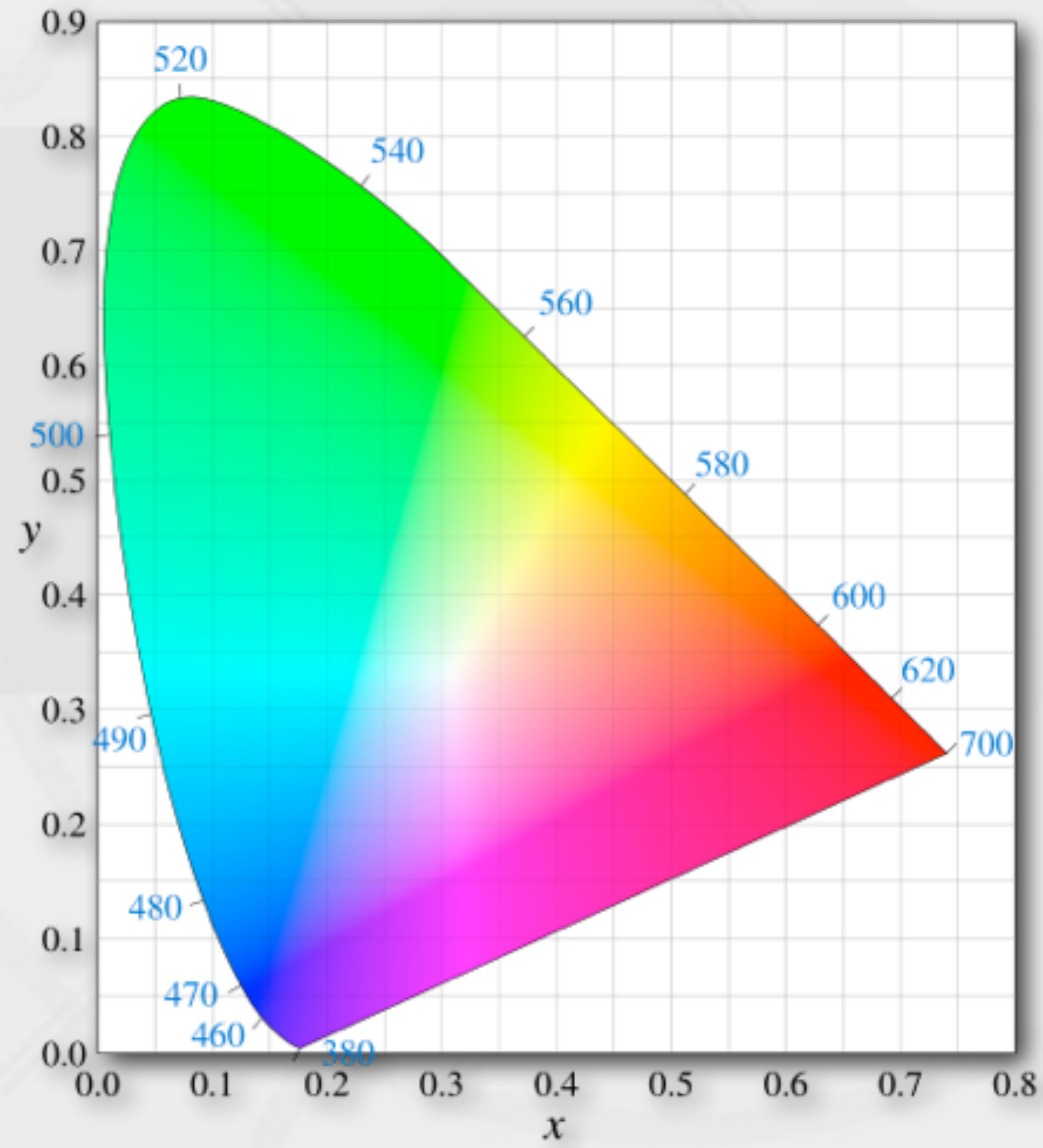


CIE XYZ space

$$x = \frac{X}{X+Y+Z}$$

$$y = \frac{Y}{X+Y+Z}$$

$$z = \frac{Z}{X+Y+Z} = 1 - x - y$$

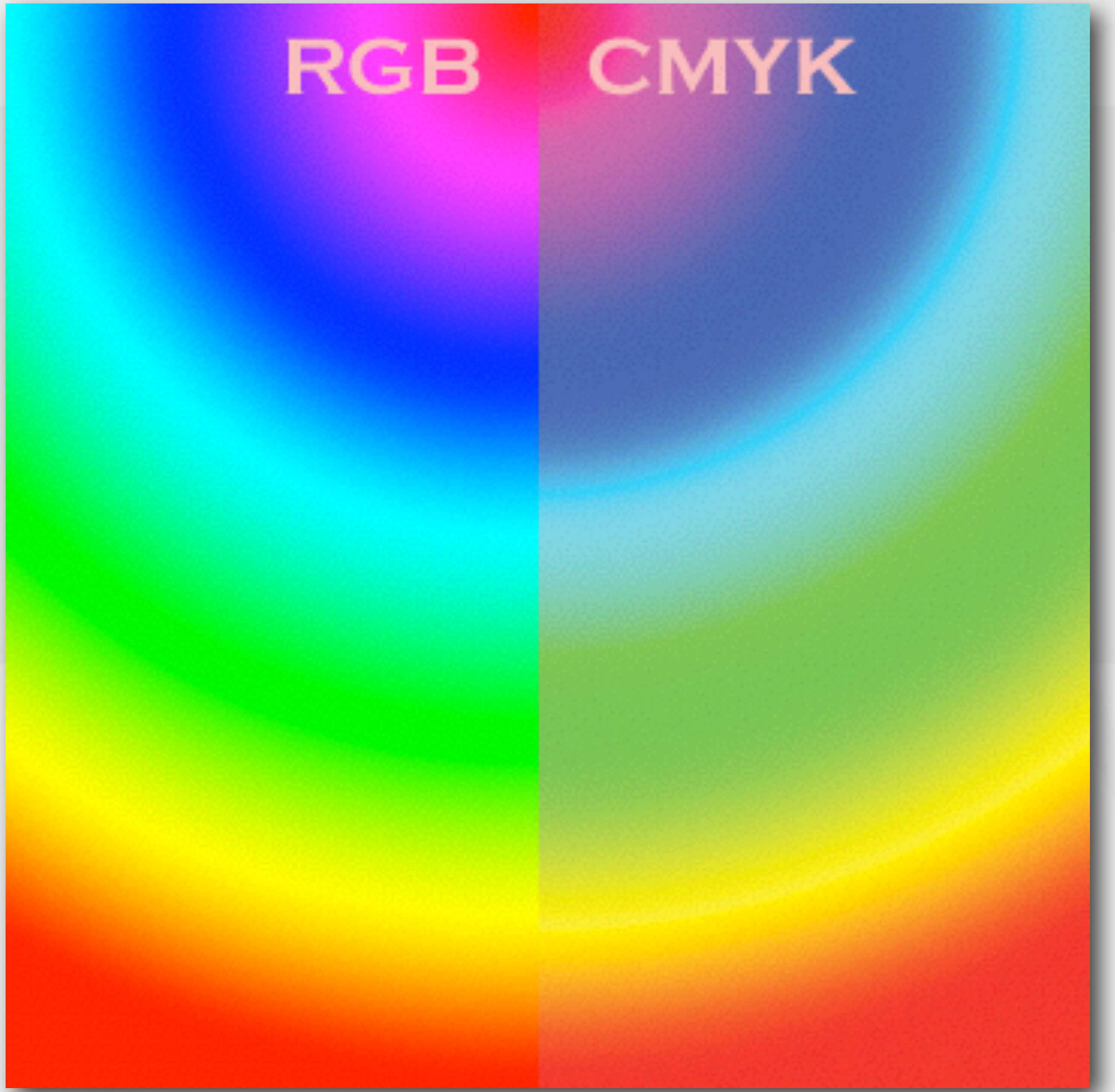


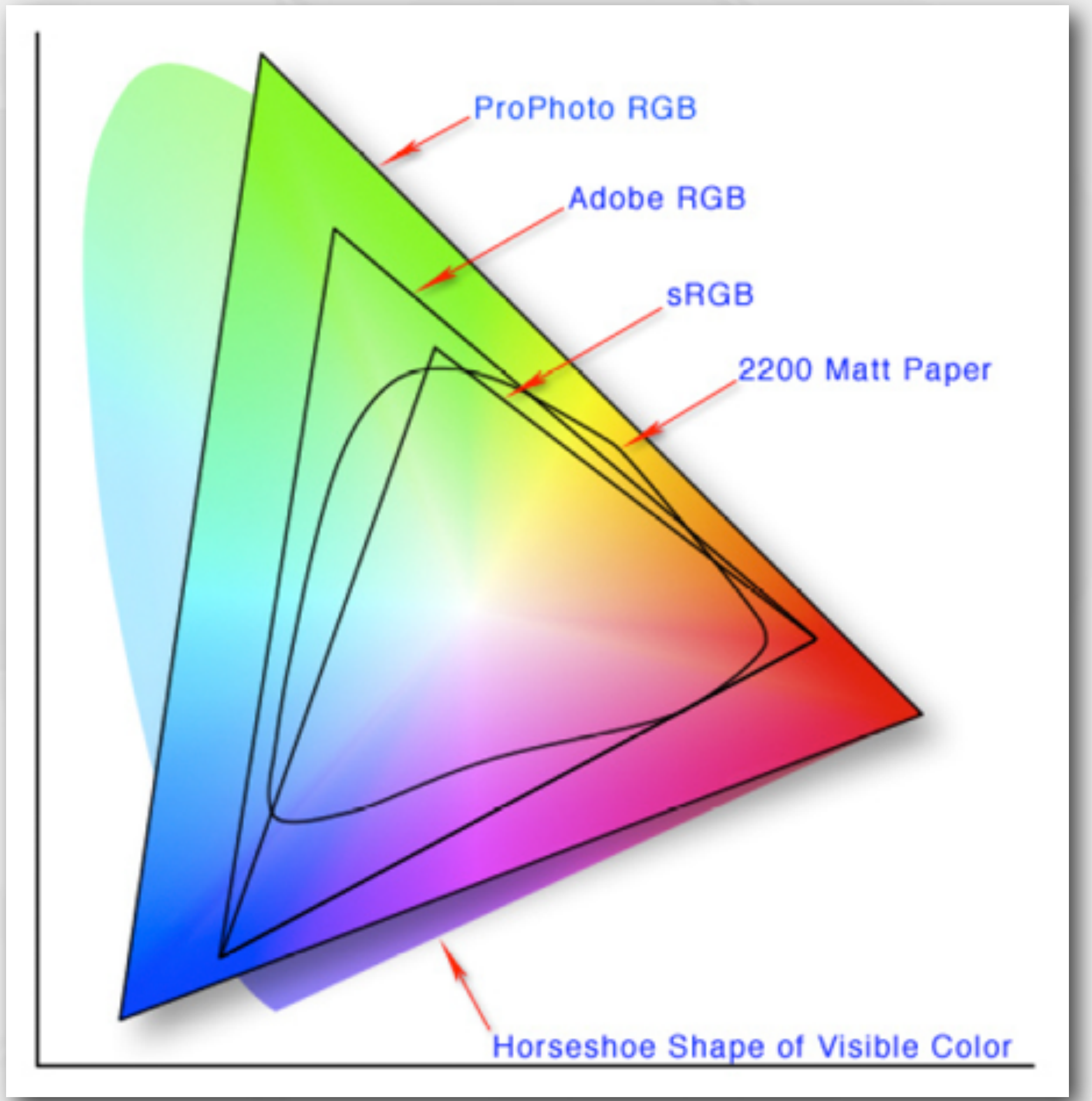
RGB vs. XYZ

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \frac{1}{b_{21}} \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} = \frac{1}{0.17697} \begin{bmatrix} 0.49 & 0.31 & 0.20 \\ 0.17697 & 0.81240 & 0.01063 \\ 0.00 & 0.01 & 0.99 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Since 1931

RGB :: CMYK :: XYZ color spaces





YUV color spaces

- used in most video capture system
- PAL television system

$$\begin{bmatrix} Y' \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.14713 & -0.28886 & 0.436 \\ 0.615 & -0.51499 & -0.10001 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Image

Y

U

V

Color spaces: reference

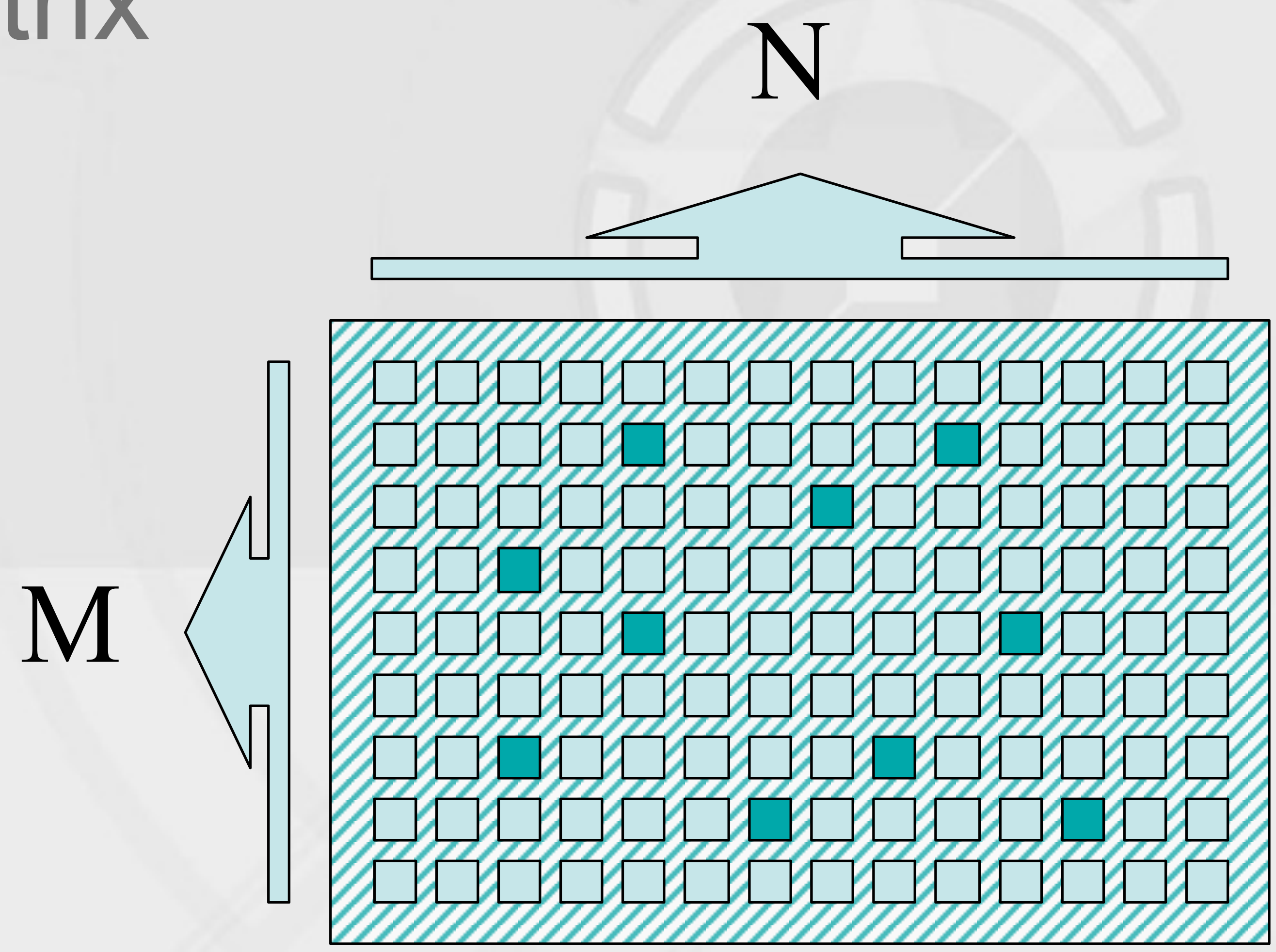
- http://en.wikipedia.org/wiki/Color_space
- http://www.cs.unc.edu/~mcmillan/comp136/Lecture4/Color.html

2.1.2 Image representations

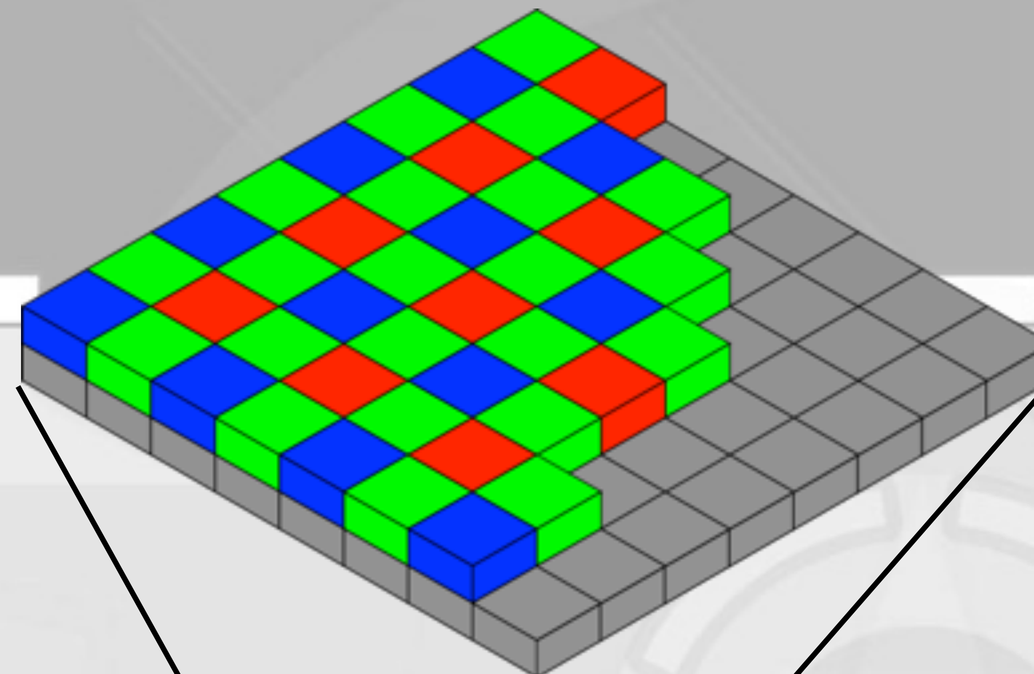
Representation of digital images

- An image can be viewed as a ***N×M*** vector matrix

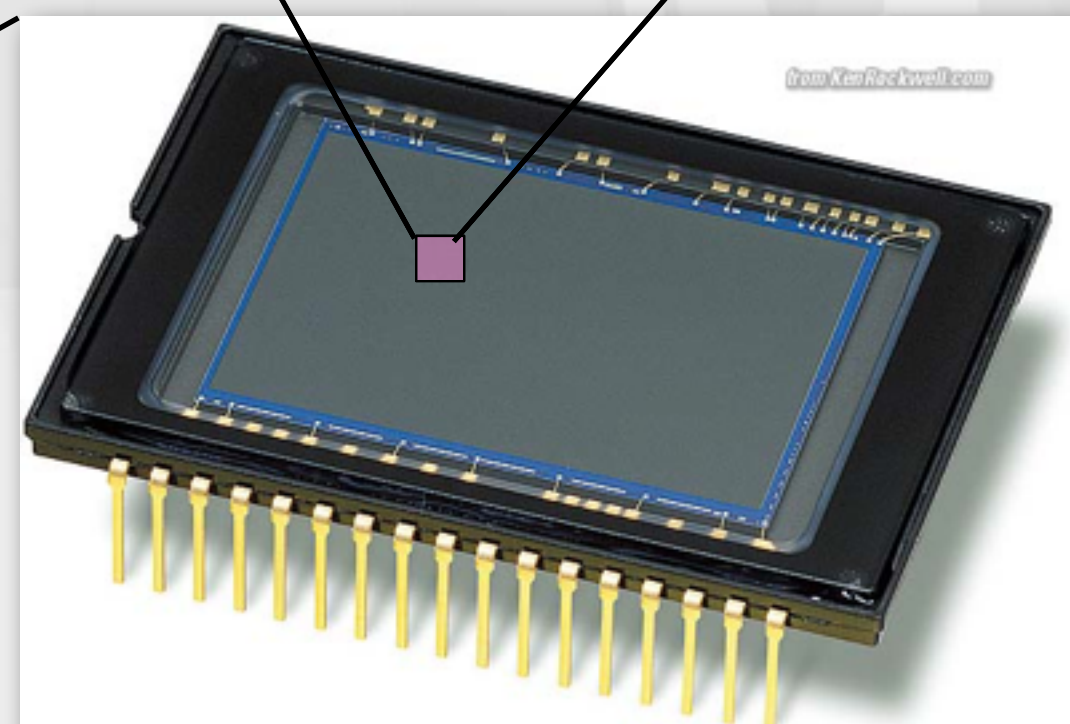
- Grayscale image
- Color image
- Palette



CCD sensors



Nikon D40



CCD sensor

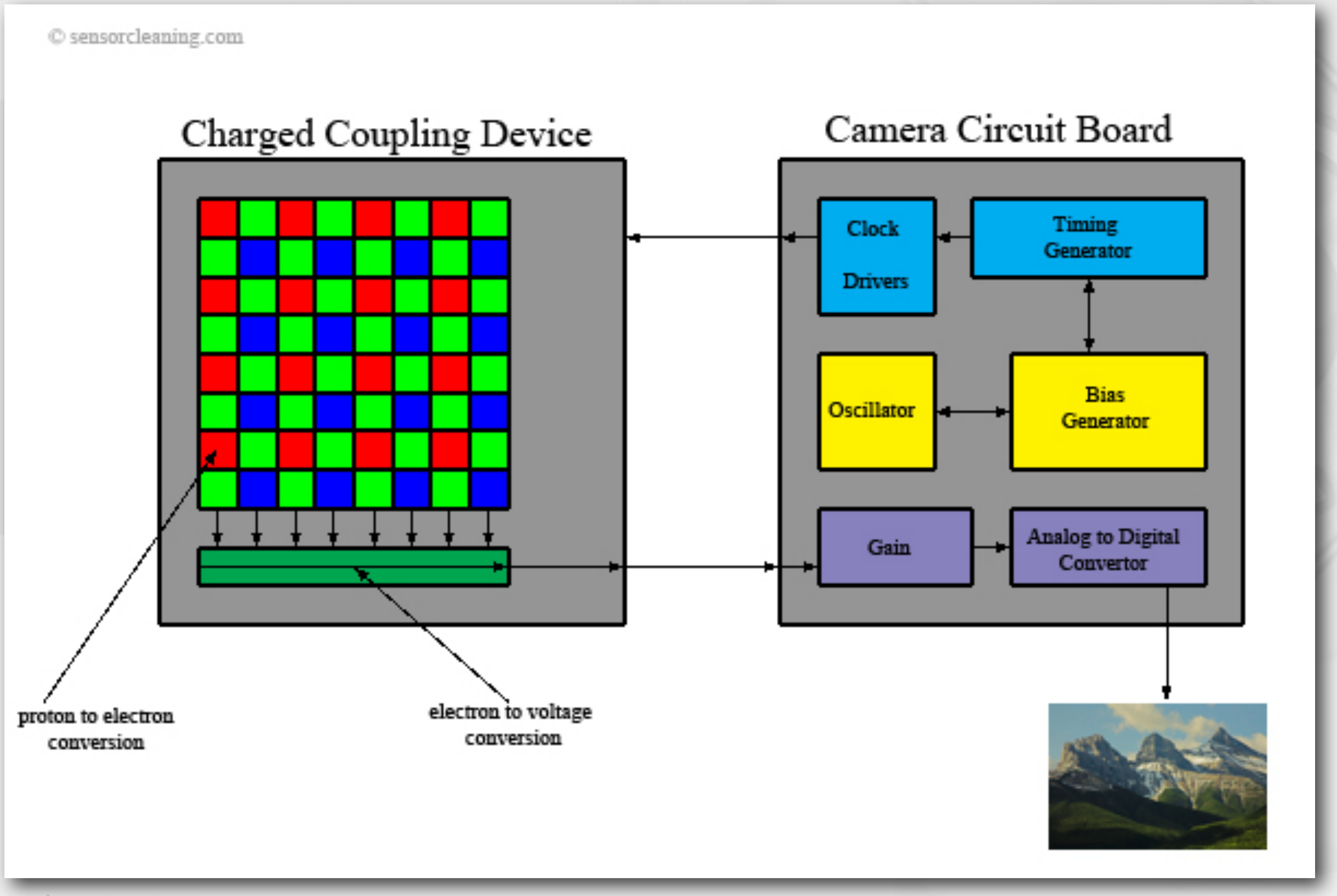
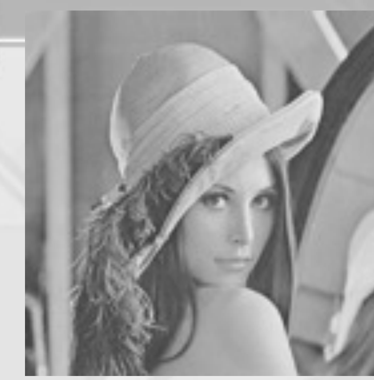
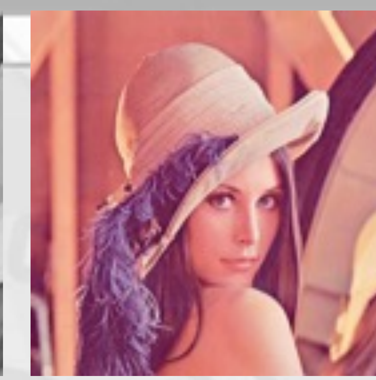

© sensorcleaning.com
Charged Coupling Device
Camera Circuit Board
Clock
Drivers
Timing
Generator
Oscillator
Bias
Generator
Gain
Analog to Digital
Convertor
proton to electron
conversion
electron to voltage
conversion

Image resolutions

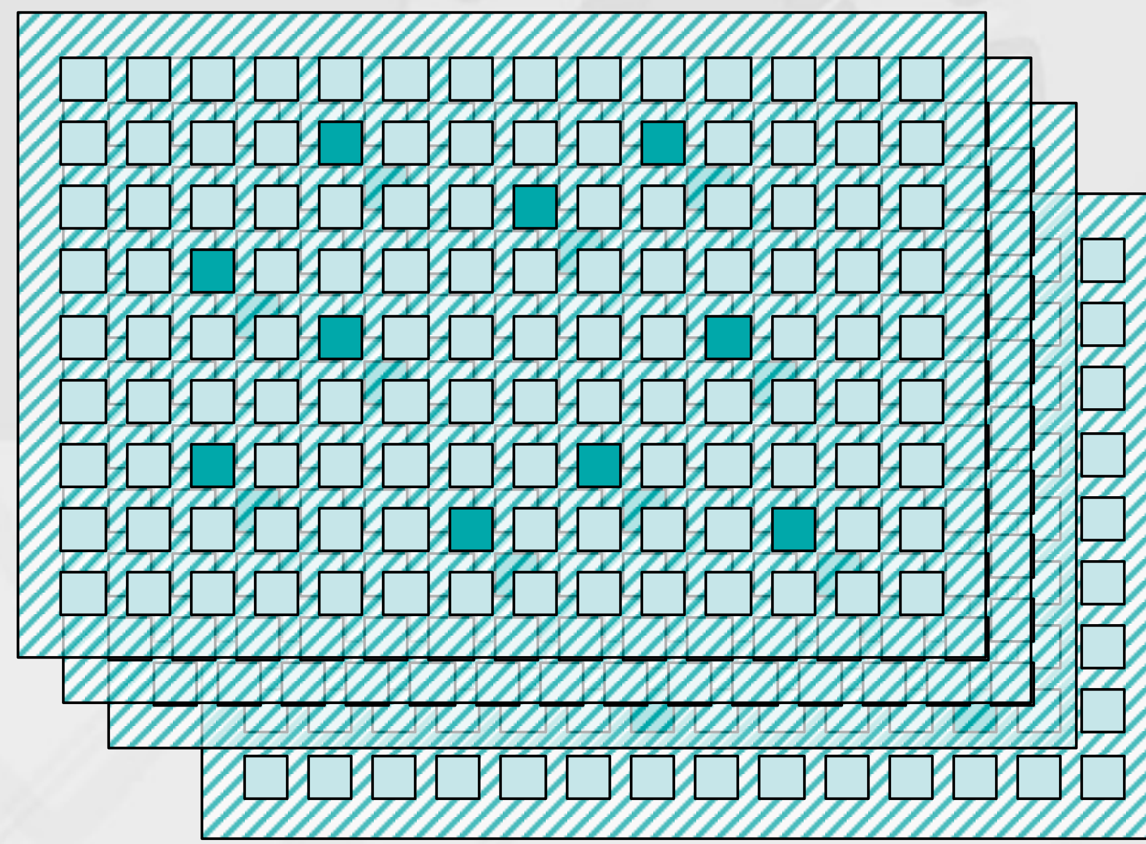
Dimensions	MEGA pixels	Name	Comments
640x480	0.3	VGA	
720x576	0.4	CCIR 601 DV PAL	PAL DV, and PAL DVDs
768x576	0.4	CCIR 601 PAL full	PAL with square sampling grid ratio
800x600	0.4	SVGA	
1024x768	0.8	XGA	
1280x960	1.2		
1600x1200	2.1	UXGA	
1920x1080	2.1	1080 HDTV	high resolution digital TV format
2048x1536	3.1	2K	Typically used for digital effects in feature films.
4096x3072	12.4	4K (!!!)	

Rep of Images

- Binary image
  - –1 bit = Boolean value
  - –One bit-planes

- Common Grayscale image:
  - –8 bits = 256 degrees of grayscale
  - –Eight bit-planes

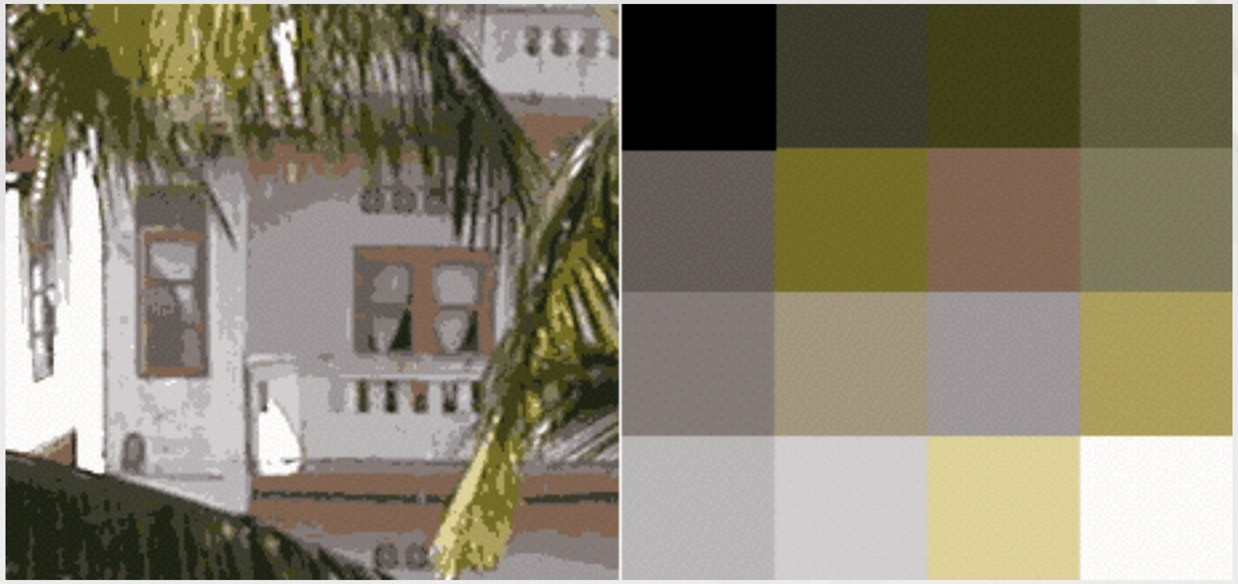


Rep of Images

- Most used color images
  - –24bit RGB
  - –Red/Green/Blue each channel has 256 degrees of grayscale
  - –Can represent 2^24 = 16,777,216 types of color

Rep of image – Palette

- Some systems and applications can only use 8-bit color images
  - –Solution： Palette (Color look-up table)



High dynamic range image

HDRI example: a New York City nighttime cityscape

High dynamic range image

- HDR pixels:
  - 16-bit or 32-bit floating point numbers
  - 10–12 bits luminance (亮度)
  - 8 bits chrominance (色度)
  - 10^{-4} to 10^{8}: the range of visible luminance values
- CMOS image sensors: up to 110dB
- Tone mapping:
  - Typical computer monitors, prints, and other methods of displaying images only have a limited dynamic range

An example of a High Dynamic Range (HDR) photography, made of three different exposures

2.1.3 Image encoding

Image compression methods

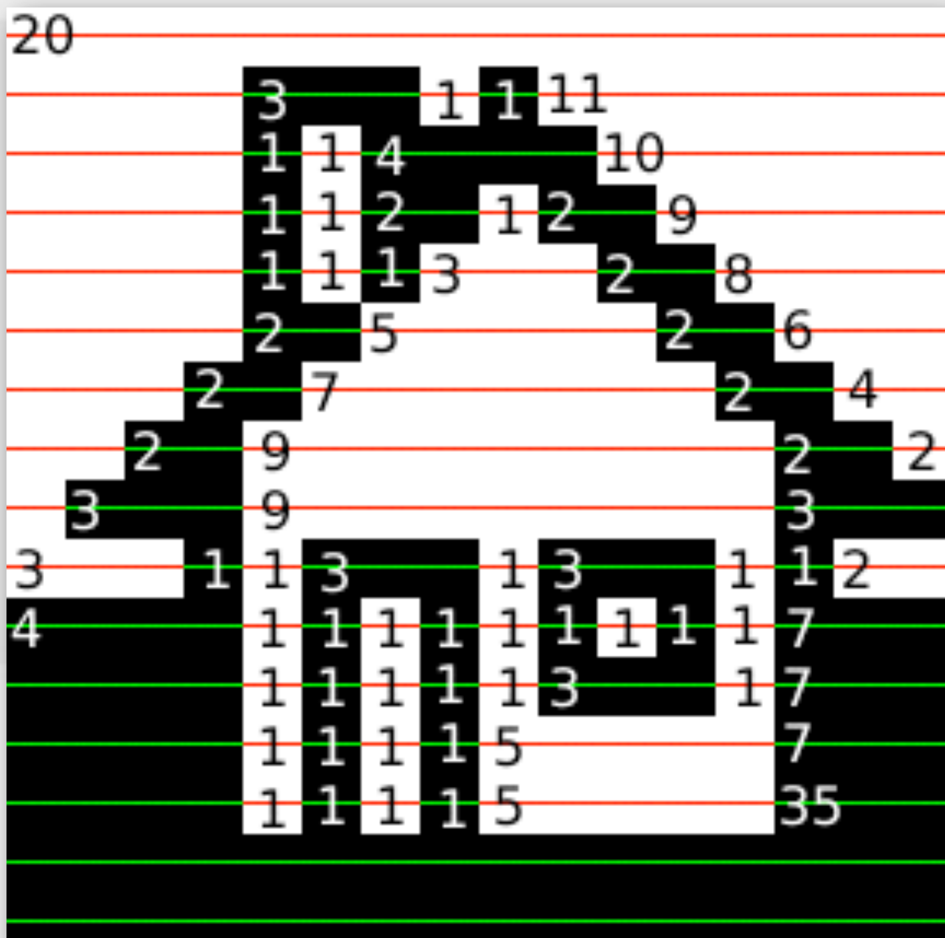
- lossless compression
- lossy compression

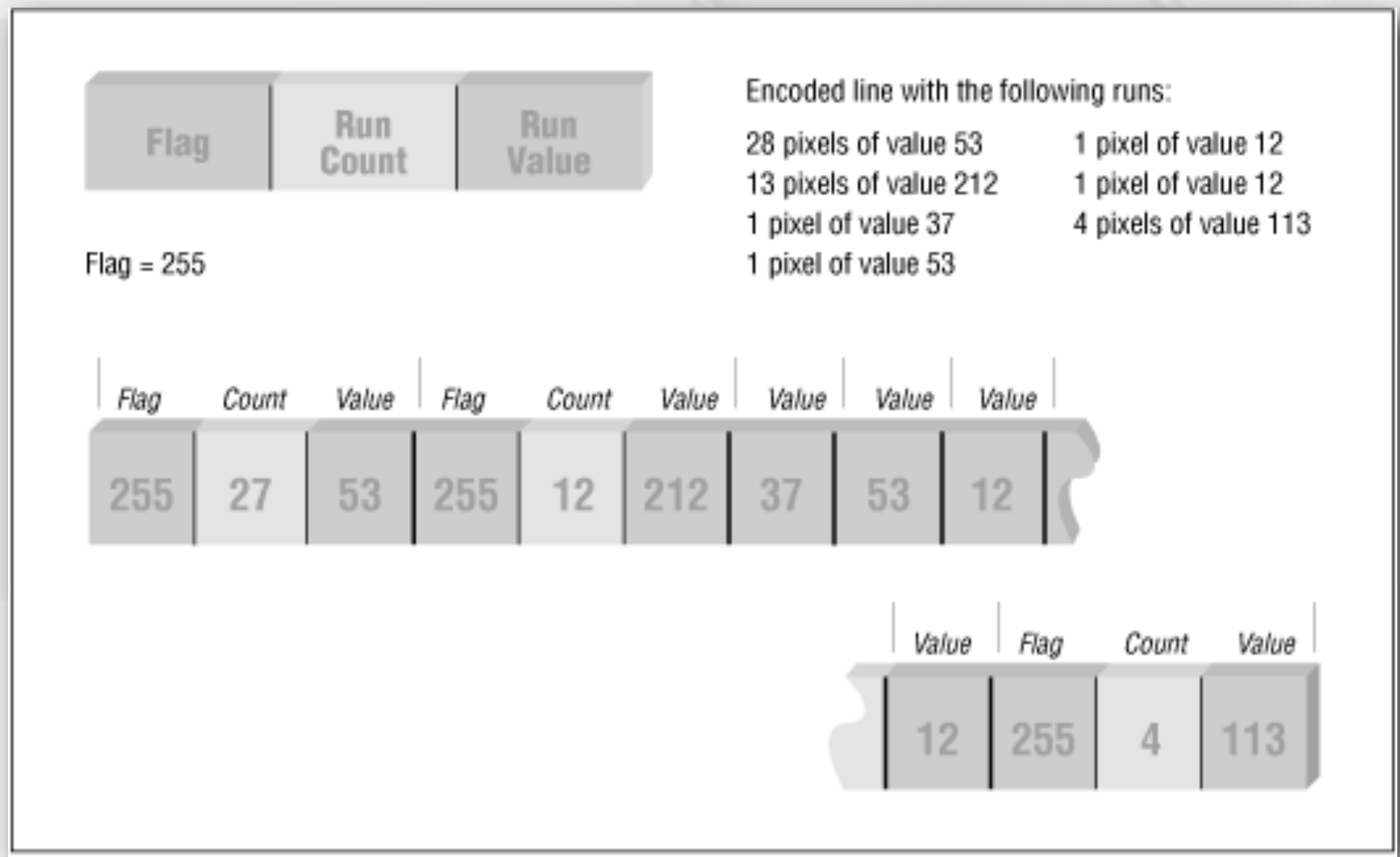
Lossless image compression methods

- Based on information theory

- General encoding methods
  - –RLC (Run-Length Coding)
  - –VLC (Variable-Length Coding)
  - –Dictionary Coding
  - –Arithmetic Coding

Run-Length Encoding





LZW : Lempel-Ziv-Welsh

- Universal lossless data compression algorithm
  - –by Abraham Lempel, Jacob Ziv, and Terry Welsh

- The compressor algorithm builds a string translation table from the text being compressed

LZW - Compressor

```
w = NIL;
add all possible char codes to the dictionary
for (every character c in the uncompressed data) do
    if ((w + c) exists in the dictionary) then
        w = w + c;
    else
        add (w + c) to the dictionary;
        add the dictionary code for w to output;
        w = c;
    endif
done
add the dictionary code for w to output;
display output;
```

● 原输入数据为:

● A B A B A B A B B B A B A B A A C D A C D A D C A B A A A B A B …

● 初始标号集为:

0	1	2	3	4	5
A	B	C	D	Clear	End

● 编码过程:

步骤	前缀	后缀	Entry	认识(Y/N)	输出	标号
1		A	(, A)			
2	A	B	(A,B)	N	A	6
3	B	A	(B,A)	N	B	7
4	A	B	(A,B)	Y		
5	6	A	(6,A)	N	6	8
6	A	B	(A,B)	Y		
7	6	A	(6,A)	Y		
8	8	B	(8,B)	N	8	9
9	B	B	(B,B)	N	B	10
10	B	B	(B,B)	Y		
11	10	A	(10,A)	N	10	11
12	A	B	(A,B)	Y		

● 编码结果:

0	1	2	3	4	5	6	7	8	9	10	11
A	B	C	D	Clear	End	AB	BA	6A	8B	BB	10A

LZW - Decompressor

```
read a char k;
output k;
w = k;
while (read a char k) do
   if (index k exists in dictionary) then
       entry = dictionary entry for k;
   else if (k == currSizeDict)
       entry = w + w[0];
   else
       signal invalid code;
   endif
   output entry;
   add w+entry[0] to the dictionary;
   w = entry;
done
```

Deflate

- a lossless data compression algorithm:
  - –LZ77 algorithm + Huffman coding.
  - –originally defined by Phil Katz for version 2 of his PKZIP archiving tool,
  - –later specified in RFC 1951.
  - –used by gzip, modern versions of zip and as part of the compression process of PNG, PPP, HTTP, SSH

Lossless image compression methods (cont.)

- Other lossless image compression methods
  - –Image different encoding （差分）
  - –Lossless JPEG (JPEG 2000)
    - discrete wavelet transform

Lossy image compression methods

- Quantization
- Transform coding
  - –Discrete Cosine Transform
  - –Discrete Wavelet Transform
  - –Karhune-Loeve Transform (Principle component analysis)

Image compression standards

- JPEG
  - –Joint picture encoding group
  - –Discrete Cosine Transform
- JPEG 2000
  - –newer standard
  - –Discrete Wavelet Transform

JPEG compression: main idea

http://zh.wikipedia.org/zh-cn/JPEG

52	55	61	66	70	61	64	73
63	59	55	90	109	85	69	72
62	59	68	113	144	104	66	73
63	58	71	122	154	106	70	69
67	61	68	104	126	88	68	70
79	65	60	70	77	68	58	75
85	71	64	59	55	61	65	83
87	79	69	68	65	76	78	94

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

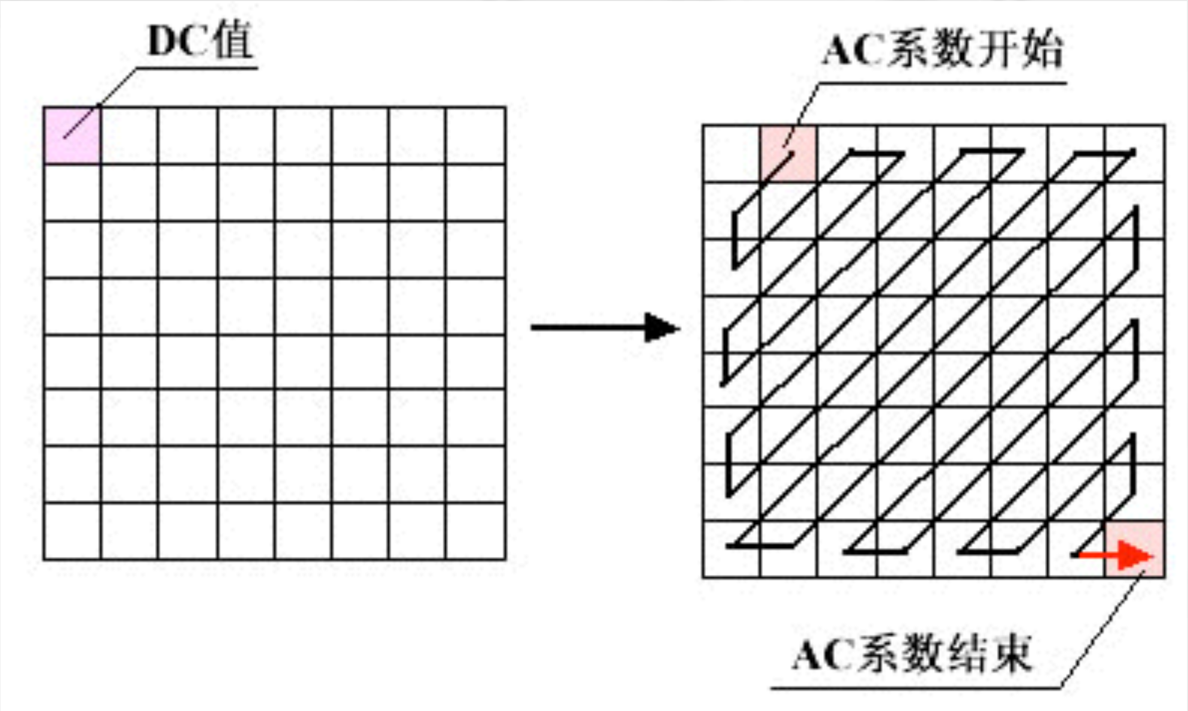
−26	−3	−6	2	2	−1	0	0
0	−2	−4	1	1	0	0	0
−3	1	5	−1	−1	0	0	0
−4	1	2	−1	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

RGB Image → YCbCr Color Model

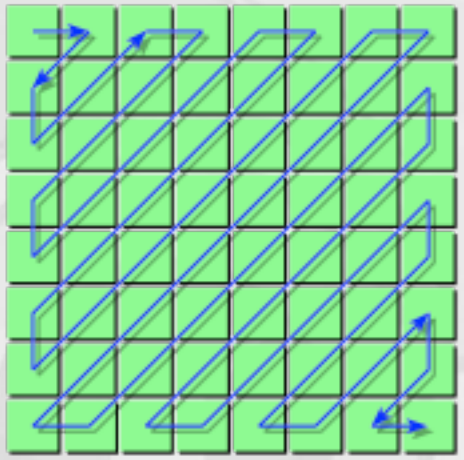
→ 8x8 image blocks

DCT → Quantization

frequency-domain representation



JPEG compression: main idea



RGB Image → YCbCr Color Model

RGB Image → 8x8 image blocks

DCT → Quantization → frequency-domain representation

52	55	61	66	70	61	64	73
63	59	55	90	109	85	69	72
62	59	68	113	144	104	66	73
63	58	71	122	154	106	70	69
67	61	68	104	126	88	68	70
79	65	60	70	77	68	58	75
85	71	64	59	55	61	65	83
87	79	69	68	65	76	78	94

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

−26	−3	−6	2	2	−1	0	0
0	−2	−4	1	1	0	0	0
−3	1	5	−1	−1	0	0	0
−4	1	2	−1	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

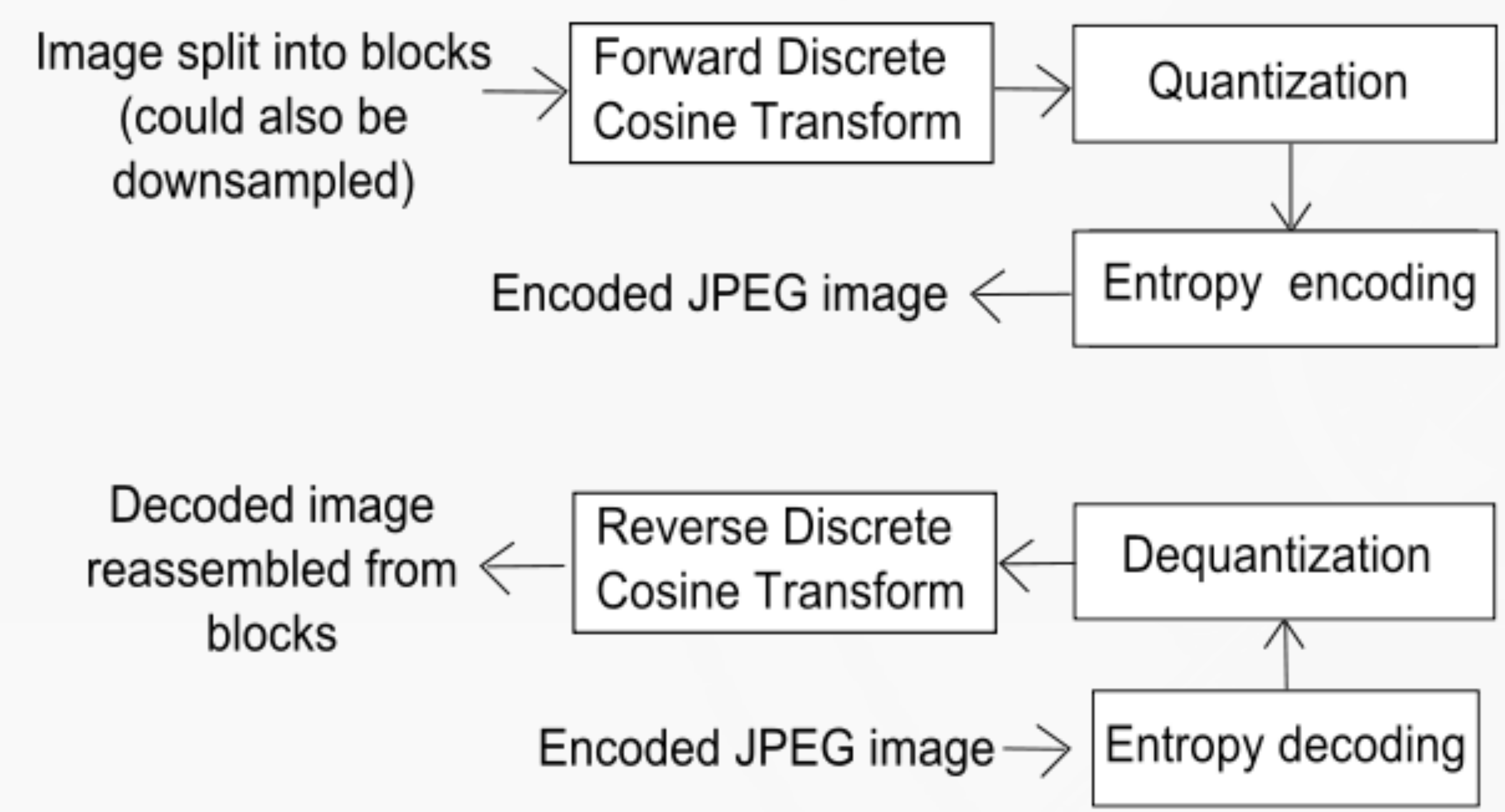
Compression Rate

bpp: bit per pixel

Upper-left: The original image.
Upper-right: Decoded at 0.5 bpp (PSNR: 35.32 dB).
Lower-left: 1.0 bpp (PSNR: 38.73 dB).
Lower-right: 1.5 bpp (PSNR: 41.62 dB).

reference: http://cobweb.ecn.purdue.edu/~ace/color-wavelet/cwavelet.html

JPEG compression: implementation



Common image formats - GIF

- Graphics Interchange Format
  - UNISYS Corporation and Compuserve
  - Lempel-Ziv-Welch compression method
  - GIF87／GIF89a
  - Features
    - Only support 8-bit (256) color image
    - Support several animation effects
    - Support interlaced image coding

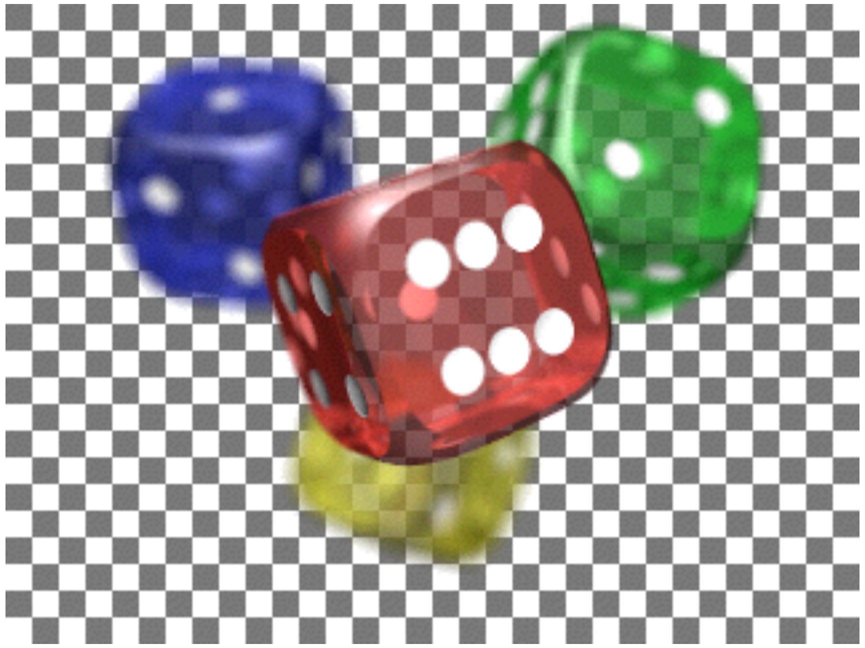
Common image formats - GIF

- Graphics Interchange Format
  - UNISYS Corporation and Compuserve
  - Lempel-Ziv-Welch compression method
  - GIF87／GIF89a

  - Features
    - Only support 8-bit (256) color image
    - Support several animation effects
    - Support interlaced image coding

Common image formats - PNG

- Portable Network Graphics
  - motivation：Compuserv owns the LZW coding patent for GIF images
  - open source
  - Transparent
  - PNG64



Common image formats - JPEG

- Lossy to lossless editing

Common image formats - TIFF (6.0)

- Tagged Image File Format
  - –flexible and adaptable
  - –handling images and data within a single file
  - –header tags: size, definition, image-data arrangement, applied image compression
  - –defining the image's geometry.

Common image formats - TIFF (6.0)

- a TIFF can be a container file
  - –compressed JPEG and RLE
  - –lossless compression
- include a vector-based Clipping path (outlines, cropping, image frames)

DNG: Digital Negative (数字负片)

- a royalty free **RAW** image format
- design by Adobe
- based on TIFF/EP
- mandates use of metadata

Summary – Essential factors of image storage

- Resolution
- Compression rate
  - 1bpp,2bpp, …
  - Compression methods
- Color representation
  - RGB, YUV, Lab …

Image converting tools

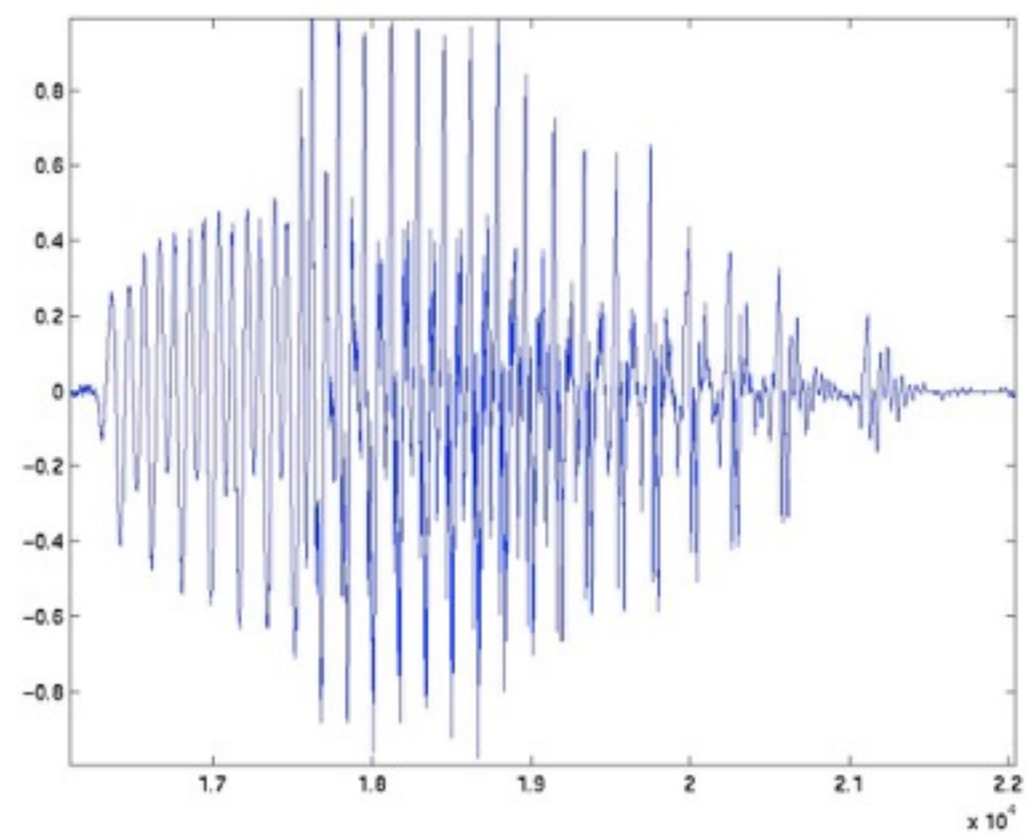
- ACDSEE
- imagemagik (Linux)
- XnView
  - http://perso.orange.fr/pierre.g/

2.2. Audio formats and compression methods

音频编码

Digitalized audio / sound

- What is sound?
  - –Knowing from ear?!?
  - –Sound wave ?!?



- Digitalization
  - –Analog signal → digital signal
  - –Quantization

Bit rate and bit

- a kind of energy wave.
- a continuous function of wave amplitude
  - Sequence is related to the X axis (the time line).
  - Amplitude is related to the Y axis.

Higher coding rate and quantization rate, better sampling quality

- discretely sampled during the digital coding period
  - Bit rate: number of samples obtained in one second
    - The highest frequency ~ 20kHz.
    - 40k samples per second (Nyquest theorem)
    - The bit rate of CD is 44.1kHz
  - Quantization rate: must be the power of 2.
    - The quantization rate of audio CD is normally 16bit.

Audio compression: lossless vs. lossy

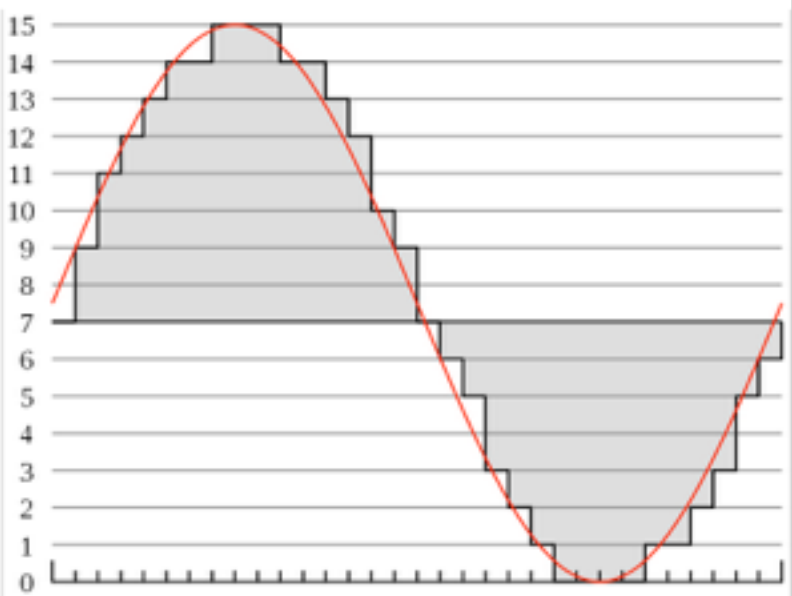
- There is no absolute lossless coding schemes!
  - –According to the definitions of bit rate and quantization rate, audio coding can only approximate to the natural sound signal as much as possible.
  - –Comparing with natural signal, all coding schemes are lossy.

- Related lossless scheme: PCM
  - –PCM can reach the highest preserving level.
  - –Widely applied in raw data saving and music data, e.g. CD, DVD and WAV files.
  - –PCM is viewed as a lossless coding scheme. However, PCM only approximate to the raw data.
  - –Comparing with the PCM coding method, we usually put MP3 coding methods into the lossy audio encoding methods.

PCM coding

- PCM - **P**ulse **C**ode **M**odulation

- PCM coding
  - –Advantage： good play back quality.
  - –Shortage: large storage space.



Sampling and quantization of a signal (red) for 4-bit PCM

- Audio CD mainly leverage the PCM coding scheme.
  - –One CD can store 72 minutes music.

PCM audio stream bit-rate

- Formula
  - –Bit rate × Quantization rate × number of sound channels (bps)。
- EXAMPLE：
- WAV file: bit rate 44.1KHz，quantization rate 16bit，stereo sound.
  - –Coding rate: 44.1K×16×2 =1411.2 Kbps.
  - –128K MP3 ~ 1411.2 K bits per second
  - –also called data width, similar to the concept of band width used in network transfer.
  - –Data speed: transferred bytes per second, = Bit rate / 8. In this example, the speed is 176.4KB/s.
  - –It takes space of 176.4KB per second. Recording 1 minute music requires 10.34M.

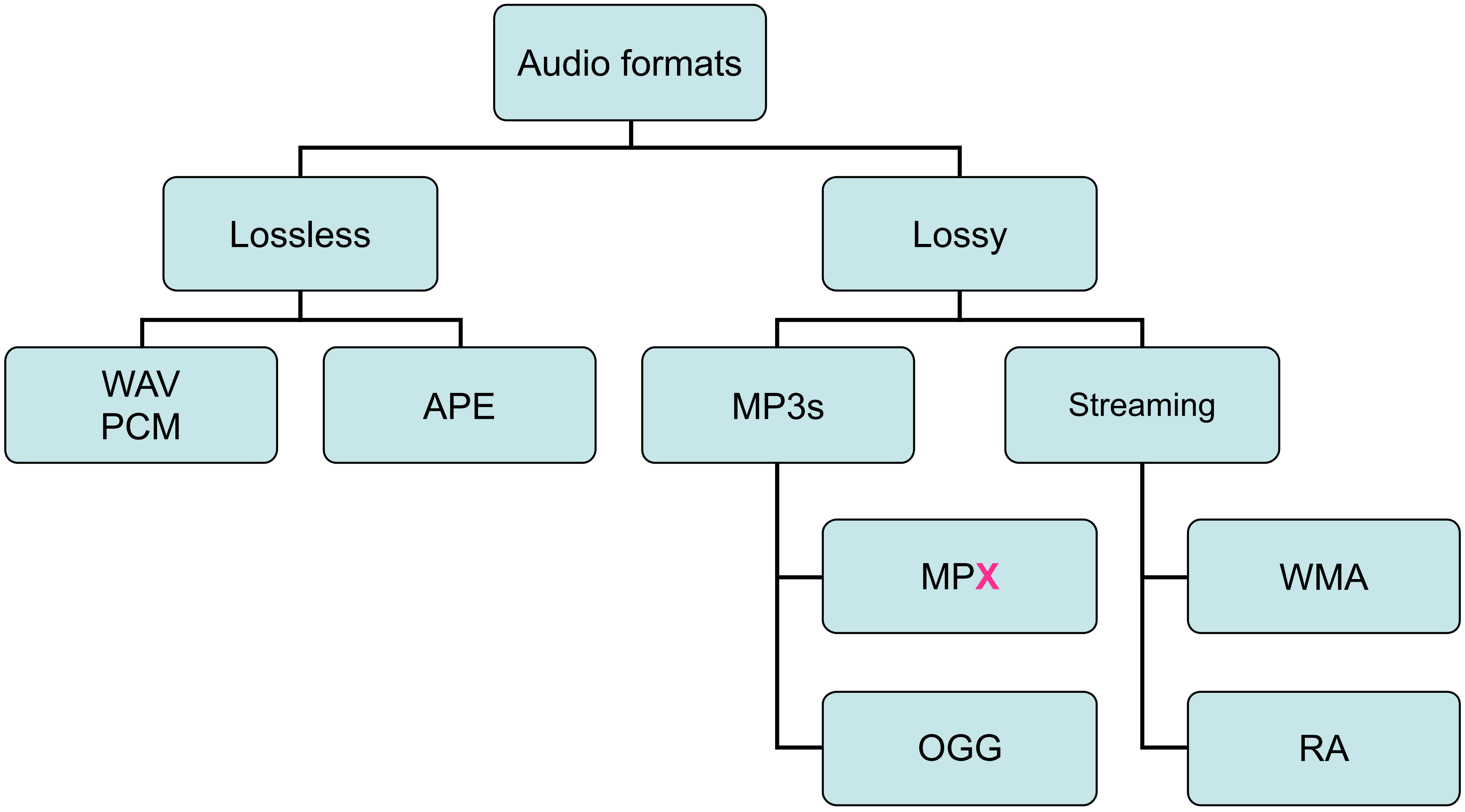
The streaming feature of audio

- The blooming of network => play on-line music.
  - –play the music meanwhile downloading.
    - Recent techniques are easy to archive this goal.

- Based on this feature, it is easy to implement:
  - –on-line direct-show
  - –DIY digital broad casting.

Common audio formats

- WAV
- MP3
- WMA
- RA
- OGG
- APE



WAV

- Developed by Microsoft
- WAV format is based on RIFF (Resource Interchange File Format) standard.
  - All WAV files have a file head which is used to record coding parameters of audio stream.
  - WAV file have no specific constraints on coding audio stream. Besides PCM, WAV can use any types of coding schemes defined by ACM.
- In Windows，PCM based WAV format is recognized as a most useful audio format.
  - WAV is good for music creating and editing, and for saving raw music data.
  - PCM based WAV file is now employed as an intermediate format for convert over different type of audio data, e.g., MP3 to WMA.

WMA

- WMA is created the Windows Media Audio coding framework, developed by Microsoft.

- WMA is designed to used for network transfer. Its main competitors are products from Real Networks.
  - –Microsoft claimed that WMA can reach the sound quality of CD in 64kbps bit rate.
  - –Provides Windows Media Rights Manager to prevent illegal copies and to count play times.
  - –Supports stream techniques and online broadcasting.

RA

- RA (RealAudio) is proposed by RealNetworks Inc.
- In network application, many music site use RealAudio for online playing.

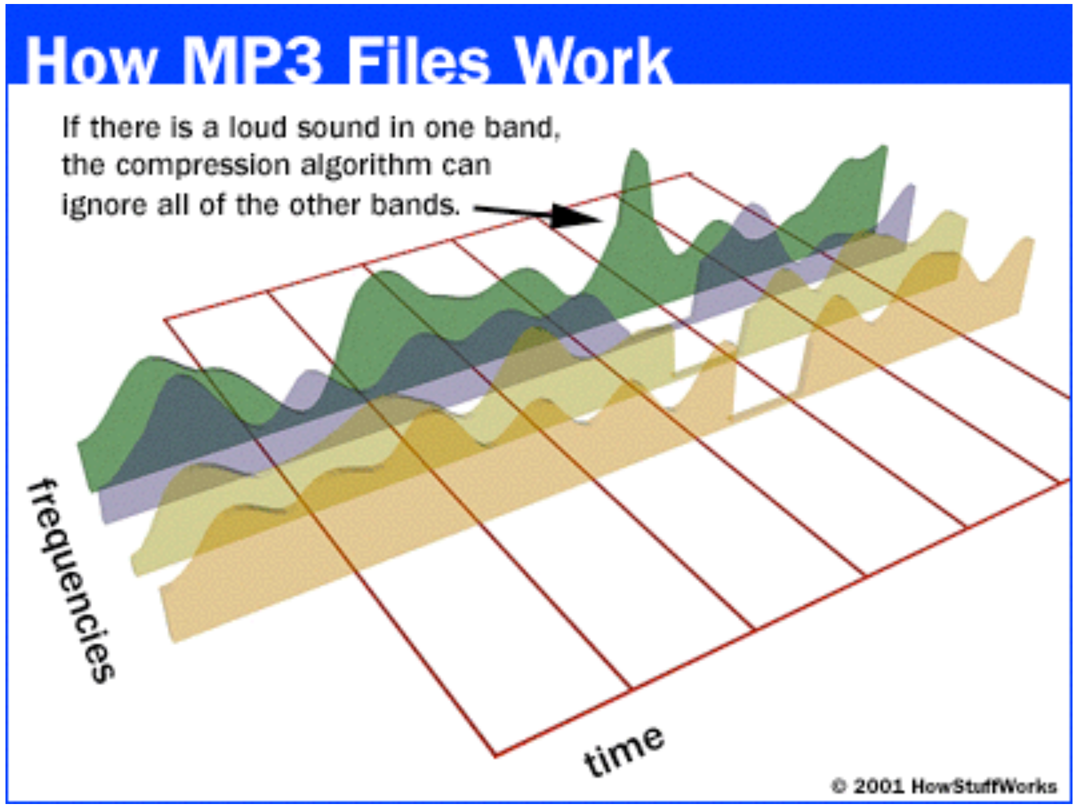
- RA mainly focus on network media market
  - Highlight: RA can alter its own coding bit rate due to the network width but keep the sound quality as much as possible.
  - RA can support many types of audio coding schemes, e.g., ATRAC3.
  - Beside the function of download-while-play, RA can also hide true internet address of sound file. It is quite useful for Music company

APE

- APE is a looseness compression format proposed by Monkey‘s Audio.
- They mainly used LZW as the compression kernel.
- High compression ratio but fast compression speed.
  – Used by many music fans to record CD and share music resources.
- Monkey‘s Audio provides a set plug-ins for different types of media players.

MP3

- From the MPEG-3 standard
- Most popular audio file format
- Special compression method for sound



perceptual noise shaping

OGG

- OGG is a huge project plan of multimedia R&D and is mainly focus on video/audio coding.
  - The total OGG project is open source and free
- Ogg Vorbis audio coding
  - Comparing with MP3, it provides lower bit rate but better play back quality.
  - Support more channels than MP3. It is suitable for recoding classical music.
  - Flexible audio coding framework

什么是MIDI

- MIDI (Musical Instrument Digital Interface即乐器数字化接口) is an international standard for general interface.
  - –It provides a set of standard interface for transferring data among different types of devices. MIDI devices shall precisely send MIDI messages.

- Wildly use in music creation, game background music and ring tone of mobile phones.

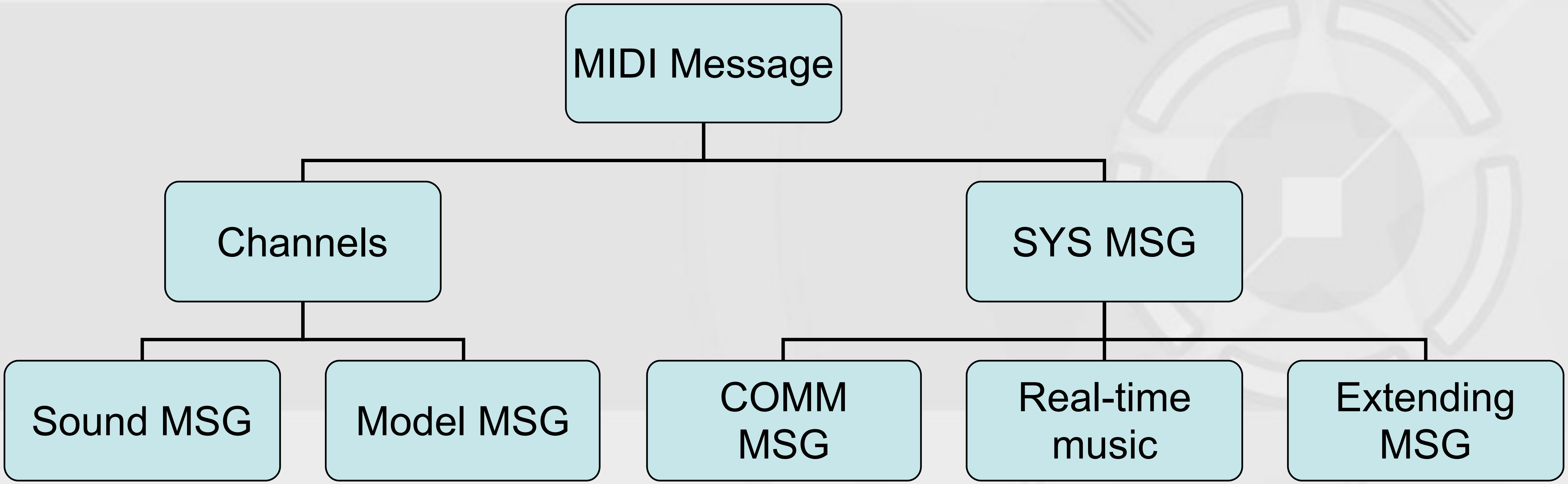
MIDI概况

- MIDI is type of description language.
  - –Different directly record digitalized sound signal
  - –Only record ‘events’ that how instruments make sound.
  - –Small storage size.
- Three elements of MIDI
  - –Synthesizer
    - Generate sound and can control the length, height, strength and other features of sound.
  - –Sequencer
    - Devices or software that store and modify MIDI information.
  - –MIDI device
    - Do not generate any sound but a sequence of MIDI commands.
    - E.g. MIDI keyboard, MIDI harp, MIDI guitar, and MIDI violin, etc.

Basic concepts of MIDI

- [Track]
  - Music is composed with several music channels.
- [Channel]
  - Each MIDI device corresponds to a channels. Each channel owns its own message sequence. Up to 16 channels
- [Voice]
  - Each channel allows multiple voice, e.g., chords when playing piano. (*Timbre* means the sum of sound in one channels)
- [Polyphony]
  - The sum of sound can be generated by Synthesizer in one moment.
- [Patch]
  - Sound feature setting up to simulate specific instrument.

Message structure of MIDI



Common MIDI file format

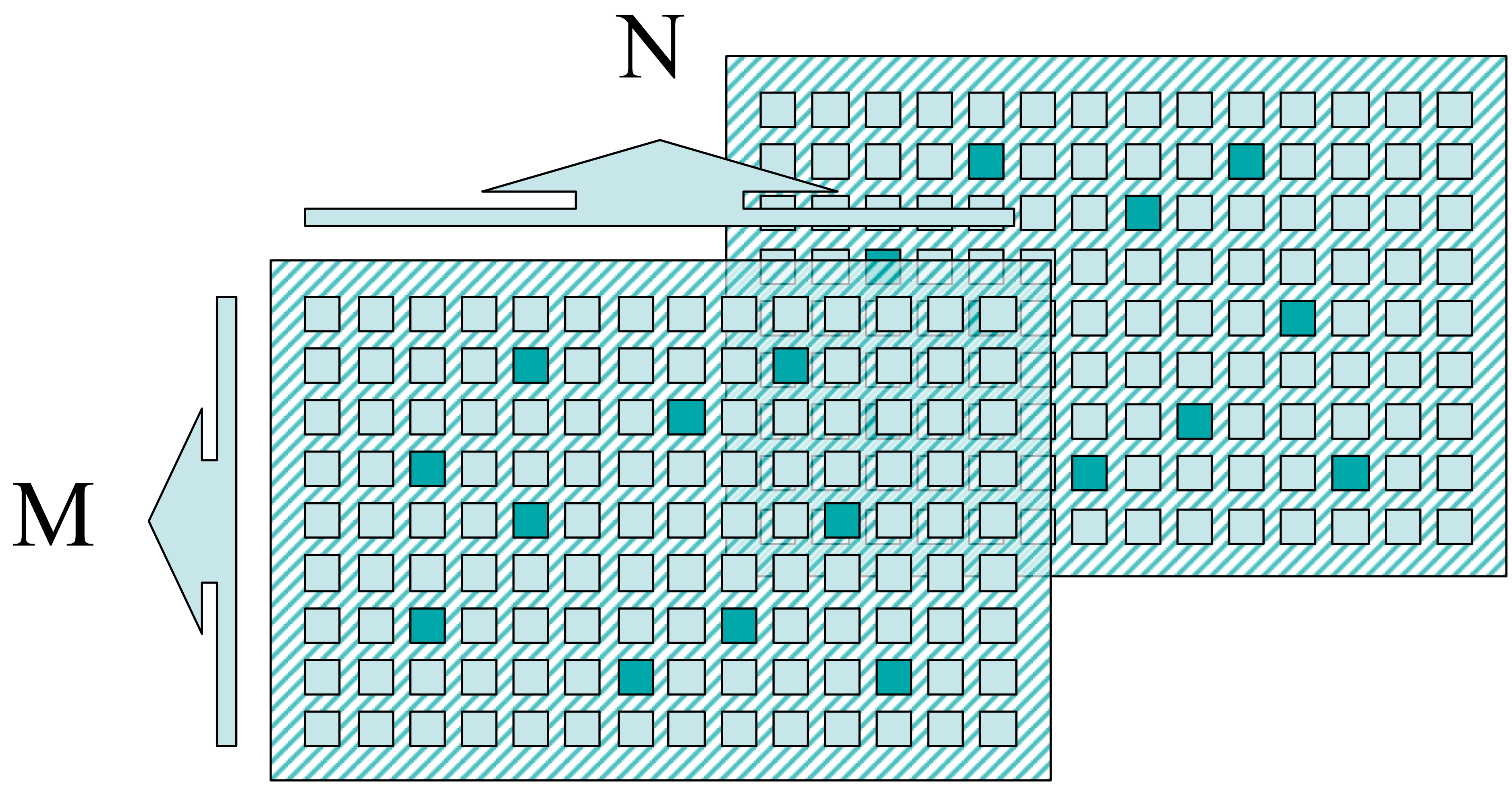
- MID
  - –General MIDI
- SMF
  - –Standard MIDI File

2.3. Video formats and coding methods

视频编码

Representations of video

- Sequence of images ?!?
  - Can be viewed as a 3-dimensional matrix
  - But it is only 50% correct



Common video formats

- AVI (Microsoft, Divx, …)
  - –avi, wmv, asf
- RM (Realplayer)
  - –rm, rmvb
- MOV (Quicktime)
  - –mov
- MPEG
  - –MPEG-1, MPEG-2, MPEG-4 …

http://www.bigbuckbunny.org/index.php/download/

Common video formats - AVI

- AVI = Audio Video Interleaved (By Microsoft)
  - A digital audio/video format according to the RIFF file format standard.
  - multimedia CDROM, store video information, movie and TV program,
  - Internet applications, download and online viewing

- Allows storing audio and video information interlaced
- But play back simultaneously

Common video formats - AVI

- AVI only defines the standard on control interface.
  - No limitation of compression approach in AVI file format
  - Supports 256 colors and RLE compression
  - AVI with specific encoding methods must be played back by matched decoding methods.
  - Many companies provide their own codecs
    - e.g., SONY

Common video formats - RM

- RM (RealVideo file): a new file format for streaming video by RealNetworks Inc.
- RealVideo techniques is used to broadcast important events over Internet.
- RealMedia: A audio/video compression standard of RealNetworks
  - Mainly used in wide range network to transform real-time video sequence in low bit rate.
  - It can alter different bit rate depends on network data transformation rate
- RealVideo can be used with RealServer. Different from most other video formats, RM can be played back while the data is downloading.

Common video formats - MOV

- A video/audio format developed by Apple Inc.

- QuickTime™ player
  – Apple Mac OS、Microsoft Windows System
- The original format supports
  – 256 color, RLE, and JPEG compression techniques.

Common video formats - MOV

- Advanced function features

  > 150 kinds of Video effects

  > 200 kinds of MIDI devices sounds.

- Internet-oriented features
  - –digitalized information stream,
  - –workflow, and
  - –play-back functions through internet.

Common video formats - MOV

- QuickTime VR (QTVR):
  - –a set of Virtual Reality (虚拟现实) techniques used in QuickTime.
  - –use mouse or keyboard
    - investigate 360 degree of scene
    - browse an object from a specific spatial angle interactively.

Video compression standards

- MPEG standards
  - Audio/Video compression, storage and play back standards
  - MPEG-1: VCD
  - MPEG-2: broadcast TV, e.g., DVD、HDTV etc.
  - MPEG-3: replaced by MPEG-2
  - MPEG-4: network video transfer，stream media
  - MPEG-7:
  - MPEG-21:

- ITU-T H.26x series

Video compression standards

- ITU-T H.26x series
  - Mainly used in video communication applications
  - Now it has H.261， H.262， H.263， H.264

  - ISDN network based H.320 standards
    - the video compression part: H.261,H.262 and H.263

  - LAN network based H.323
  - PSTN network based H.324
    - the video compression part: H.261 and H.263

MPEG概况

- MPEG = **M**otion **P**icture **E**xpert **G**roup

- **ISO/IEC JTC1/SC29**
  - WG11:Motion Picture Experts Group (MPEG)
  - WG10:Joint Photographic Experts Group (JPEG)
  - WG7: Computer Graphics Experts Group (CGEG)
  - WG9: Joint Bi-level Image coding experts Group (JBIG)
  - WG12: Multimedia and Hypermedia information coding Experts Group (MHEG)

MPEG概况

- MPEG-1,2 standards were started at 1988
  - 需求 [Requirement]
  - 系统 [System]
  - 视频 [Video]
  - 音频 [Audio]
  - 实现 [Implementation]
  - 测试 [Testing]

- Newest MPEG standards: MPEG-4, MPEG-7, MPEG-21

MPEG-1 Standard ISO/IEC 11172-2 (1991)

"Coding of moving pictures and associated audio for digital storage media"

- **Video**
  - –optimized for bit rates around 1.5 Mbit/s
  - –originally optimized for SIF picture format,
  - –but not limited to it:
    - •[**NTSC based**] : 352x240 pixels at 30 frames/sec
    - •[**PAL based**] : 352x288 pixels at 25 frames/sec
  - –progressive frames only
    - •no direct provision for interlaced video applications, such as broadcast television

MPEG-1 Standard ISO/IEC 11172-2 (1991)

- Audio
  - –joint stereo audio coding at 192 kbit/s (layer 2)
- System
  - –mainly designed for error-free digital storage media
  - –multiplexing of audio, video and data
- Applications
  - –CD-I, digital multimedia, and
  - –video database (e.g. video-on-demand)

MPEG-2 Standard ISO/IEC 13818-2 (1994)

- **Video**
  - –2-15 or 16-80 Mbit/s bit rate (target bit rate: 4…9 Mbit/sec)
  - –TV and HDTV picture formats
  - –Supports interlaced material
  - –MPEG-2 consists of *profiles* (类) and *levels* （级）
    - Main Profile, Main Level (MP@ML)
      - –720x480 resolution video at 30 frames/sec
      - – < 15 Mbit/sec (typical ~4 Mbit/sec)
      - –for NTSC video
    - Main Profile, High Level (MP@HL)
      - –1920x1152 resolution video at 30 frames/sec
      - –< 80 Mbit/sec (typical ~15 Mbit/sec)
      - –HDTV

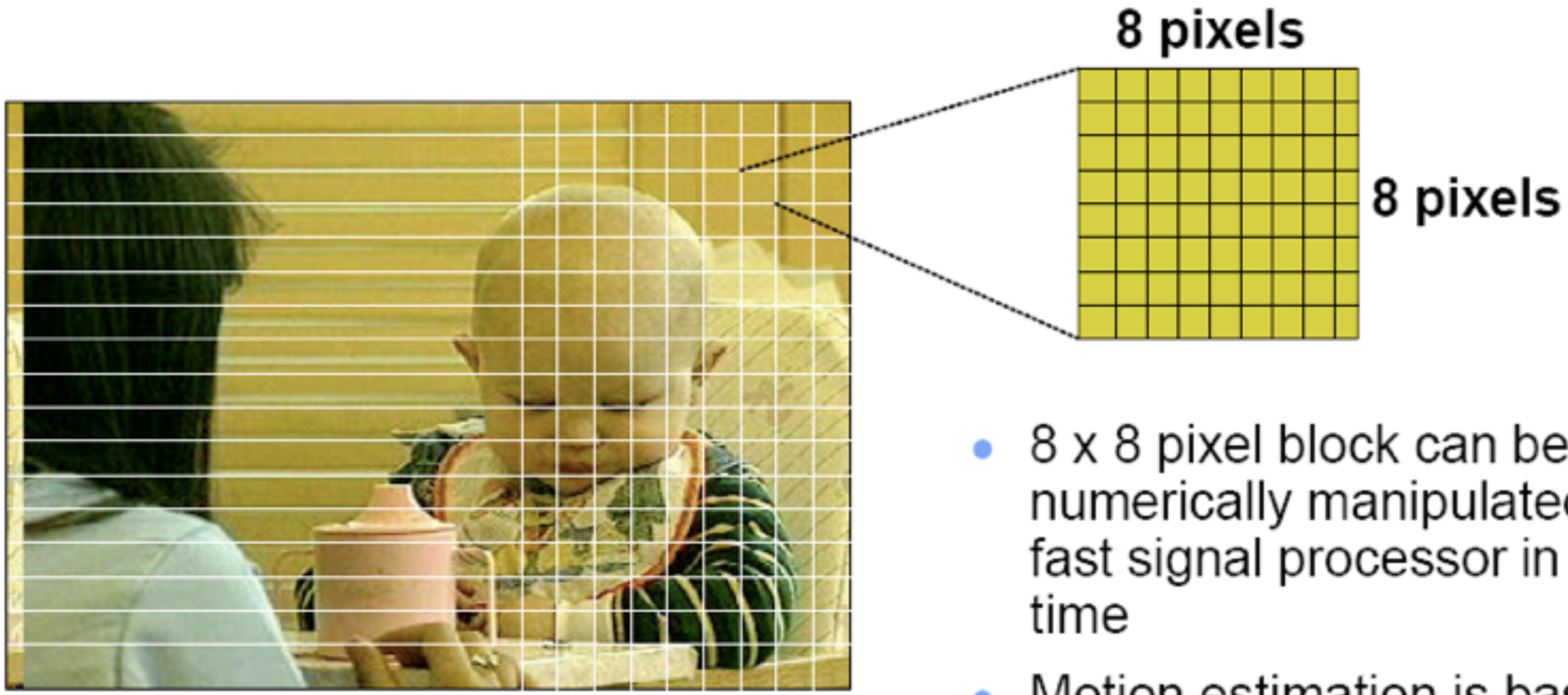
MPEG-2 Standard ISO/IEC 13818-2 (1994)

- Audio
  - –compatible multichannel extension of MPEG-1 audio

- System
  - –video, audio and data multiplexing defines tow presentations:
    - ***Program Stream*** for applications using near error free media
    - ***Transport Stream*** for more error prone channels

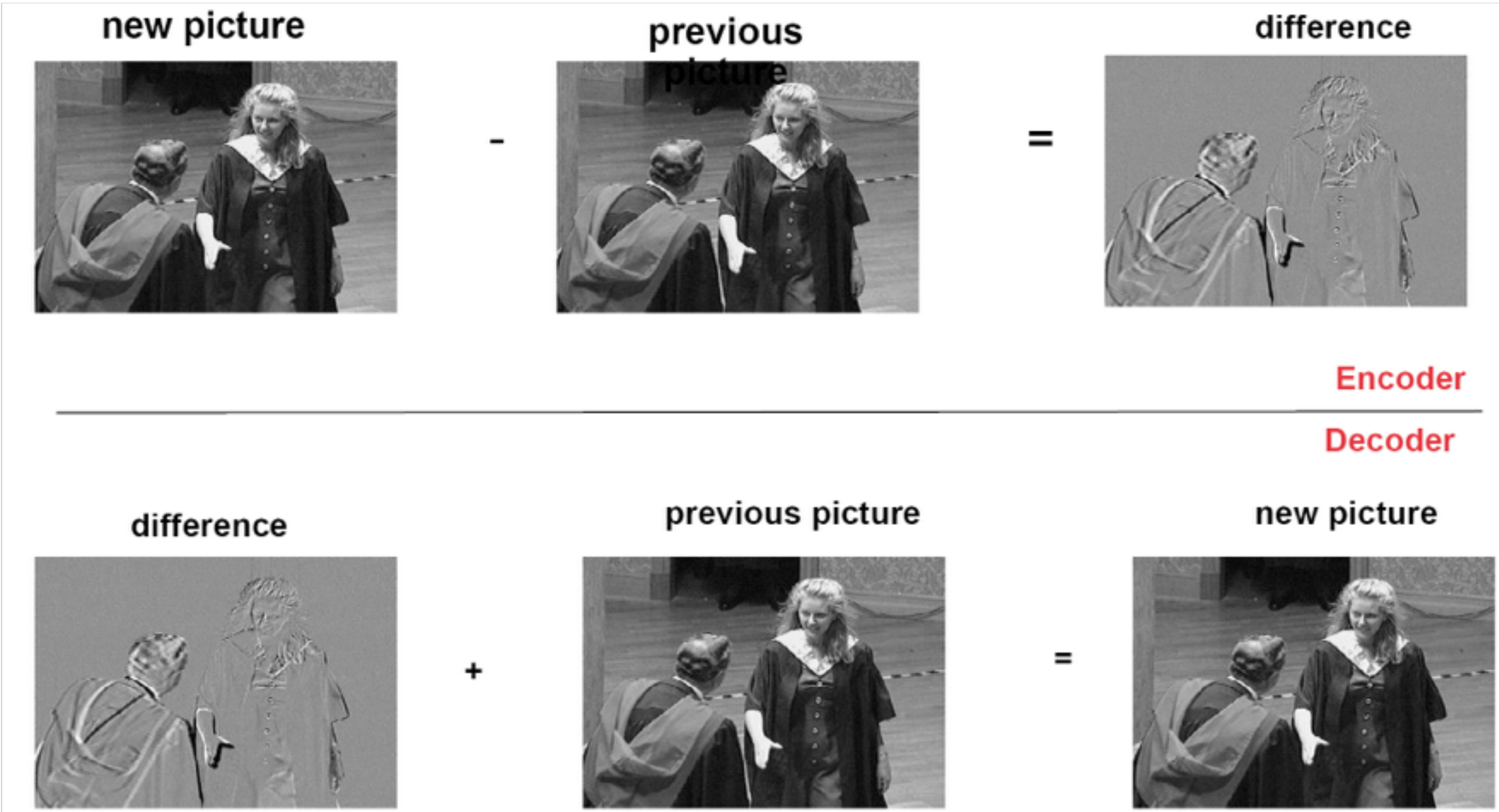
- Applications
  - –satellite, cable, and terrestrial broadcasting,
  - –digital networks, and
  - –digital VCR

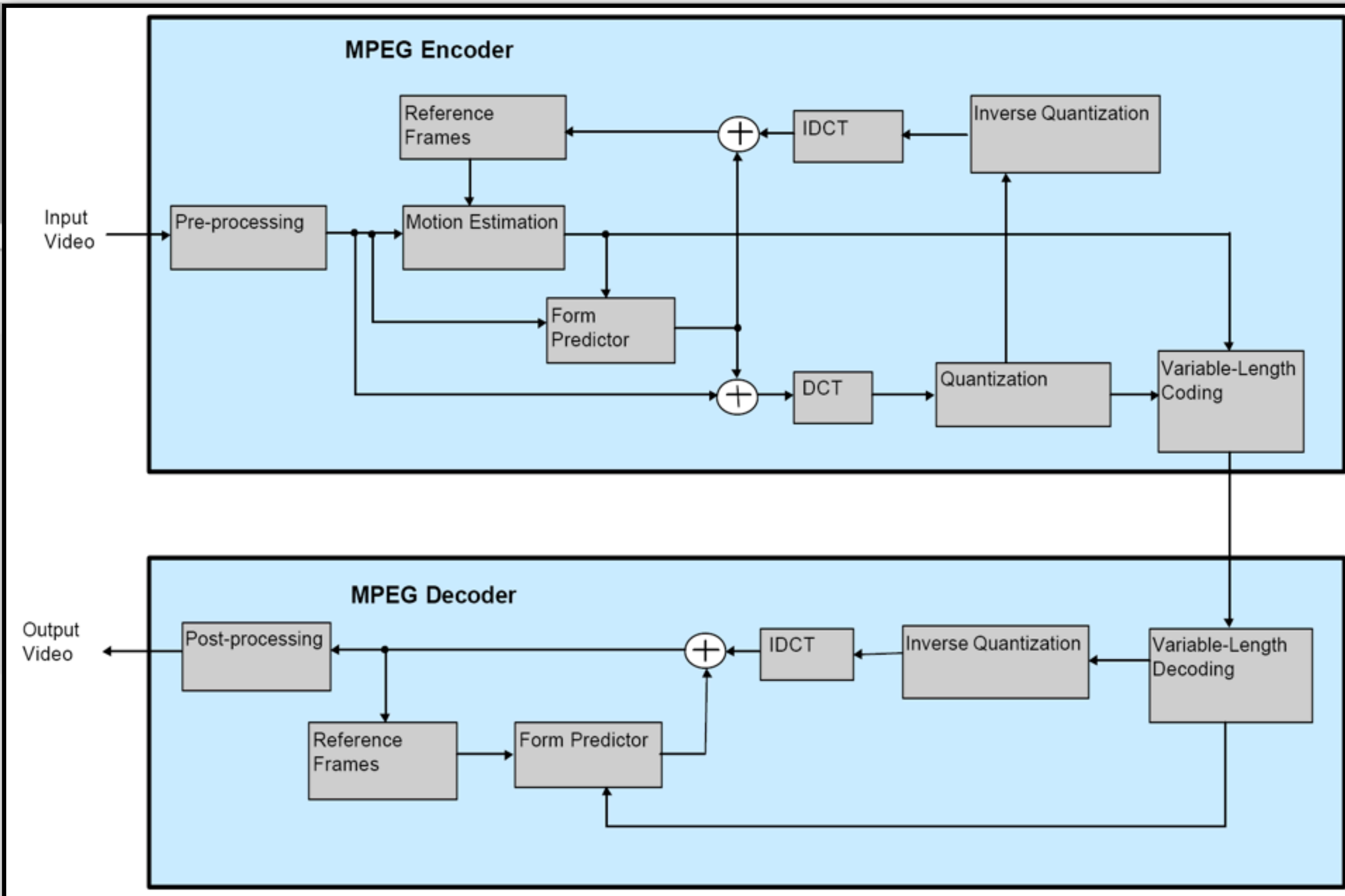
MPEG compression is based on 8 x 8 pixel block processing



- 8 x 8 pixel block can be numerically manipulated by fast signal processor in real time
- Motion estimation is based on comparing the blocks between series of pictures

MPEG: only compress moving parts

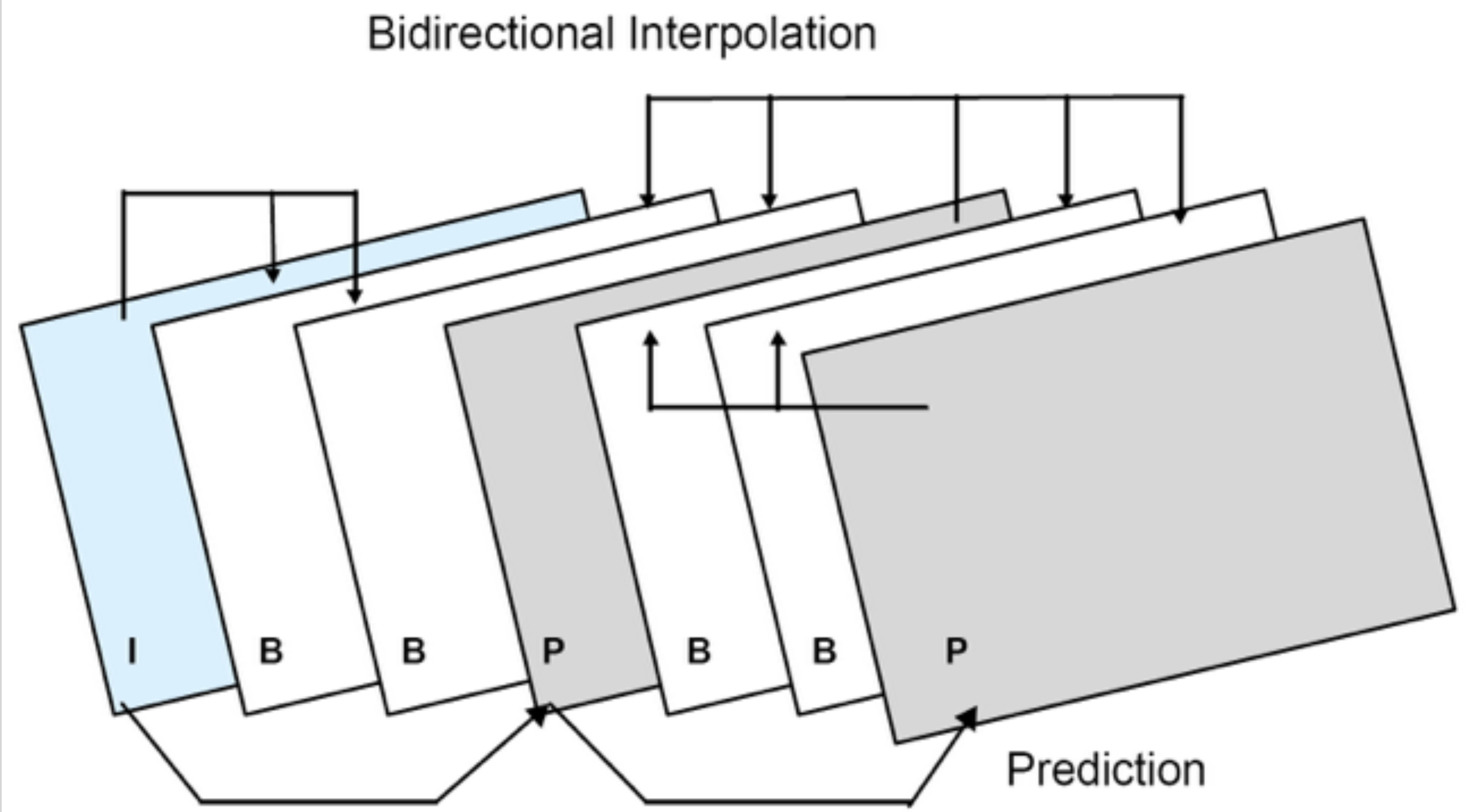




MPEG encoding and decoding

MPEG: motion compression

I = Intra-Frame
P = Predicted frame
B = Bi-directionally interpolated frame



Typical sequence: IBBPBBPBBI

Video signal: stream of picture, it is not necessary to send every picture

- Whole picture is needed only when all the content is changed!
- Several pictures has to be buffered to memory to make prediction forward and backward

MPEG: other issues

- Motion compensating
- Intra-frame transfer order

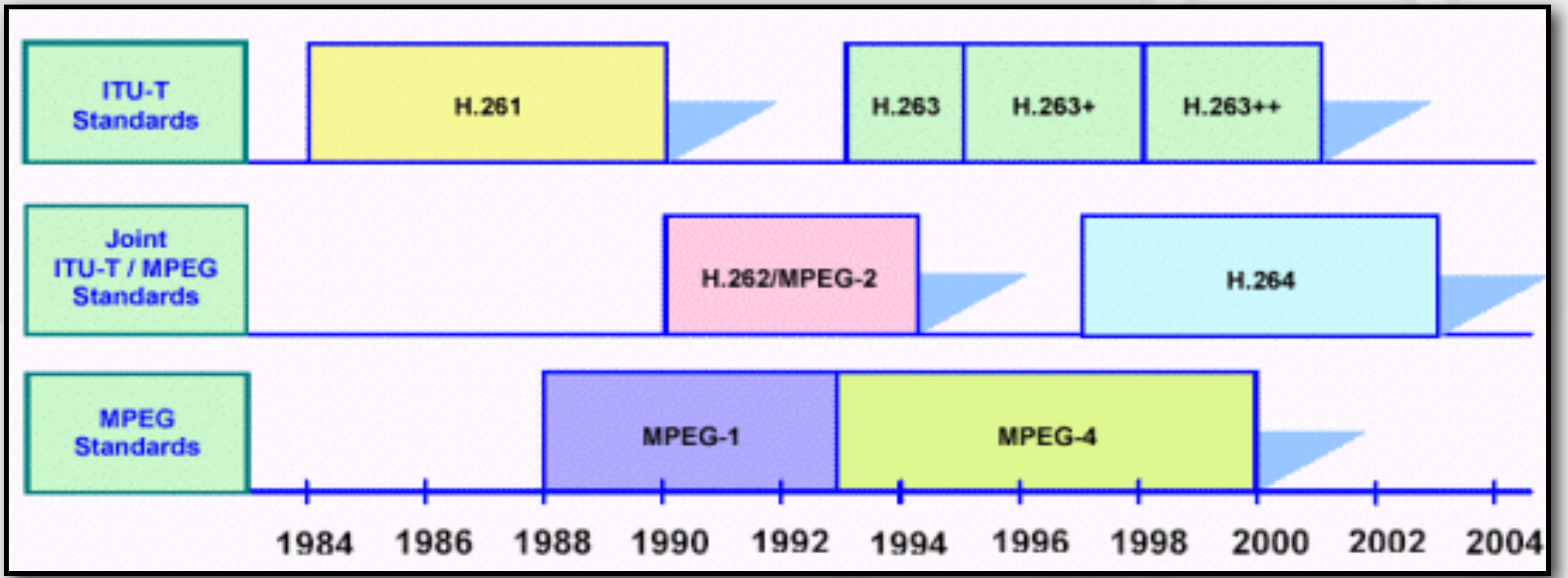
Color video coding

- 抽样和编码整个模拟（彩色）视频信号
  - 例如，复合编码
- 对亮度和色度分别编码
  - 例如，组件编码
  - 亮度比色度更重要，可根据应用场合采用4:2:2，4:2:0, 4:4:4等不同的编码比率

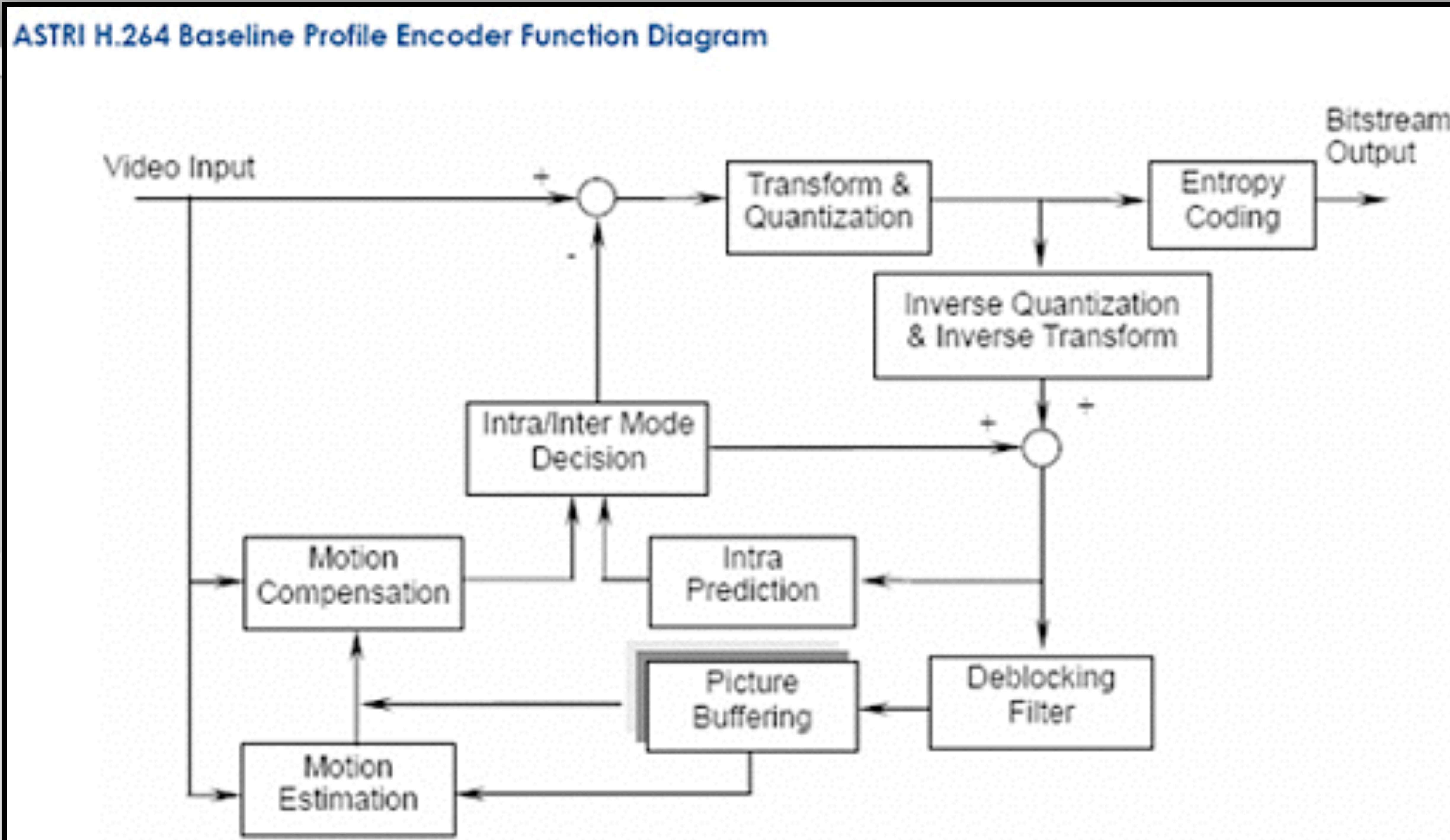
Overview of H.264

- JVT（Joint Video Team）
  - –founded on December 2001, Pattaya Thailand.
  - –video coding specialists from ITU-T and ISO, the two international standards organizations
  - –goal: define a new video coding standards to achieve high compression rate, high image quality, good network adaptive coding frame.
- H.264: A new video compression standard
  - –accepted by ITU-T
  - –accepted by ISO
    - called AVC (Advanced Video Coding) standard
    - as the 10th part of MPEG-4

Major history of digital video standard



H.264 coding principle



浙江大学计算机学院
数字媒体与网络技术

H.264的主要技术特点

1. 4类DCT整数变换以及相应的量化方法

2. 7种宏块预测模式
   - 16×16, 16×8, 8×16, 8×8, 8×4, 4×8, 4×4
   - 运动估计和补偿更加精确

3. 多参考帧
4. 帧内预测
5. 改进的去块效应滤波器（Deblocking filter）
6. 增强的熵编码方法
   - UVLC（Universal VLC）、CAVLC（Context adaptive VLC）和CABAC

7. 1/4像素插值
8. 宏块级逐行、隔行自适应编码MBAFF

Advantages and shortages of H.264

☑ High compression rate

– In the same image quality, H.264 can be compressed as size of
  - 36% of MPEG-2， 61% of MPEG-4 ， 51% of H.263

– Low bit stream, high quality

☑ High error correctness rate

– H.264 provides necessary tools to solve the error coding problem in unstable network environments

☑ Network adaptation

– H.264 provides Network Adaptation Layer so as to make files of H.264c can be easily transferred in different network environments.

☐ High computation price

– In the same image quality, H.264 is twice of MPEG-2 in computation complexity.

Applications of H.264

- H.264 standards added a NAL（Network Abstraction Layer）
  - to face the network connection and interface problem in the real applications.
- video communication
  - In real-time communication, POLYCOM、TANDBERG、VCON、SONY etc. claimed their own H.264 based TV-meeting products.
- digital TV broadcasting
  - MPEG has already finished defining the MPEG-2 compatible standard on H.264 stream coding content
- video storage-and-play-back
  - For High resolution DVD（HD DVD）application, H.264/MPEG-4 AVC solution.

Summary of video coding

- Resolution
- Coding rate
- Motion coding
- Transfer performance

2.4. HTML and XML

结构化文档概览

Overview of HTML

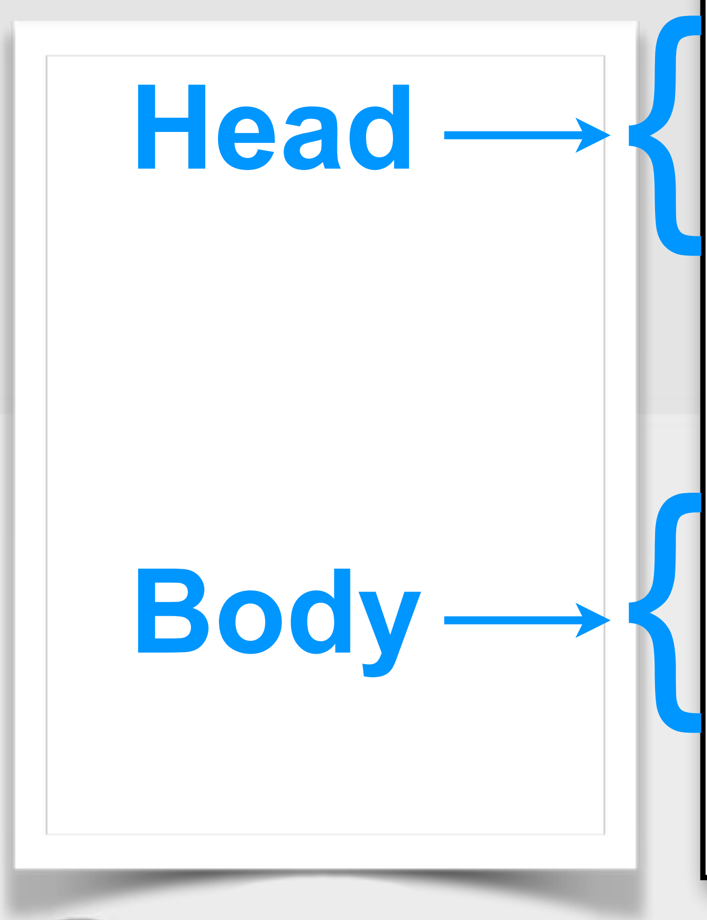
- Hypertext Markup Language
  - Developed by Tim Berners-Lee
    - lightweight markup language vs. complex SGML.
  - Based on pure text format

  - Rich abilities to display multimedia information.
    - Later added tags to support image and videos.

  - HTML 3.2 => HTML 4.0 => HTML 5.0
    - Different browser has their own display effects.

Overview of all HTML elements

Reference: http://htmlhelp.com/reference/wilbur/overview.html




```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
    <title>Apple中国</title>
    <meta http-equiv="content-type" content="text/html;
charset=gb2312">
    ...
</head>

<body>
<!-- Tag for Activity Group: General, Activity: Apple China -
Homepage -->
...
</body>
</html>
```

http://www.apple.com.cn/

Overview of HTML - Head elements

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
     <title>Apple中国</title>
     <meta http-equiv="content-type" content="text/html; charset=gb2312">
     ...
     <link rel="home" href="http://www.apple.com/">
     ...
     <script src="http://images.apple.com/global/scripts/lib/prototype.js" type="text/javascript" charset="utf-8">
     </script>
     ...

     <style type="text/css" media="all">
     ...
     #billboard { width: 1100px; margin: 0 auto 15px; overflow: hidden; position: relative; }
     #ticker { margin-bottom: 15px; }
     #homefooter { margin: 60px auto 50px; }
     ...
     </style>
</head>
```

- TITLE - Document title
- ISINDEX - Primitive search
- META - Meta-information

- LINK - Site structure
- BASE - Document location
- SCRIPT - Inline script
- STYLE - Style information

Overview of HTML - Body elements

```
<html>
<head> ... </head>
<body>
    <H1> Hello, world </H1>
    <P> Digital Asset management is cool! </P>

</body>
</html>
```

- **Block level elements**
  - Headings: H1 => H6
  - Lists: UL, OL, DIR, MENU, LI, DL, DT, DD
  - Text Containers: P, PRE, BLOCKQUOTE, ADDRESS
  - others: DIV, CENTER, FORM, HR, TABLE

Overview of HTML - Body elements

```
<html>
<head> ... </head>
<body>
    <H1> Text-level elements </H1>
    <A href="http://www.google.com"> GOOGLE  <IMG src=" ... ">  </A>

</body>
</html>
```

- **Text-level elements**
  - –Logical markup: EM ...
  - –Special markup: A, IMG, APPLET ...
  - –Physical markup: B, ...
  - –Forms: INPUT ...
  - –Tables: CAPTION, TR, TH, TD

About CSS
叠样式表 (Cascading Style Sheets)

- http://www.w3school.com.cn/css/css_intro.asp

- 样式定义如何显示 HTML 元素
- 样式通常存储在样式表中（*.css）
- 把样式添加到 HTML 4.0 中，是为了解决内容与表现分离的问题
- 外部样式表可以极大提高工作效率
- 外部样式表通常存储在 CSS 文件中
- 多个样式定义可层叠为一个

HTML 5.0

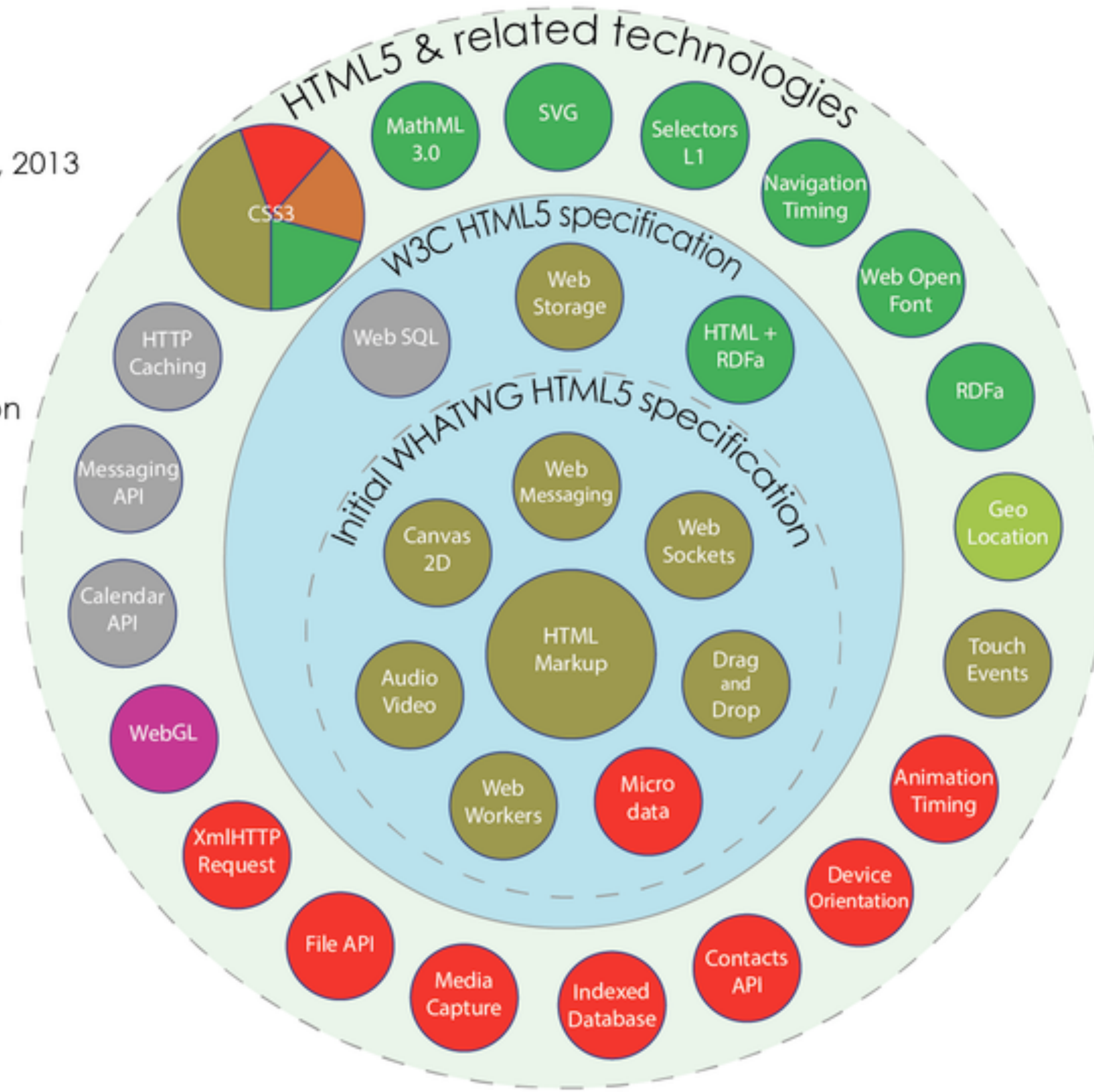
- 学习参考：http://www.w3school.com.cn/html5/index.asp
- 实例解释：http://directguo.com/html5
  - http://html5-slide-template.googlecode.com/svn/trunk/html5-slide-template.html#slide1

- 一套Web富客户端开发的工业标准
  - 许多新特性：内建的视频、音频标记，元素拖放功能
  - 最新的 Safari、Chrome、Firefox 以及 Opera 支持某些 HTML5 特性，Internet Explorer 也已经逐步支持

HTML5

Taxonomy & Status on January 20, 2013

- W3C Recommendation
- Proposed Recommendation
- Candidate Recommendation
- Last Call
- Working Draft
- Non-W3C Specifications
- Deprecated



About JavaScript

- http://www.cad.zju.edu.cn/home/zhx/DAM/2014/doku.php?id=js
- JavaScript

Using JavaScript

- What is JavaScript

- Web programing with JS
  - –Node.JS
  - –Sails
  - –Yeoman: http://yeoman.io/codelab/setup.html

  - –…

Overview of XML

- Extensible Markup Language
  - Aim at data searching

  – Similar to HTML
    - More restrict grammar checking
    - User defined tags to describe data structure
    - Flexible data displaying schemes
    - Cross-platform, language and application independent
    - DTD and XML Schema.

- http://www.brics.dk/~amoeller/XML/overview.html

XML Example

```xml
<recipe id="117" category="dessert">
  <title>Rhubarb Cobbler</title>
  <author><email>Maggie.Herrick@bbs.mhv.net</email></author>
  <date>Wed, 14 Jun 95</date>

  <description>
    Rhubarb Cobbler made with bananas as the main sweetener.
    It was delicious.
  </description>

  <ingredients>
    ...
    <item><amount>2 tablespoons</amount><type>sugar</type></item>
    ...
  </ingredients>

  <preparation>
    Combine all and use as cobbler, pie, or crisp.
  </preparation>

  <related url="#GardenQuiche">Garden Quiche</related>
</recipe>
```

HTML v.s. XML

```
<h1>Rhubarb Cobbler</h1>
<h2>Maggie.Herrick@bbs.mhv.net</h2>
<h3>Wed, 14 Jun 95</h3>

Rhubarb Cobbler made with bananas as the main sweetener.
It was delicious.  Basicly it was
  <table>
  <tr><td> 2 1/2 cups <td> diced rhubarb
  <tr><td> 2 tablespoons <td> sugar
  <tr><td> 2 <td> fairly ripe bananas
  <tr><td> 1/4 teaspoon <td> cinnamon
  <tr><td> dash of <td> nutmeg
  </table>
Combine all and use as cobbler, pie, or crisp.
Related recipes: <a href="#GardenQuiche">Garden Quiche</a>
```

```
<recipe id="117" category="dessert">
  <title>Rhubarb Cobbler</title>
  <author><email>Maggie.Herrick@bbs.mhv.net</email></author>
  <date>Wed, 14 Jun 95</date>

  <description>
    Rhubarb Cobbler made with bananas as the main sweetener.
    It was delicious.
  </description>

  <ingredients>
    <item><amount>2 1/2 cups</amount><type>diced rhubarb</type></item>
    <item><amount>2 tablespoons</amount><type>sugar</type></item>
    <item><amount>2</amount><type>fairly ripe bananas</type></item>
    <item><amount>1/4 teaspoon</amount><type>cinnamon</type></item>
    <item><amount>dash of</amount><type>nutmeg</type></item>
  </ingredients>

  <preparation>
    Combine all and use as cobbler, pie, or crisp.
  </preparation>

  <related url="#GardenQuiche">Garden Quiche</related>
</recipe>
```

A conceptual view of XML

```
...<foo attr="val" ...>...</foo>...
```

a matching element end tag

the contents of the element

an attribute with name attr and value val, values enclosed by ' or "

an element start tag with name foo

```
...<foo attr="val" ... />...
```

XML documents as text with markup

A conceptual view of XML

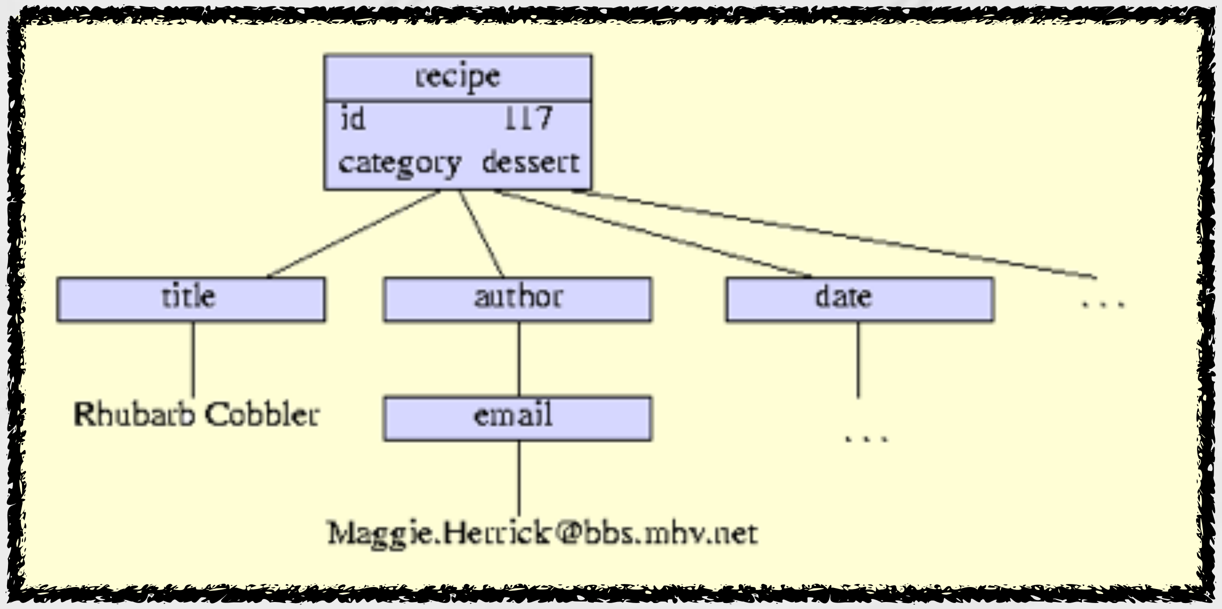
- An XML document is a (Unicode) text with markup tags and other meta-information.
- An XML document must be well-formed:
  - start and end tags must match
  - element tags must be properly nested
  - + some more subtle syntactical requirements

- XML is case sensitive!
- Special characters can be escaped using Unicode character references:
  - < and < both yield <

A conceptual view of XML

- An XML document is an ordered, labeled tree:
  - character data leaf nodes contain the actual data (text strings)
    - usually, character data nodes must be non-empty and non-adjacent to other character data nodes
  - elements nodes, are each labeled with
    - a name (often called the element type), and
    - a set of attributes, each consisting of a name and a value,

XML documents as labeled trees



A conceptual view of XML

- XML trees may contain other kinds of leaf nodes:
  - –processing instructions - annotations for various processors
  - –comments - as in programming languages
  - –document type declaration

XML documents as labeled trees

- The XML vision offers:
  - common extensions to the core XML specification
    - a namespace mechanism, document inclusion, etc.
  - schemas
    - grammars to define classes of documents
  - linking between documents
    - a generalization of HTML anchors and links
  - addressing parts of read-only documents
    - flexible and robust pointers into documents
  - transformation
    - conversion from one document class to another
  - querying
    extraction of information, generalizing relational databases

To use XML

- Define your XML language
    - use XML Schema to define its syntax
- Exploit the generic XML tools
    - XSLT and XQuery processors
- As a generic protocols, and the generic programming frameworks
    - DOM or SAX to build application tools

Summary: HTML and XML

- Both of them are useful today for different applications

JSON

- http://www.w3schools.com/json/
- JavaScript Object NotationJSON Example

- {"employees":[
- {"firstName":"John", "lastName":"Doe"},
- {"firstName":"Anna", "lastName":"Smith"},
- {"firstName":"Peter", "lastName":"Jones"}
-]}

2.5. Graphics formats

Graphics ≠ Images

- Representation ability
  - Graphics are usually described in vectors which can provide arbitrary precision
  - Images are usually sampled in fragments/pixels which can only provide limited precision

- Application area
  - Graphics are mainly applied in CAD, model design, computer animation, system simulation and printing.
  - Images are mainly used for photo display and image processing etc.

Classification of different graphics formats

Graphics	2D Graphics	Printing	PDF, PS,AI
		Animation	Flash, SVG
	3D Graphics	Scene	VRML/X3D SGI Inventor
		Rendering	.pov .rib
		Modeling	.obj .3ds .off .ply .md5 .dwg …

Overview of SVG

- http://www.w3.org/Graphics/SVG/About.html

- 什么是SVG?
  - –SVG 指可伸缩矢量图形 (Scalable Vector Graphics)
  - –SVG 用来定义用于网络的基于矢量的图形
  - –SVG 使用 XML 格式定义图形
  - –SVG 图像在放大或改变尺寸的情况下其图形质量不会有所损失
  - –SVG 是万维网联盟的标准
  - –SVG 与诸如 DOM 和 XSL 之类的 W3C 标准是一个整体

Elements of 3D graphics format

- Global scene description
  - –Parameters of light and camera, other system configurations
- Geometric model description
  - –Curves and surfaces
    - Line, plane, quadratic surface, spline …
  - –Mesh surfaces ＝ vertex coordinates＋ topology connectivity
  - –Texture coordinates, normals
- Material description
  - –Reflectance model, texture image
- Animation description
  - –Skeleton model …

Main problems for 3D graphics format

- CAD and computer animation software
  - –Different application area
  - –Different system design principles
  - –Different types of geometric representation combinations

- Mainstream commercial software employ different types of 3D graphics model.
  - –It is hard to obtain a uniform graphics format.
  - –Data exchange and sharing become key issues for 3D designing system.

Overview of X3D

- X3D [Extensible 3D] is an international standard of 3D graphics. It defines how to integrate and access interactive 3D content in a multimedia environment.

- The former of X3D is VRML which is established on 1998 as a network graphics ISO standard (ISO/IEC14772).

- X3D decompose scene descriptions of VRML97 into components. Therefore it is very convenient to extend original VRML functions by adding new components.

New 3D graphics standard-X3D

- Ten years from VRML to X3D

1994.10 通过VRML1.0 三维文件格式
1996.7 公布VRML2.0 草案加入交互特性
1998.1 通过VRML97国际标准
1998.11 改名为Web3D联盟，推荐结合
1999.2 启动X3D
1999 - 2002 实现了 gzip、Universal-Media-Libraries、GeoVRML、DIS-Java-VRML、H-Anim、EAI
2002.4 VRML标准修订，正式加入UTF-8、EAI、GeoVRML、NURBS 曲面特性
2002.7 X3D 宣布草案
2002.12 X3D 进入ISO审议
2003.2 X3D 编码规格进入ISO审议
2003.3 X3D 语言结合标准进入ISO的最后审议阶段
2004 通过 X3D ISO 国际标准

X3D 教程

- http://x3d.esoe.ntu.edu.tw/

- Use FreeWRL as a player

COLLADA (SONY)

- Wildly used in PS3/PSP games
- XML based, and similar to X3D

- http://www.khronos.org/collada/
- http://www.opencollada.org/home.html

3D mesh surface compression

- Terrain data can be compressed by JPEG related methods

- MPEG-4 defines a compression method:
  - –Compress topological connectivities: relationships among vertices
  - –Compress geometric position information: vertex positions, normal vectors, texture coordinates …
  - –Compress texture images ...

浙江大学计算机学院
数字媒体与网络技术

Digital Asset Management
数字媒体资源管理

3. Multimedia Database

3.1. Challenges

Outline

- What is Multimedia?
- Why do we need multimedia?
- Classification of Media Types
- Types of Media
- Characteristics of MM Data
- Definition- MMDBMS
- MMDBMS Characteristics

What's Multimedia?

- Multi => Many

- Media:
  - A means to distribute and represent information: Text, graphics, pictures, voice, sound and music..
    - Perception media (how do humans perceive information?)
      - Audio/visual media
    - Representation media (how is information encoded?)
      - ASCII, JPG, MPEG, PAL.
    - Presentation media (medium used for output/input)
      - Input/output media (keyboards, papers)
    - Storage media (Where is information stored?)
      - Magnetic disk, optical disk

What's Multimedia?

- Multimedia:
  - To distribute and present information coded as
    - Text, Graphics, animation, audio and video..
  - By
    - Computer, TV, phone, etc.

What is Multimedia?

Multimedia (MM) has pervaded the worlds of entertainment, education and business.

- Different view to define MM:
  - Technologist: MM system
  - Games Player: Not only see the blood but hear the screams
  - Artist: The potential to create life

Definitions are contextual and depend on the perspective of the user.

What is Multimedia?

- Multimedia: **a working definition**
  - A combination of two or more categories of information having different transport signal characteristics
  - Typically, one medium is a continuous medium while another is discrete
  - Image, audio, video and graphics are usually the examples of media

What is Multimedia?

Some serious definitions:

"From a user's perspective - multimedia enables computer *information* to be *represented* through audio, video, text, images, graphics and animation."

"Multimedia is defined as an interactive computer-mediated presentation that included at least two of the following elements: text, sound, still graphic images, motion graphics and animation."

What is Multimedia?

- Most definitions agree that there are a number of key aspects of multimedia:
  - Multimedia involves "Many Media"
    - THE MEDIA DOMAIN
  - Multimedia involves Computers
    - THE SYSTEMS DOMAIN
  - Multimedia enhances the presentation and communication of information
    - THE APPLICATION DOMAIN

Multimedia: Computers v.s. Humans

Machine-Centred View

- People are:
  - vague (含糊）
  - disorganized（无序）
  - distractible （不专注）
  - emotional （情绪化）
  - illogical （非逻辑）

- Machines are:
  - precise
  - orderly
  - undistractible
  - unemotional
  - logical

VS

Human-Centred View

- People are:
  - multimedia
  - creative
  - attentive to change（着力改变）
  - resourceful （富想象力）
- Machines are:
  - rigid & text oriented
  - dumb
  - insensitive to change
  - unimaginative

Multimedia: Computers → Humans

- Machines cannot do many *mundane* things which can be performed routinely by humans （机器不擅长）
  - Natural language processing
  - Recognizing objects in images, vision
- Humans (untrained) cannot do many *clever* things which can be performed routinely and efficiently by machines
  - Numerical and symbolic computation, theorem proving
  - Searching information backwards
  - Chess playing

Why do we need multimedia?

- A computer database supports following basic data types:
  - numeric (e.g. student number); character string (e.g. surname, course title);
  - alphanumeric (e.g. course code, telephone number);
  - Boolean (e.g. female or male); date (e.g. date of birth);
  - text (e.g. description of a particular course, address).
- Inadequate to hold documents such as letters, application forms or image (student photograph).
- Need a document image processing system including:
  - a scanner or digital camera to capture (input) the images;
  - image database to store and retrieve the image.
- Further enhancement to handle other types of information (voice)
- A short video clip

Classification of Media Types

- Media types can be divided into two groups:
  - Temporal (Continuous media)
    - Time or more exactly time-dependency between information items, is part of the information itself.
    - dynamic, time-based, continuous
  - Non-temporal (Discrete media)
    - Time is not part of the semantics of the media.
    - static, non-time-based, discrete

Classification of Media Types

- Temporal media types:
  - audio, video, music, animation
- Non-temporal media types:
  - text, graphics, images

Media Type: Text

Media Type: Text

Different Representation

- ASCII
- ISO character sets
- Marked-up text
- Structured text
- Hypertext

Different Operations

- Character operations
- String operations
- Editing
- Formatting
- Pattern-matching
- Searching
- Sorting
- Compression
- Encryption

Media Type: Text

- Plain Text
  - American Standard Code for Information Exchange (ASCII)
  - Each ASCII code uses seven bits for; 8 bits are used to store each character with the extra bit being 0.
  - ***Unicode and UTF-8***
- Structured Text
  - SGML, XML, HTML
  - Latex,
  - Office Document Architecture (ODA)

Media Type: Images

Media Type: Image

Different Representation

- Color model
- Alpha channels
- Number of channels
- Channel depth
- Interlacing
- Indexing
- Pixel aspect ratio
- Compression

Different Operations

- Editing
- Point operations
- Filtering
- Compositing
- Geometric Transforms
- Conversion

Media Type: Images

- Several Issues
  - Is the image as high-quality as I need? (**Representation**)
  - Is the image efficiently stored and transmitted? (**Compression, transmission**)
  - Can I find the similar images in a database (**Retrieval engine, Internet**)? How? (**Analysis, processing**)

Media Type: Graphics

Media Type: Graphics

Different Representation

- Geometric models
- Solid models
- Physically based models
- Empirical models
- Drawing models
- External format for mode

Different Operations

- primitive editing
- Structural editing
- Shading
- Mapping
- Lighting
- Viewing
- Rendering

Media Type: Analog Video

Media Type: Analog Video

Different Representation

- Frame rate
- Number of scan lines
- Aspect ratio
- Interlacing
- Hypertext
- Quality
- Component/Composite

Different Operations

- Storage
- Retrieval
- Synchronization
- Editing
- Mixing
- Conversion

Media Type: Digital Video

Media Type: Digital Video

Different Representation

- Sampling
- Sampling rate
- Sampling size and quantization
- Data rate
- Frame rate
- Compression
- Support for interactivity
- Scalability

Different Operations

- Storage
- Retrieval
- Synchronization
- Editing
- Effects
- Conversion

Media Type: Digital Video Formats

- AVI
  - A format developed by Microsoft for storing video and audio information. AVI files are limited to 320 x 240, and 30 frame/sec.
- Quicktime
  - A video and animation system developed by Apple Computer. It is built into the Macintosh operating system.
  - QuickTime supports most encoding formats, including JPEG and MPEG.
- MPEG
- ActiveMovie
  - A new multimedia streaming technology developed by Microsoft, supporting most multimedia formats, including MPEG.
- RealVideo
  - A streaming technology developed by RealNetworks for transmitting live video over the Internet.
  - RealVideo uses a variety of data compression techniques and works with both normal IP connections as well as IP Multicast connections

Media Type: Videos

- Issues
  - The format of the video? (Compression)
    - MPEG (.mpg), AVI (.avi), Realplayer (.rm).
  - What do I need to stream video across the network?
    - Watch video online.
  - Are the different media synchronized?
    - Multimedia description and characterization.
  - How can I describe a video retrieval request like a query?
    - Video processing, database management, retrieval

Media Type: Digital Audio

Media Type: Digital Audio

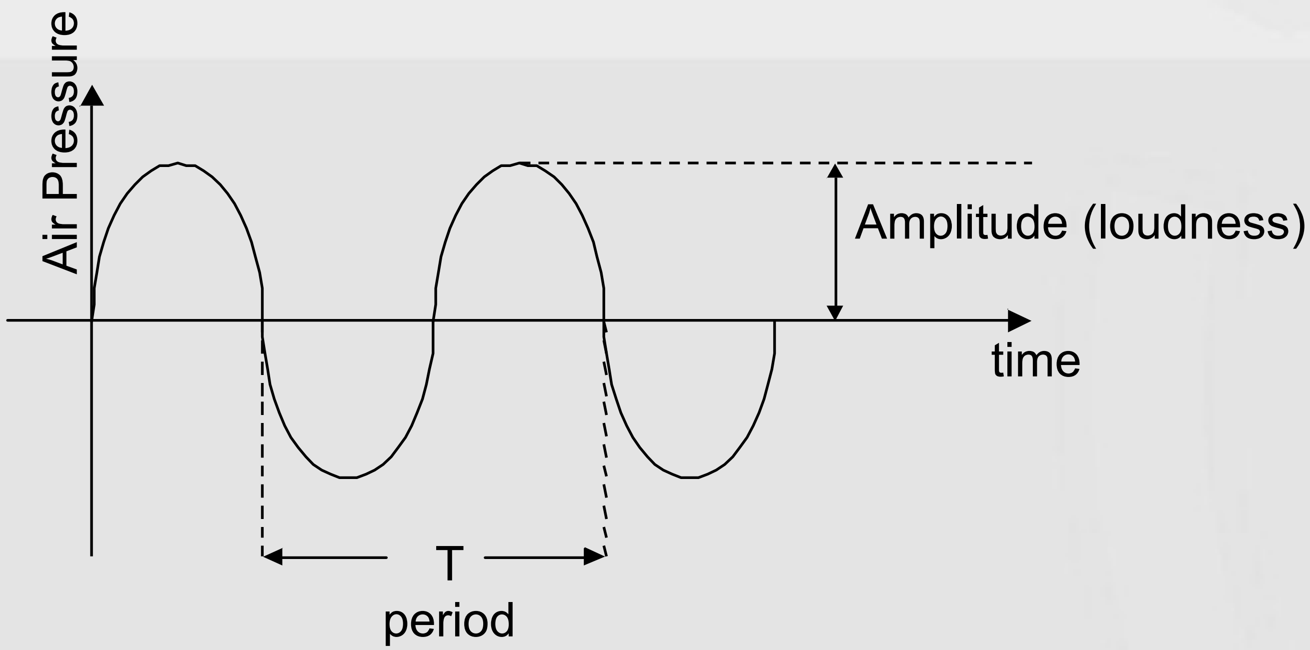
Representation

- Sampling frequency
- Sampling size and quantization
- Number of channels
- Interleaving
- Negative samples
- Encoding

Operations

- Storage
- Retrieval
- Editing
- Effects and filtering
- Conversion

Media Type: Audio



- The frequency range is divided into:
  - Human hearing frequency: 20 Hz to 20 KHz,
  - Infra-sound: 0 to 20 Hz,
  - Ultrasound: 20 KHz to 1 GHz,
  - Hypersound: 1 GHz to 10 THz

- **Sound** consists of pressure waves that move through a compressible medium.
- The ***frequency*** of a sound is the reciprocal value of the period (wavelength).
- Wavelength is the distance between identical points in the adjacent cycles of a waveform signal

  Terahertz (THz) is used; 1 THz = 1,000,000,000,000 cycles per second

Media Type: Music

Media Type: Music

Different Representation

Operational/Symbolic

MIDI

(Musical Instrument Digital Interface)

Different Operations

Playback and synthesis

Timing

Editing and composition

Media Type: Animation

Media Type: Animation

Different Representation

- Cel models
- Scene-based models
- Event-based models
- Key frames
- Articulated objects and hierarchical models
- Scripting and procedural models
- Physically based and empirical models

Different Operations

- Graphics operations
- Motion and parameter control
- Rendering
- Playback

Other Media Types

- Surface images
- Volume images
- Digital ink
- Speech encoding
- Speech synthesis
- Non-temporal video and animations

Characteristics of MM Data
-- challenges about managing MM data

- ***Huge Size***
  - multimedia objects are large in size (compared with traditional alphanumerical data) and are not readily accommodated by ‘old’ DBMSs
  - E.g.,
    - a video – each second contains, say, 30 frames; each frame may require, depending on video quality, several megabytes of storage.
    - a color picture of 1280 x 960 pixels using 32-bit color requires about 5MB of memory

      ≈ 4,915,200 = 1280 x 960 x 4 *bytes*
- Needs fast and powerful processors
- Large storage capacities
- Multimedia data have to move very quickly through the different components of the computer

Characteristics of MM Data
-- challenges about managing MM data

Data Volume

- Data dynamic in time inherently leads to a high data volume for single data elements (cf. Table I):

MediaType	SampleFormat	DataVolume	TransferRate
Text	ASCII	1 MB / 500 pages	2 KB / page
Color Image	GIF, TIFF;	1.6GB / 500 images	3.2 MB / image
	JPEG	0.2GB / 500 images	0.4 MB / image
CD-music	CD-DA	52.8 MB / 5 min.	176 KB / sec.
Consumer video	PAL	6.6 GB / 5 min.	22 MB / sec.
High quality Video	HDTV	33 GB / 5 min.	110 MB / sec.
Speech	m-law, linear;	2.4 MB / 5 min.	8 KB / sec.
	ADPCM;	0.6 MB / 5 min.	
	MPEG audio	0.2 MB / 5 min.	

Characteristics of MM Data

Data Volume

- Concerning when designing a MMDB system:
  - MM applications: dealing with **huge amount of data** under **real time constraints**
    - Design of hardware, OS, and networks ...
  - Perform the **processing** on **the references to the values**.
    - E.g., video script editing.
  - Dynamic data
    - cannot be performed over references (e.g., copying) on one hand
    - cannot be executed in the standard way as for alphanumeric data because dynamic data exceeds the physical resources.

Characteristics of MM Data

Data Volume

- Specific form of dynamic data management
  - speed the process over time
    - s.t., at each distinct moment, only a limited amount of physical resources are needed.
  - This kind of dynamic operation heavily affects the behavior of a system, hence must not be transparent

Characteristics of MM Data

- The requirements regarding data volumes
  - appropriate referencing mechanisms to refer to MM data units
  - dynamic data management for very large objects

Characteristics of MM Data
-- challenges about managing MM data

- ***Quality of Service (QoS)***
  - multimedia applications differ from traditional DB applications with respect to **performance requirement**
  - multimedia applications, in general, require **high throughput and a constant delivery** of information
  - e.g, real-time requirement for audio

note that lost audio samples are perceived much sooner by a human user than lost video frames:

the ear is more "sensitive" than the eye!

Characteristics of MM Data
-- challenges about managing MM data

- ***Quality of Service (QoS)***

- To offer a certain quality of service, the system has to **reserve** an adequate number of resources
- the higher the required service, the more resources the system has to reserve, and the more '**expensive**' the service will be.

Characteristics of MM Data
-- challenges about managing MM data

- ***Synchronization***
  - multimedia objects: composed of several components,
  - require synchronization of them
    - e.g., a film consists of moving pictures, speech, and subtitles
    - *It'll be odd if the lip movements of actors/actresses do not synchronize precisely with their voices and with the text of the subtitles.*

- Research results have shown that
  - **video/audio** or **video/video** synchronization is **less critical**, expressed in time constraints, than **audio/audio** synchronization!

Characteristics of MM Data
-- challenges about managing MM data

- ***Content-Based Retrieval***
- Hard to use traditional "exact-match" queries to retrieve multimedia objects:
  - interpretations of multimedia data is a rather **unexplored area**
  - multiple interpretations are, in general, possible (=> inexact match)
  - multimedia objects may contain multiple attributes of type audio, video, or text
- Obviously, content-based retrieval of multimedia objects is far more complex than Information Retrieval (IR) on text objects. (Note that IR is already more complex than traditional database accesses.)

Characteristics of MM Data

- ***Similarity-based Search***
  - Unlike traditional 'exact search' in relational database, users usually ask for similar objects based on their contents
    - Finding an image with similar color
  - A multimedia object may contain multiple features
    - Video may contain text, image, audio, etc.
    - Similarity search may be based on multiple features, i.e., integrating content and semantic features

Characteristics of MM Data
-- challenges about managing MM data

Temporal Aspects

- Certain multimedia data types like audio and video have a canonical mapping to real-world time (i.e., their time scale is absolute).

- When processing dynamic data, involving dynamic data types like audio or video, parallel tasks can occur. Also, the dynamics of a multimedia document are based on the temporal relationships between the constituent media types.

  E.g., consider an application to play back a video on a screen: it gets the audio simultaneously from a different device and allows for user interaction to control presentation (like, adding annotations to the video at certain points without interrupting the video presentation).

- There is also a need to support for media-specific synchronization

  e.g., playing back the video frames and the sound track of a movie requires fine-grained synchronization, such as lip synchronization.

Characteristics of MM Data
-- challenges about managing MM data

Temporal Aspects

- There is a need to model, store, and process temporal relationships between media components of a multimedia document or presentation, including synchronization mechanism
- Methods available for controlling parallelism of tasks :
  - **relative scheduling (e.g., one task triggers the other)**
  - **absolute scheduling (placing events on an absolute time scale)**
  - **combination of the above**

Characteristics of MM Data
-- challenges about managing MM data

Temporal Aspects

- Requirements
  - incorporation of **time-related** concepts into the data model
  - **non-transparent parallelism** for explicit control of parallel tasks
  - scheduling and (media-specific) synchronization mechanisms to provide for the description of temporal relationships and their enforcement

Characteristics of MM Data
-- challenges about managing MM data

Media Representation

- The basis data types like **alphanumeric ones are not appropriate** to reflect the structure of multimedia data
- **new built-in data types**
  – e.g, bitmaps or audio samples ...

- **type constructors** taking into account the temporal nature of multimedia data will be needed in some form, and **appropriate support for processing** these data types has to be provided

Characteristics of MM Data
-- challenges about managing MM data

Media Representation

- To be able to use **different formats** for the **same multimedia types** is crucial for multimedia data:
  - **different compression techniques**
    - for different applications, for different resources may need different formats, and each format should be convertible to another in principle
  - the internal representation may not be appropriate to be presented to the user, so special representations for different users to provide different views of the same data may be needed (either generated on the fly or stored persistently)

Characteristics of MM Data

Media Representation

- In SQL3 (or, SQL:1999), the repertoire of built-in types are extended: in addition to the types known from SQL2, there are
  - **Boolean type**
  - **Large character objects**
  - **Large binary objects**
- The latter two allow for character strings and bitstrings in the Gigabyte ranges, thus matching the storage requirements of certain MM applications.

Characteristics of MM Data

E.g., in Oracle 9i…

Name	Data type	Size	characteristics
BLOB	*Binary*	*4Kb in table space* *4GB in ext. space*	*random access;* *transaction support*
CLOB	*Character*	*4GB*	*random access;* *transaction support*
NCLOB	*National character sets*	*4GB*	*random access;* *transaction support*
BFILE	*Binary*		*Read only;* *external file*

Characteristics of MM Data

e.g., in Oracle 9i, after the following:

```
CREATE  TABLE  Grape
            (grape_name VARCHAR2(25) primary key, picture BFILE);
CREATE DIRECTORY “PHOTO_DIR” AS ‘C:\PICTURES’;
```

The user can then do insert:

```
INSERT INTO Grape(grape_name, picture)
     VALUES (‘chardonnay’, BFILENAME(‘PHOTO_DIR’, ‘chardonnay.jpeg’))
```

Characteristics of MM Data

Media Representation

- Requirements:
  - new built-in **data types** and **operations** for MM data
  - modular and efficient representation of **various formats**
  - data **representation** should be **transparent** to users
  - **different views** of the same data should be possible

Characteristics of MM Data
-- challenges about modeling MM data

- High level content abstraction is natural to the way humans think

- Effective modeling of MM data is critical
  - support for semantically rich conceptual contents
  - ability to represent diverse aspects of the data to be modeled
  - facilities for dynamic concept enrichment and expansion
  - incorporation of knowledge of low level data
  - isolation of the user from the low level representation and storage levels

Characteristics of MM Data
-- challenges about modeling MM data

- Low level representation of multimedia data, which encodes the physical reality, leads to the problem of huge amounts of data.
  - references to (as *abstractions* of) MM data are needed to process it efficiently to avoid copies;
  - more complex abstractions, like references enriched with more information than that needed for identification, can be used to index data to provide fast access.
- Another reason for introducing abstractions (provided by the user/system) is to allow the user to refer to the data in terms of abstractions which make up his model of the application domain.

Characteristics of MM Data
-- challenges about modeling MM data

- However, traditional (relational/object-oriented) data models fall short in representing MM data adequately

Data model features	Relational model	Object oriented model	"Multimedia" model
Structural model	Relations (tables)	Objects attributes	Continuous data streams
Behavioral model	SQL	Methods	Time-dependent operations
Constraints	Primary keys, foreign keys etc.	Referential integrity, pre-/post-conditions, etc.	QoS parameters

Characteristics of MM Data
-- challenges about modeling MM data

- MM Data Abstraction for Time-Dependent Data
  - to reflect the time-related characteristics
  - and requirements of continuous data should address the description, the processing of temporal relationships, and synchronization constraints.

Time-dependent data description:

- sequential-composition,
- parallel-composition

Two approaches for describing MM data compositions are:

- time-critical medium (e.g., audio) as the reference for the others
- an abstract temporal dimension ref. *common time line*

Characteristics of MM Data
-- challenges about modeling MM data

MM Data Abstraction for Time-Dependent Data

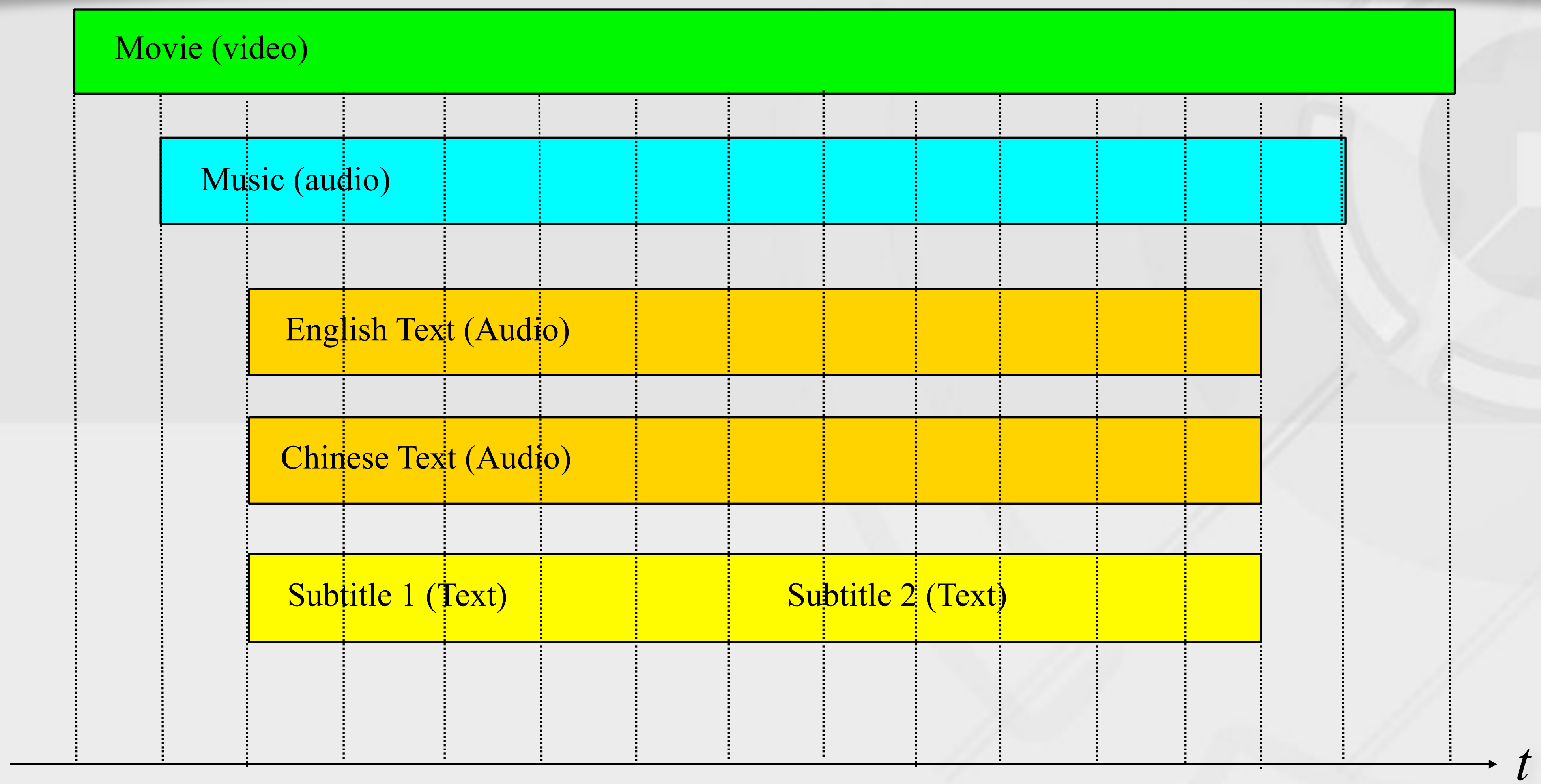


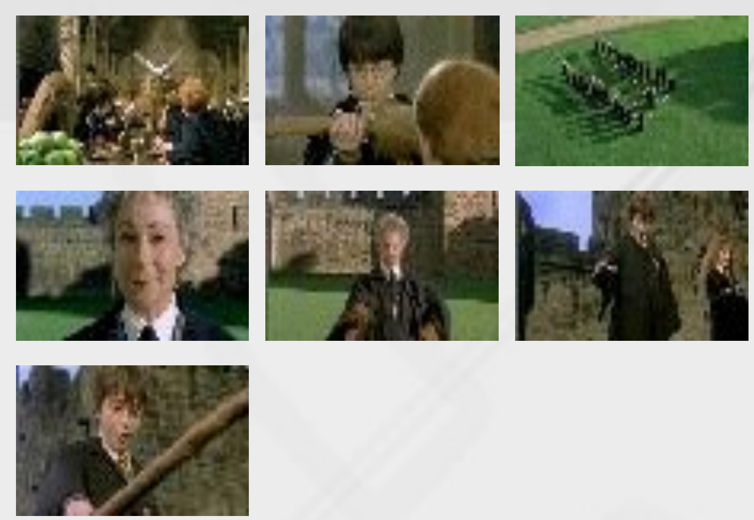
Fig. Different Media Bound to a Common Time Line

Characteristics of MM Data
-- challenges about modeling MM data

- **Semantic nature** of MM data is more complex…
  - A picture often means different things to different people => even manual annotation is hard, not to mention tedious and expensive;
  - Difficulties of cross-media search

E.g., an image query (a) and semantically relevant results (b)

image



video

Harry Potter has never been the star of a Quidditch team, scoring points while riding a broom far above the ground. He knows no spells, has never helped to hatch a dragon, and has never worn a cloak of invisibility.

text

Characteristics of MM Data
-- challenges about modeling MM data

- Operations pertinent to individual media are needed
- Data objects in MMDB can be divided into "**static**" and "**time-dependent**" ones
  - Two groups of operations:
    1. Operations on static data (eg, image)
    2. Operations on dynamic data (eg, audio/video)
  - **static**:

    op: P x DB => R x DB (P: parameters; R: domain)

    eg, *op(p,db)*: a compressed image file is passed to a viewer provided/ incorporated by the MMDBMS. Zoom and rotate are examples of operations for images.

Characteristics of MM Data
-- challenges about modeling MM data

- **time-dependent**:

  op: P x DB x T => R x DB x T

  eg, *op(p,db, t):* when presenting a video on a video viewer (or delivering it to any other applications), operations such as play, pause, fast forward, reverse etc. would be applicable and need to be devised.

Characteristics of MM Data
-- challenges about modeling MM data

- Providing users with a semantically effective and performance **efficient query language** is critical
  - **exact match** query of conventional databases not suitable
  - specific properties of MM data, esp.
    - the temporal and spatial aspects, the concepts of vagueness or unsharpness
  - Additional concepts are required/useful to query a MMDBMS:
    1. **content-based access**, including query-by-example (QBE)
    2. **domain-specific knowledge** (subject to the high costs on acquisition and the expensive maintenance of the domain knowledge)
    3. **speech recognition and retrieval**

Characteristics of MM Data
-- challenges about modeling MM data

- **Multi-layers of abstraction** for the retrieval and organization of the MM data
  - Example: a database of video
    - first layer: identify single scenes in videos
    - second layer: identify geometric objects in these scenes
    - third layer: geometric objects could be related to real world entities in the scenes.

  A user may search for such entities based on attribute values stored elsewhere in the DB, and so, may access a video in which this entity occurs using indices at each layer.

Characteristics of MM Data
-- challenges about modeling MM data

- In order to have these features available in a MMDB system, appropriate solutions have to be found for
  - **indexing mechanisms**
  - **semantic and consistent modeling of abstractions**

- abstraction and modeling facilities (hardware resources, e.g., compression chips, presentation devices like speakers, monitors, windows, and software resources)
  - device transparency
  - resource sharing
  - device classification and modeling
  - distribution of data should be transparent to applications and users.

Definition- MMDBMS

- MMDBMS
  – *a framework that manages different types of data potentially represented in a wide diversity of formats on a wide array of media sources*

MMDBMS Characteristics

- **Uniformly query** data represented in different formats
- Query data represented in **diverse media**
- Retrieve media objects from a local storage devices in a smooth, jitter-free manner
- Provide audio visual presentation of a query result
- Deliver presentation to satisfy quality of service requirements

MMDBMS Characteristics

- Uniformly query data represented in different formats
  - Integrate data from different relational databases with different schemas
  - Query flat files, OO and spatial databases, legacy data sources
  - Ability to handle fuzzy searching

MMDBMS Characteristics

- Query data represented in **diverse (多变的) media**
  - Queries spanning multiple media types
  - → ability to merge them together

MMDBMS Characteristics

- Retrieve media objects from a local storage devices in a smooth, jitter-free manner
  - Media stored in up to 10 gigabytes
  - Stored on mix of storage devices
  - → variable performance characteristics

MMDBMS Characteristics

- Provide audio visual presentation of a query result
  - The user should have the ability to specify the form and content of the presentation he/she would like to obtain

MMDBMS Characteristics

- Deliver presentation to satisfy quality of service requirements
  - Buffer availability and bandwidth need to be taken into account

Multimedia equation

Multimedia = presentation + context

presentation: sensory, aesthetic part （美学）

context = convergence + information + architecture

- convergence = data +platform + distribution
- information = storage and retrieval
- architecture = compression + components + connectivity

http://www.notre-dam.org
an open source DAMS

多媒体数据库系统的挑战

- 多媒体数据的独特数据特性
  - 单个条目数据规模和吞吐大
  - 多通道数据需要同步
  - 连续媒体数据需以流媒体形式提取
  - 对于不同客户的QoS需求
  - 相似性比较和搜索困难

3.2. Architecture

Outline

1. MM content organization

2. MM database system architecture

3. MM system service model

4. Multimedia Data Storage

5. Multimedia application

3.2.1. Multimedia Content Organization

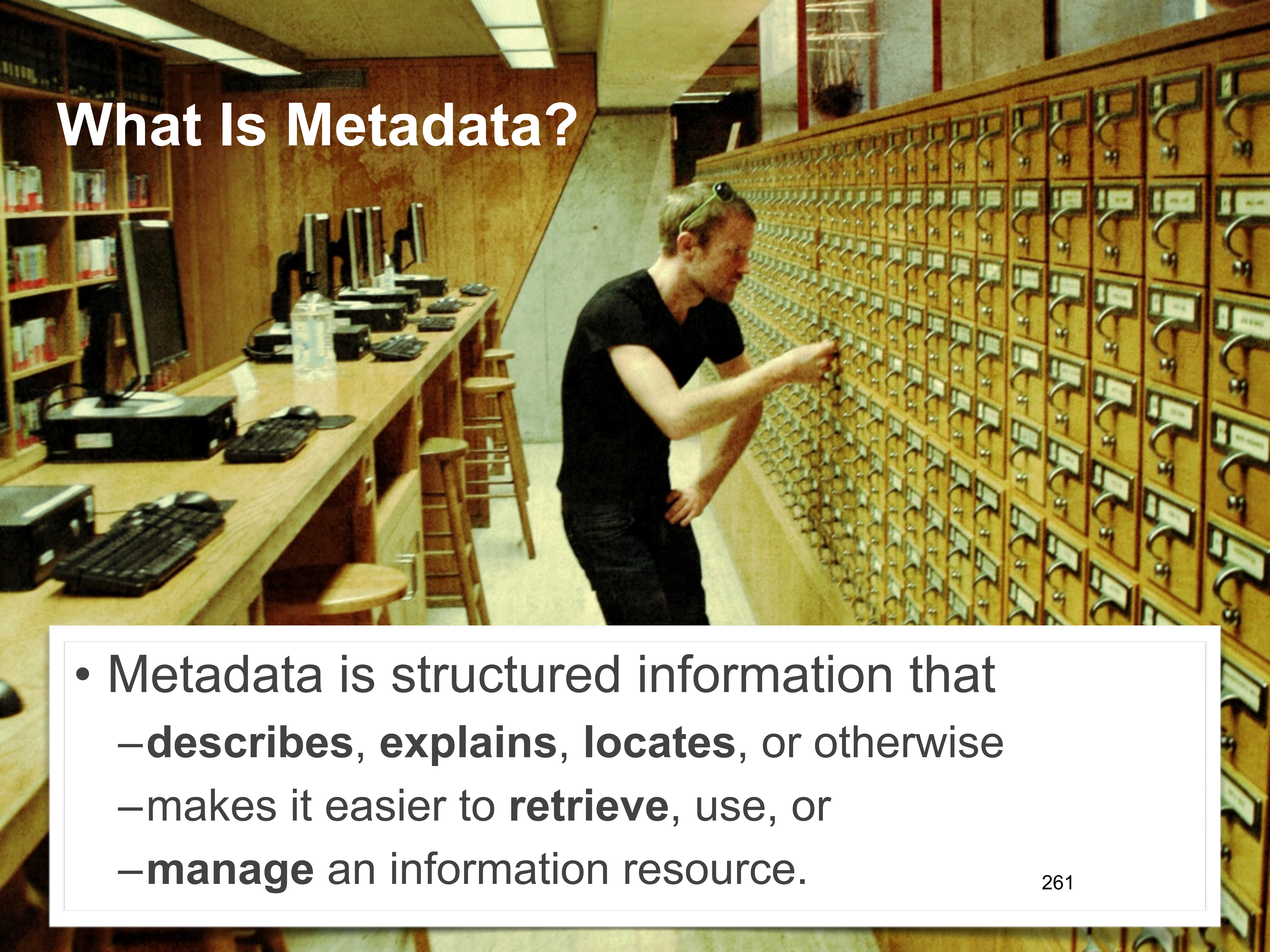
Metadata Model Organization

- Content-dependent Metadata
- Content-descriptive Metadata
- Content-independent Metadata

Metadata Model

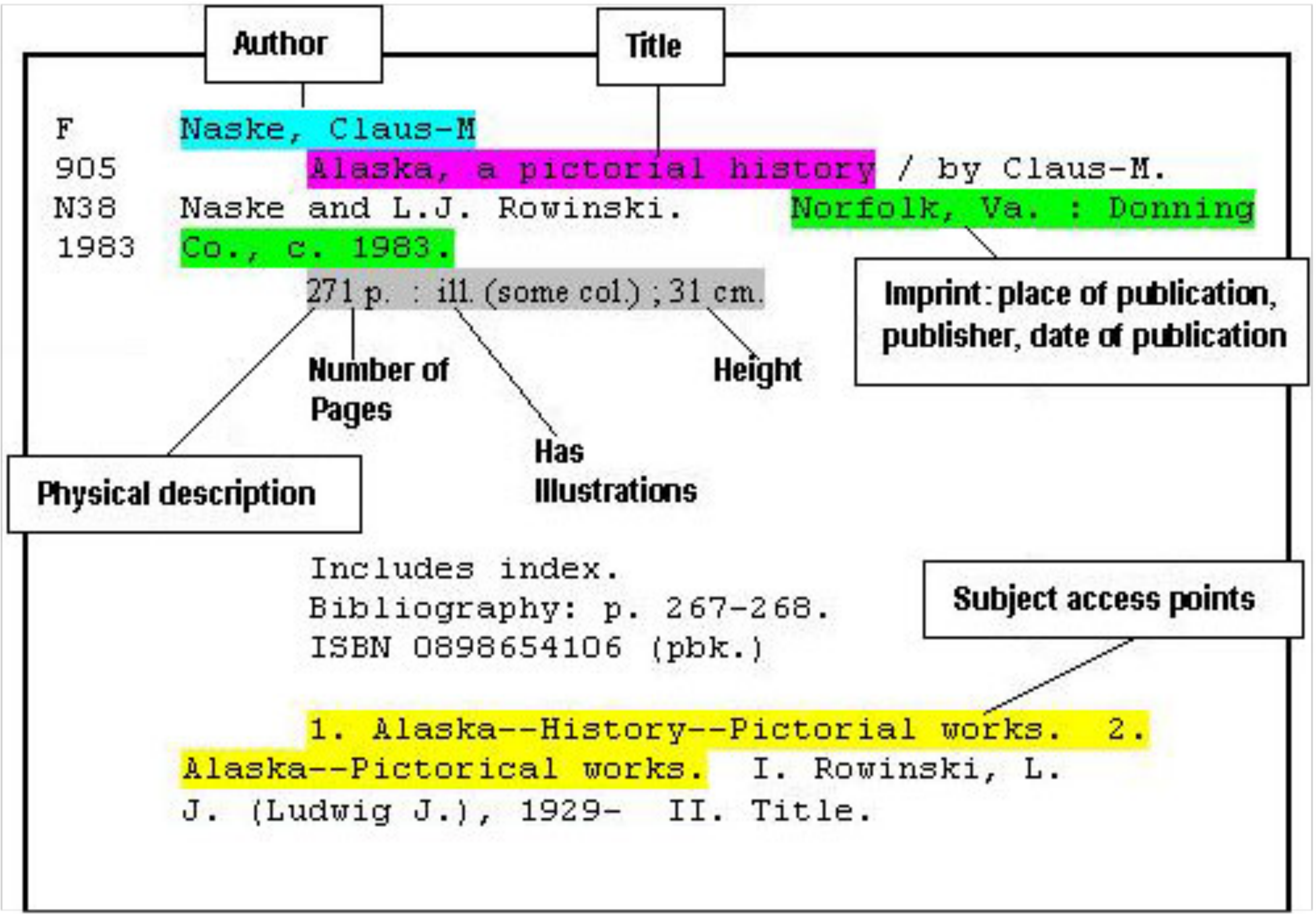
- Metadata => **data about data**
  - –forms an essential part of any database
    - providing **descriptive data** about each stored object, and
    - is the key to **organizing** and **managing** **data** objects
  - –critical for describing essential aspects of content:
    - main topics, author, language, publication, etc.
    - events, scenes, objects, times, places, etc.
    - rights, packaging, access control, content adaptation, …

What Is Metadata?



- Metadata is structured information that
  - **describes**, **explains**, **locates**, or otherwise
  - makes it easier to **retrieve**, use, or
  - **manage** an information resource.

What Is Metadata?



Metadata Model

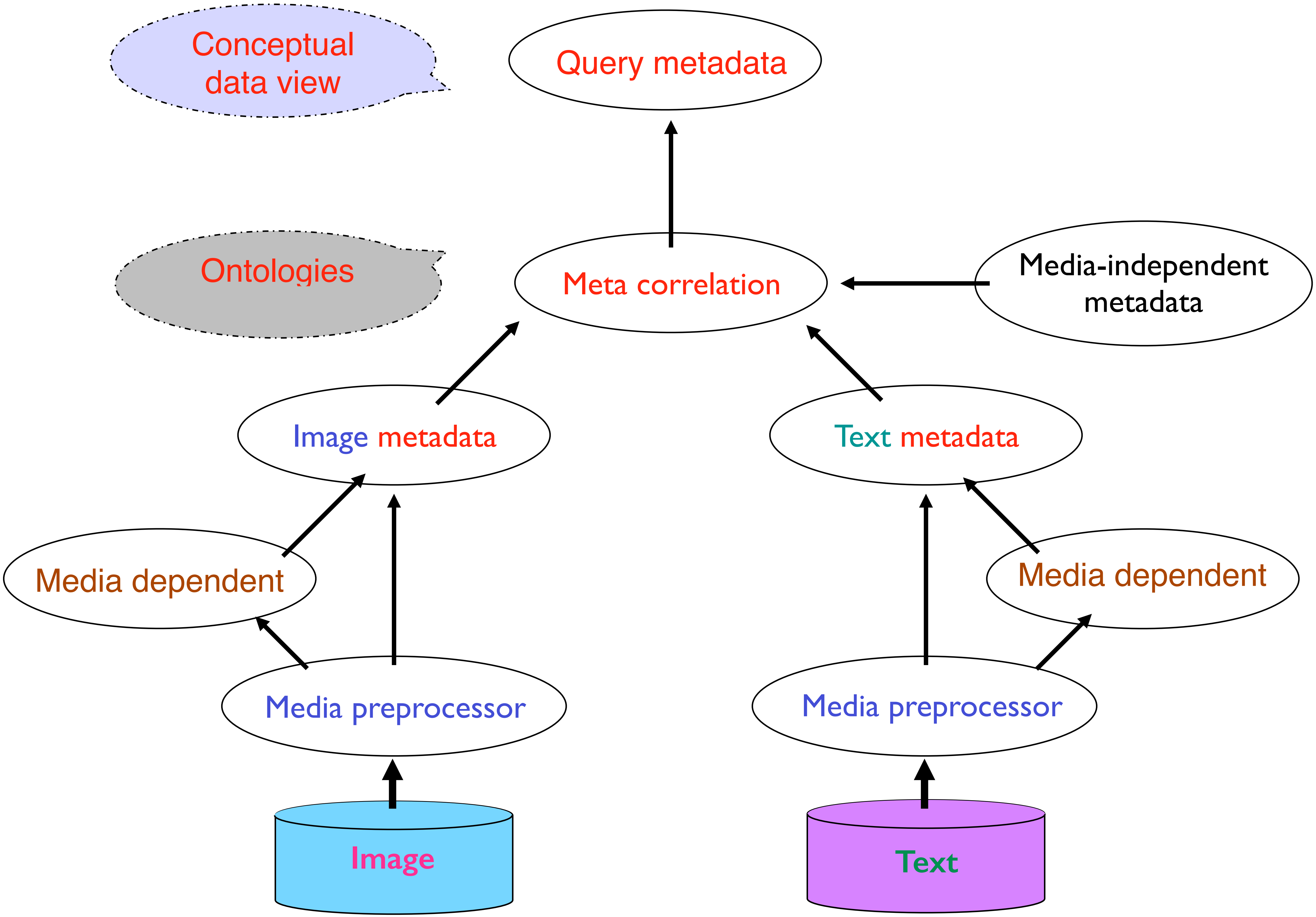
- **Purposes** of metadata:
  - **Administrative**
    - managing and administrating the data collection process
  - **Descriptive**
    - describing and identifying for retrieval purpose, creating indices
  - **Preservation**
    - managing data refreshing and migration
  - **Technical**
    - formats, compression, scaling, encryption, authentication and security
  - **Usage**
    - users, their level and type of use, user tracking, versioning (e.g., a high resolution version and corresponding thumbnail).

Metadata Model

Conformity （一致性）with open metadata standard will be a vital:

- Faster design and implementation
- Interoperability (互操作性）with broad field of competitive standards-based tools and systems
- Leveraging of rich set of standards-based technologies for critical functions
  - e.g., content extraction, advanced search, and personalization

The "role" of metadata in query processing:



Classifying Metadata

Classification of metadata can be:

1. Specific to the **media involved**
2. Specific to the **processing**
3. **Content** specific metadata

Image object

- Image capture
- Image storage
- Caption
- Genre
- Period
- Subjects
- Photographer
- IP rights
- Texture

Text object

- title
- author
- abstract
- Full text indices

Video

- time based
- play rate
- camera motion
- camera lighting

Sample Metadata

Metadata Classification

Metadata can be classified as:

- **Content dependent** (e.g., face features; used in CBR)
- **Content-descriptive** (used in TBR)
  1. Domain-independent metadata: independent of the application or subject topic
  2. Domain-dependent metadata: specific to the application area
- **Content-independent** (e.g., photographer's name; used in ABR)

- ABR: Auction-based Retrieval (基于竞价的检索，较少见)
- CBR: Content-based Retrieval
- TBR: Text-based Retrieval

Metadata Classification

Media	Content independent	Content descriptive	Content dependent
Text	status, location, date of update components	keywords, formats, categories, language	subtopic boundary word image spotting
speech	start, end time location confidence of word recognition	speakers	speech recognition speaker recognition prosodic cues change of meaning
Image	creator title date	keywords, formats	feature selection image features (e.g., histogram, segmentation)
Video	product title data distributor	camera shot action distance close-up	shot boundary frame features (e.g., histogram, motion lighting level, height)

Domain-dependent Metadata

- Standards for domain-specific metadata
  - Digital geospatial metadata
    - US Geographic Data Committee
    - http://www.fgdc.gov/metadata/metahome.html
  - Environmental data (UDK)
    - the European Environmental Catalog
  - Product data exchange (PDES)
    - an ANSI standard for the exchange of product model data
  - Rich Site Summary (RSS)
    - a lightweight XML vocabulary for describing websites, ideal for news syndication
  - Medical information (HL7)
    - provides specification for hospital records and medical information management
    - accredited by ANSI

Domain-independent Metadata Standards

- ISO/IEC 11179 (http://metadata-standards.org/11179/)
  - Intended to provide:
    - conceptual framework,
    - logical explanations of the processes for an organization to describe data semantics consistently, and
    - the exchange of data and metadata across organizational units
  - The standard divides data elements into 3 parts:
    - **Object class** – the thing the data describes (e.g., person, airplane)
    - **Property** – a peculiarity that describes/distinguishes objects
    - **Representation** – the allowed values and other information

Domain-independent Metadata Standards

- ISO/IEC 11179

Attribute	Description
Name	the label assigned to the **data element (d.e.)**
Id	the unique identifier assigned to the d.e.
Version	the version of the d.e. (e.g., 1.1 for Dublin Core)
Registration Authority	the entity authorized to register the d.e.
Language	the language in which the d.e. is specified (e.g., English)
Definition	a statement representing the d.e. concept and nature
Obligation	indicates if the d.e. is required to be not null
Data type	indicates the data type that can be represented in d.e.
Maximum Occurrence	indicates any limit to the repeatability of the d.e.
Comment	a remark concerning the application of the d.e.

Domain-independent Metadata Standards

- The Dublin Core Metadata set
  http://purl.org/metadata/dublin_core
  - Originally for resource description records of **online libraries** over Internet
  - version 1.1
    - broaden to other media with a link to the ISO/IEC 11179 standard
      - Each Dublin Core element is defined using a set of 10 attributes from the ISO/IEC 11179
      - Six of them are common to all the Dublin Core element (3-5, 7-9)
    - 15 metadata elements (the Dublin Core) has been proposed
      - which are suggested to be the minimum number of metadata elements to support retrieval of a document-like object (DLO) in a networked environment

The Dublin Core Metadata set

ID	Core element	Semantics
1	Subject	topic addressed by the work
2	Title	the name of the object
3	Creator	entity responsible for the intellectual content
4	Publisher	the agency making the object available
5	Description	an account of the content of the resource
6	Contributor	an entity making contributions to the resource content
7	Date	associated with an event in the life cycle of the resource
8	Resource type	the nature/genre of the resource content
9	Format	physical/digital manifestation of the resource; format of the file (e.g., postscript)
10	Id	unique identifier
11	Relation	a reference to a related resource
12	Source	a ref. to a resource from which the current resource is derived
13	Language	language of the intellectual content
14	Coverage	extent/scope of the resource content; typically include location, period
15	Rights	Information about rights held in and over the resource

Domain-independent Metadata Standards

- Resource Description Framework (RDF)
  - Being developed by the W3C as a foundation for processing metadata
  - Allows multiple metadata schemes to be read by human and parsed by machines
  - Specific objectives include:
    - **Resource discovery** – to provide better search engine capabilities
    - **Cataloging** – for describing the content and relationships available through intelligent software agents
    - **Content rating** – describing collection of pages that represent a single logical "document"
    - **IP rights** – describing the intellectual property of web pages
    - **Privacy preferences and policies** – for users and website
    - **Digital signatures** – to create a "web of trust" for e-commerce, collaboration, and other applications

Resource Description Framework (RDF)

- The formal model of the **RDF framework**:
  - *Resources* (set).
  - *Literals* (set).
  - a subset of resources called *Properties*
  - There is a set called *Statements*, each element of which is a triple of form <**pred, sub, obj**>, where
    - **pred** is a property,
    - **sub** is a resource (member of *Resources*)
    - **obj** is either a resource or a literal
- The preferred language for writing RDF schemas is XML

XML

- Defined by the WWW Consortium (W3C)
- Originally intended as a document markup language not a database language
  - Documents have tags giving extra information about sections of the document
    - <title> XML </title> <slide> Introduction …</slide>
    - <?xml … ?> (document declaration)
    - <!-- definition of elements --> (comments)
  - Derived from SGML (Standard Generalized Markup Language), but simpler to use than SGML
  - **Extensible**, unlike HTML
    - Users can add new tags, and *separately* specify how the tag should be handled for display

XML

- The ability to specify new tags, and to create nested tag structures made XML a great way to exchange **data**, not just documents.
  - Much of the use of XML has been in data exchange applications, not as a replacement for HTML
- Tags make data (relatively) self-documenting

```
<bank>
   <account>
      <account-number> A-101     </account-number>
      <branch-name>      Downtown </branch-name>
      <balance>              500         </balance>
   </account>
   <depositor>
      <account-number> A-101    </account-number>
      <customer-name> Johnson </customer-name>
   </depositor>
</bank>
```

Structure of XML

- **Tag**: label for a section of data
- **Element**: section of data beginning with **<tagname>** and ending with matching **</tagname>**
- Elements must be properly ***nested***
  - ✓Proper nesting
    - <account> … <balance> …. </balance> </account>
  - ➡Improper nesting
    - <account> … <balance> …. </account> </balance>
  - Formally: every start tag must have a unique matching end tag, that is in the context of the same parent element.
- Every document must have a single top-level element

Structure of XML

– Mixture of text with sub-elements is legal in XML
  - Example:

```
<account>
   This account is seldom used any more.
    <account-number> A-102</account-number>
    <branch-name> Perryridge</branch-name>
    <balance>400 </balance>
</account>
```

  - Useful for document markup, but discouraged for data representation

Attributes

– Elements can have **attributes**

```
<account acct-type = “checking” >
        <account-number> A-102 </account-number>
         <branch-name> Perryridge </branch-name>
         <balance> 400 </balance>
 </account>
```

– Attributes are specified by *name*=*value* pairs inside the starting tag of an element

– An element may have several attributes, but each attribute name can only occur once

```
<account  acct-type = “checking”  monthly-fee=“5”>
```

Attributes vs. Subelements

- Distinction between subelement and attribute
  - In the context of documents
    - attributes: are part of markup
    - subelements: contents are part of the basic document contents
- Some information can be represented in two ways
  - <account account-number = “A-101”> .... </account>
  - <account>
    <account-number>A-101</account-number> ...
    </account>

attribute

subelement

- Suggestion: use attributes for identifiers of elements, and use subelements for contents

More on XML Syntax

- Elements without subelements or text content can be abbreviated by ending the start tag with a /> and deleting the end tag
  - <account number="A-101" branch="Perryridge" balance="200 />
- To store string data that may contain tags, without the tags being interpreted as subelements, use CDATA as below
  - <![CDATA[<account> … </account>]]>

    Here, <account> and </account> are treated as just strings

Namespaces

- XML data has to be exchanged between organizations
- Same tag name may have different meaning in different organizations, causing confusion on exchanged documents
- Specifying a unique string as an element name avoids confusion
- Avoid using long unique names all over document by using XML Namespaces

```
<bank Xmlns:FB=‘http://www.FirstBank.com’>
    …
    <FB:branch>
            <FB:branchname>Downtown</FB:branchname>
        <FB:branchcity>   Brooklyn   </FB:branchcity>
    </FB:branch>
    …
</bank>
```

XML Document Schema

- Database schemas constrain
  - what information can be stored, and
  - the data types of stored values
- not necessary in a XML document
- very important for XML data exchange
  - Otherwise, a site cannot automatically interpret data received from another site
- **Two mechanisms** for specifying XML schema
  - Document Type Definition (**DTD**)
  - **XML Schema**

XML Document Schema

- The type of an XML document can be specified using a DTD
- DTD constraints structure of XML data
  - What elements can occur
  - What attributes can/must an element have
  - What subelements can/must occur inside each element, and how many times.
- DTD does not constrain data types
  - All values represented as strings in XML
- DTD syntax
  - <!ELEMENT element (subelements-specification) >
  - <!ATTLIST element (attributes) >

Element Specification in DTD

– Subelements can be specified as
  - names of elements, or
  - #PCDATA (parsed character data), i.e., character strings
  - EMPTY (no subelements) or ANY (anything can be a subelement)
– Example

```
<! ELEMENT depositor (customer-name  account-number)>
<! ELEMENT customer-name (#PCDATA)>
<! ELEMENT account-number (#PCDATA)>
```

– Subelement specification may have regular expressions

```
<!ELEMENT bank ( ( account | customer | depositor)+)>
```

– Notation:
  - » "|" - alternatives
  - » "+" - 1 or more occurrences
  - » "*" - 0 or more occurrences

IDs and IDREFs

- An element can have at most one attribute of type ID
- The **ID attribute value** of each element in an XML document must be **distinct**
  - Thus the ID attribute value is an object identifier

- An attribute of type IDREF must contain the ID value of an element in the same document
- An attribute of type IDREFS contains a set of (0 or more) ID values.
- Each ID value must contain the ID value of an element in the same document

Bank DTD with ID and IDREF attribute types

```
<!DOCTYPE bank-2[
    <!ELEMENT account (branch, balance)>
    <!ATTLIST account
            account-number ID        # REQUIRED
            owners               IDREFS # REQUIRED>
    <!ELEMENT customer(customer-name, customer-street,
                                          customer-city)>
    <!ATTLIST customer
            customer-id      ID        # REQUIRED
            accounts           IDREFS # REQUIRED>
    … declarations for branch, balance, customer-name,
                         customer-street and customer-city
]>
```

XML data with ID and IDREF attributes

```
<bank-2>
    <account account-number="A-401" owners="C100 C102">
        <branch-name> Downtown </branch-name>
        <balance>        500 </balance>
    </account>
    <customer customer-id="C100" accounts="A-401">
        <customer-name>Joe        </customer-name>
        <customer-street> Monroe  </customer-street>
        <customer-city>     Madison</customer-city>
    </customer>
    <customer customer-id="C102" accounts="A-401 A-402">
        <customer-name> Mary     </customer-name>
        <customer-street> Erin      </customer-street>
        <customer-city>     Newark </customer-city>
    </customer>
</bank-2>
```

Limitations of DTDs

- No typing of text elements and attributes
  - All values are strings, no integers, reals, etc.
- Difficult to specify unordered sets of subelements
  - Order is usually irrelevant in databases
  - (A | B)* allows specification of an unordered set, but
    - Cannot ensure that each of A and B occurs only once
- IDs and IDREFs are untyped
  - The *owners* attribute of an account may contain a reference to another account, which is meaningless
    - *owners* attribute should ideally be constrained to refer to customer elements

Dublin Core Data in XML

- http://dublincore.org/documents/dc-xml-guidelines/

```
<?xml version="1.0"?>
<metadata
  xmlns="http://example.org/myapp/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://example.org/myapp/ http://example.org/myapp/schema.xsd"
  xmlns:dc="http://purl.org/dc/elements/1.1/">

  <dc:title>
    UKOLN
  </dc:title>
  <dc:description>
    UKOLN is a national focus of expertise in digital information
    management. It provides policy, research and awareness services
    to the UK library, information and cultural heritage communities.
    UKOLN is based at the University of Bath.
  </dc:description>
  <dc:publisher>
    UKOLN, University of Bath
  </dc:publisher>
  <dc:identifier>
    http://www.ukoln.ac.uk/
  </dc:identifier>
</metadata>
```

Alternative solution: metadata.JSON

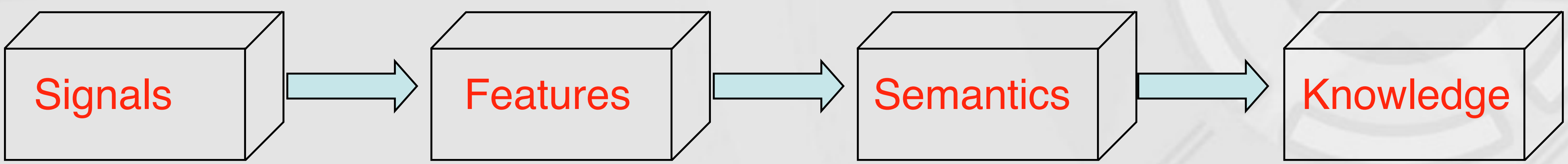
Generic metadata example

```
{
  // (rest of the document object)
  "metadata": {
    "field1__double": [ 1.0 ], // (single atomic type)
    "field2": [ "1", "2", "3", "4" ], // (array of atomic types)
    "field3": [ { "type": "simple" } ], // (single simple object)
    "field4": [ { "type": { "nested": true } } ], // (single nested object)
    "field3": [ { "type": "simple", "index": 1 }, { "type": { "nested": true }, "index": 2 } ], // (array of objects)
    // etc
  }
}
```

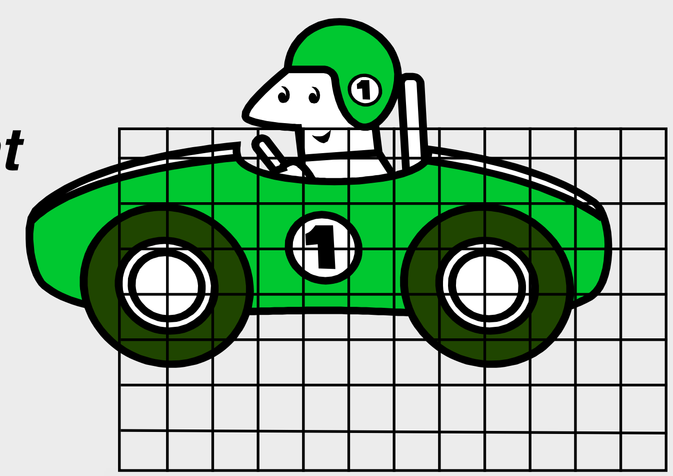
浙江大学计算机学院
数字媒体与网络技术

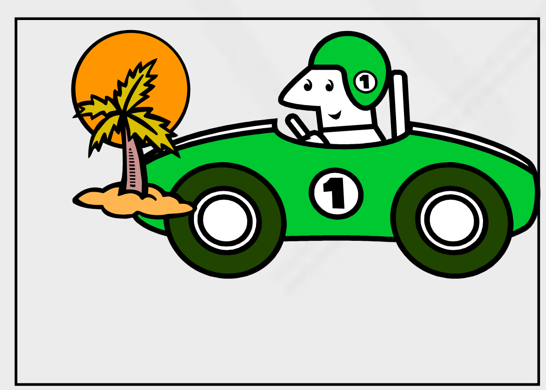
Domain-dependent Metadata Standards

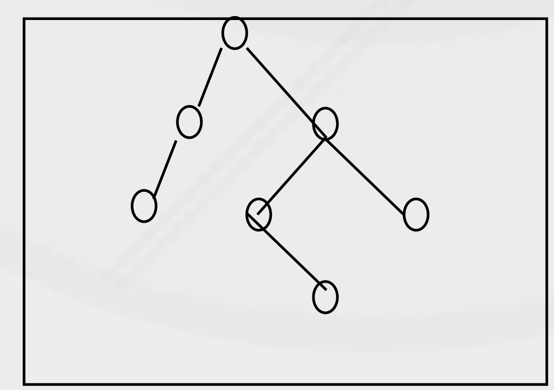
- MPEG series
  - Moving Picture Experts Group (MPEG) since 1998
  - responsible for developing standards of the coded representation of moving pictures and associated audio



Recent past







Near future

浙江大学计算机学院
数字媒体与网络技术

Domain-dependent Metadata Standards

Applications			
MPEG-1,-2,-4 Video storage Broadband, streaming video delivery	MPEG-4,-7 CBR Multimedia filtering Content adaptation	MPEG-7 Semantic-based retrieval and filtering Intelligent media services (iTV)	MPEG-21 Multimedia framework e-Commerce
Problems and Innovations			
Compression coding communications	Similarity search object- & feature- based coding	Modeling & classifying, personalization, summarization	Media mining, decision support

MPEG-1,-2 ,

MPEG-4 ,

MPEG-7 ,

MPEG-21 ,

MPEG-7

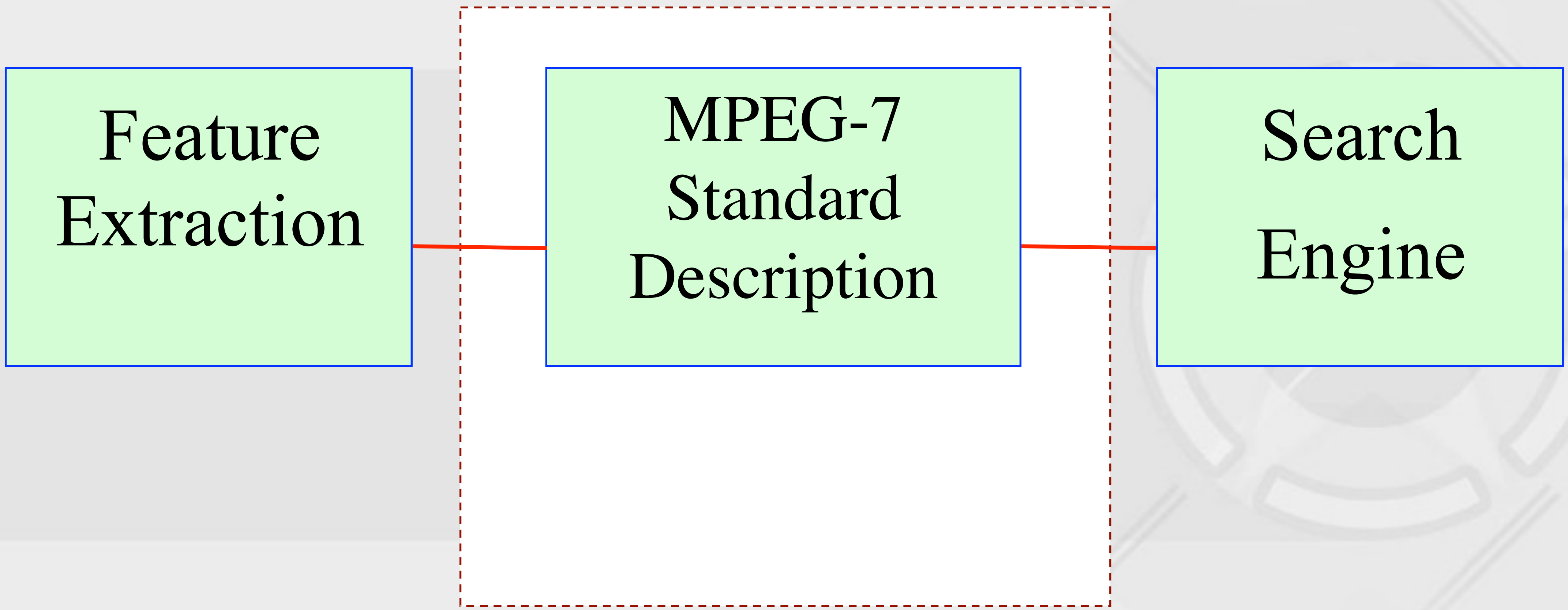
- ***Multimedia Content Description Interface***
  - Representation of information **about** the content
    - still pictures, graphics, 3D models, audio, speech, video & their combination
  - Goal:
    - to **support efficient search** for multimedia content using **standardized descriptions**
    - desirable to use textual information for the descriptions

</description>

<xml>

<resource>

Domain-independent Metadata Standards

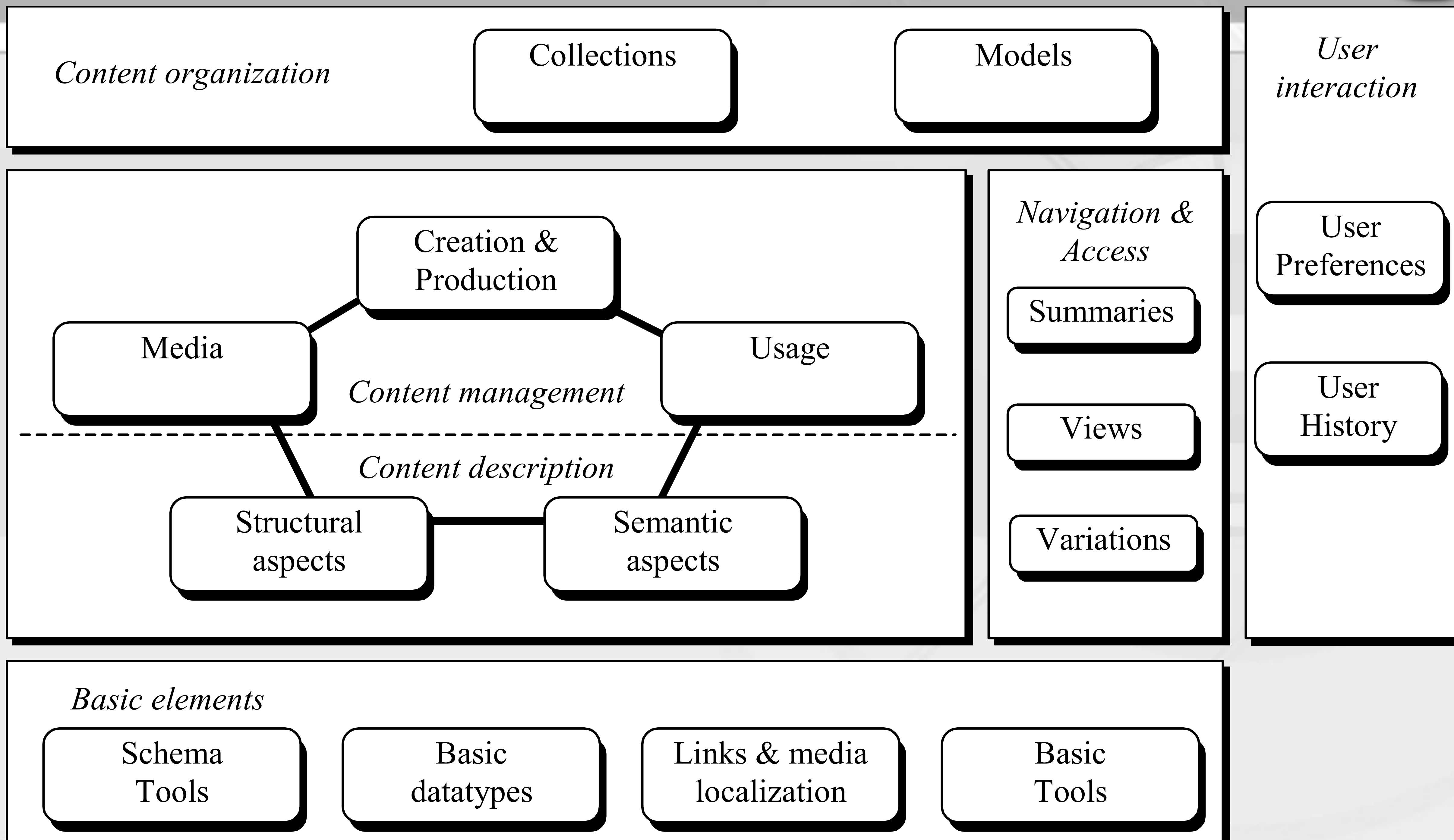


Scope of MPEG-7

MPEG-7

Set of description tools	Functionality
Media	Description of the storage media: typical features include the storage format, the encoding of the multimedia content, the identification of the media. Note that several instances of storage media for the same multimedia content can be described.
Creation & Production	Meta information describing the creation and production of the content: typical features include title, creator, classification, purpose of the creation, etc. This information is most of the time author generated since it cannot be extracted from the content.
Usage	Meta information related to the usage of the content: typical features involve rights holders, access right, publication, and financial information. This information may very likely be subject to change during the lifetime of the multimedia content.
Structural aspects	Description of the multimedia content from the viewpoint of its structure: the description is structured around segments that represent physical spatial, temporal or spatial-temporal components of the multimedia content. Each segment may be described by signal-based features (color, texture, shape, motion, and audio features) and some elementary semantic information.
Semantic aspects	Description of the multimedia content from the viewpoint of its semantic and conceptual notions. It relies on the notions of objects, events, abstract notions and their relationship.

MPEG-7

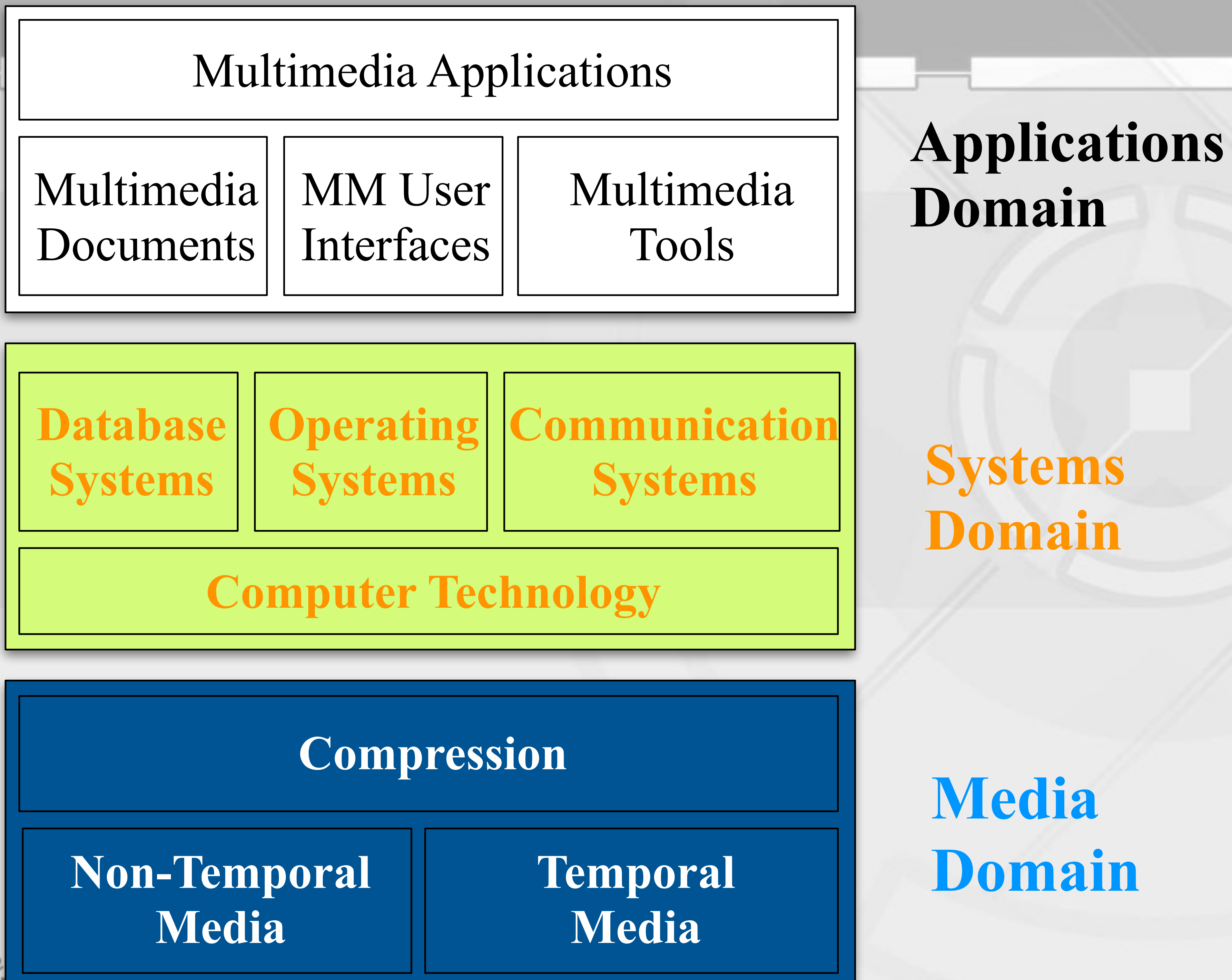

Content organization
Collections
Models
User interaction
Navigation & Access
Summaries
Views
Variations
User Preferences
User History
Creation & Production
Media
Usage
Content management
Content description
Structural aspects
Semantic aspects
Basic elements
Schema Tools
Basic datatypes
Links & media localization
Basic Tools

MPEG-7 Standard Elements

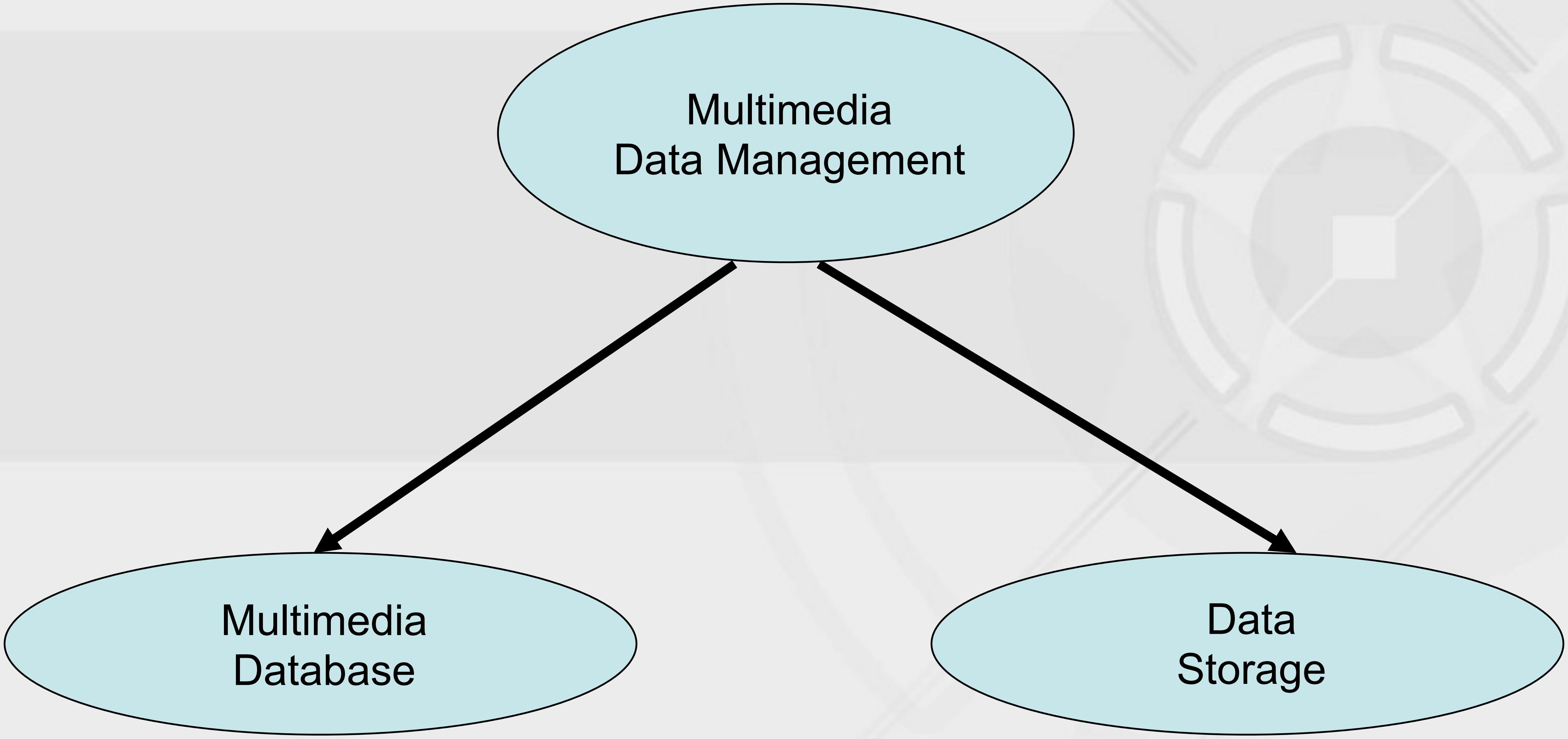
- **Descriptors** (Ds)
  - describe features, attributes, or groups of attributes of MM content
- **Description Schemes** (DSs)
  - a DS specifies the structure and semantics of the components (which may be other DSs, Ds, or datatypes)
- **Datatypes**
- **Classification Schemes** (CS):
  - lists of defined terms and meanings
- **System Tools**
- **Extensibility**
  - e.g., new DS's and D's; registration authority for CS

3.2.2. Multimedia Database System

Multimedia Architecture



Multimedia Database System

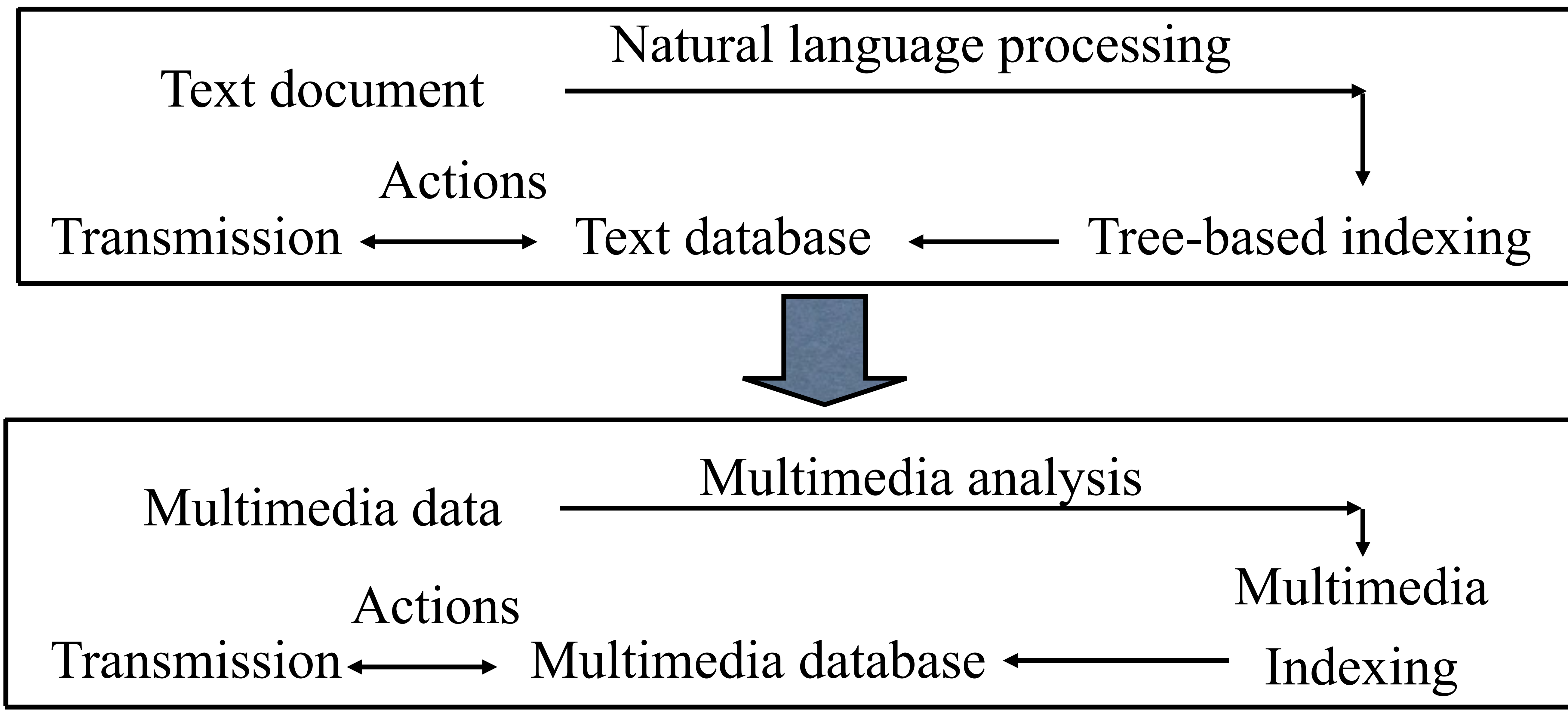


Multimedia Database System

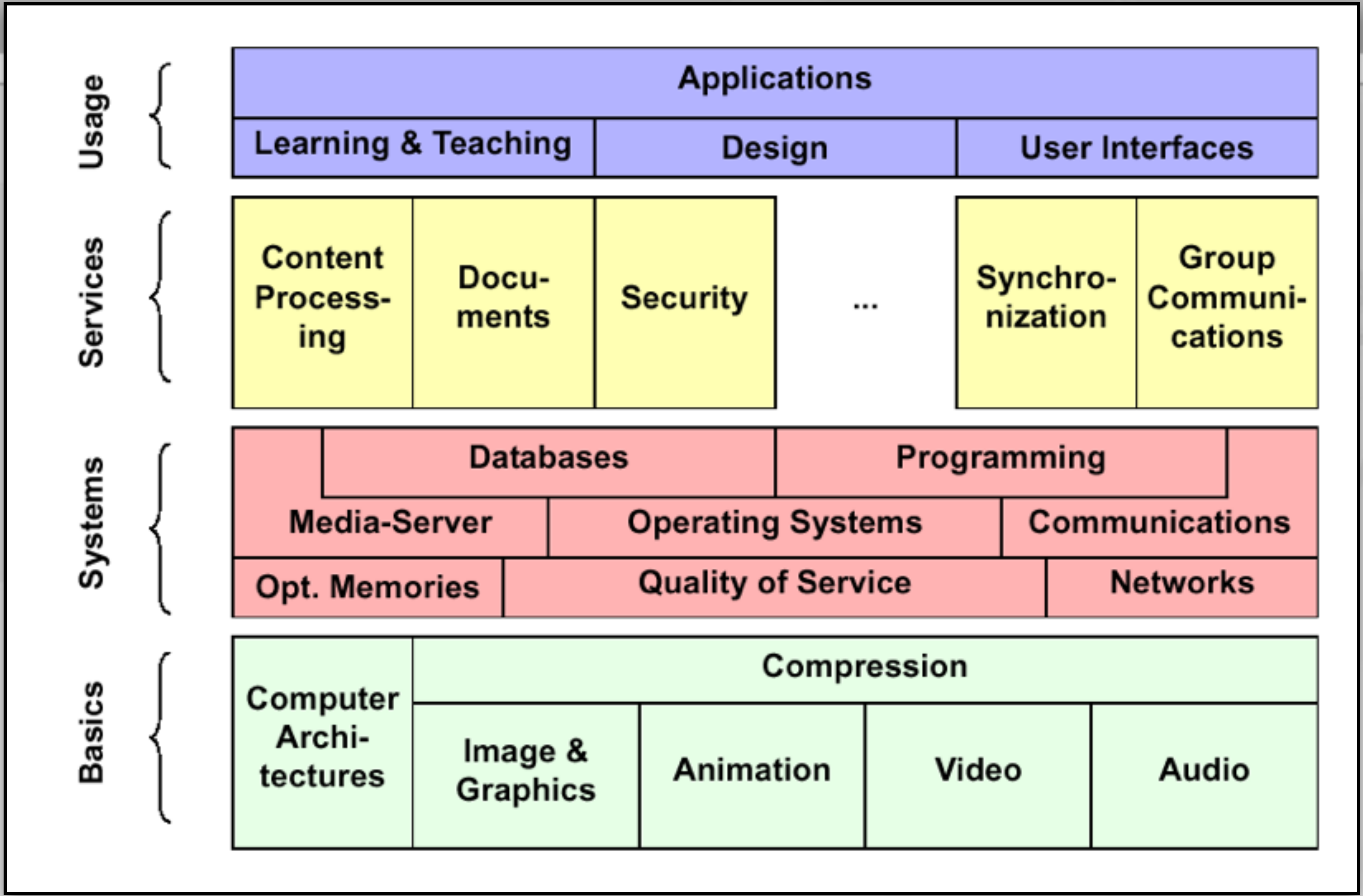
- **Multimedia database** v.s. **text database**
  - **Temporal data**: Requires temporal modeling
  - **Huge amount of data**: Compression helps get around this.
  - Data is **not easily indicative** of the information
  - Requires a lot of **pre-processing** in order to store data efficiently:
    - PCA, feature extraction and segmentation
  - **Novel Query mechanisms**
  - **Hypermedia**: The ability to interactively move around in the data.

How to Build Multimedia Database Systems?

How to build text database? *Yahoo, Google*



Scope



Usage
Applications
Learning & Teaching
Design
User Interfaces
Services
Content Process-ing
Docu-ments
Security
...
Synchro-nization
Group Communi-cations
Systems
Databases
Programming
Media-Server
Operating Systems
Communications
Opt. Memories
Quality of Service
Networks
Basics
Computer Archi-tectures
Compression
Image & Graphics
Animation
Video
Audio

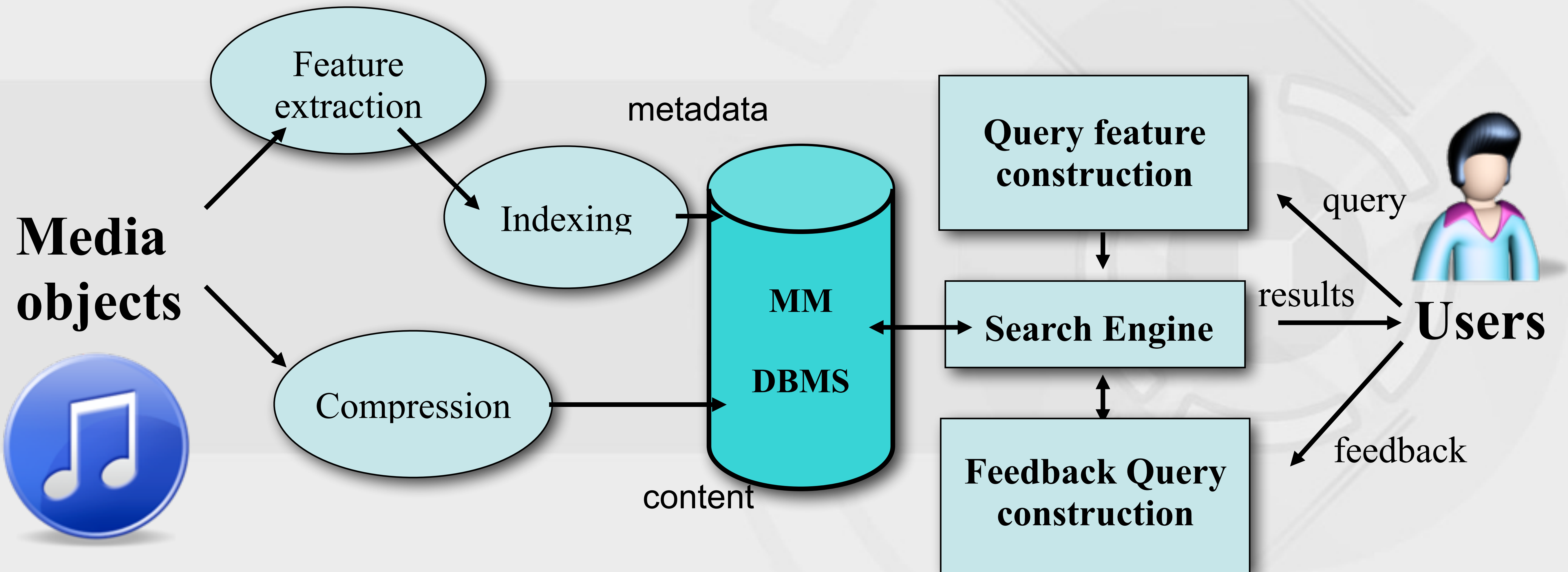
A Reference Architecture for MMDB System

- **Considerations:**
  - **Real time aspects/constraints impose strong demands on the systems**
    - Simultaneous presentation of multimedia objects may cause performance problems.
  - **Data Sharing**
    - Due to the possibly very large multimedia data, traditional replicated data technique may not be applicable, hence data sharing is essential
  - **Multiple Client/ Multiple Server Architecture**

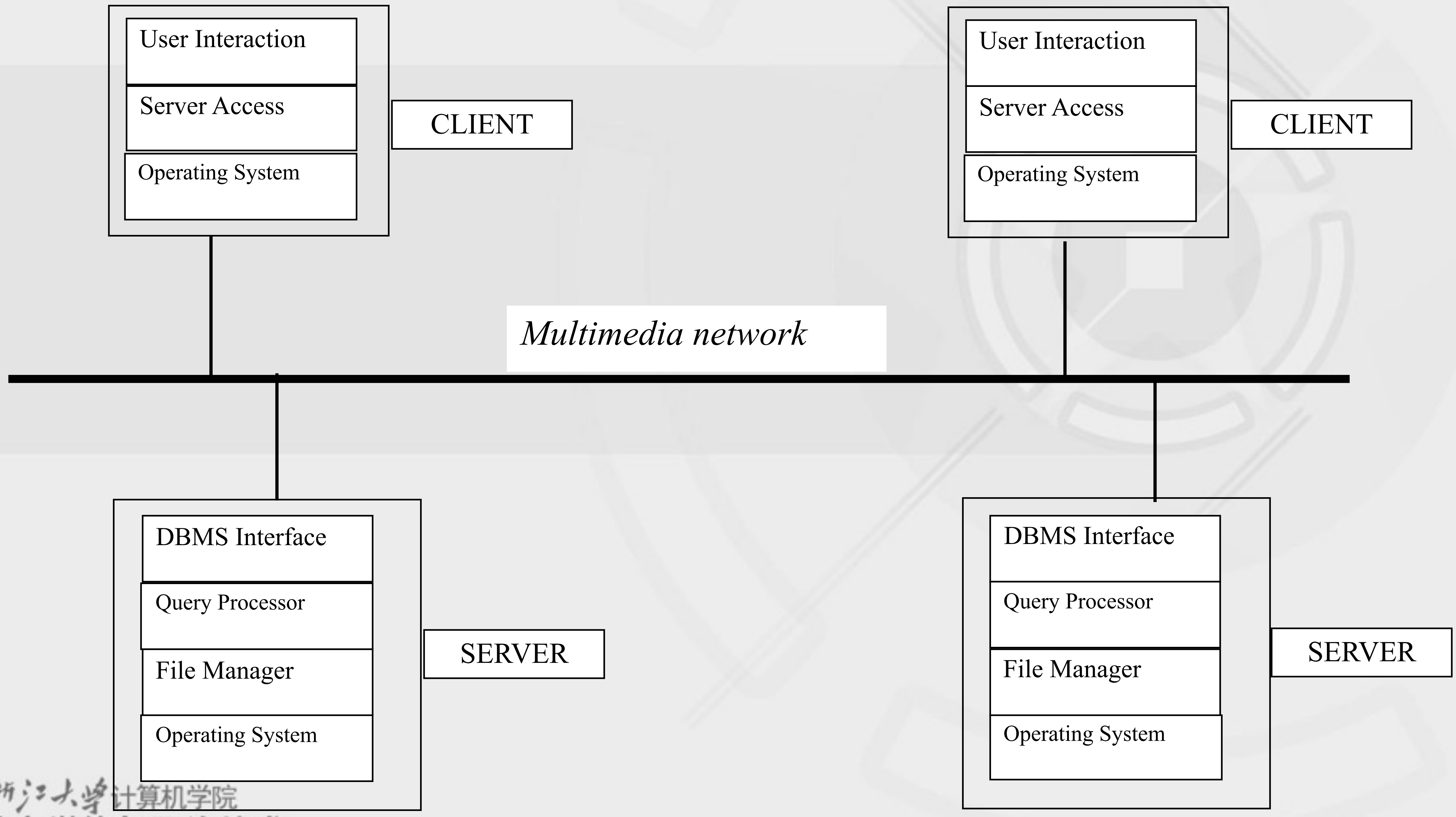
A Reference Architecture for MMDB System

- **Considerations:**
  - **Real time aspects/constraints**
  - **Data Sharing**
  - **Multiple Client/ Multiple Server Architecture**
    - Many multimedia applications work with data that are stored on remote sites (e.g, VOD, tele-learning), which suggests for client / server architecture.
    - A **client** consists of **three** layers…
      - **User Interaction** – takes care of input and output of multimedia data
      - **Server Access** – allows searching of servers by the client
      - **Operating System** – not a real part of the MMDBS
    - A **server** consists of **four** layers:
      - **DBMS Interface**
      - **Query Processor**
      - **File Manager**
      - **Operating System**

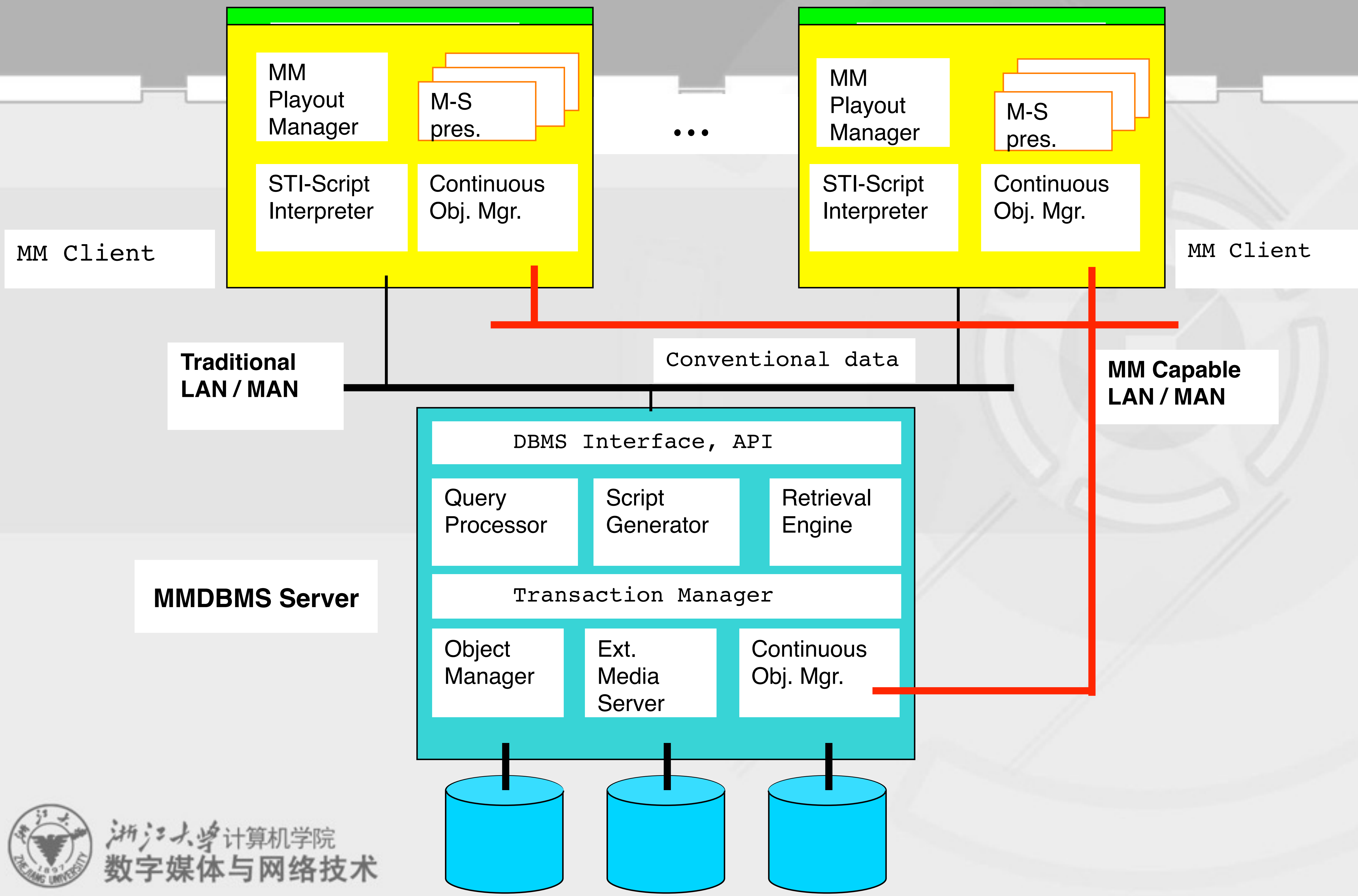
A Generic Architecture of MMDBMS



MMDB Reference Architecture: "Simplified View"



Detailed View of MMDB Architecture



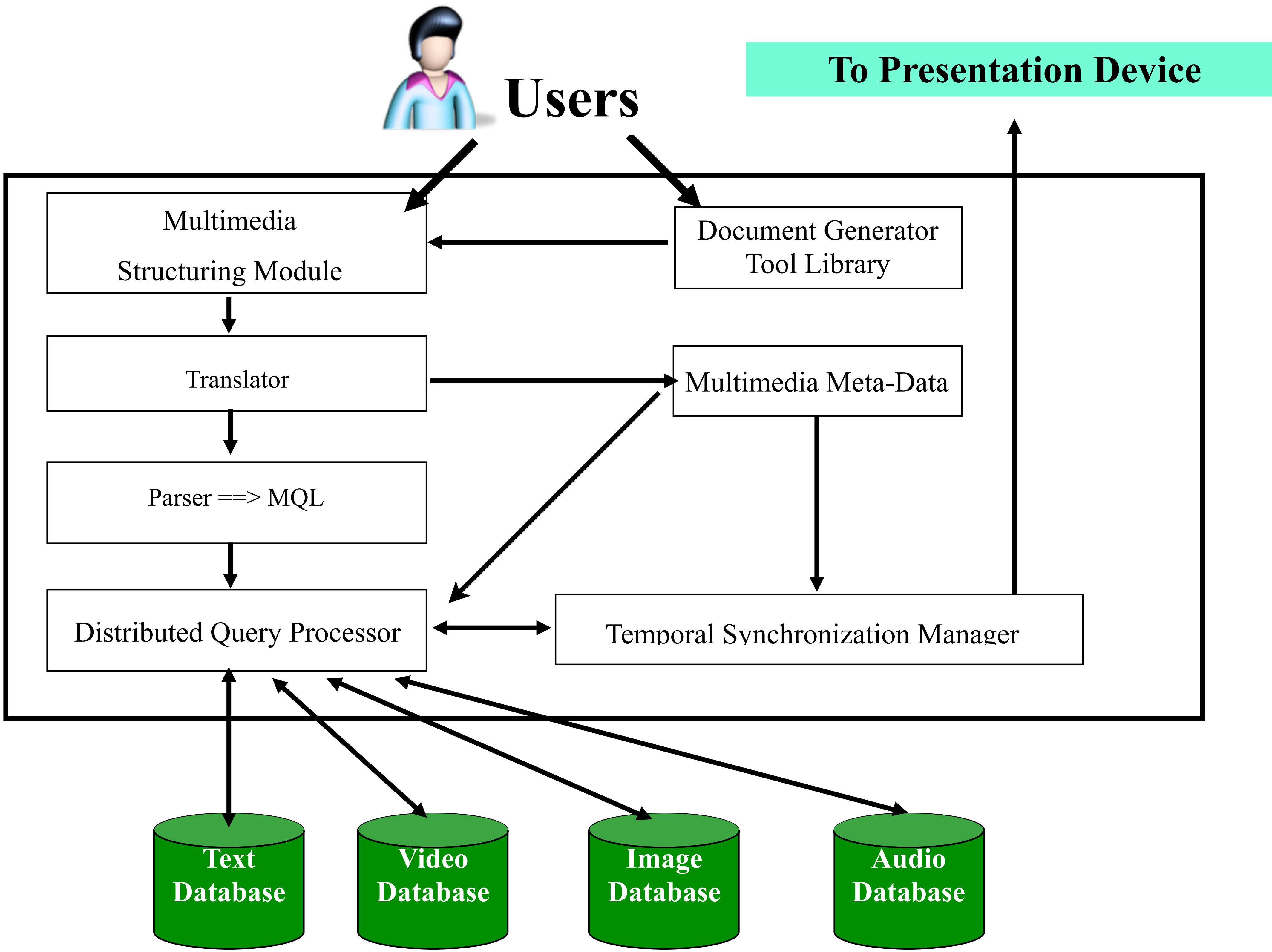
浙江大学计算机学院
数字媒体与网络技术

MMDBMS Development

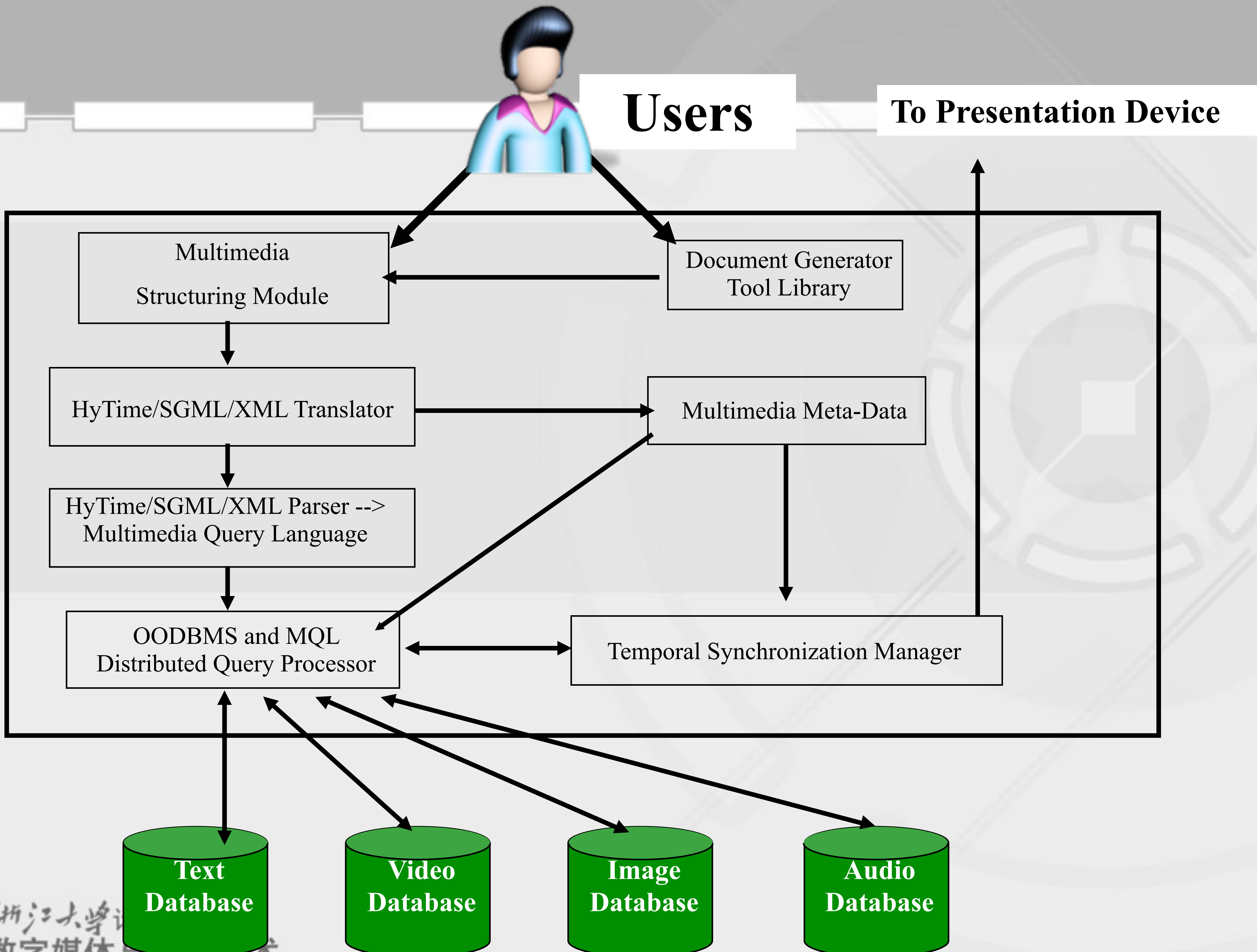
Major steps in developing MMDBMS

1. **Media acquisition:**
   - collect media data from various sources, such as WWW, CD, TV, etc.
2. **Media processing:**
   - extract media representations and their features, including noise filtering, rending, etc.
3. **Media storage:**
   - store the data and their features in the system based on application requirement.
4. **Media organization:**
   - organize the features for retrieval. i.e., indexing the features with effective structures.
5. **Media query processing:**
   - Accommodated with indexing structure, efficient search algorithm with similarity function should be designed.

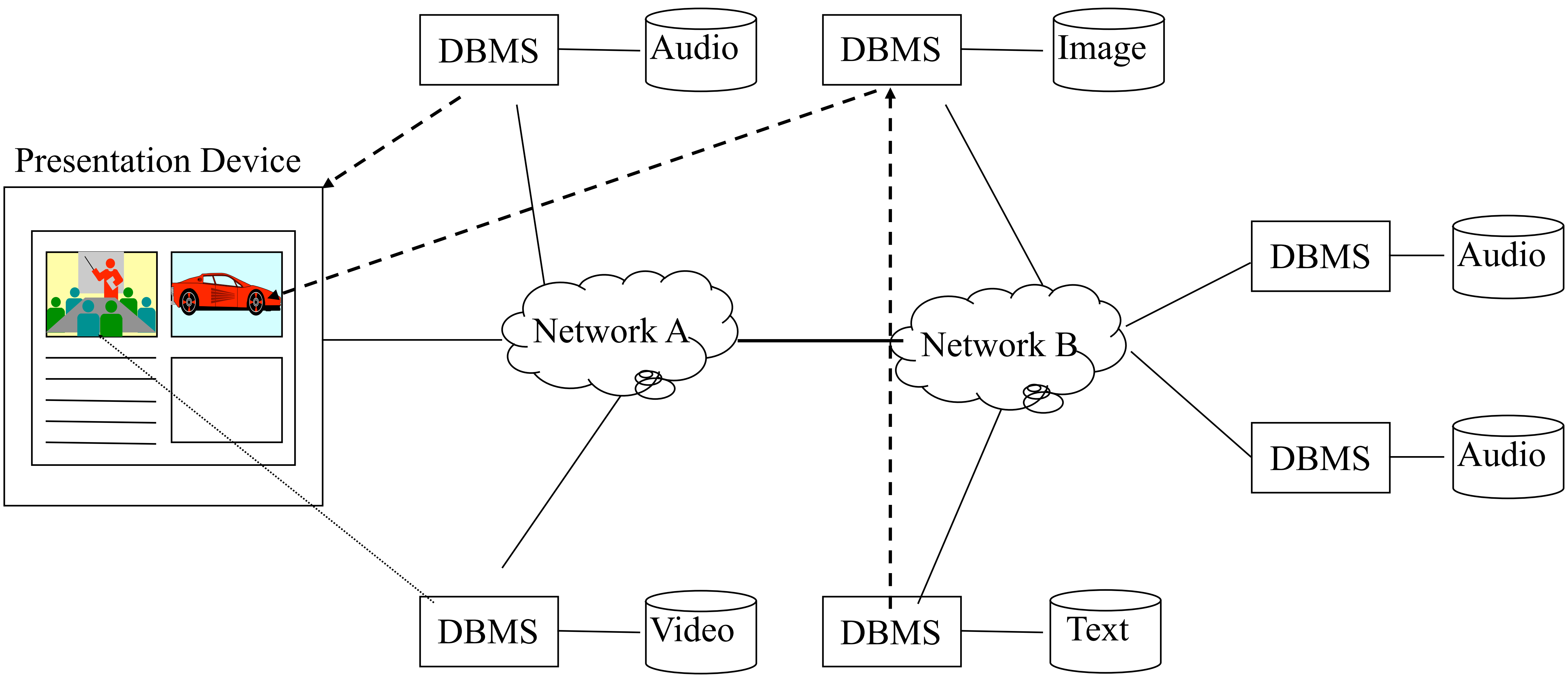
Software Architecture of MMDBMS



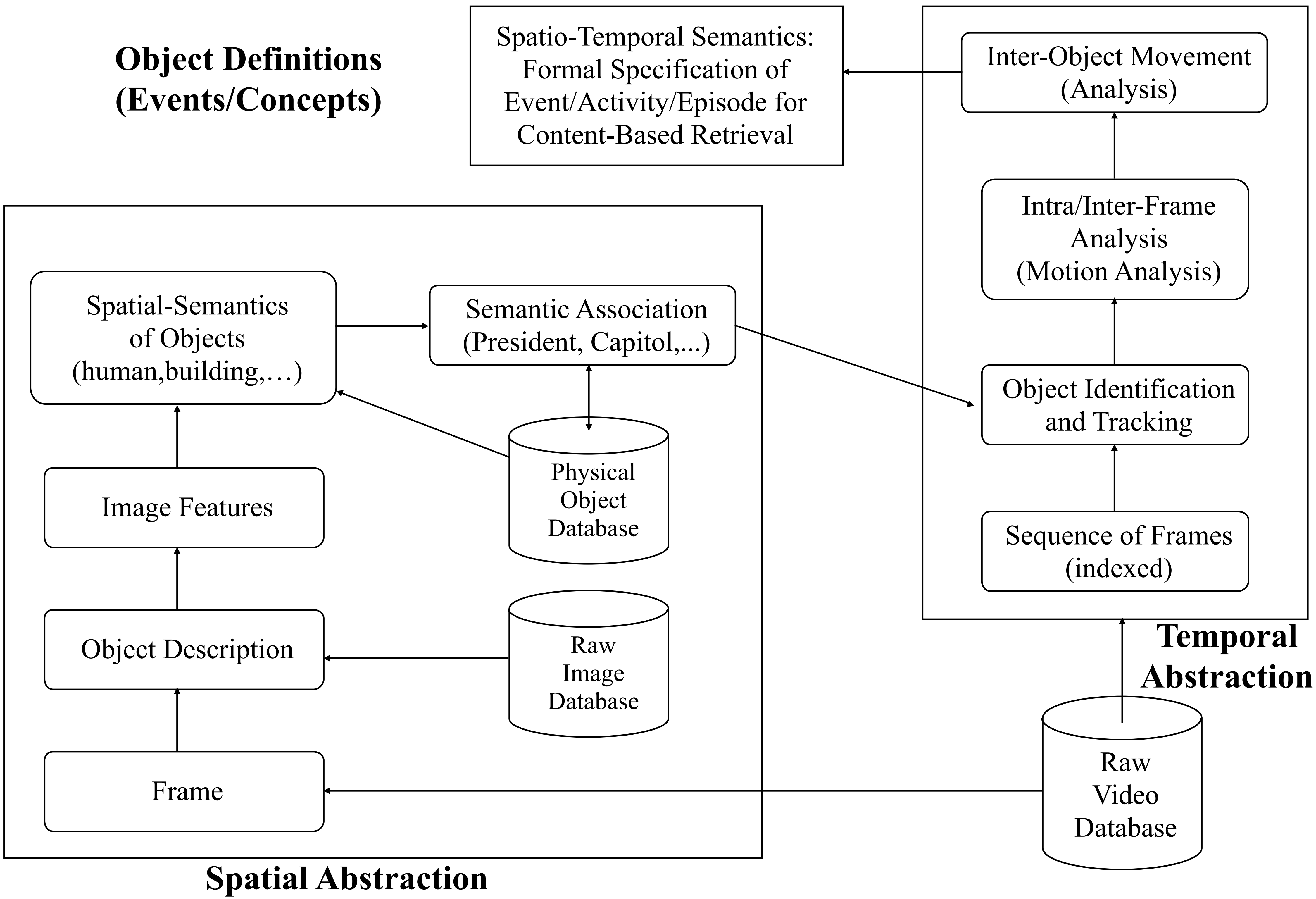
Software Architecture of MMDBMS



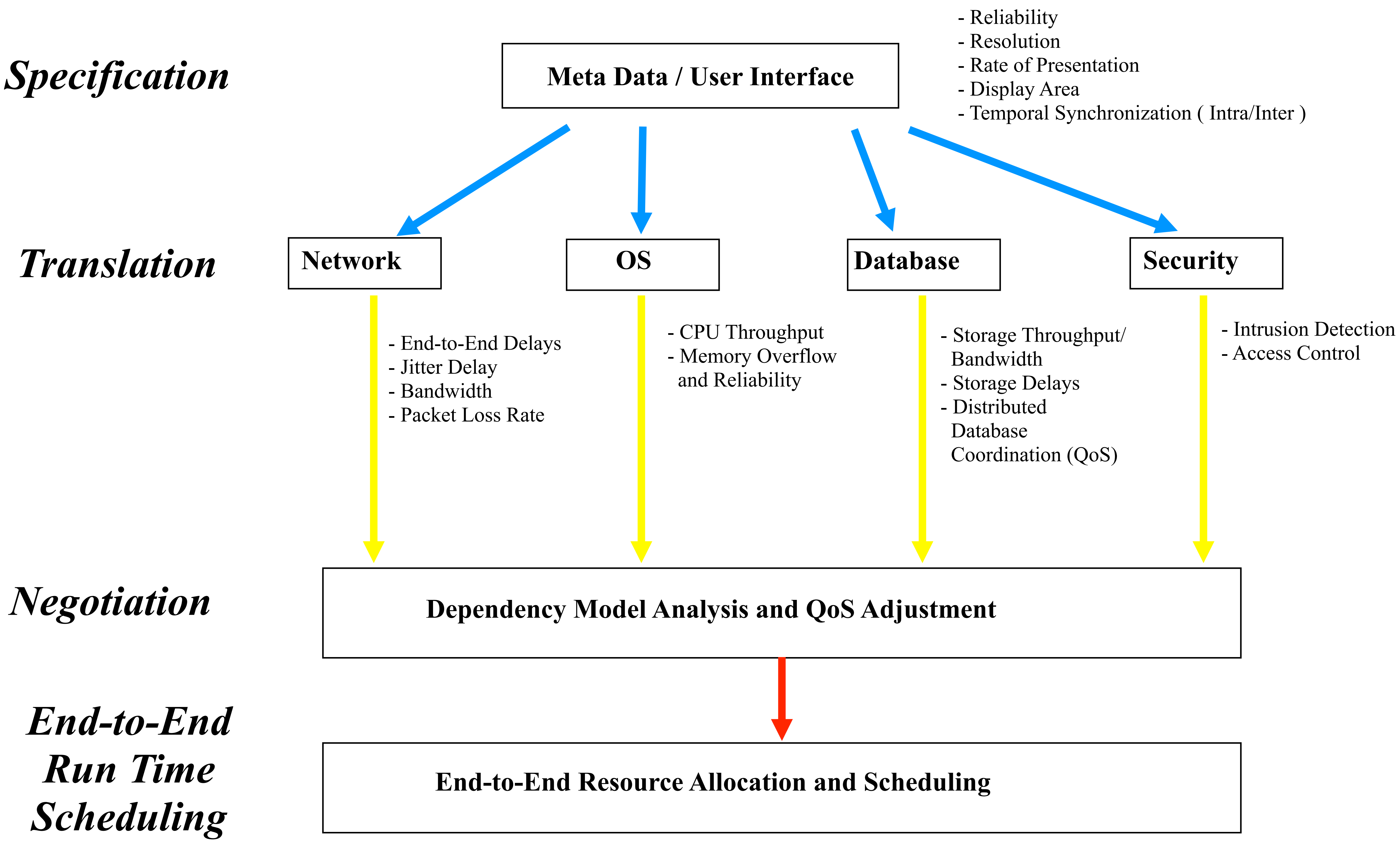
Distributed Multimedia Database Systems



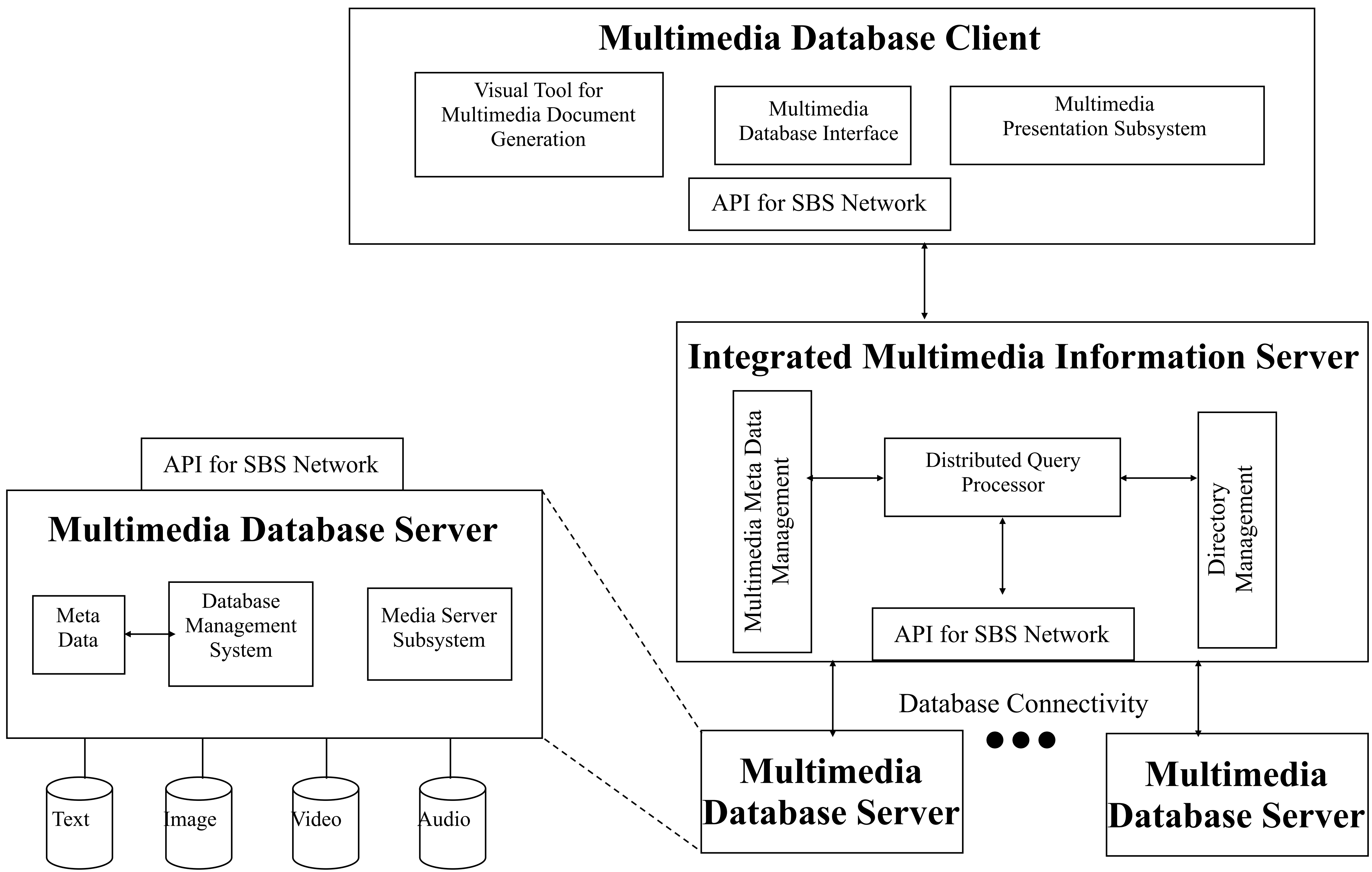
An Architecture for Video Database System



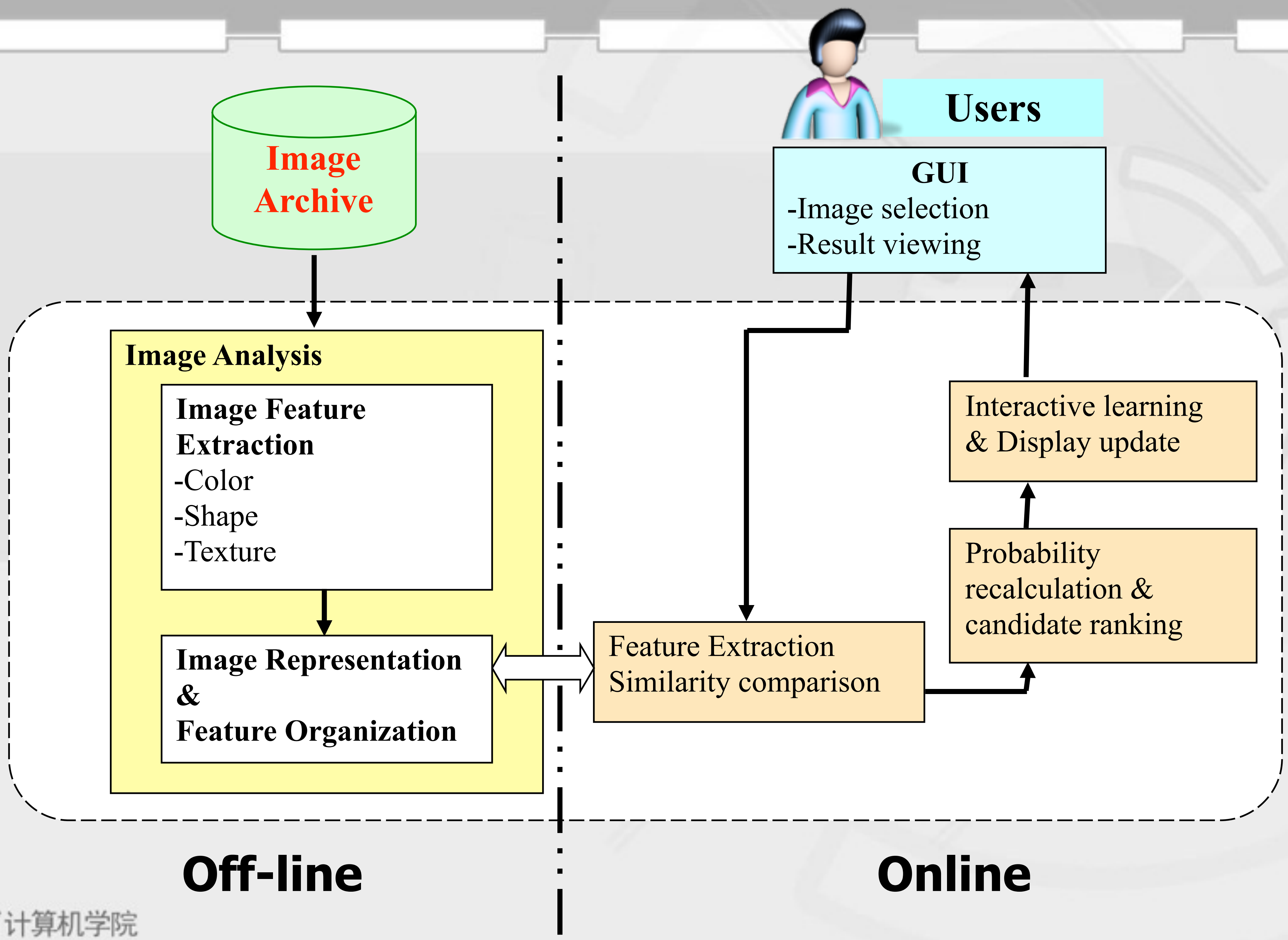
End-to-End QoP / QoS Management



Architecture of a Distributed Multimedia Database Management



Overview of the System



3.2.3 Multimedia System Service Model

What is a Media Service/Server?

- A scalable storage manager
  - **Allocates multimedia data optimally** among disk resources
  - Performs memory and disk-based **I/O optimization**
- Supports
  - real-time and non-real-time **clients**
  - presentation of **continuous-media data**
  - mixed workloads: schedules the **retrieval of blocks**
- Performs **admission control**

Service Models

- Random Access
  - **Maximize the number of clients** that can be served concurrently at any time with a low response time
  - **Minimize latency** (等待时间)

- Enhanced Pay-per-view (EPPV)
  - **Increase the number of clients** that can be serviced concurrently beyond the available disk and memory bandwidth, while guaranteeing a constraint on the response time

Service Models

- Example
  - –**Server**

> - 50 movies, 100 min. each
> - Request rate: 1 movie/min
> - Max. capacity: 20 streams

- Random Access Model
  - Case 1: after 20 movies, no more memory left. 21st movie waits for 80 minutes, 22nd movie waits for 81 minutes …
  - Case 2: after 20 movies, more memory can be allocated. 21st movie has to wait (initial latency) till one round of the previous 20 movies each has been served.

- EPPV Model:
  - At any time 20 movies are served, movies are initiated every 5 minutes
  - Streams are distributed uniformly during these 20 minutes

3.3 Multimedia Data Storage

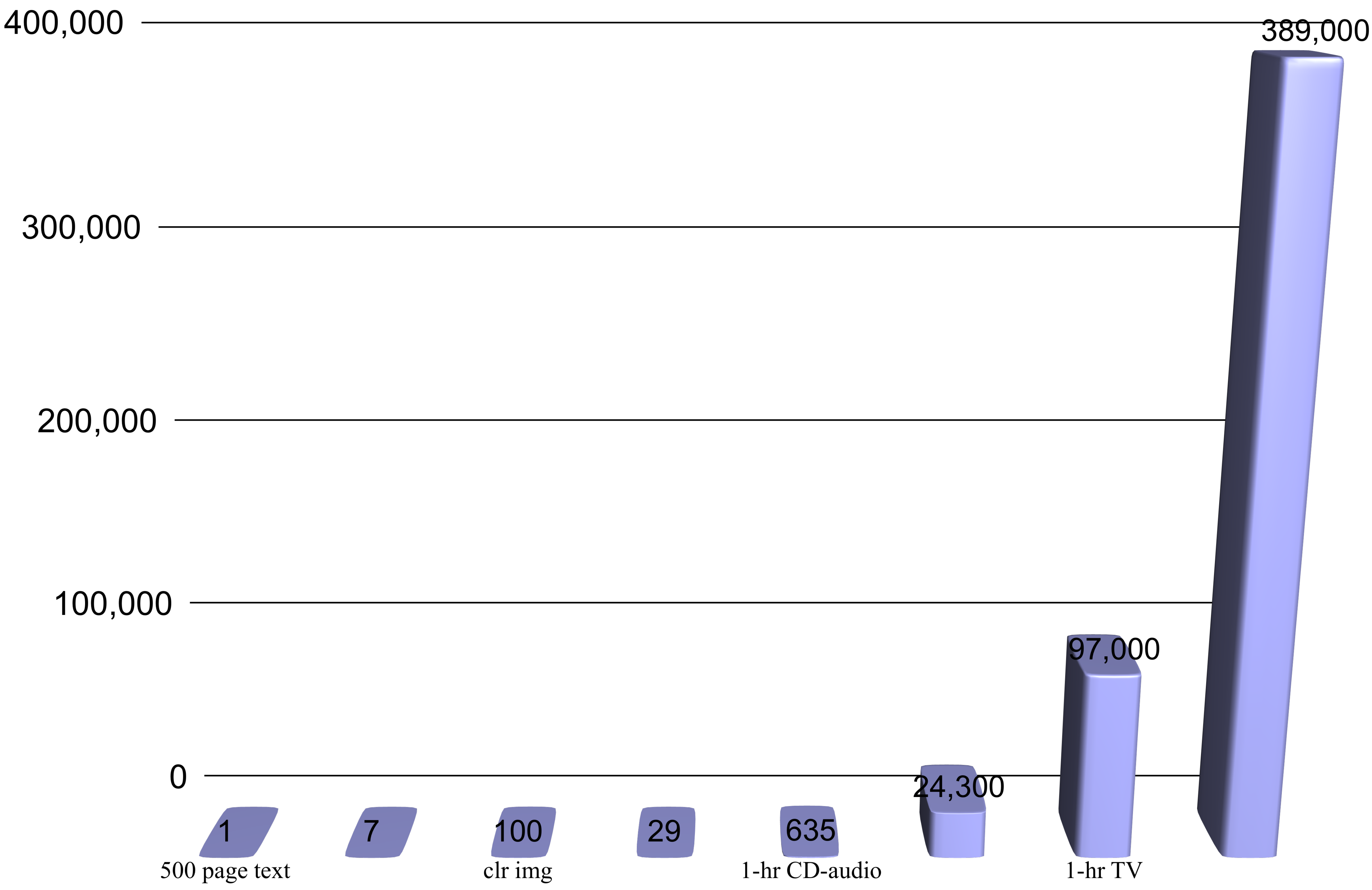
Multimedia Data Storage

- Storage Requirements
- RAID Technology
- Optical Storage Technology

Requirements of MM Information

- Storage and Bandwidth Requirement
  - measured in bytes or Mbytes for storage
  - measured in bits/s or Mbits/s for bandwidth
    - An image 480 x 600 (24 bits per pixel),
      - 864k bytes (without compression).
      - To transmit it within 2 sec => 3.456Mb/s.
    - 1GB Hard-disk
      - 1.5 hr. of CD-audio or
      - 36 seconds of TV quality video
      - require 800s to be transferred (10Mbits/s network).

Storage & Bandwidth Requirements



Delay and Delay Jitter Requirements

- Digital audio and video are **time-dependent** continuous media
- **dynamic media** => achieve **a reasonable quality playback** of audio and video, media samples must be received and played back at regular intervals.
- E.g. audio playback, 8K samples/sec have to be achieved

- **End-to-end delay is the sum of all delays** in all the components of a MM system, disk access, ADC, encoding, host processing, network access & transmission, buffering, decoding, and DAC

  In most conversation type applications, end-to-end delay should be kept below **300ms**

- Delay variation is commonly called **delay jitter**. It should be small enough to achieve smooth playback of continuous media, e.g.,

  < **10ms** for telephone-quality voice and TV-quality video,

  < **1ms** for stereo effect in high quality audio.

Other Requirements

Quest for Semantic Structure

- For alphanumeric information, **computer can search & retrieve alphanumeric items** from a DB or document collection.
- It is **hard to automatically retrieve** digital audio, image, & video as no semantic structure is revealed from the series of sampled values

Spatial-Temporal Relationship Among Related Media

- **Retrieval and transmission of MM data must be coordinated and presented** so that their specified temporal relationship are maintained for presentation
- A synchronization scheme therefore defines **the mechanisms used to achieve the required degree of synchronization**
- Two areas of works: user-oriented and system-oriented synchronization

Other Requirements

Error and Loss Tolerance

- Unlike alphanumeric information, we can tolerate some error or loss in MM
- For voice, we can tolerate a bit error rate of 10^{-2}
- For images and video, we can tolerate a bit rate from 10^{-4} to 10^{-6}.
- Another parameter: **packet loss rate** - a much more stringent requirement

Text v.s. MM Data Requirements

Characteristics	Text-based Data	Multimedia Data
Storage Req.	Small	Large
Data Rate	Low	High
Traffic Pattern	Bursty	Stream-oriented, highly bursty
Error/Reliability Req.	No loss	Some loss
Delay/Latency Req.	None	Low
Temporal Relationship	None	Synchronized Trans.

Quality of Service (QoS)

- To provide a uniform framework to specify and guarantee these diverse requirement, a concept called QoS has been introduced.
- QoS is **a set of requirement, but there is no universally agreed one**.
- QoS is **a contract** negotiated and agreed **among MM applications and MM system** (service provider)
- The QoS requirement is normally specified in two grades: the preferable quality and the acceptable one.
- The QoS guarantee can be in one of three forms: hard or deterministic (fully satisfied), soft or statistic (guaranteed with a certain probability), and best effort (no guarantee at all)
- A lot of research issues are involved and still undergoing!!

File Systems

- **The most visible part** of an operating system.
- **organization of the file system**
  - an important factor for the usability and convenience of the operating system.
- Files are stored in secondary storage, so they can be used by different applications.
- In **traditional file systems**, the information types stored in files are sources, objects, libraries and executables of programs etc.
- In **multimedia systems**, the stored information also covers digitized video and audio with their related real-time "**read**" and "**write**" demands.
- ===>>> additional requirements in the design and implementation

File Systems

Traditional File Systems

- The main goals of traditional files systems are:
  - to **provide** a comfortable interface for file access to the user
  - to **make** efficient use of storage media
  - to **allow** arbitrary deletion and extension of files

Multimedia File Systems

- the main goal is to provide a **constant and timely retrieval** of data.
- It can be achieved through providing enough buffer for each data stream and the employment of disk scheduling algorithms, especially optimized for real-time storage and retrieval of data.

Multimedia File Systems

- The much greater size of continuous media files and the fact that they will usually be retrieved sequentially are reasons for an *optimization of the disk layout*
- Continuous media streams predominantly **belong to the *write-once-read-many nature* (ROM?)**, and streams that are recorded at the same time are likely to be played back at the same time.
- Hence, it seems to be reasonable to *store continuous media data in large data blocks contiguously on disk*.
- Files that are likely **to be retrieved together are grouped together on the disk**.
- With such a disk layout, the buffer requirements and seek times decrease.
- The disadvantage of the continuous approach is **external fragmentation** and **copying overhead** during insertion and deletion.

Data Management & Disk Spanning

Data Management:

- **Command queuing**: allows execution of multiple sequential commands with system CPU intervention. It helps in minimizing head switching and disk rotational latency.
- **Scatter-gather**: scatter is a process whereby data is set for best fit in available block of memory or disk. Gather reassembles data into contiguous blocks on disk or in memory.

Disk Spanning

- Attach **multiple devices** to a **single host adapter**.
- good way to increase storage capacity by adding incremental drives.

RAID Redundant Arrays of Inexpensive Disks

- **By definition RAID has three attributes:**
  - a set of disk drives viewed by the user as one or more logical drives
  - data is distributed across the set of drives in a pre-defined manner
  - redundant capacity or data reconstruction capability is added, in order to recover data in the event of a disk failure

- **Objectives of RAID**
  - Hot backup of disk systems (as in mirroring)
  - Large volume storage at lower cost
  - Higher performance at lower cost
  - Ease of data recovery (fault tolerance)
  - High MTBF (mean time between failure)

Different Levels of RAID

- **Eight discrete levels of RAID functionality**
    - Level 0 - disk striping
    - Level 1 - disk mirroring
    - Level 2 - bit interleaving and Hamming Error Correction (HEC) parity
    - Level 3 - bit interleaving and XOR parity
    - Level 4 - block interleaving with XOR parity
    - Level 5 - block interleaving with parity distribution
    - Level 6 - Fault tolerant system
    - Level 7 - Heterogeneous system
- Data is spread across the drives in units of **512 bytes** called **segments**. **Multiple segments form a block.**

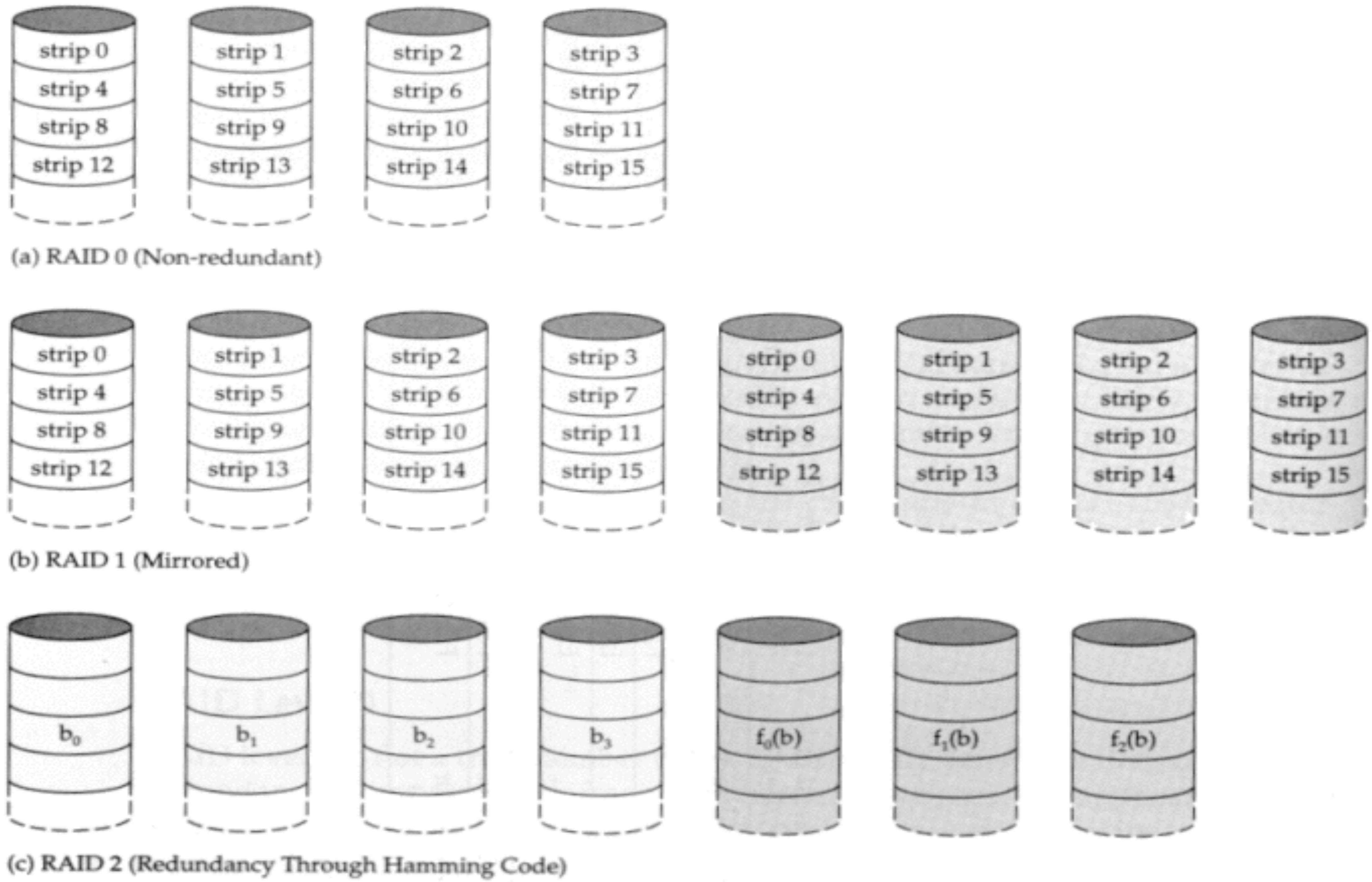
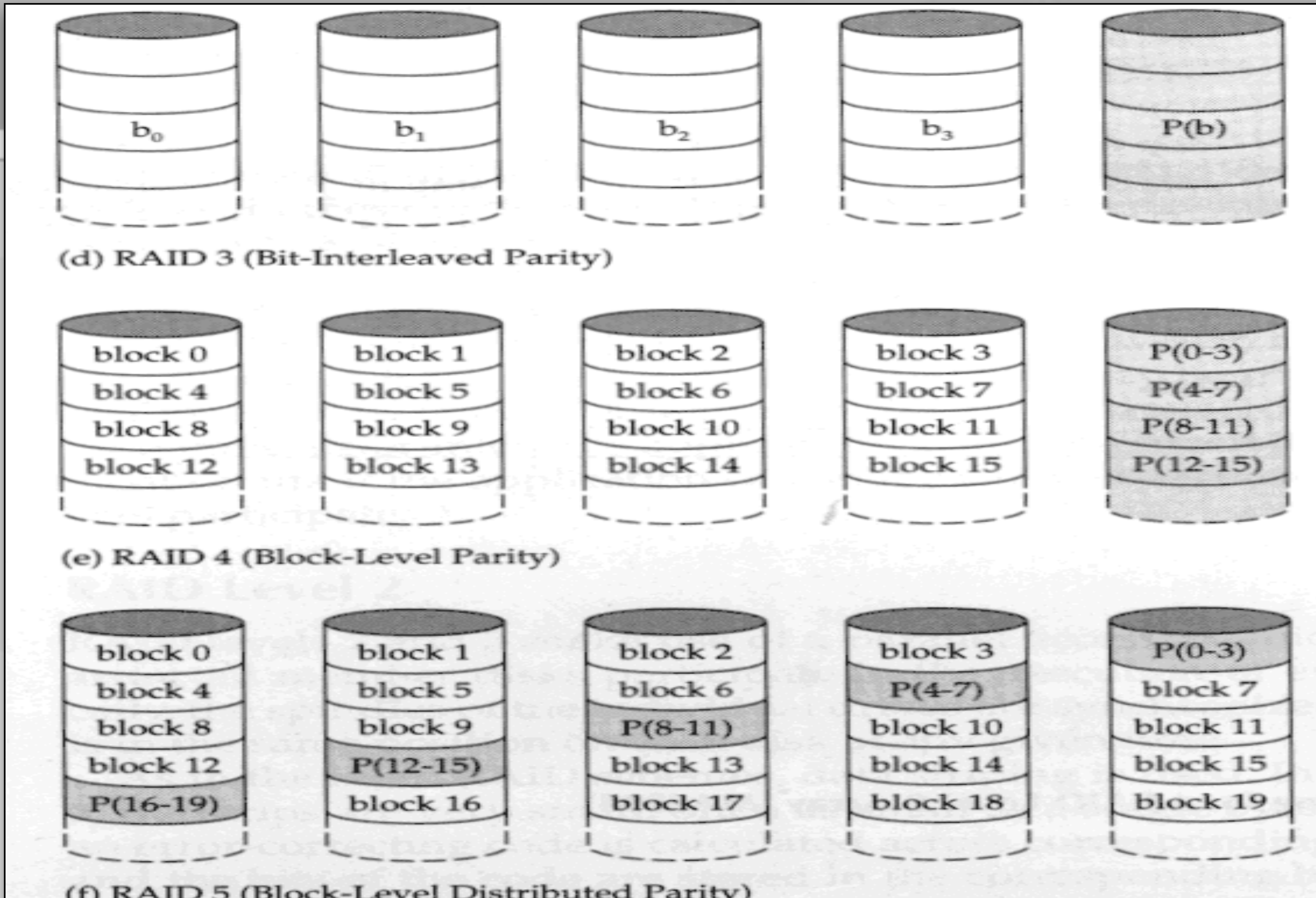
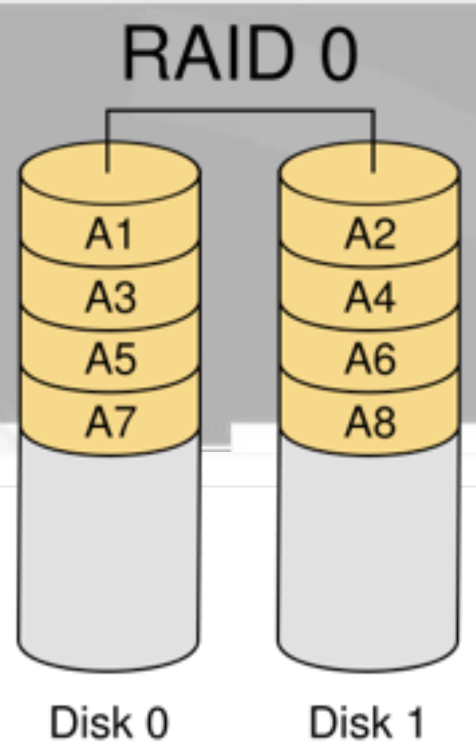


FIGURE 5.5. RAID Levels


b0
b1
b2
b3
P(b)
(d) RAID 3 (Bit-Interleaved Parity)
block 0 block 1 block 2 block 3 P(0-3)
block 4 block 5 block 6 block 7 P(4-7)
block 8 block 9 block 10 block 11 P(8-11)
block 12 block 13 block 14 block 15 P(12-15)
(e) RAID 4 (Block-Level Parity)
block 0 block 1 block 2 block 3 P(0-3)
block 4 block 5 block 6 P(4-7) block 7
block 8 block 9 P(8-11) block 10 block 11
block 12 P(12-15) block 13 block 14 block 15
P(16-19) block 16 block 17 block 18 block 19
(f) RAID 5 (Block-Level Distributed Parity)

RAID Level 0 - Disk Striping

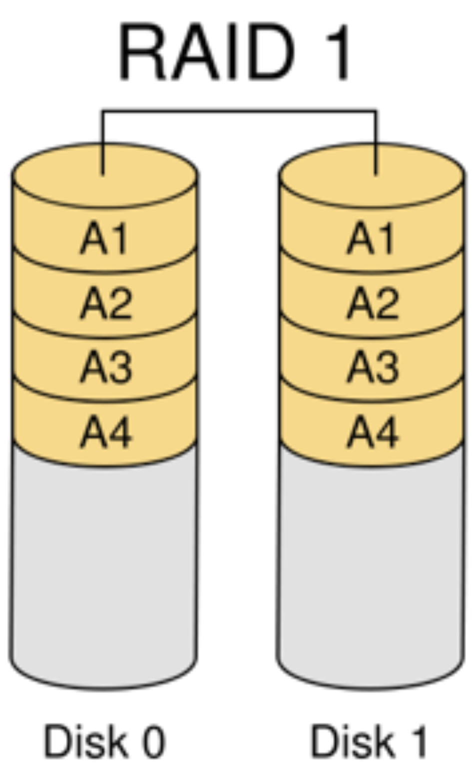


- To improve performance by overlapping disk reads and writes
- Multiple drives connected to a single disk controller
- Data is striped to spread segments of data across multiple drives in block sizes ranging from 1 to 64 Kbytes
- Disk striping provides a higher transfer rate for write and retrieve block of data
- Typical application: database applications
- Drawbacks:
  - If one drive fails, the whole drive system fails
  - Does not offer any data redundancy, no fault tolerance

RAID Level 1 - Disk Mirroring

- Each main drive has a **mirror drive**
- Two copies of every file will write to two separate drives

  complete redundancy



- Performance:
  - * Disk write : take almost twice time
  - * Disk read : can be speed up by overlapping seeks
- Typical use:
  - * in file servers provides backup in the event of disk failure
- Duplexing:
  - * Use two separate controllers
  - * The second controller enhances both fault tolerance and performance
  - * Separate controllers allow parallel writes and parallel reads

RAID Level 2
- Bit Interleaving and HEC Parity

- Contain arrays of multiple drives connected to a disk array controller.
- Data is written interleaved across multiple drives (often one bit at a time) and multiple check disks are used to detect and correct errors.
- **Hamming error correction (HEC)** code is used for error detection and correction.
- The drive spindles must be **synchronized** as a single I/O operation accesses all drives
- Benefits:
  * High level of data integrity and reliability (error correction feature)
  * Mainly use for **supercomputers** to access large volumes of data with a small number of I/O request.

RAID Level 2

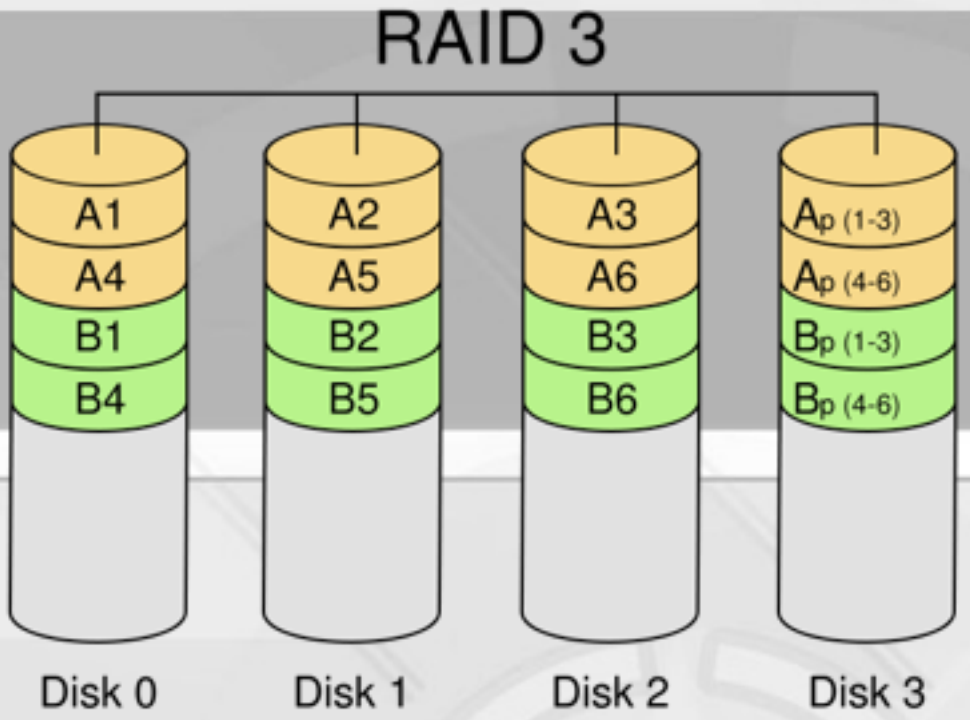
- Bit Interleaving and HEC Parity

Drawbacks:

- **Expensive** - requires **multiple drives** for error detection and correction
- **Error-correcting scheme**: **slow** and **cumbersome**
- Multimedia applications can afford to **lose occasional bit** or there without any significant impact on the system or the display quality.
- Each sector on a drive is associated with sectors on other drives to form a single storage unit, it takes multiple sectors across all data drives to storage even just a few bytes, **resulting in waste of storage**.
- Should not be used for transaction processing where **the data size of each transaction is small**.

RAID Level 3

- Bit Interleaving with XOR Parity

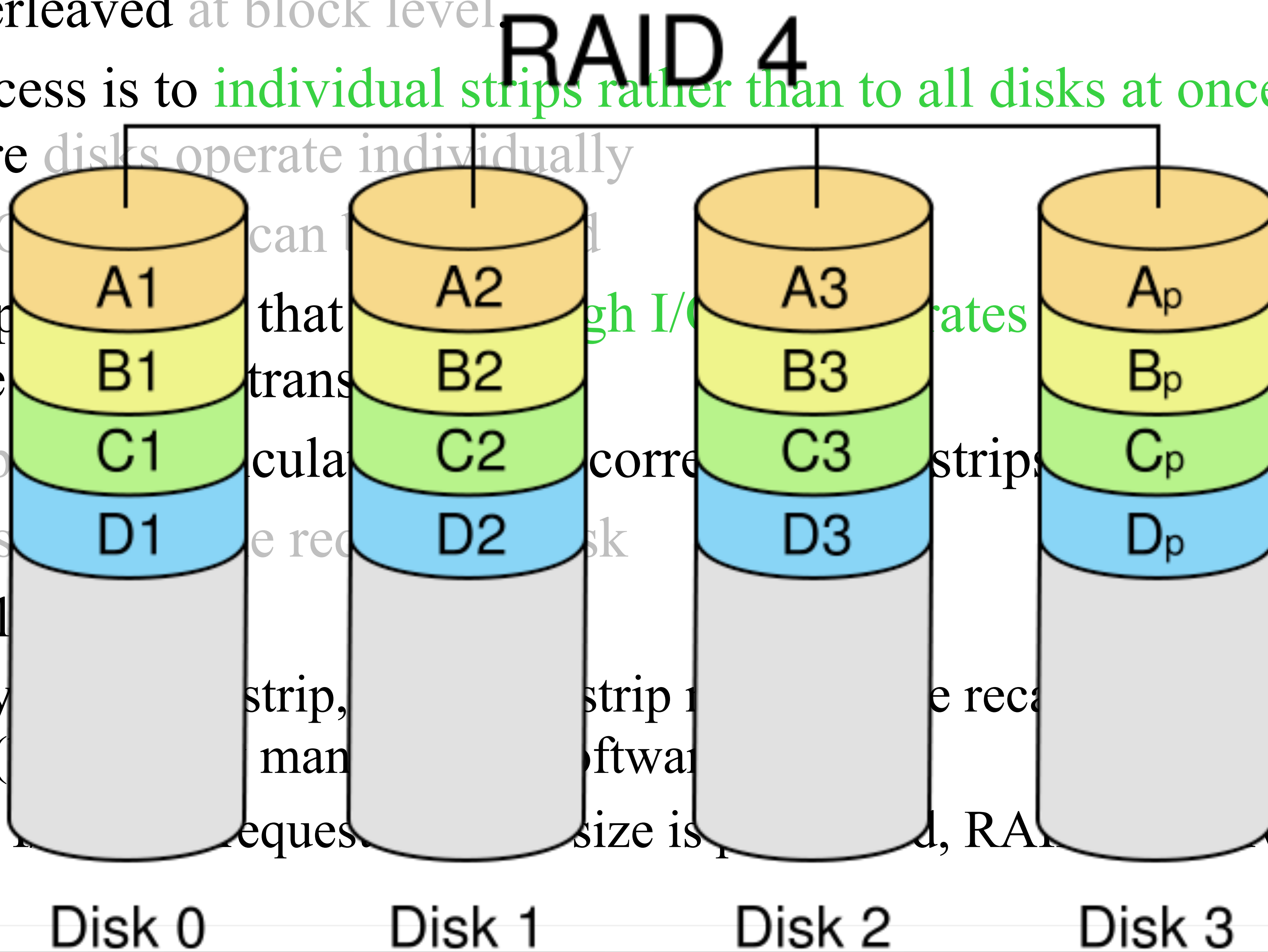


- **Bit interleaved** across multiple drives
- **Only offer error detection** - not error correction
- **More efficient than RAID 2**: parity bits are written into the data stream and only one parity drive is needed to check data accuracy.
- Parity generation and parity checking performed by hardware
- **Not suitable for small transaction**
- **Good for supercomputer and data server**: large sequential I/O request

RAID Level 4

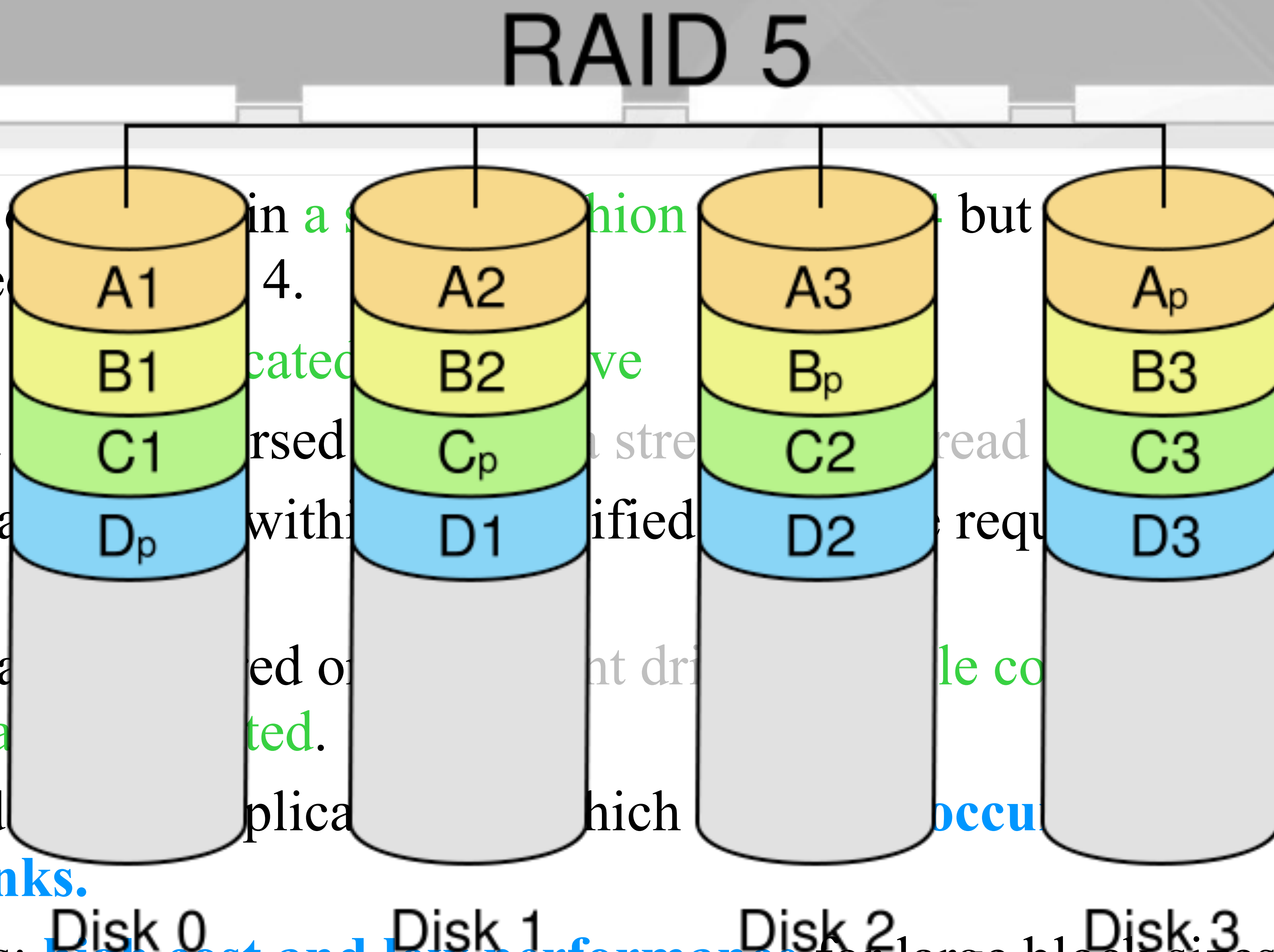
- Block Interleaving with XOR Parity

- **Write successive blocks of data on different drives.**
- Data is interleaved at block level.
- RAID 4 access is to individual strips rather than to all disks at once (as in RAID 3); therefore disks operate individually
- Separate I/O ... can ... d
- Good for ap... that ... gh I/... rates ... r applications that require ... trans...
- Bit-by-bit p... cula... corre... strip... disk
- Parity bits s... e rec... k
- Write penal...
  - For every ... strip, ... strip ... e reca... d written, i.e., updated (... man... ftwa...
  - When an ... reques... size is ... d, RA... ves a write penalty.



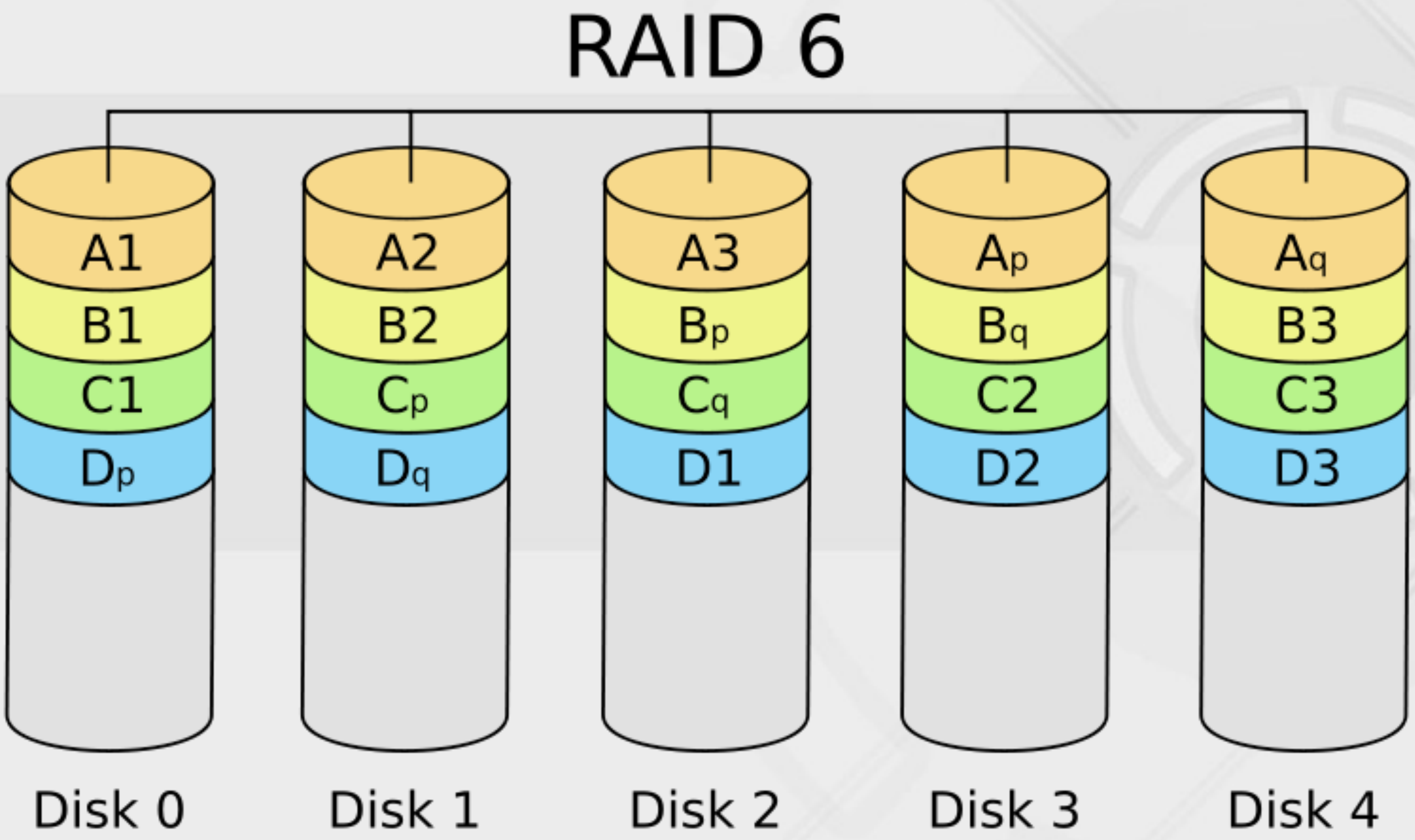
RAID Level 5

- Block Interleaving with Parity Distribution



- RAID 5 is … in a … hion … but … bottleneck encountere… 4.
- It does not … cated … ve
- Parity data … rsed … stre … read … ltiple drives.
- Block of da… withi… ified … requ… a single I/O access.
- Block of da… ed o… nt dri… le co… lock-sized accesses ca… ted.
- Good for d… plica… hich … occu… **nly and in small chunks.**
- Drawbacks: **high cost and low performance** for large block sizes objects such as audio and video.

RAID Level 6-7
- Fault-Tolerant and Heterogeneous System



RAID Level 6-7

- Fault-Tolerant and Heterogeneous System

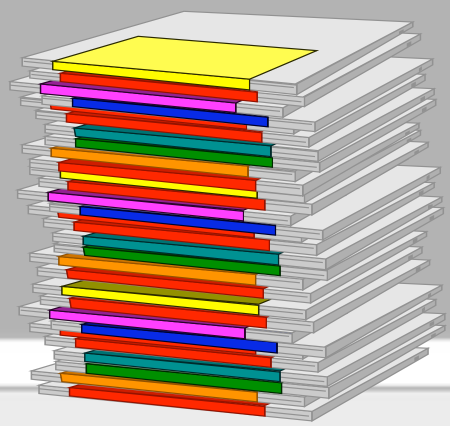
- RAID 6 has become a common feature in many systems. RAID 6 is an improvement over RAID 5 model through the addition error recovery information.
- Conceptually, the disks are considered to be in a matrix formation and the parity is generated for rows and for columns of disks in the matrix. The multi-dimensional level of parity is computed and distributed among the disks in the matrix.
- RAID 7 is the most recent development in the RAID taxonomy. Its architecture allows **each individual drive to access data as fast as possible** by incorporating a few crucial features.
- With the growth in **the speed of computers** and communications in response to **the demands for speed & reliability**, the RAID theme has begun to attract significant attention as a potential **mass storage solution** for the future.

Data Storage

- The strategy adopted for data storage will depend on the **storage technology, storage design,** and **the nature of data itself**.

- Any storage has the following parameters:
  - Storage capacity
  - Standard operations of Read and Write
  - Unit of transfer for Read and Write
  - Physical organization of storage units
  - Read-Write heads, Cylinders per Disc, Tracks per Cylinder, and Sectors per Track
  - Read time and seek time

- Of the storage technologies that are available as computer peripherals, the optical medium is the most popular in the multimedia context.

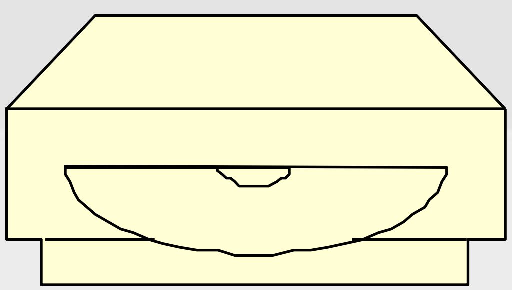


Magnetic

- **Hard Disk**
- **Floppy Disk**
- **PCMCIA**

Advantages:
- Faster than tape
- Allows direct access to data

Disadvantages:
- Performance relies on speed of mechanical heads
- Neither fault nor damage resistant



Optical

- **CD-ROM, DVD**
- **Magneto-Optical Disk**

Advantages:
- More data capacity than magnetic disk
- High quality storage of sound and images

Disadvantages:
- Data capacity is small for videos in CD and DVD are better
- Limited Data densities

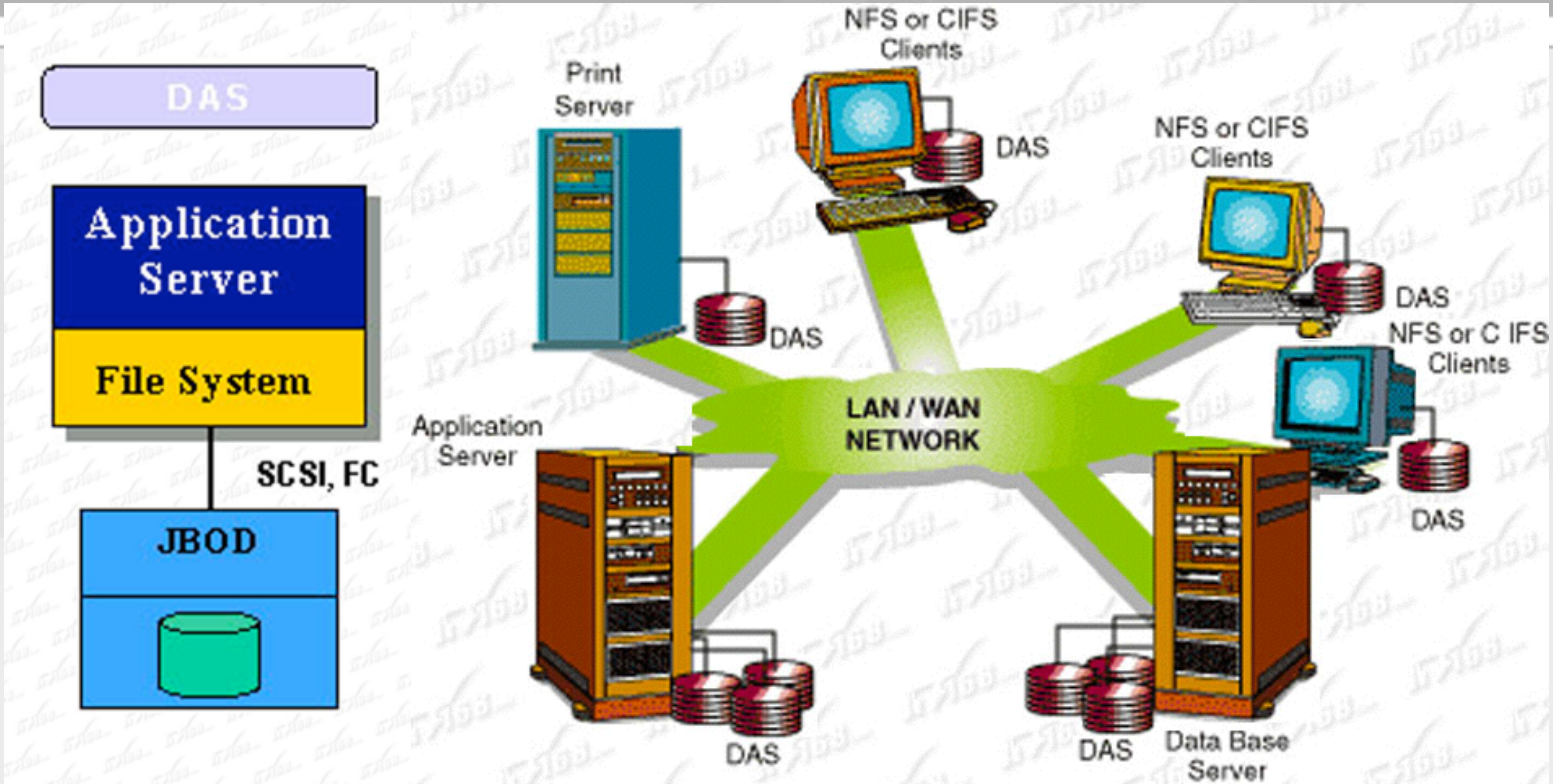
Storage solutions

- DAS: Direct Attached Storage
  – 直联存储

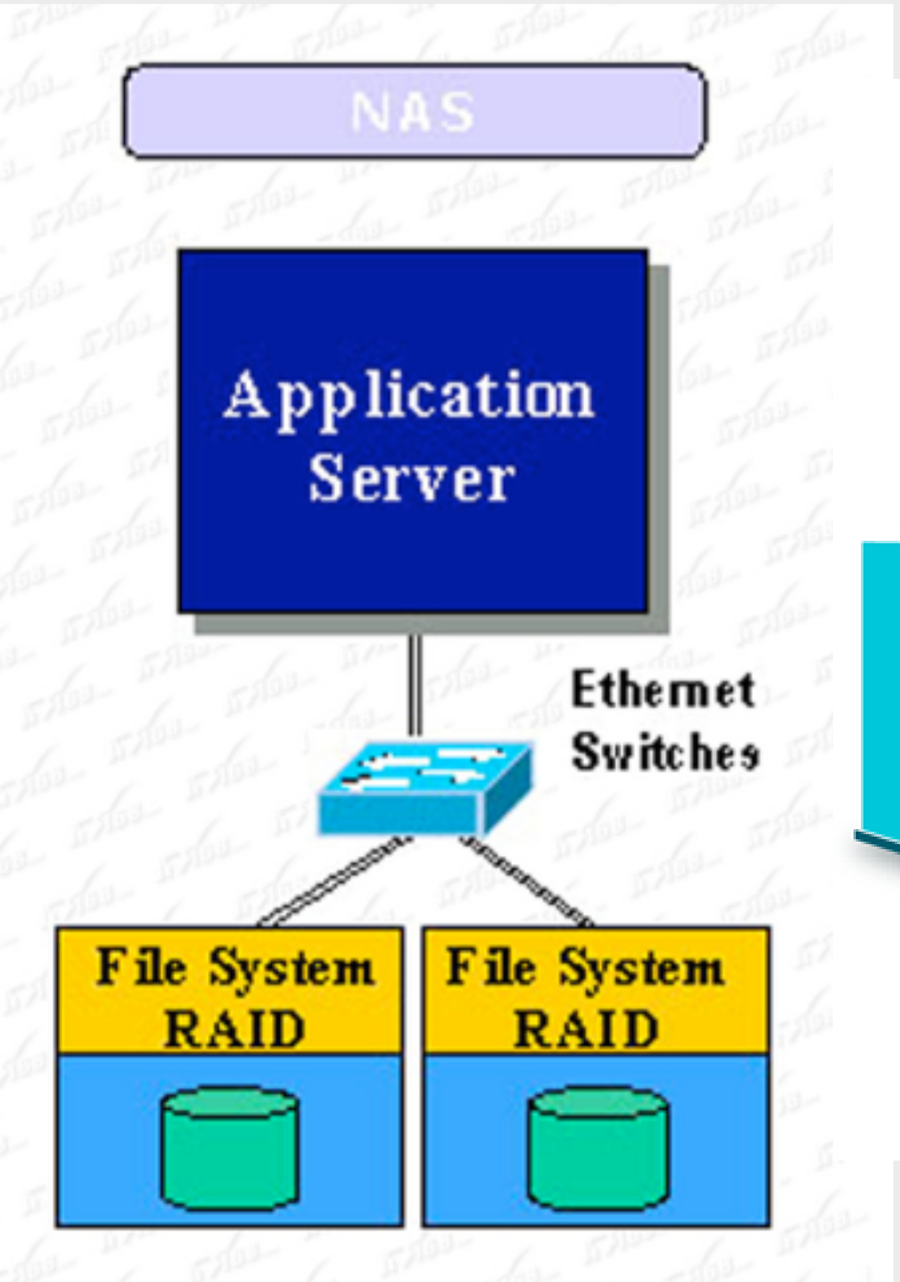
参考：
http://publish.it168.com/2004/0819/20040819005703.shtml
http://www.storagesearch.com/auspexart.html

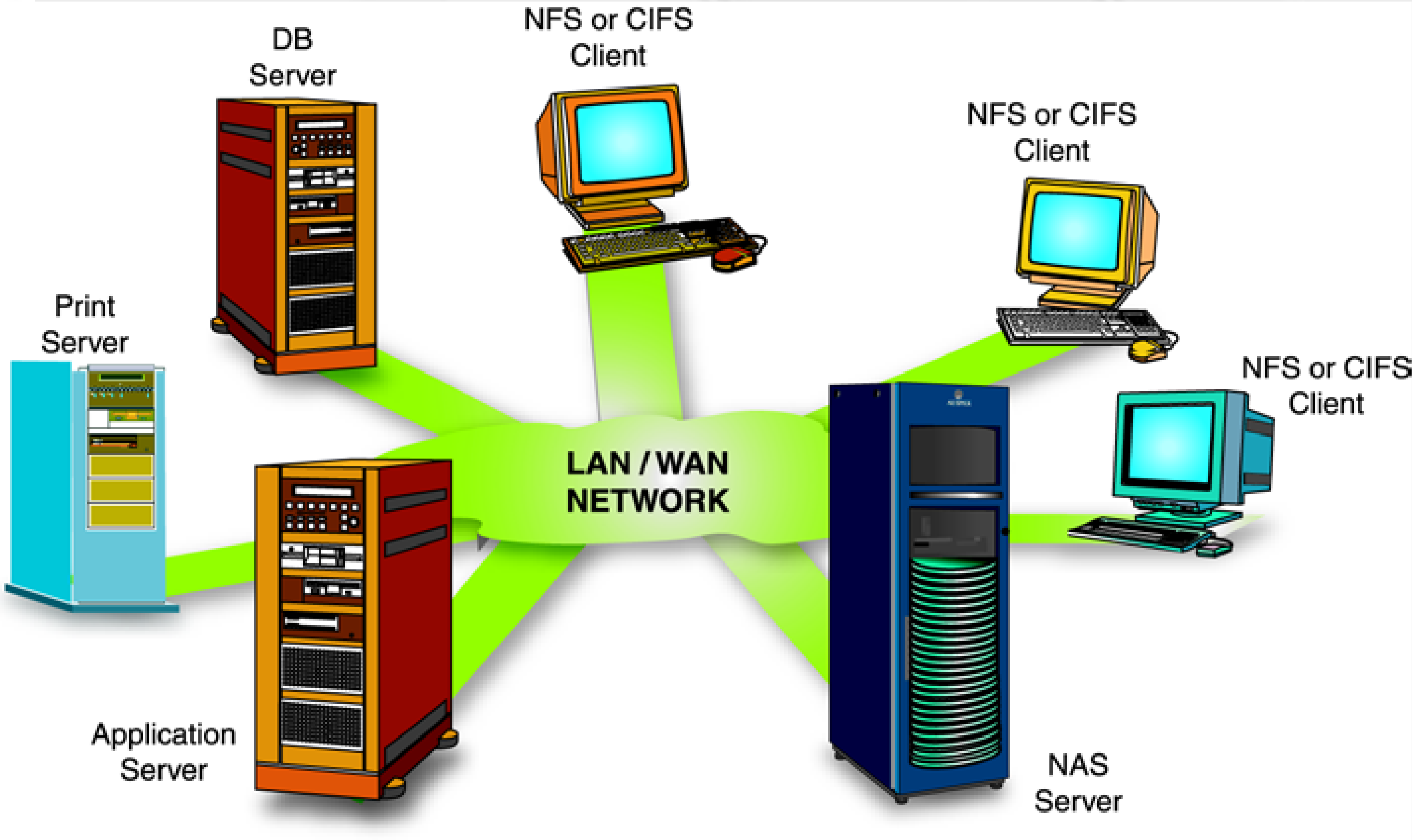
- NAS: Network Attached Storage
  – 网络附加存储
- SAN: Storage Area Network
  – 存储区域网络

DAS: Direct Attached Storage



NAS: Network Attached Storage

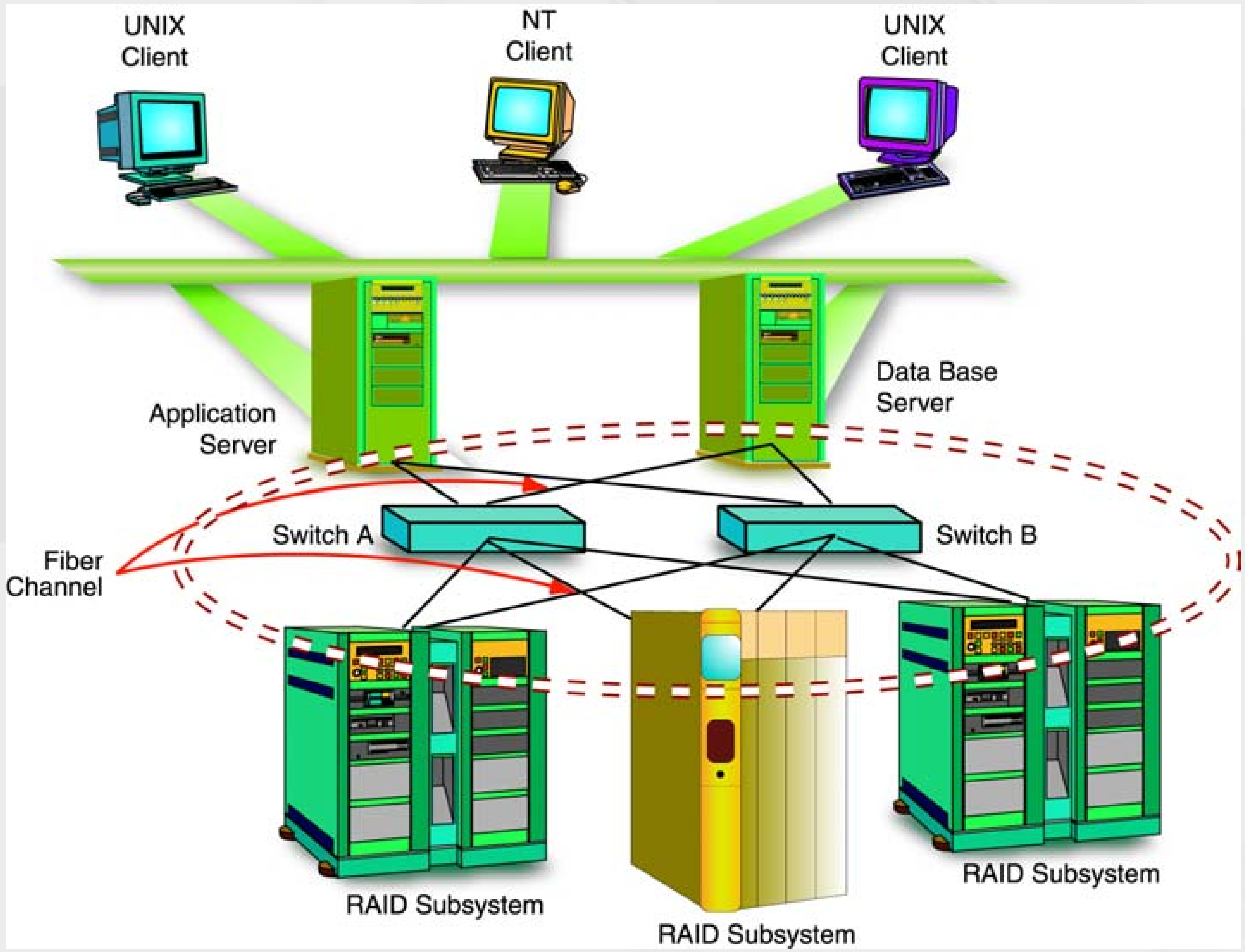




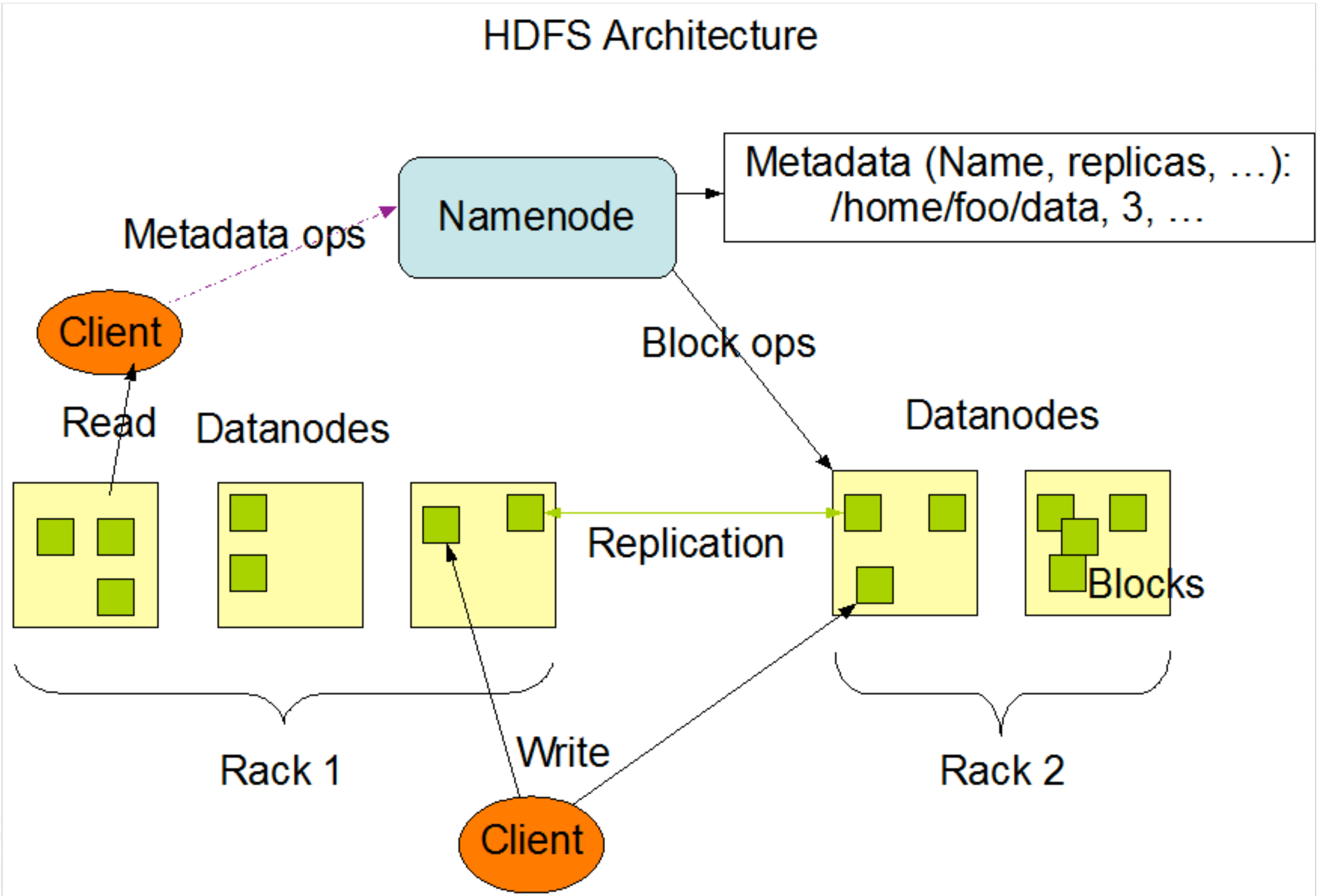
适于文件共享存储

SAN: Storage Area Network

SAN是独立出一个数据存储网络，网络内部的数据传输率很快，但操作系统仍停留在服务器端，用户不是在直接访问SAN的网络



云存储 GFS/HDFS/OpenStack ...



实例

- 阿里云：渲染云项目
  - –存储使用OSS
  - –http://oss.aliyun.com

- 课后思考题:
  - –学习使用OSS
  - –可作为作业1，2的加强版

No (Not Only) SQL

- http://sebug.net/paper/databases/nosql/Nosql.html
  - e.g. MongoDB, HBase

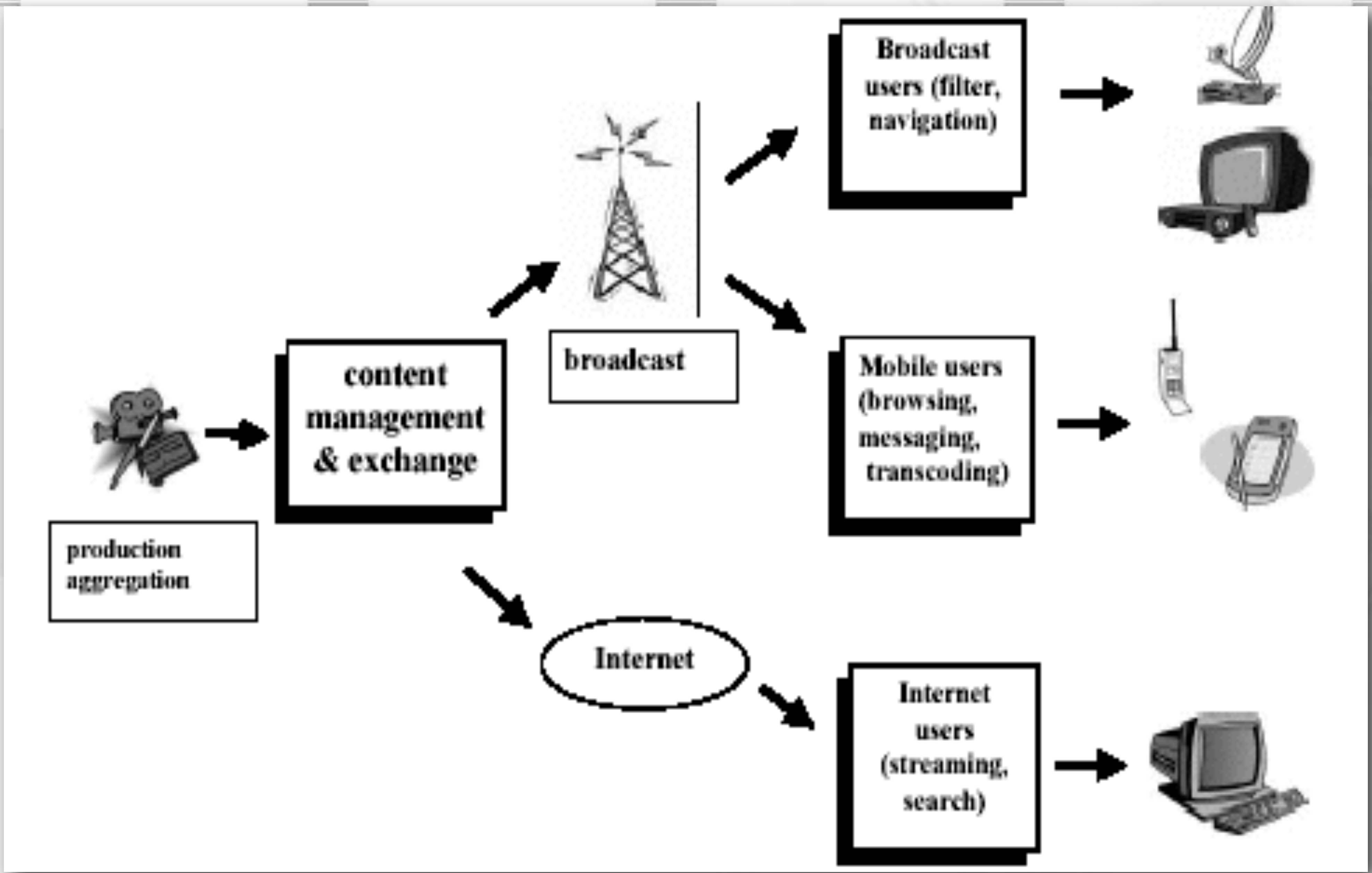
  - Key-Value database
  - Huge size
  - High scalability
  - distribution

更多参考

《大话存储II》

3.4 Multimedia System Application

Multimedia Systems Application Chain



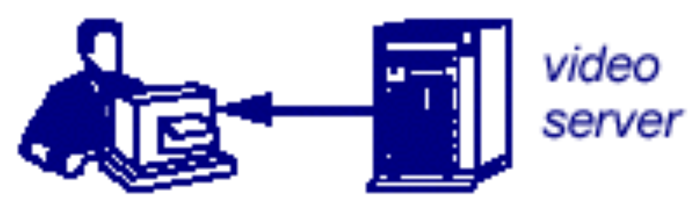
Applications of Multimedia

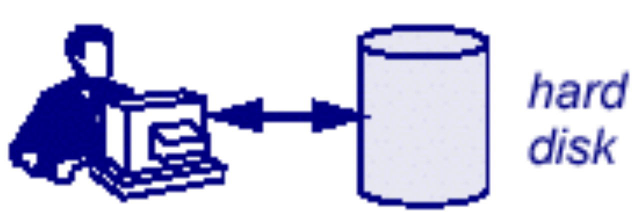
Application classes:

- Local
- Distributed

Basic multimedia services:

- Interpersonal communication
- Information retrieval
- Information recording and editing





Application Areas, Industries and Usage

Application areas:

- Learning and education
- Simulation
- Visualisation
- Presentation
- Documentation
- Archivation
- Customer information
- Cooperative work
- Supervision and control
- Entertainment

Industries:

- Bank
- Trade
- Insurance
- Research
- Education
- Manufacturing
- ...

Information

- Books on CDs
- Electr. Newspaper
- Kiosks

Communication

- CSCW
- Video conferences
- Remote diagnosis

Entertainment

- Interactive TV
- Interactive Audio
- Games

Multimedia Applications

- Hypermedia courseware
- Video conferencing
- Video on demand
- Interactive TV
- Home shopping
- Game
- Digital video editing and production systems

Digital Asset Management
数字媒体资源管理

4. Digital Rights Management

Agenda

- Overview
- Introduction of DRM (Sony & DRM)
- Protecting Digital Intellectual Property
- Rights Expression Language (REL)
- Case Study – Existing DRM systems

DRM - Overview

- History and background of DRM
- Functional Architecture for DRM
- Key Players in DRM
- Business Imperatives for DRM
- The divide between content industry and IT industry
- DRM standards initiatives: decoding the alphabet soup
- Digital copyright law developments
- State of DRM market development and its future

History of DRM

- In the pre-digital era, people's ability to do various things to or with content were limited.
- The networked digital age makes it possible to do just about anything to digital content, instantaneously and at virtually no cost.
- While this is a great opportunity for new content business models, it threatens the livelihood of content creators by making rampant piracy possible
- Also, more and more public and private entities are going digital and doing business online. Information is increasing retrieved through computer networks by customers, employees and partners etc.

➢ Need:
  - A technology that enables the **secure** creation, management, distribution and promotion of digital content on the Internet.

DRM is everywhere

Play Station
Game Store

iTunes: Apple's store

Shengda
Bambook

JailBreak & Hack?

- Why people want to hack PSP, iPhone and many other devices?
- Why jailbreak becomes legal?

DRM history

- Disk copy protection

- DVD/CD copyright protection

DRM

- **Set of technologies that enable content owners to specify and control:**
  - the access they want to give consumers and
  - the conditions under which it is given.
- **It includes:**
  - **Persistent Protection:**
    - Technology for protecting files via encryption and allowing access to them only after the entity desiring access has had its identity authenticated and its rights to that specific type of access verified
  - **Business rights:**
    - Capability of associating business rights with a content by contract, e.g. author's rights to an article or musician's rights to a song
  - **Access tracking:**
    - Capability of tracking access to and operations on content
  - **Rights licensing:**
    - Capability of defining specific rights to content and making them available by contract

DRM Functional Architecture

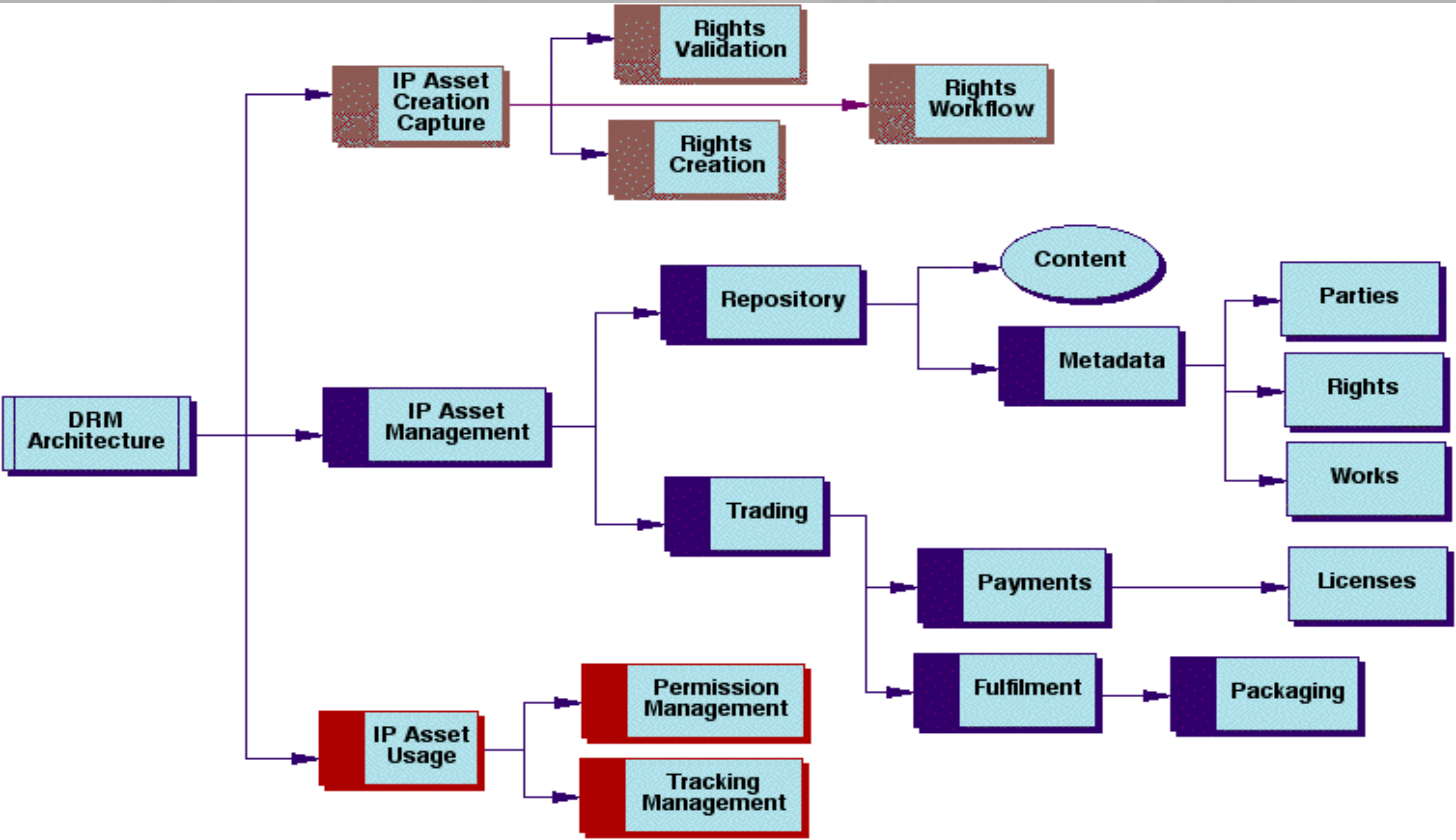
- **IP Asset Creation and Capture Module**
  - **Rights Validation** to ensure content being created includes the rights to do so
  - **Rights Creation** to allow rights to be assigned to new content
  - **Rights Workflow** to allow for content to be processed through series of workflow steps
- **IP Asset Management Module**
- **IP Asset Usage Module**

DRM Functional Architecture

- **IP Asset Creation and Capture Module**
- **IP Asset Management Module**
  - **Repository functions** to enable the access/retrieval of content in potentially distributed databases and the access/retrieval of metadata
  - **Trading functions** to enable assignment of licenses to parties who have traded agreements for rights over content, including payments from licensees to rights holders (e.g., royalty payments)

- **IP Asset Usage Module**
  - **Permissions Management** to enable usage environment to honor the rights associated with the content, e.g., if user only has the right to view the document, then printing will not be allowed
  - **Tracking Management** to enable monitoring of usage of content where such tracking is part of the agreed to license conditions, e.g., user has license to play video ten times

DRM Functional Architecture



Interested Players in DRM

- **Government Agencies**
  - Interested in controlled viewing and sharing of highly secure and confidential documents, audio and video data.
  - "Need to know basis"
- **Private Corporations**
  - Want to limit the sharing of their proprietary information
  - Track accesses and any modifications made to it.
  - E.g. news agencies like Reuters
- **Owners of commercial content**
  - Content owners, artists, and publishers want to gain revenue through sale and promotions
  - Concerned about protecting their copyrighted works from illegal use

Interested Players in DRM (cont.)

- **Intermediaries (service providers, content distributors etc.)**
  - Concerned about minimizing costs of providing services
  - Cautious about protecting themselves from lawsuits over illegal distribution
- **Producers of end user equipment (PCs, players, etc.)**
  - Concerned about minimizing design and production costs
  - Unwilling to pay for features that only some users need
- **End users**
  - Interested in immediate access to desired content
  - Want to use the content conveniently

Business Imperatives (业务需求) for DRM:

- Control Access During Workflow
- Downstream Use
- Modification of Rights Over Time
- Regulatory and Business Standards
- Outsourcing （外包）
- Protection throughout Content Life-cycle

Business Imperatives for DRM: Downstream Use

- Companies need to deliver controlled access downstream so that content can be licensed, deployed and repurposed by business partners in accordance with the terms of agreements.

  – CASE:
  Music publishers license DRM-enabled content to online transactional or subscription services. The DRM-enabled content allows both distributors and consumers to choose from multiple fee/free business models.

  For example, the content could be included in both the free-play list for one-time use on multiple devices, or it could be licensed on a fee-for play use by media companies, publishers corporate, government or institutional users.

  Further, with DRM-enabled content, owners may chose to permit licensees the ability to re-distribute or enter into republication agreements.

Business Imperatives for DRM: Modification of Rights Over Time

- Systems must be able to accommodate changes by updating parameters of rights and usage as needed to accommodate new distribution models.
- Lack of ability to change access rights to content can be a serious business liability, cost a lot of money and be a disincentive to customers.
    - *Example*: The U.S. Supreme Court decision in *New York Times v. Tasini* (2001) compelled content industry vendors to remove or modify core research records in database archives, because creators of content in those archives were not being properly compensated. Compliance costs for vendors included additional staffing to re-code or remove records, systems development expenses, along with increased demand on customer service and marketing departments.

➢ DRM, in such cases, can facilitate collaboration, creating the 'trusted environment' needed for collaboration by persistently protecting critical Intellectual Property (IP) beyond the boundaries of business processes and corporate organizations.

Business Imperatives for DRM: Regulatory and Business Standards

- *Integrity, authentication, security, privacy* and *accountability* are 'watchwords' for new legislative and regulatory standards.
  - *Example*: HIPPA regulations mandate new levels for privacy and authentication for document management in healthcare institutions and the medical community.
  - *Example*: Warranties and liability requirements demand strict assurances that the latest, most comprehensive, and appropriate instructions, product information and warning of potential hazards are in the hands of the users.

➢ Integrated DRM-CMS solutions
can offer corporations, public sector institutions and regulated industries enterprise-wide assurance that content and document operations comply with current regulatory regimes, accountability, privacy, and security legislation.

Business Imperatives for DRM: Outsourcing

- Offshore processing and data-conversion service bureaus have long been a staple of trade, technical, professional and database publishers
- Software and entertainment products are routinely outsourced. There is a growing trend to rely on outsourced personnel for the roles companies traditionally reserved for employees

- Bottom line is many people working on digital content products and processes do not have long-term commitment or loyalty to the company.
- Security and communication become large issues and require a level of embedded knowledge within core business processes.

➢ DRM ensures that information expressed in a standard format to minimize ambiguity, provide an efficient and accurate way to update operational routines, and assure appropriate levels of accountability.

Business Imperatives for DRM: Protection throughout Content Life-cycle

- Piracy, whether of software, music, film, images, or text, costs billions of dollars each year.
- It squanders valuable company time and resources by requiring costly efforts to detect and deter theft.
- Further, it creates an atmosphere of distrust that can become counterproductive to developing new business models for digital content.

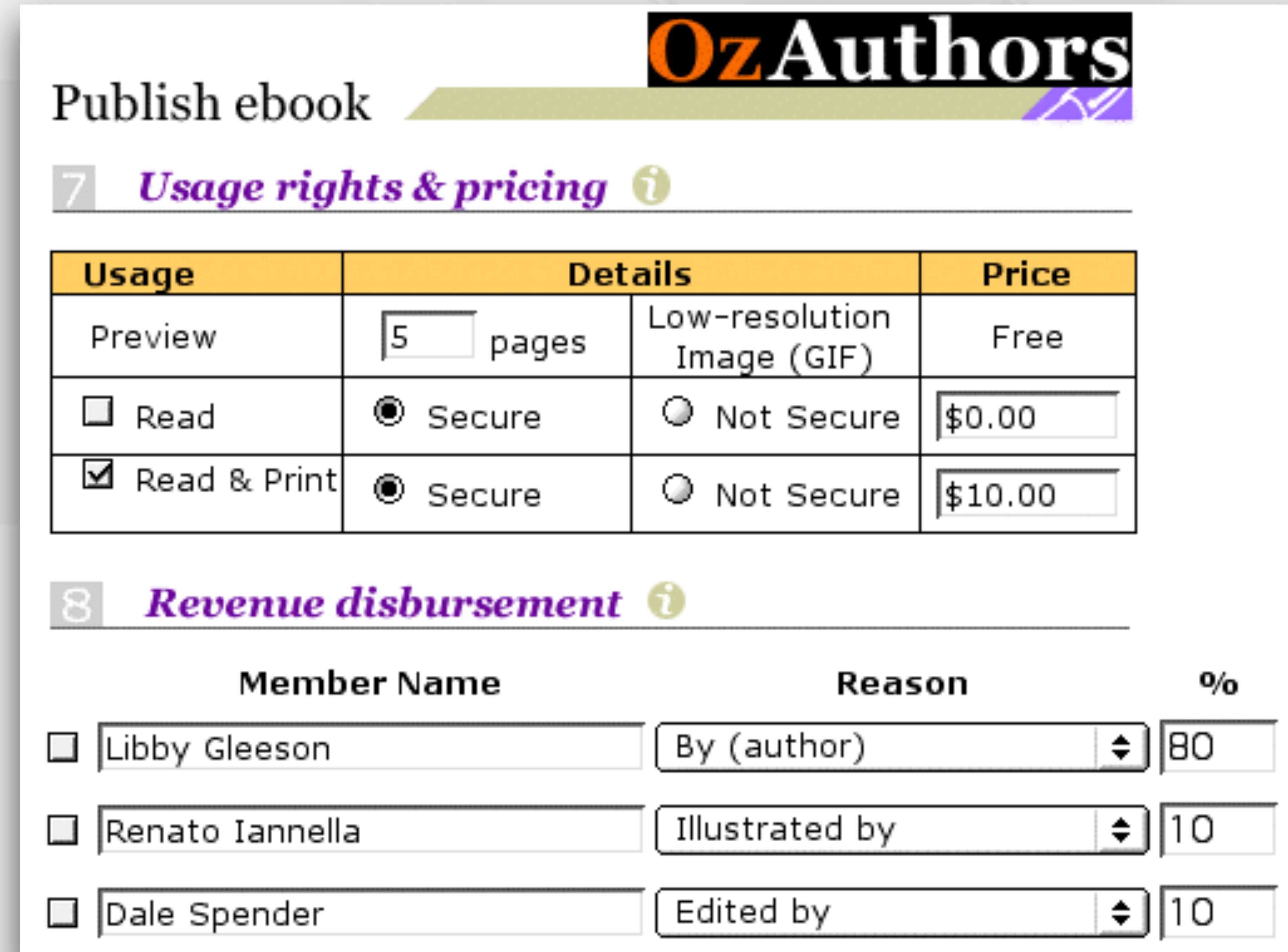
➢ DRM-enabled protection continues throughout the distribution of the content, auditing its use and accounting for its fees and licenses.

An Example of DRM Implementation

OzAuthors

(An online e-book store)

Rights Interface



OzAuthors

Publish ebook

7 *Usage rights & pricing*

Usage	Details		Price
Preview	5 pages	Low-resolution Image (GIF)	Free
☐ Read	◉ Secure	○ Not Secure	$0.00
☑ Read & Print	◉ Secure	○ Not Secure	$10.00

8 *Revenue disbursement*

	Member Name	Reason	%
☐	Libby Gleeson	By (author)	80
☐	Renato Iannella	Illustrated by	10
☐	Dale Spender	Edited by	10

The Division

- **The content development industry:**

  (e.g., the recording industry and the movie studios)

  – the need for immediate DRM solutions that stop all unauthorized copying and distribution.

- **The IT industry:**

  – DRM solutions should support the concept of "fair use,"

    • allows consumers to make copies of some types of copyrighted content for their own personal use.

The Division

content provider

IT

- In the US, these disagreements have led to an increase in both DRM-related **lawsuits** and new legislative initiatives.

Fair Use (合理使用)

- Copyright principle based on **the belief**:
  - the public is entitled to freely use **portions of copyrighted materials** for purposes of **commentary and criticism** (评论、批评)

- Unfortunately, if the copyright owner disagrees with your fair use interpretation, the dispute will have to be resolved by courts or arbitration.
- The four factors for measuring fair use:
  - the purpose and character of your use
  - the nature of the copyrighted work
  - the amount and substantiality of the portion taken, and
  - the effect of the use upon the potential market.

Examples of Fair Use Court Cases (I)

- **Universal City Studios v. Sony Corp., 464 U.S. 417 (1984)**
  - The Supreme Court determined that home videotaping of a TV broadcast was fair use.
  - One of the few occasions when copying a complete work was accepted as a fair use.
  - Evidence indicated that most viewers were "time-shifting" and not "library-building“
  - *Important factors:* The Supreme Court reasoned that the "delayed" system of viewing did not deprive the copyright owners of revenue
  -

Examples of Fair Use Court Cases (2)

- **Kelly v. Arriba-Soft, 03 C.D.O.S. 5888 (9th Cir. 2003)**
  - A search engine's practice of creating small reproductions ("thumbnails") of images and placing them on its own website (known as "inlining") did not undermine the potential market for the sale or licensing of those images.
  - *Important Factors.* The thumbnails were much smaller and of much poorer quality than the original photos and served to index the images and help the public access them.
  -

Examples of Fair Use Court Cases (3)

- **Religious Technology Center v. Lerma, 40 U.S.P.Q. 2d 1569 (E.D. Va. 1996)**
  - Entire publications of the Church of Scientology were posted on the Internet by several individuals without Church permission.
  - The court held that the use was not fair, since fair use is intended to permit the borrowing of portions of a work, not complete works.
  -

Digital Copyright Millennium Act (DCMA)

- 1998 law designed to increase copyright holders' rights.
- Creates civil and criminal penalties for creation or distribution of DRM circumvention tools.
- As a result, a user attempting to circumvent copyright protection, even for legitimate reasons, violates federal law.
- What this means?
  - Open-source software developers rely on reverse engineering to write programs that can interact with hardware. This practice is illegal under the DCMA.
  - Reverse Engineering and Cryptanalysis can also be interpreted as illegal under the DCMA.
  - Is Norton Anti-Virus illegal?

A Media Consumption Culture Shift: Pay-Per-Use and the Marginalization of Content Sharing

- **DRM could also acclimate users to a system where sharing of content is not permitted. In 1996 in *The Right to Read*, Stallman envisioned a world where copy protection prevented the anonymous reading of books, lending books to others, or the mere possession of software tools that could be used to bypass copyright law:**
  - *This put Dan in a dilemma. He had to help her--but if he lent her his computer, she might read his books. Aside from the fact that you could go to prison for many years for letting someone else read your books, the very idea shocked him at first. Like everyone, he had been taught since elementary school that sharing books was nasty and wrong--something that only pirates would do.*

    --*The Right to Read*, Richard Stallman, 1996.

Microsoft Palladium

- A system that combines software and hardware controls to create a **"trusted" computing platform**.
  - purports to stop viruses
  - store personal data within an encrypted folder.
  - depend on hardware that has
    - either a digital signature
    - or a tracking number.
  - filter spam.
  - incorporate DRM technologies for media files of all types (music, documents, e-mail communications).
  - \+ purports to transmit data within the computer via encrypted paths.

Major Legal Developments : The SSSCA and the CBDTPA

- In September 2001, Senator Fritz Hollings announced plans to introduce the Security Systems Standards and Certification Act (SSSCA), which would require equipment manufacturers to embed government-approved copy protection systems into all computer equipment.
- In February 2002, Sen. Hollings scheduled hearings. All of the hearing panelists represented large corporations and there was no testimony taken from consumer advocates.
- In March 2002, Sen. Hollings introduced the Consumer Broadband and Digital Television Promotion Act (CBDTPA). This copyright control would force manufacturers to embed copy protection in all devices that can receive digital media.
- Opposition to the CBDTPA has been vigorous both from individual users and from business interests.

Major Legal Developments: Dmitry Sklyarov and Adobe eBook Copy "Protection"

- In June 2001, a Russian programmer named Dmitry Sklyarov published a program that can defeat a DRM technology used to secure Adobe eBooks.
- In July, at the behest of Adobe, the Department of Justice arrested Sklyarov for violating the DMCA shortly after he presented a paper on cracking Adobe ROT-13 copy protection.
- Sklyarov remained in jail for several weeks and was released on $50,000 bail. The Electronic Frontier Foundation (EFF) assisted in his defense and in December 2001, federal authorities dropped charges against him.
- Federal authorities have now pursued ElcomSoft, Dmitry Sklyarov's employer. The case is being litigated in Federal District Court in California.

Major Legal Developments: Ed Felten and Suppression of Academic Inquiry into DRM Systems

- In April 2001, a team of researchers headed by Princeton Prof. Felten announced that they could defeat a DRM system developed by the Secure Digital Media Initiative (SDMI).
- SDMI and the Recording Industry Artists of America (RIAA) threatened Felten and his team with a lawsuit under the DMCA. Felten's team decided not to publish the paper.
- Ultimately, SDMI and RIAA retreated from the treat of lawsuit, fearing that the DMCA may have been stricken as constitutionally overbroad when applied against a group of professors presenting an academic paper.
- In June 2001, the Electronic Frontier Foundation (EFF) bought suit against RIAA to obtain a declaratory judgment that Felten could present the SDMI research. Additionally, EFF sought the invalidation of the DMCA as an unconstitutional restriction on free expression.
- In August 2001, Felten presented the SDMI paper at the USENIX conference. In November 2001, a Federal District Court dismissed EFF's case. In February 2002, Felten decided not to appeal the dismissal.

References

- *"Integrating Content Management with Digital Rights Manangement"*, Bill Rosenblatt and Gail Dykstra, May 14 2003
- http://www.epic.org/privacy/drm/
- http://www.dlib.org/dlib/june01/iannella/06iannella.html
- http://www.eff.org/IP/DRM/

Agenda

- Overview
- Introduction of DRM (Sony & DRM)
- Protecting Digital Intellectual Property
- Rights Expression Language (REL)
- Case Study – Existing DRM systems

DRM – Introduction (Sony & DRM)

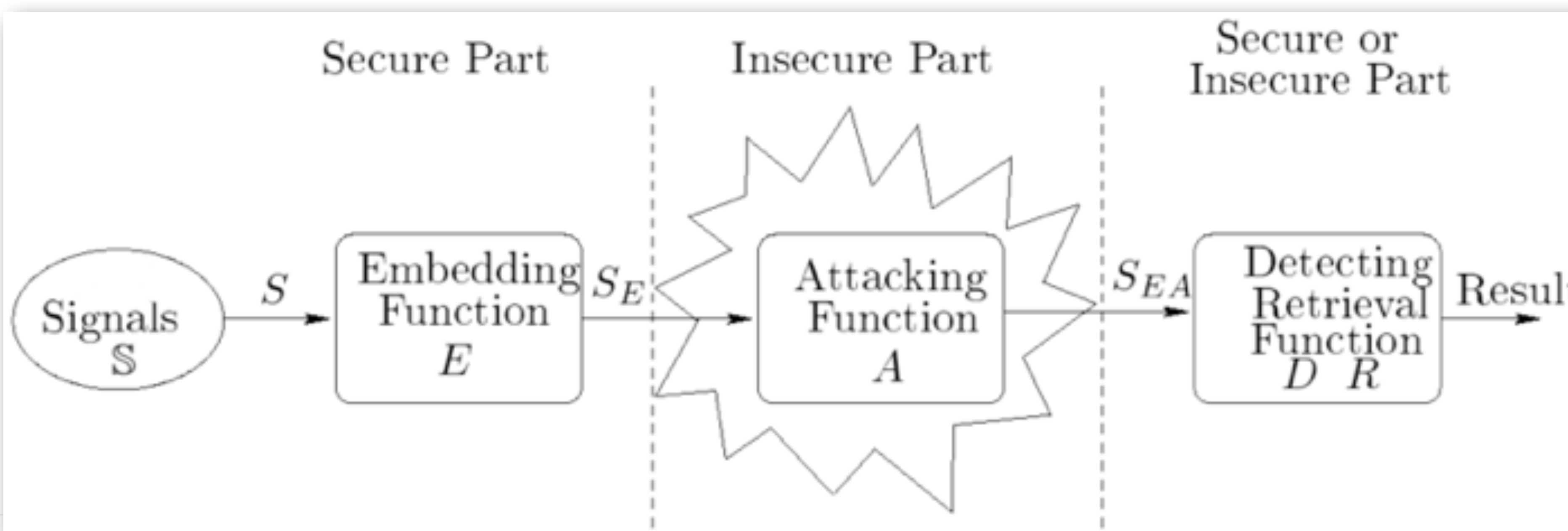
- History
- DRM & Sony
- DRM Technology
- Sony Aftermath
- Review - Moral of DRM

Previous Technologies

- PKI – Public Key Infrastructure
- PGP – Pretty Good Privacy
- S/MIME
- Access Control Systems
- Smart Cards
- Biometrics

Preventing Copying With Watermarking (水印)

- digital art
- 票据防伪
- 数据隐藏
- 隐蔽通讯



Stenography

1. removing all but the last 2 bits of each color component
2. X 85

Digital Watermark

- Invisible ink on multimedia data
  - image
  - video
  - music
  - graphics

How are these technologies **different** to DRM?

- Only protect the data in transit
  - E.g. over the Internet or on CD
- Once the data is opened, it can be:
  - edited
  - copied
  - printed
  - saved as an unprotected file

  And then
- Redistributed to anyone else in an unprotected format.

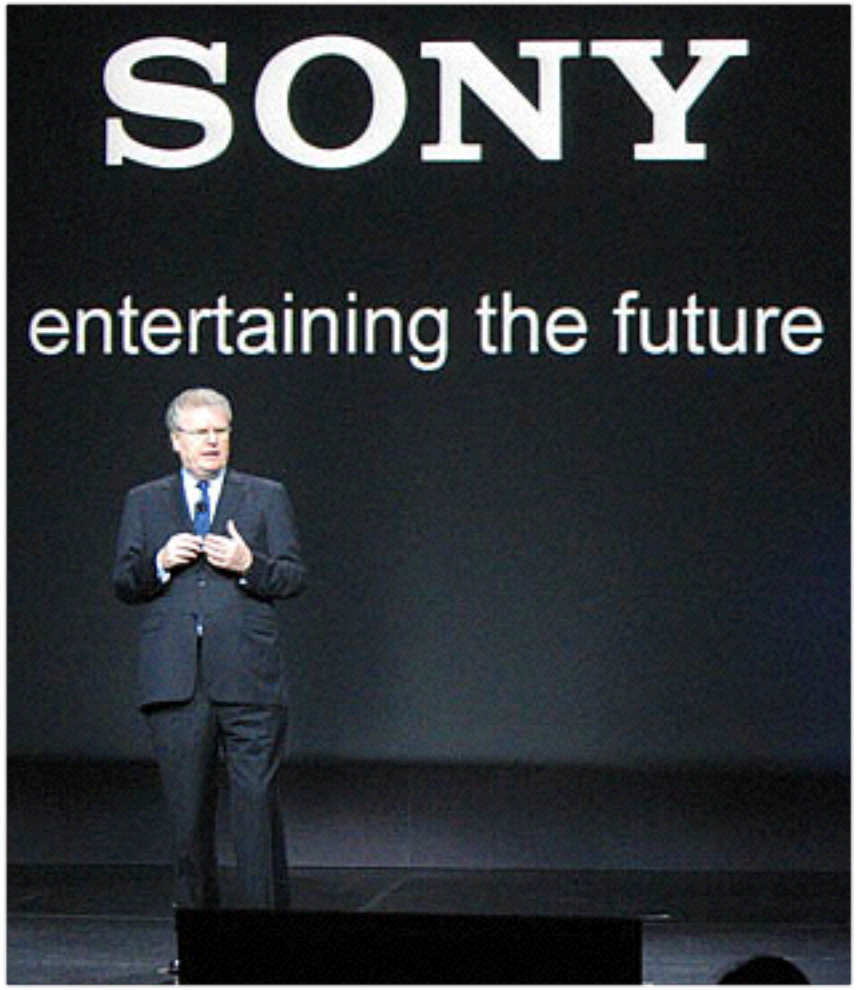
Rely on TRUST once the content is delivered

DRM in short

Any technology used to protect the interests of owners of content and services (such as copyright owners). Typically, authorized recipients or users must acquire a license in order to consume the protected material—files, music, movies—according to the rights or business rules set by the content owner.

浙江大学计算机学院
数字媒体与网络技术

Sony & DRM



Sony's Problem:

- One of the big 4 music companies.
  - Copies of its music are easily made by **ripping** from CD's.

- BMG's music was continuously being illegally downloaded and shared across the internet.
  - Large sales being lost.
  - Hard to track popularity data

Sony's Response

"The industry will take whatever steps it needs to protect itself and protect its revenue streams...It will not lose that revenue stream, no matter what … **Sony is going to take aggressive steps to stop this**."

Sony's Solution:

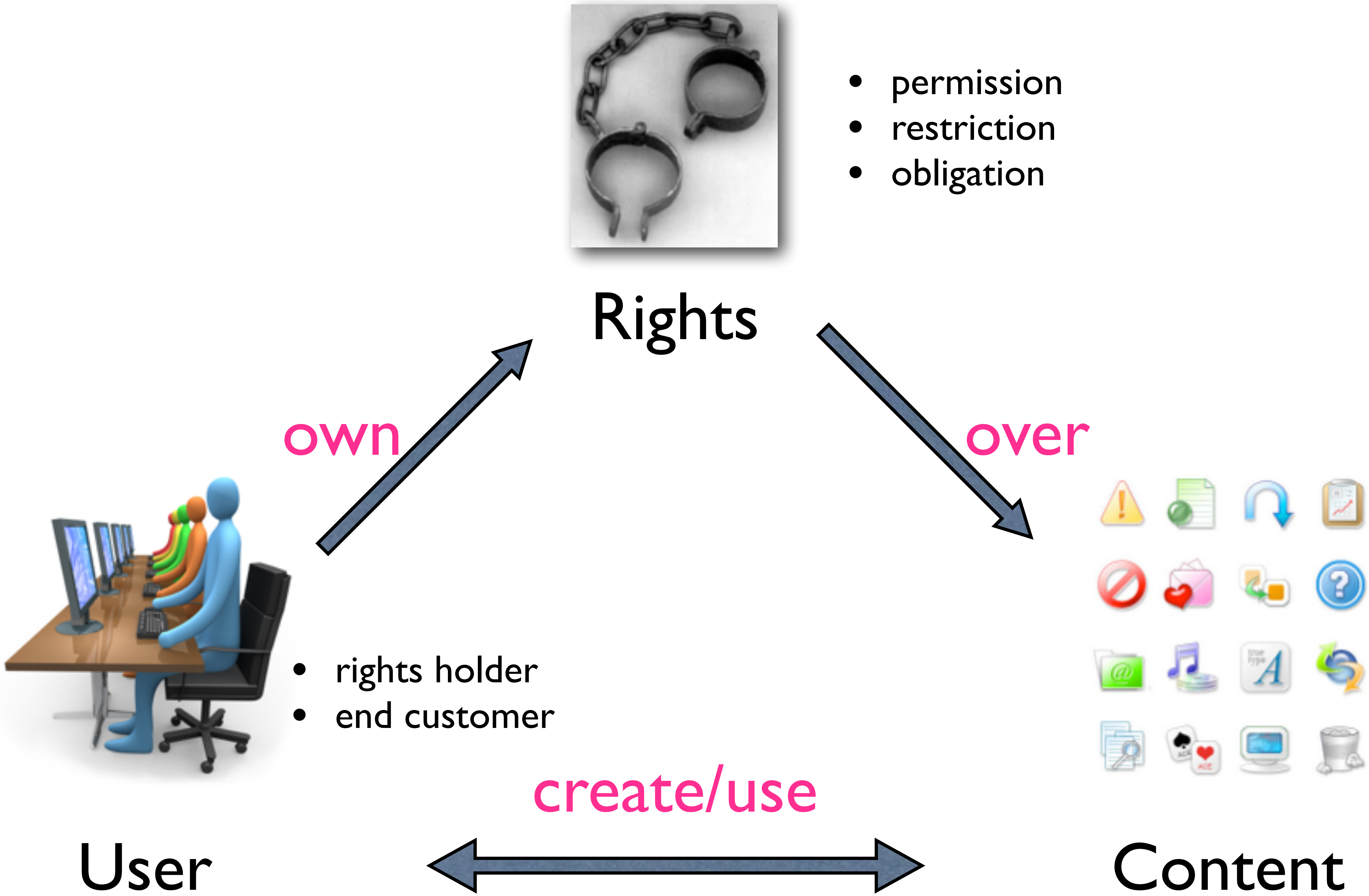
- DRM!
- More specifically:
  - A **rootkit** concealing a software called 'Extended Copy Protection' was installed on every CD user's machine.

The Rootkit:

- Remains resident in the user's system
  - intercepting all accesses of the CD drive to prevent any media player or ripper software other than the one included with XCP-Aurora from accessing the music tracks of the Sony CD.
- Player software will:
  - Play songs
  - Allow only a limited degree of other actions
    - e.g., burning the music onto a certain number of other CDs or
    - loading it onto certain supported devices, e.g., a few portable music players. (iPod not supported)

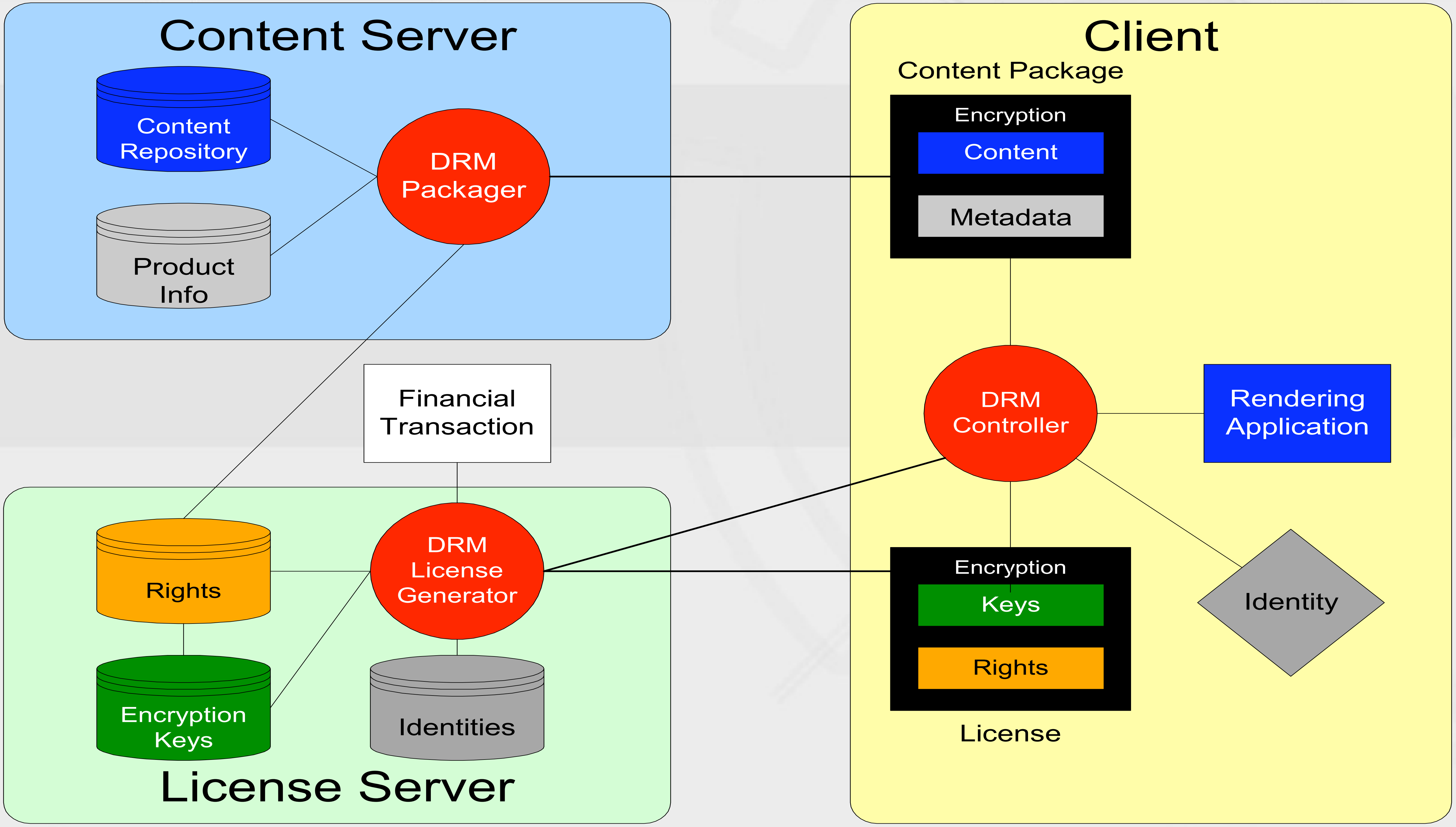
So how technically does the Rootkit act as DRM?

Lets take a look at:
the DRM Technology Building Blocks



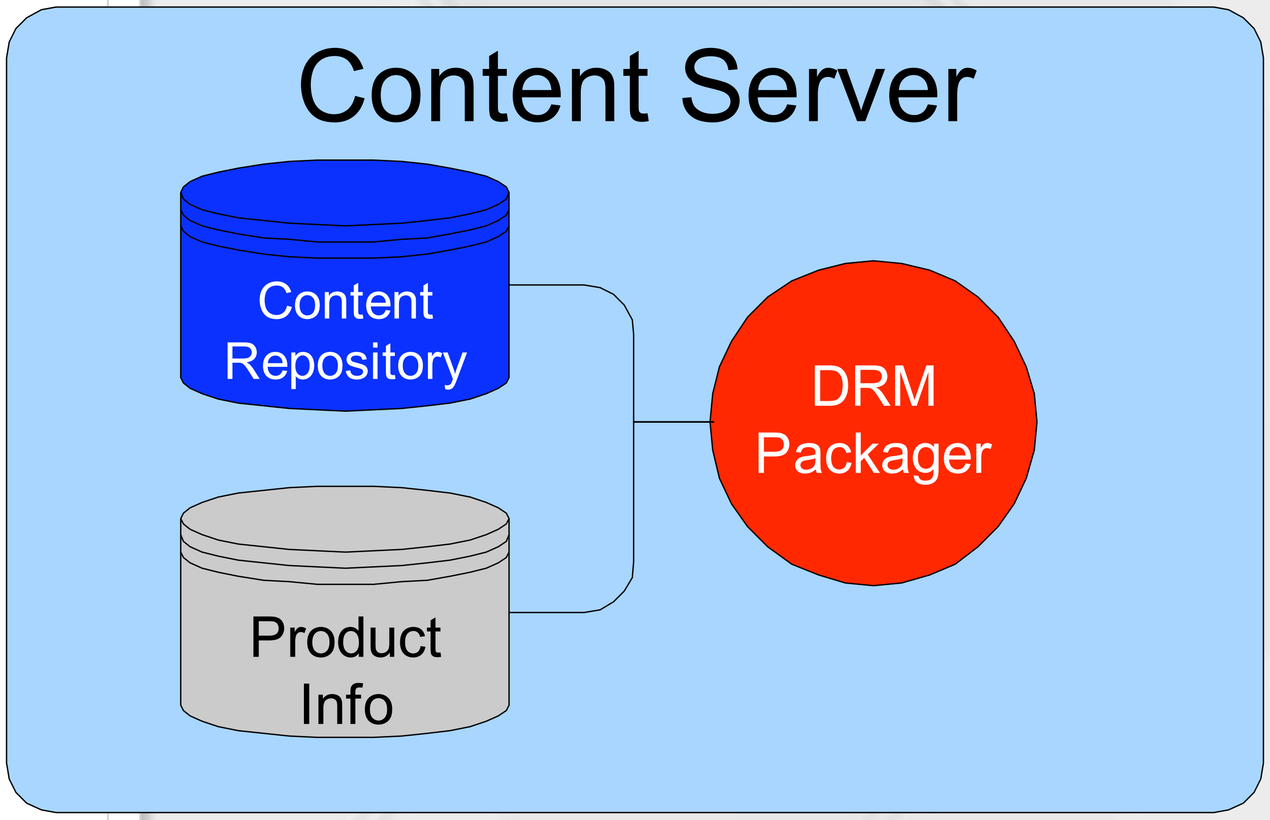
DRM basic Model

DRM Reference Architecture



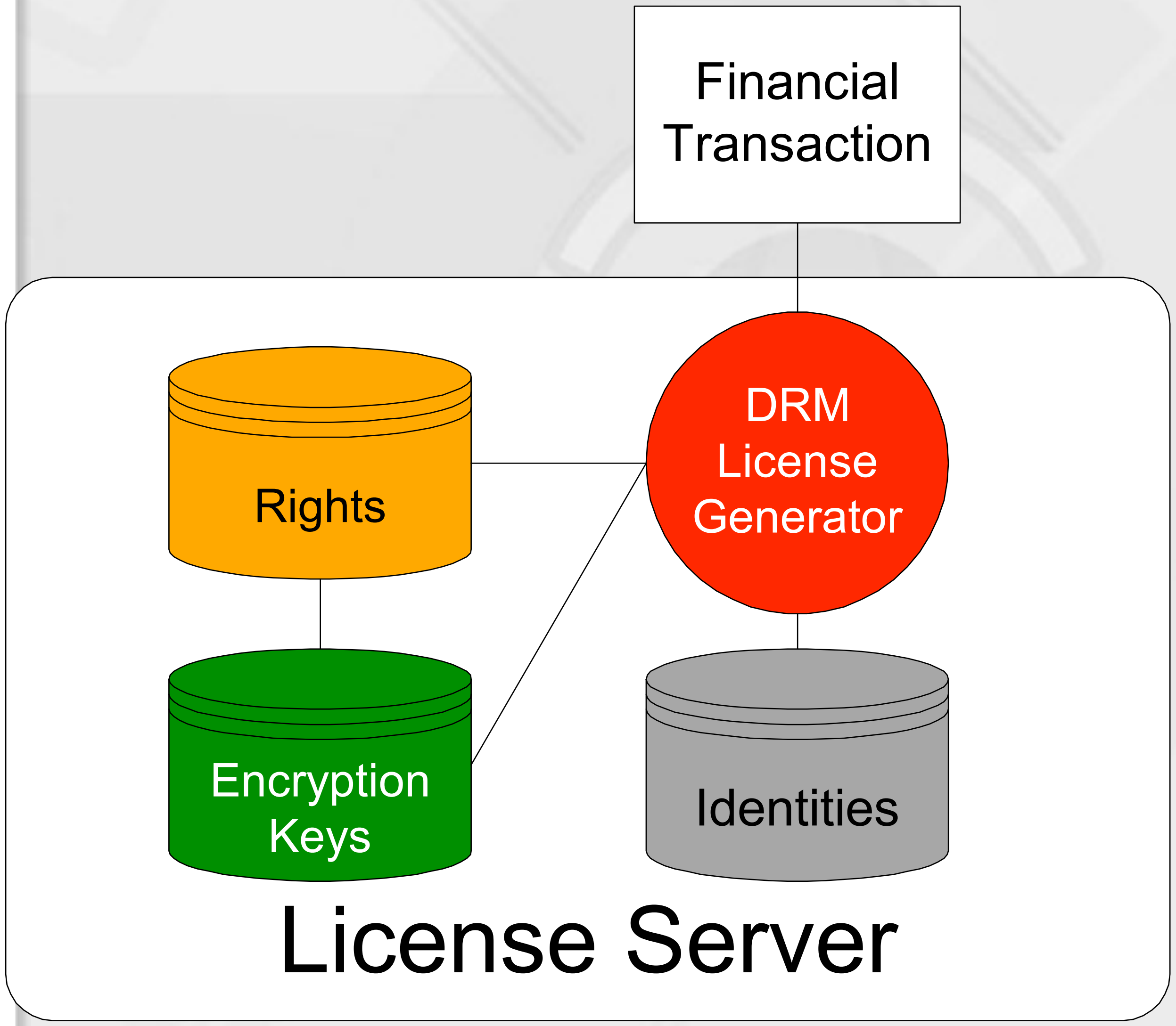
Content Server

- Content Repository
  - Content Management system
  - Digital Asset Management system
  - File server
- Product Info
  - Rights
  - Product metadata
- DRM Packager
  - Packages content with metadata
  - Encrypts



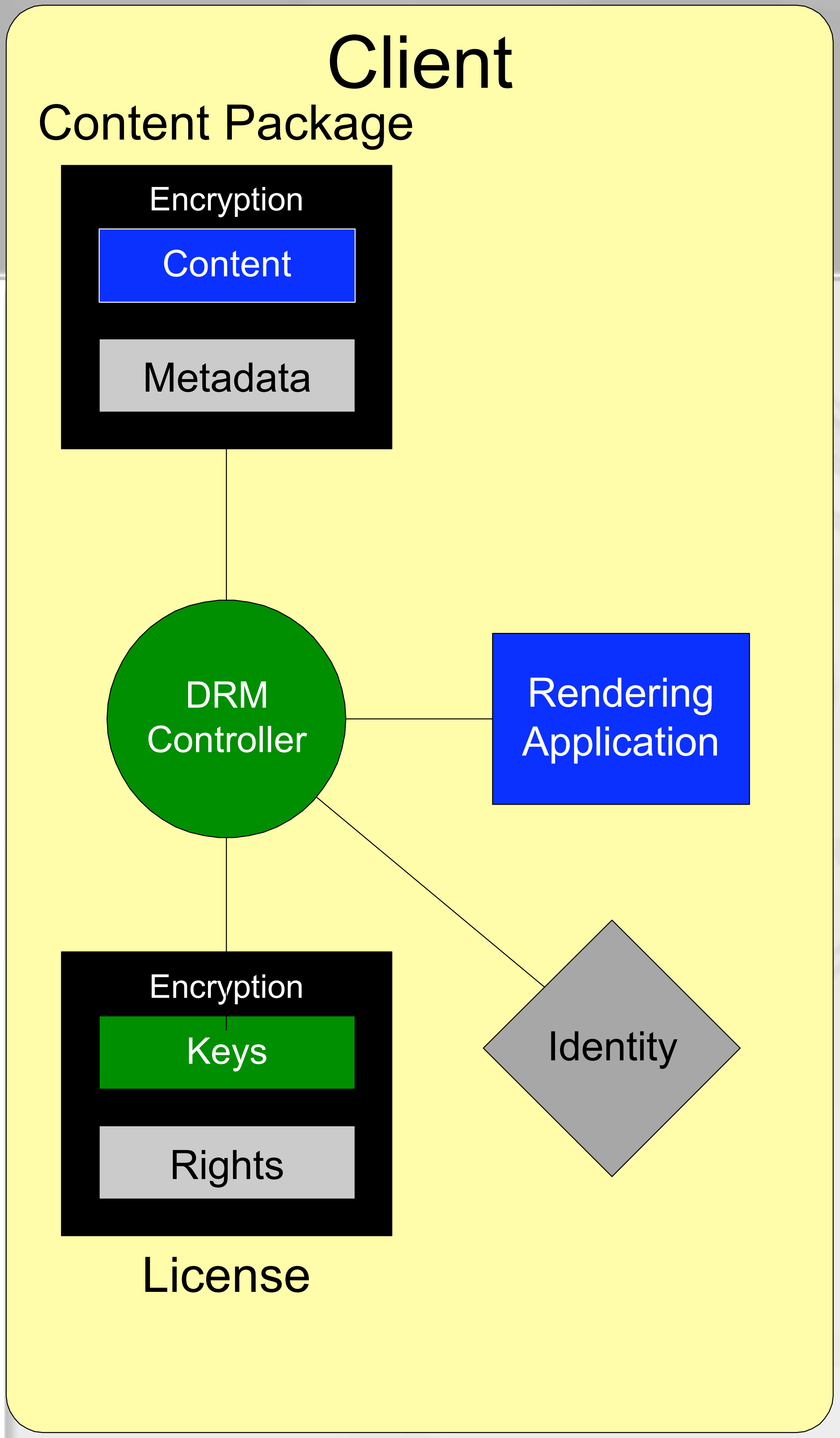
License Server

- Encryption key repository
- User identity database
  - Usernames
  - Machine IDs
- DRM License Generator



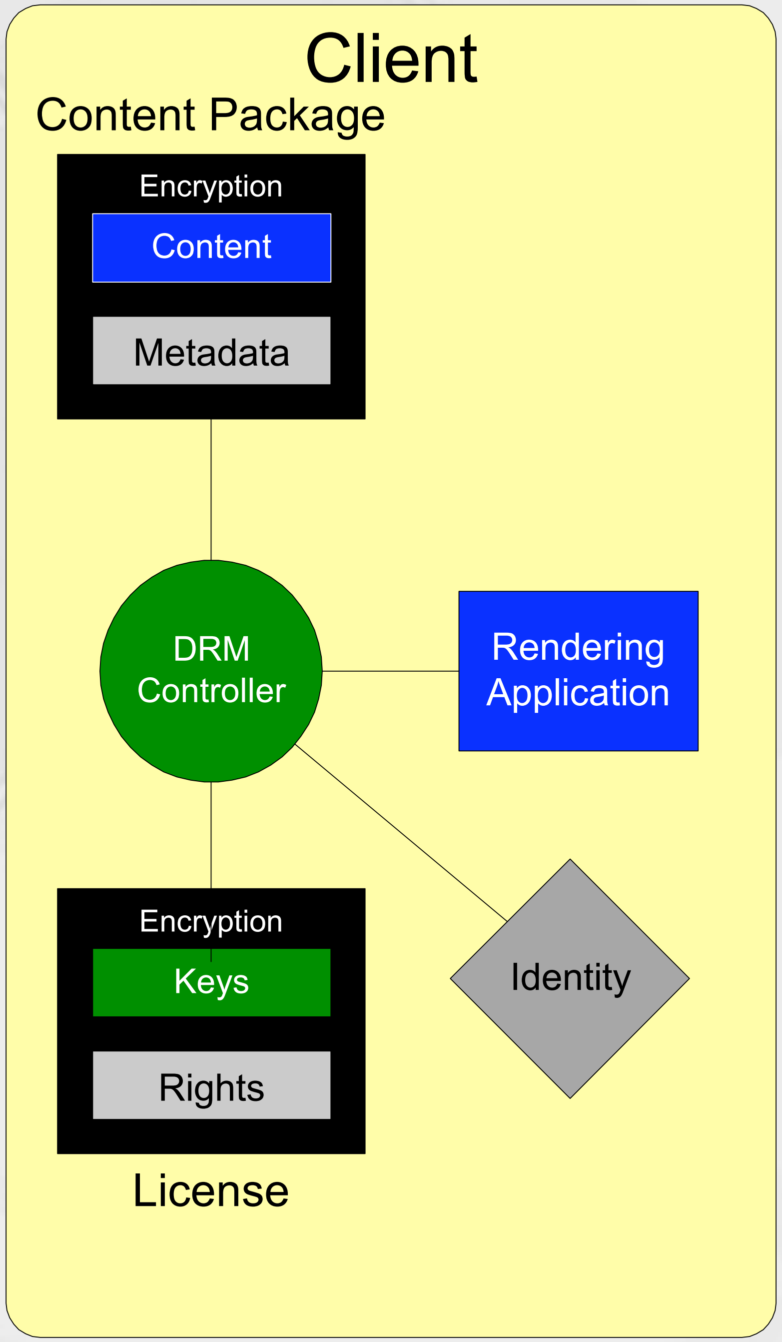
Client

- DRM Controller
  - Nerve center of process
- Rendering application
- Content packages
- Licenses
- Identity



Processes - User Initiation

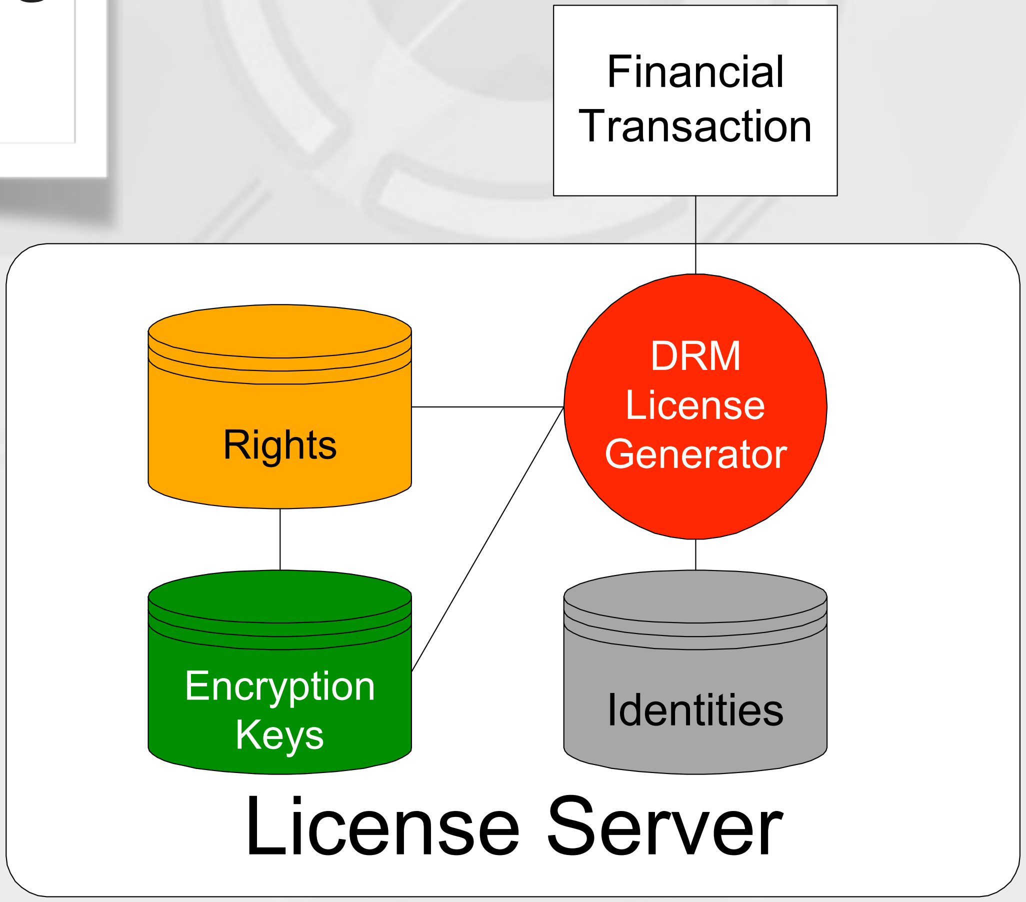
- User obtains content package
- User requests operation
  - view, play
- DRM controller collects info
  - Content
  - Identity
  - Requested rights
- DRM controller:
  - license generator



Processes - License Generation

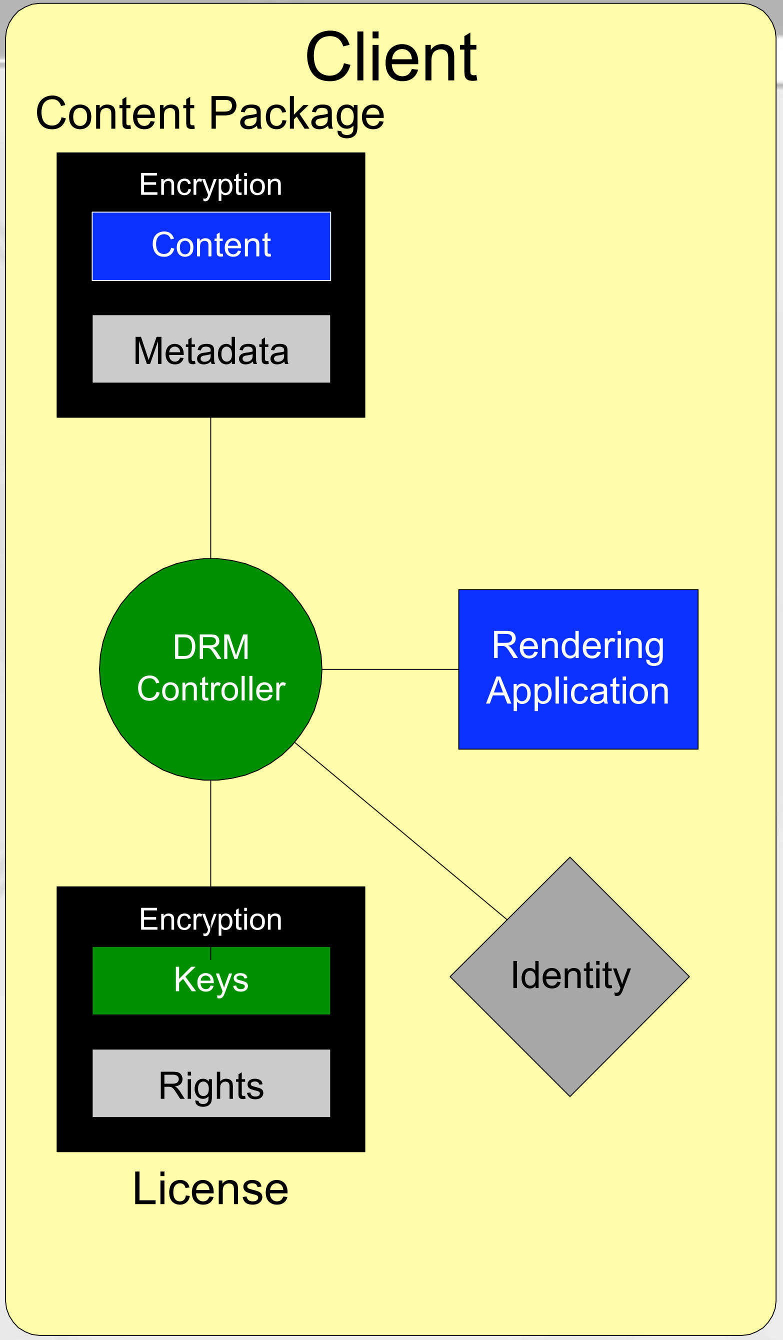
DRM License Generator…

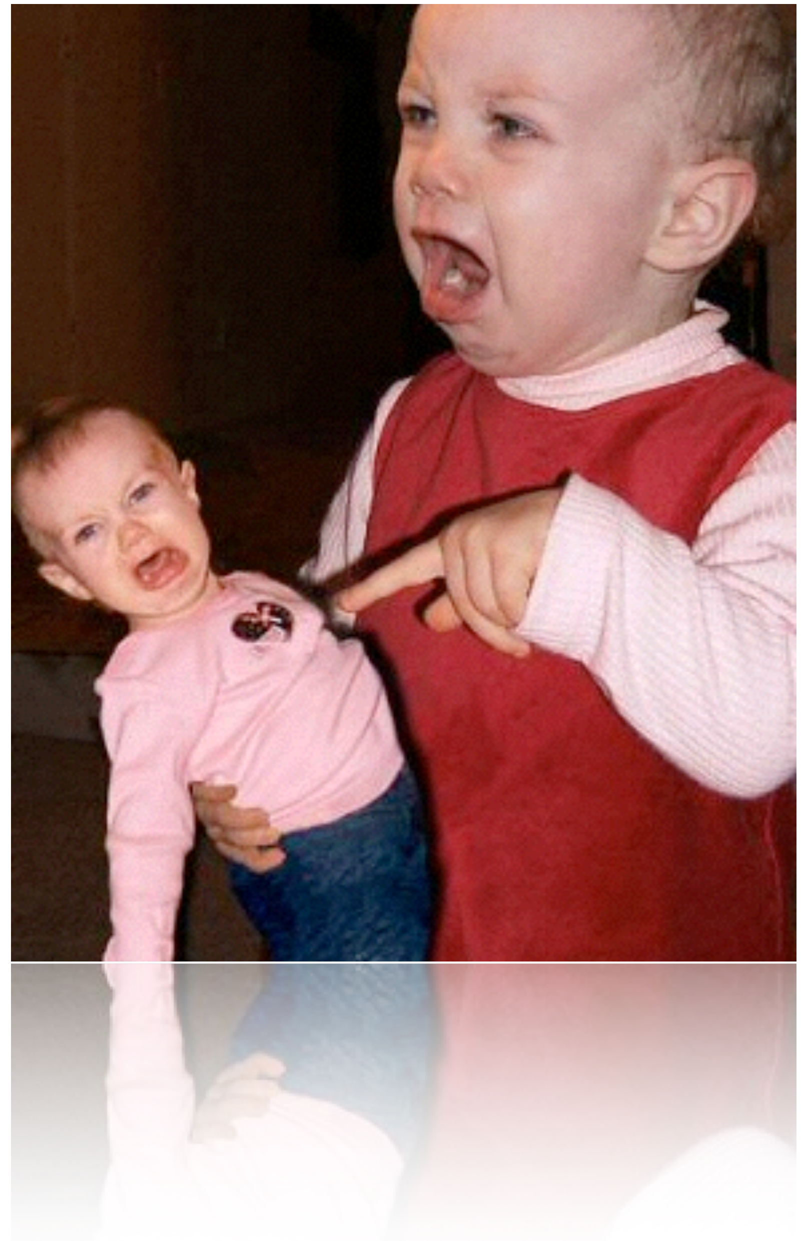
- Checks content & identity
- Obtains keys from key repository
- Creates & sends license to client
- Generates financial transaction, where necessary



Processes - User Completion

- DRM Controller…
  - Receives license
  - Extracts keys from license
  - Decrypts content
  - Generates financial transaction, where necessary
  - Hands content to rendering application
- Rendering application plays content





What were the results of Sony's attempt at DRM?

Results of Sony's DRM Project:

- The rootkit was uninstallable
- Installing software without permission is illegal in many countries
- Rootkit left backdoors that could be exploited by viruses
- Sony under class action lawsuit

Public Outcry

- The head of Sony BMG's global digital business, Thomas Hesse, told National Public Radio

> "Most people, I think, don't even know what a rootkit is, so why should they care about it?"

- Turns out people did care
  - Class action lawsuit in place against Sony
  - Uninstaller finally released for the rootkit

Digital Rights Management Revisit

- DRM and movie industry: DVD CSS
- DRM and music industry:
  - audio CD: from sony BMG
  - internet music: iTunes store
- E-Books: Adobe Acrobat, M$ Reader, Kindle

- 守护数字文档,数字版权管理:一个商业难题 [新华网 2006年7月7日]
  - 在国内某著名兵工厂的一次老总级别会议后,一份电子版的会议纪 要被秘密地发送到了几个有权限的重要人物手中,三个小时之后,这份文件将会自动销毁.
  - 一个商业难题
  - 新销售体系
  - 待填补的市场

Thus, we see ...

- DRM can help ensure companies, corporations, and other entities who share similar business that:
  - Rights are tracked at ingestion
  - Access is controlled during production processes
  - Protection for the content extends throughout product life-cycles

Thus, we see ...

- Additionally, **DRM** can integrate persistent content protection with content management to ensure:
  - Proper business practices
  - Implementation of new business models
  - Compliance with regulatory requirements in industries such as financial services, healthcare, and government

- 首批广播影视数字版权管理标准完成起草
  - http://news.cctv.com/china/20081108/105830.shtml
  - http://space.tv.cctv.com/video/VIDE1226188087000110

Previous Technologies

- PKI – Public Key Infrastructure
- PGP – Pretty Good Privacy
- S/MIME
- Access Control Systems
- Smart Cards
- Biometrics

How are these technologies different to DRM?

- Only protect the data in transit
  - E.g. over the Internet or on CD
- Once the data is opened, it can be:
  - edited
  - copied
  - printed
  - saved as an unprotected file

  And then
- Redistributed to anyone else in an unprotected format.

Rely on TRUST once the content is delivered

Protecting Digital Intellectual Property

- Preventing Copying with Encryption
  - 加密
- Preventing Copying with Watermarking
  - 水印

Preventing Copying With Encryption (加密)

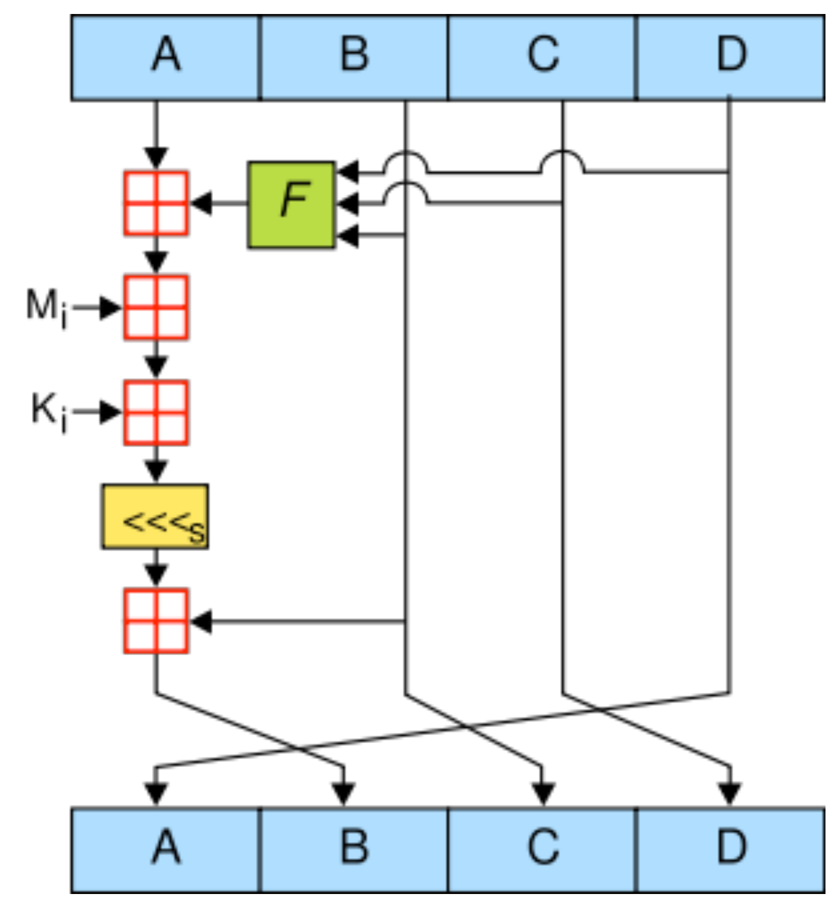
- Encryption is the scrambling of a message
  - Simple one is Caesar encryption
  - To decrypt (decode) message, you need one or more *Keys*
  - Also need an encryption *algorithm*, that specifies how to apply the key to the message to produce the scrambled message
- Symmetric key crypto: same key used for encrypt/decrypt
- Public key (we'll talk about the details later…):
  - Keys come in matched pairs: one encrypts, other decrypts
  - Given one key, you cannot deduce the other

Encryption

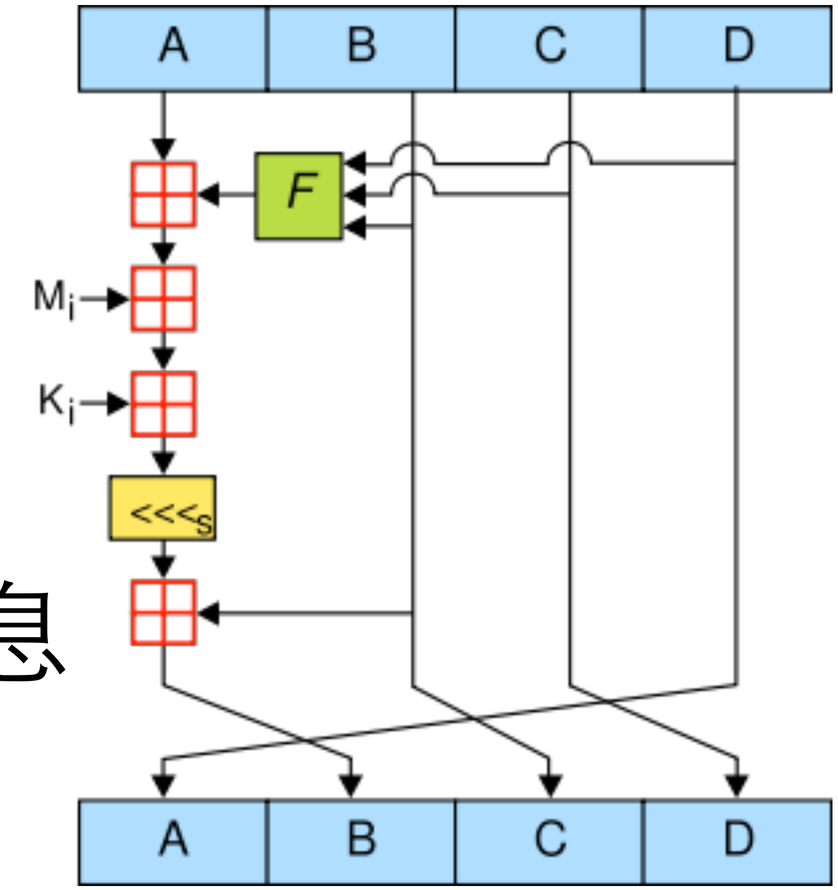
- RSA
- DES
- MD5

MD5
(Message Digest Algorithm version 5)

- MD5 is widely used in the open source world
  - Enough for data sharing
  - But not so safe

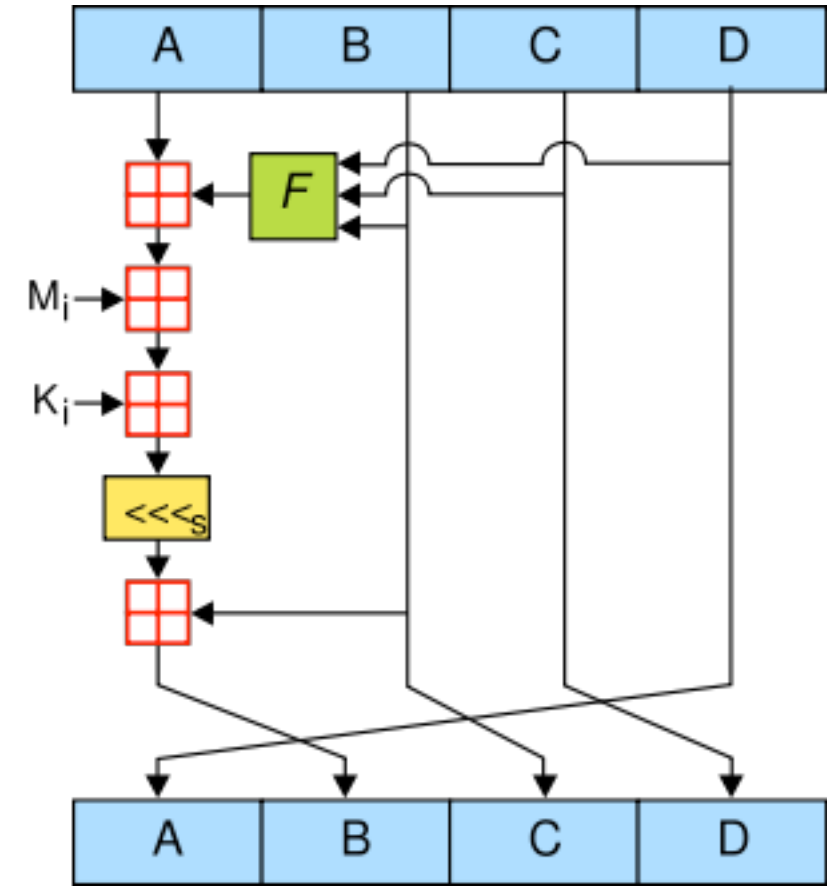


MD5 算法



- 输入：以512位分组来处理的信息
  - 每一分组又被划分为16个32位子分组
  - 对信息进行填充，使其位长对512求余的结果等于448
  - N*64+56个字节
- 输出：四个32位分组，构成128位散列
  - Hashing：散列

MD5算法



- Chaining Variable
  - A=0x01234567， B=0x89abcdef， C=0xfedcba98， D=0x76543210
- 循环运算
  - A到a，B到b，C到c，D到d
  - 主循环有四轮
    - 一轮进行16次操作
    - 每次操作对a、b、c和d中的其中三个作一次非线性函数运算

MD5 算法

- 基本函数
  - F(X,Y,Z) =(X&Y)|((~X)&Z)
  - G(X,Y,Z) =(X&Z)|(Y&(~Z))
  - H(X,Y,Z) =X^Y^Z
  - I(X,Y,Z)=Y^(X|(~Z))
  - & 表示“与”，|表示“或”，
    ~表示“非”，^表示“异或”
  - 基本操作

- FF(a, b, c, d, Mj, s, ti)
  a = b + ((a + F(b, c, d) + Mj + ti) << s)
- GG(a, b, c, d, Mj, s, ti)
  a = b + ((a + G(b, c, d) + Mj + ti) << s)
- HH(a, b, c, d, Mj, s, ti)
  a = b + ((a + H(b, c, d) + Mj + ti) << s)
- II(a, b, c, d, Mj, s, ti)
  a = b + ((a + I(b, c, d) + Mj + ti) << s)
- Mj表示消息的第j个子分组（从0到15）

MD5算法

- 在第i步中，ti是4294967296*abs(sin(i))的整数部分，i的单位是弧度。

- 完成上述64步操作之后，将A、B、C、D分别加上a、b、c、d。然后用下一分组数据继续运行算法

- 最后的输出是A、B、C和D的级联。

- 例：（可试用python中的md5实现：hashlib）

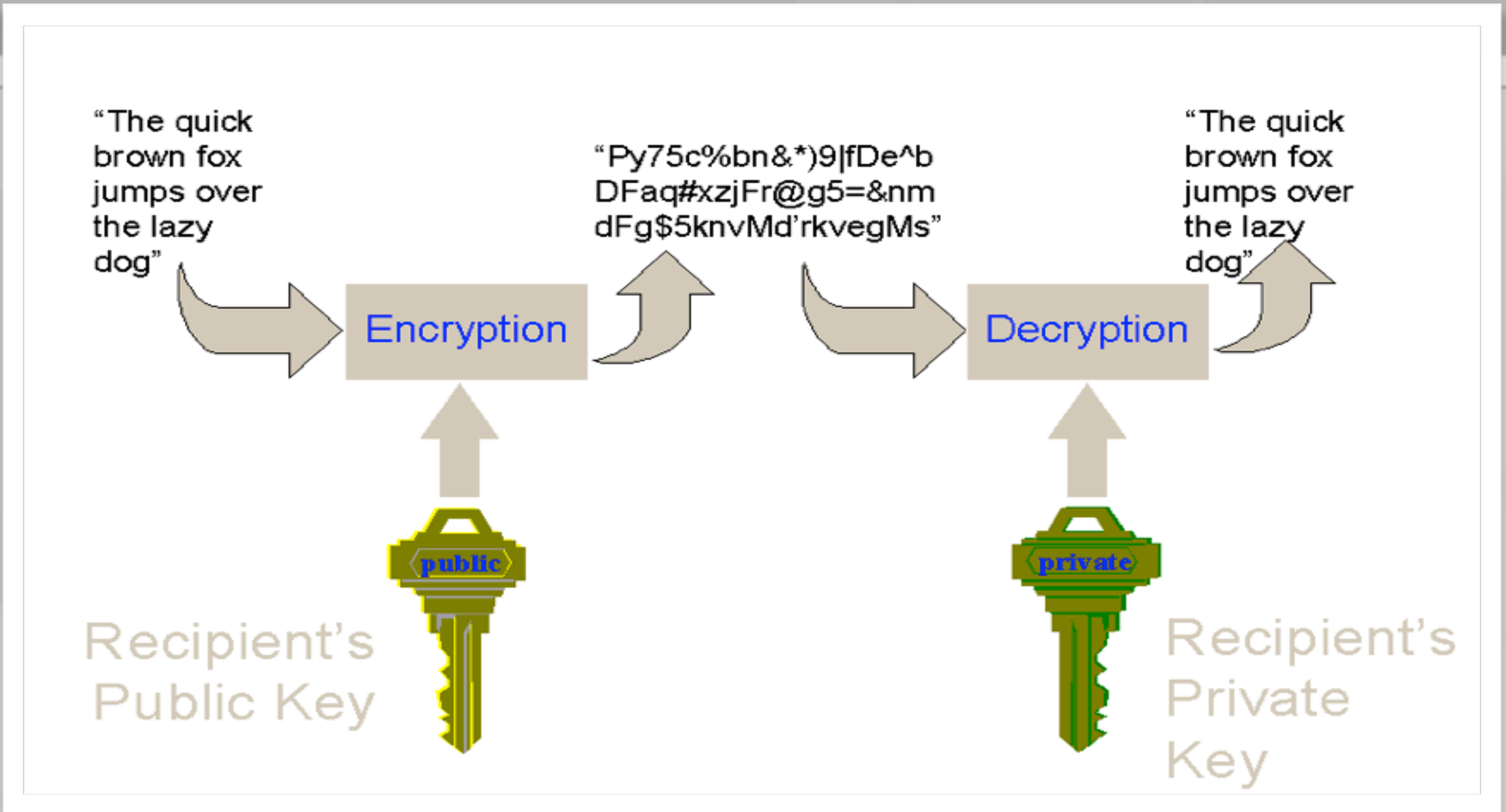
  - MD5 ("") = d41d8cd98f00b204e9800998ecf8427e

  - MD5 ("abc") = 900150983cd24fb0d6963f7d28e17f72

开源密码体系的崩溃

- 山东大学的王小云教授
  - [Crypto 2004], 利用hash碰撞原理，攻破MD5、HAVAL-128、 MD4和RIPEMD算法
  - 2005年8月, 给出攻击SHA－1的算法

Basic Idea of Cryptography



Think of encryption key as sealing an envelope, and decryption key as unsealing it.

How do you “break” encryption?

- Usual assumptions of cryptography…
  - Adversary knows details of algorithm (not in WWII!)
  - Adversary may know something about nature of messages (why would this help?)
  - Adversary *doesn’t know decryption key(s)*
- Hard: exploit mathematical weakness in the algorithm
- Hard: guess key by (educated) trial and error
- Usually easier: attack some weaker part of the system
  - Usually, trick system into revealing a key
  - Chain is only as strong as weakest link!

DVD Content Scrambling System (CSS)

- To each **licensed DVD player** corresponds a **decryption key**:
  - P1, P2, …, Pn
- Each disc is encrypted under its own key, call it D
  - n copies of D are stored on the disc; each copy encrypted with one player's P
  - Player finds a D that it can decrypt, then uses D to play disc
- DVD player is a trusted client
  - It's not supposed to ever reveal any D, or its own P
  - What happens if either of these occur?
  - Why can't you convert DVD to another format?
  - Why can't you make direct copies of a DVD onto another disc (copying the D keys along with the content?)

Early DeCSS timeline...

- Sep ’99, DeCSS released as open-source Linux DVD player
- Dec ’99, DVDCCA sues 500 individuals in California for hosting DeCSS, alleging trade-secret violations
- Jan ’00, MPAA sues 2600.com in New York under DMCA’s copyright protection circumvention laws
- Jan ’00, DVD Source Code Distribution Contest
- Jan ’00 Jon Johansen arrested in Norway, later released
- Aug 00 MPAA wins DMCA suit in NYC

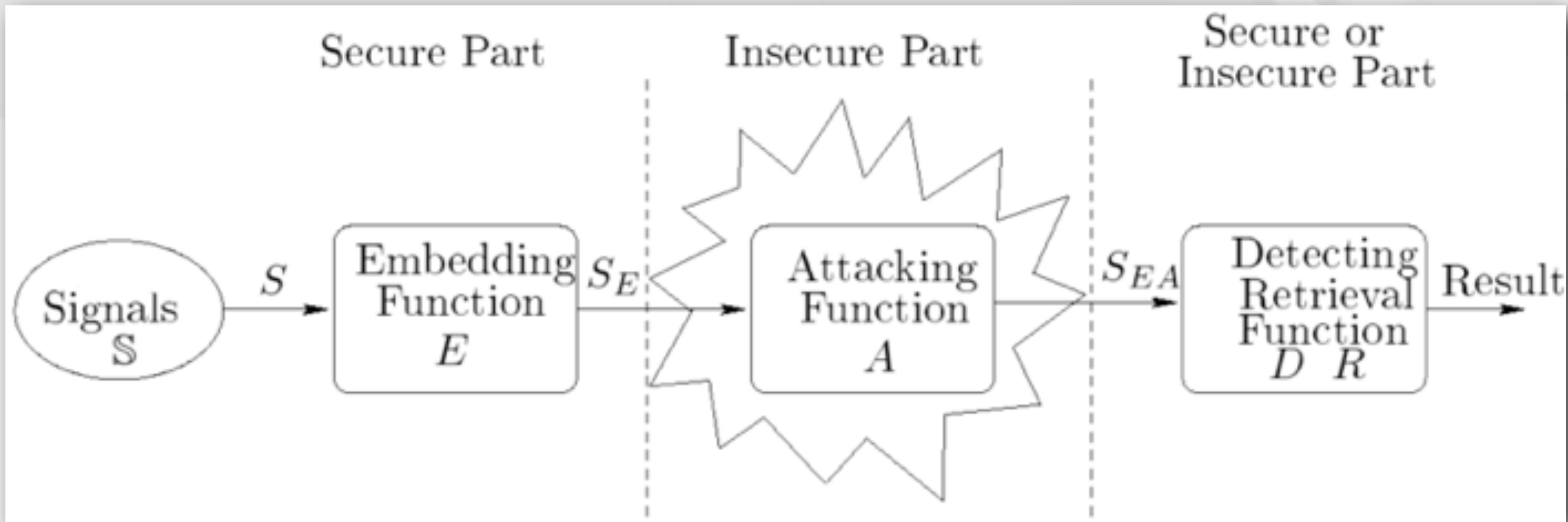
How Was CSS cracked?

- Idea =>
- P must appear somewhere in the decryption code of a trusted player
  - –Hardware players difficult to reverse-engineer/probe
  - –Software players maybe easier? …turns out yes!
  - –Later analysis revealed weaknesses in CSS…it probably could have been broken *without* first recovering a key

- Original goal of CSS: even if one P is compromised, others are still sound
- Flaw: weakness in the algorithm allowed *all* P's to be compromised once a single P was found
  - –Why wasn't this flaw discovered *before* the algorithm went into production players?

Preventing Copying With Watermarking (水印)

- digital art
- 票据防伪
- 数据隐藏
- 隐蔽通讯



Stenography

1. removing all but the last 2 bits of each color component
2. X 85

Digital Watermark

- Invisible ink on multimedia data
  - image
  - video
  - music
  - graphics

Digital Watermark

- Image

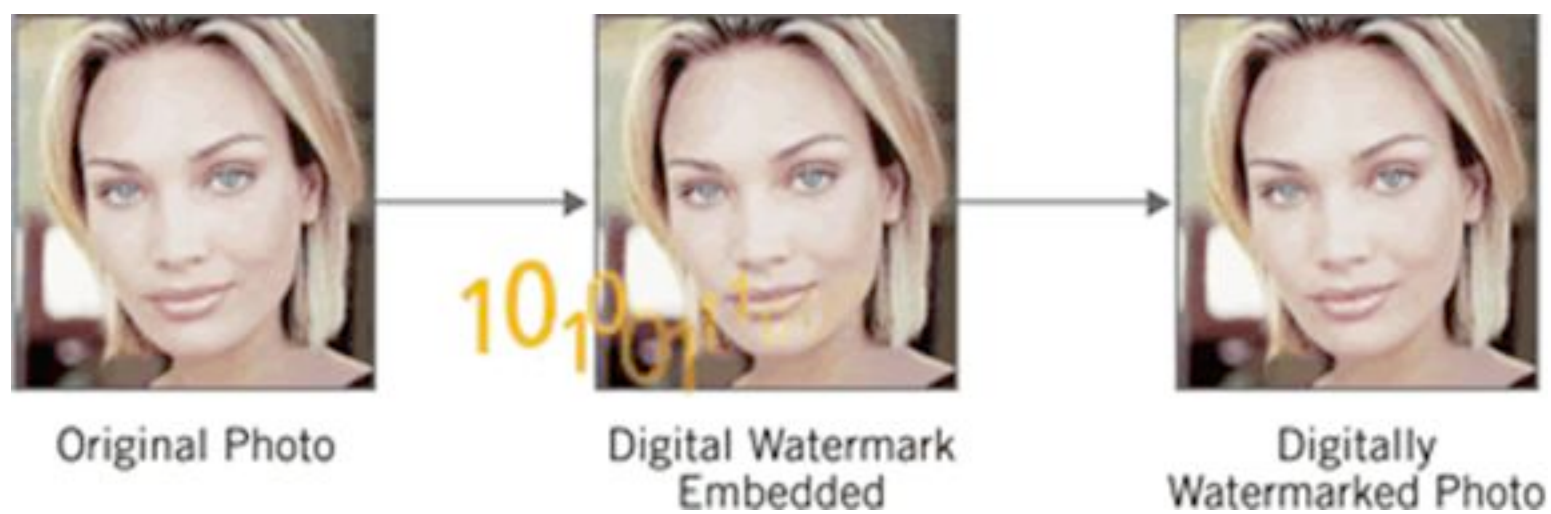
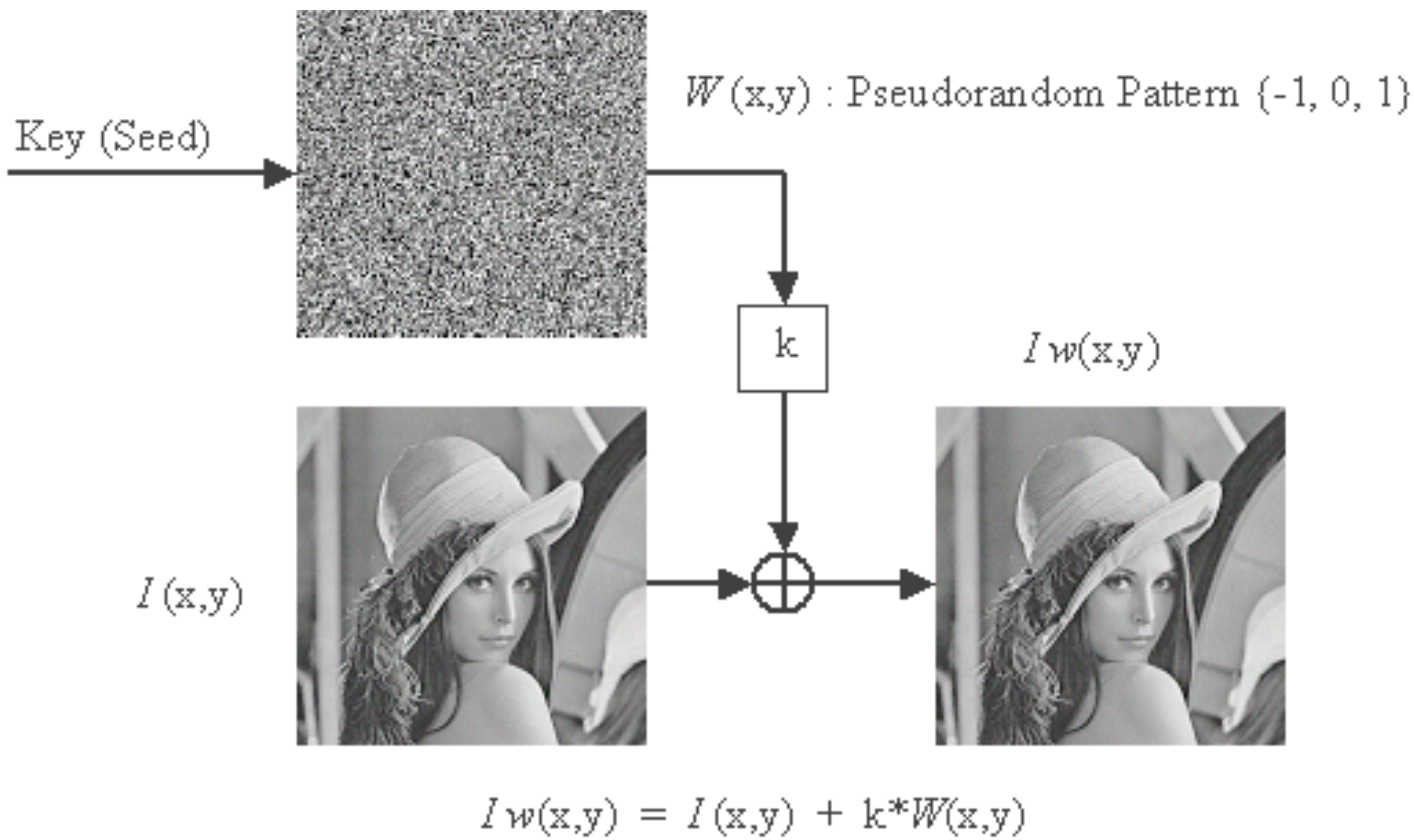
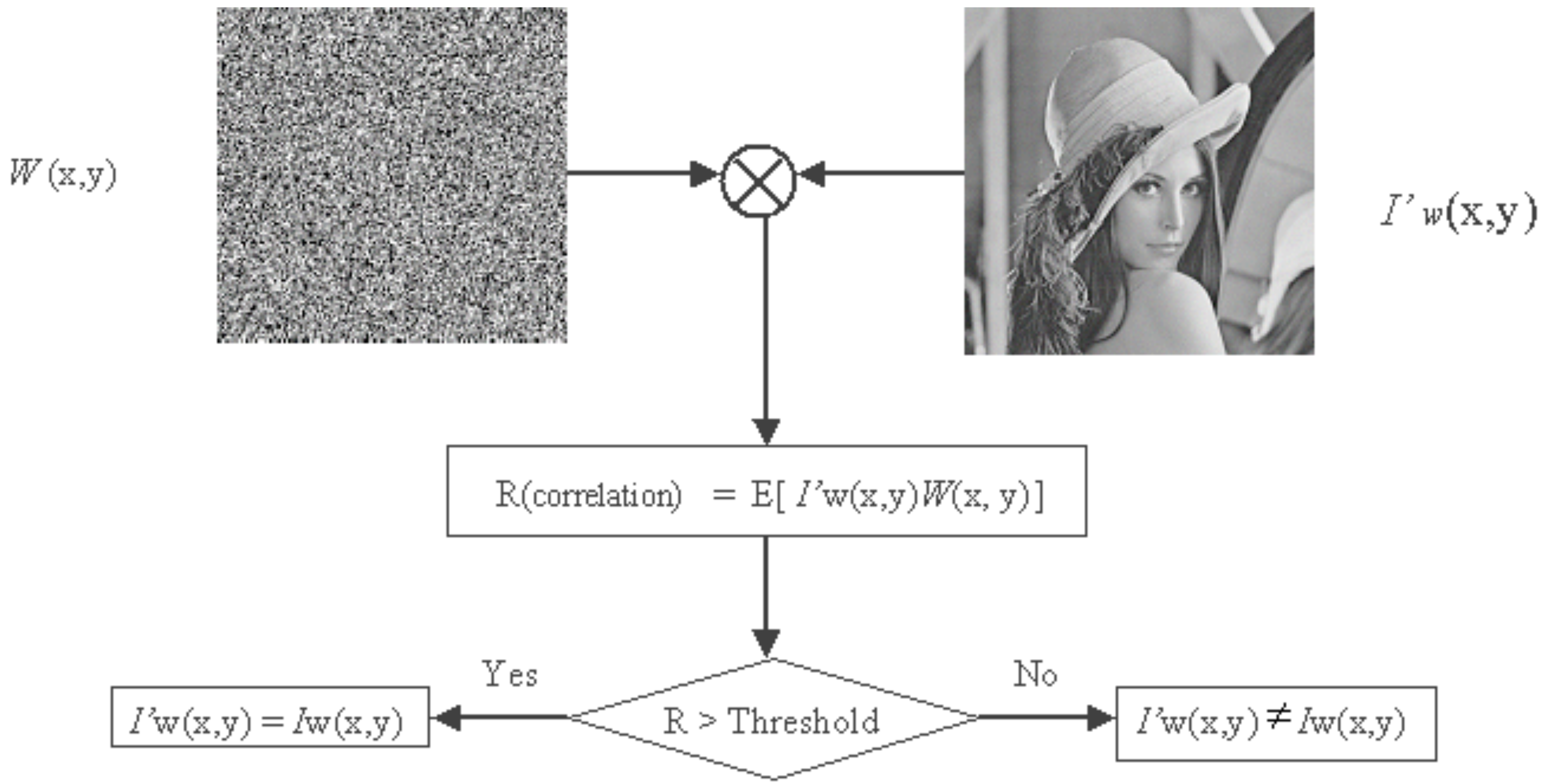


Image watermarking



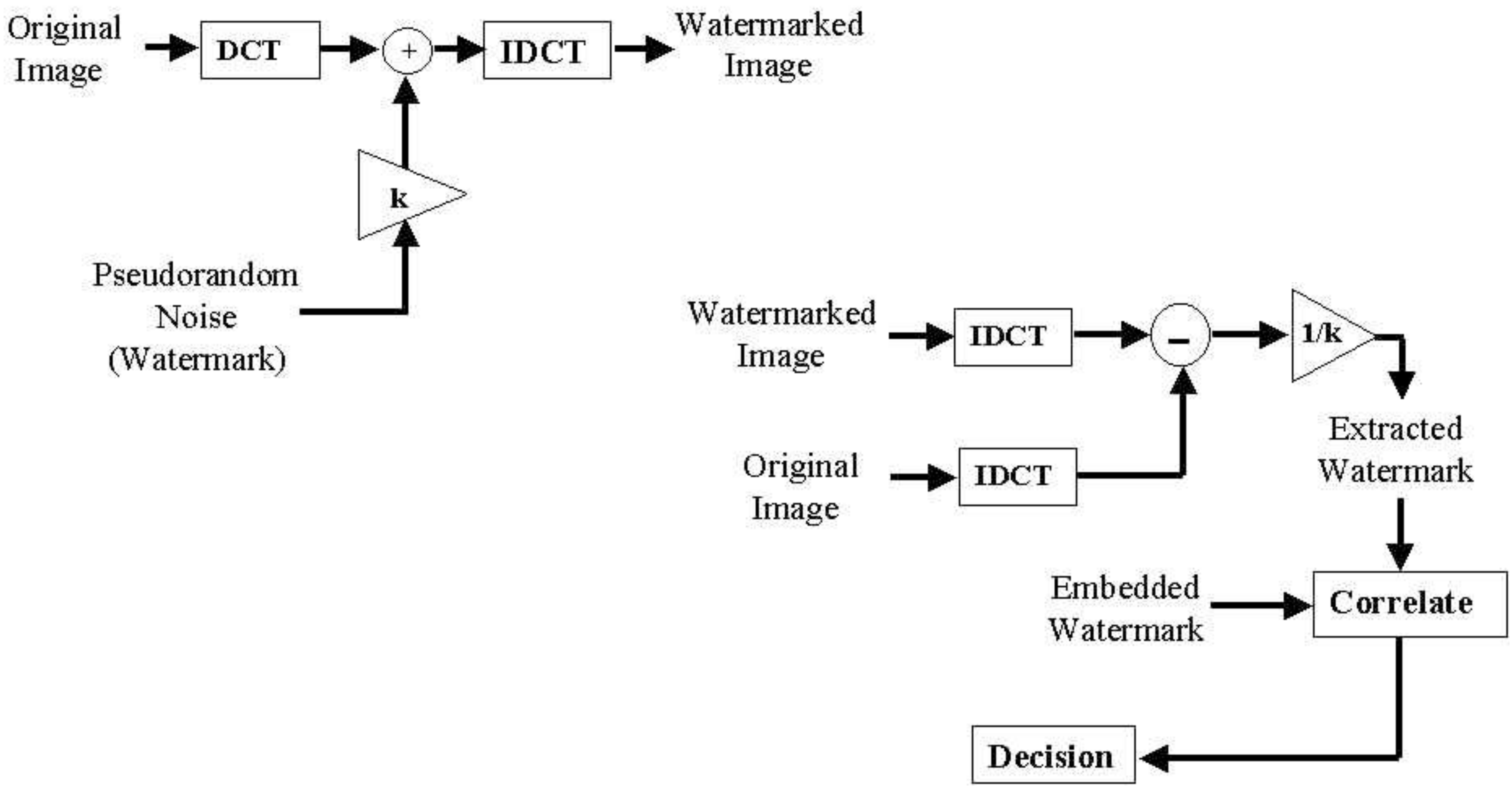
Embedding

Image watermarking



Detecting

DCT based algorithm



REF: http://scien.stanford.edu/pages/labsite/2001/ee368/projects2001/dropbox/project06/

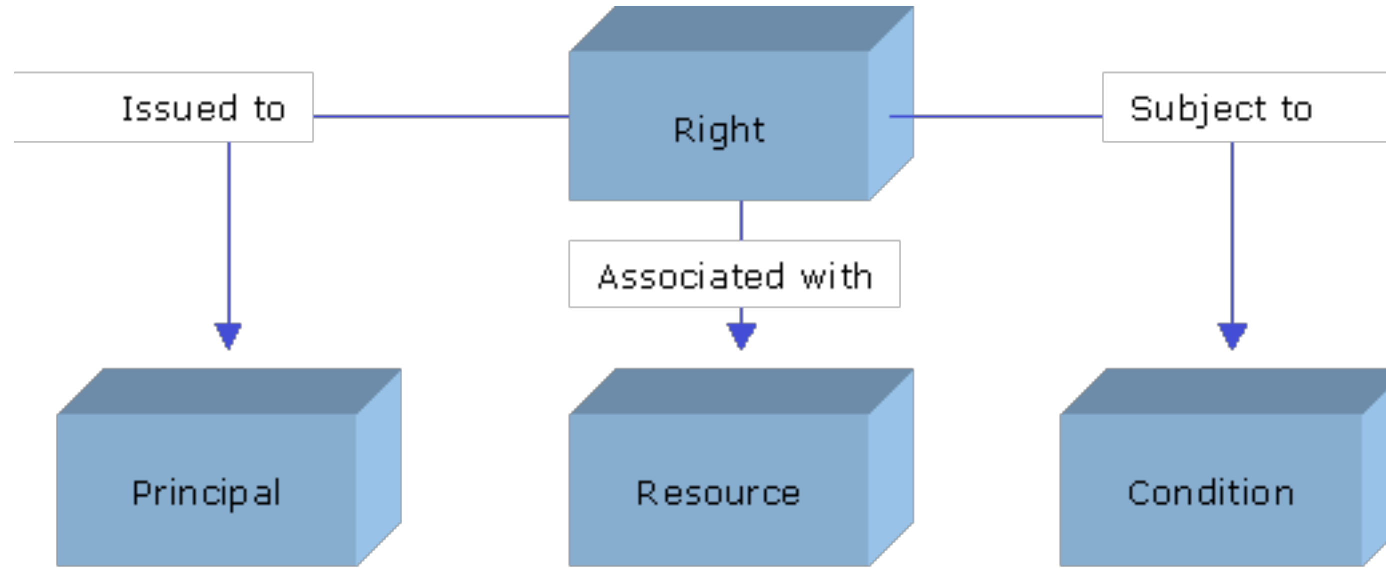
Digital Watermark

Music: mp3stego

http://www.petitcolas.net/fabien/steganography/mp3stego/index.html

Digital Rights Management
– Rights Expression Language (REL)

Metadata for DRM



- MPEG-21: REL
  - A digital item
    - is a structured digital object with a standard representation, identification and metadata
  - User
    - is any entity that interacts in the MPEG-21 environment or makes use of digital items

Rights model

- Render rights
  - –View, Print, Play or Execute
- Transport rights
  - –Copy, Move, Loan
- Derivative work rights
  - –Edit, Embed, Extract
- Utility rights
  - –Backup, Caching, Data integrity

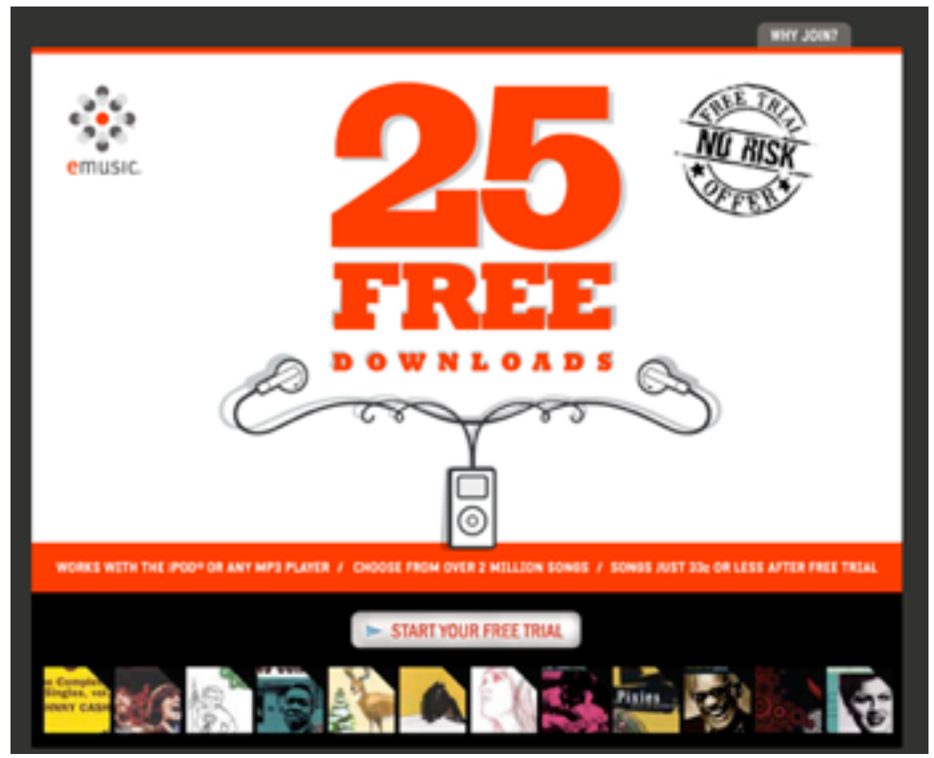
DRM technologies and associated devices

Name	Used in	Date to use	Description
Fairplay	ipod, iphone, itunes	2003+	The purchased music files are encoded as AAC, then encrypted with an additional format that renders the file exclusively compatible with iTunes and the iPod
3-play	Microsoft Zune	2006+	Music files that are received wirelessly from other Zune devices can be played only a maximum of three times on the device.
Janus WMA DRM	All PlaysForSure Devices	2004+	Janus is the codename for portable version of Windows Media DRM for portable devices.
OMA DRM	Implemented in over 550 phone models	2004+	A DRM system invented by the Open Mobile Alliance to control copying of cell phone ring tones

DRM opposition

digital rights management
= digital restrictions management ?

DRM-free



- Apple began selling "DRM-Free" music through their iTunes store in April of 2007
- the DRM-Free iTunes files were still embedded with each user's account information

Digital Rights Expression Languages

- Rights may be managed using digital rights expression languages.

- DRELs specify the permissions given to
  - users, distributors and repositories
  - and the conditions and obligations that have to be satisfied for these permissions to be exercised.

Rights Expression Language (REL)

- A standard way to express and interpret rights specification for interoperability.
- Comprehensive, generic, precise and extensible.

- eXtensible rights Markup Language (**XrML**).
  - –XrML 2.0 : MPEG REL
- Open Digital Rights Language (**ODRL**).
  - –ODRL 1.1 : OMA (Open Mobile Alliance) REL

General description of RELs

- A rights expression language (REL) is a type of policy authorization language.
  - Focus is on expressing rights granted by one party to another.
  - Issuance and delegation rights for other grants are core concepts.
    - Can be used to model lending, loans, transfers of rights.

- REL design goals:
  - Provide a flexible, extensible mechanism for expressing authorizations.
  - Enable interoperability across various policy evaluation systems.
  - Make it easy for policy authors (e.g. content owners) to express their desired policies.

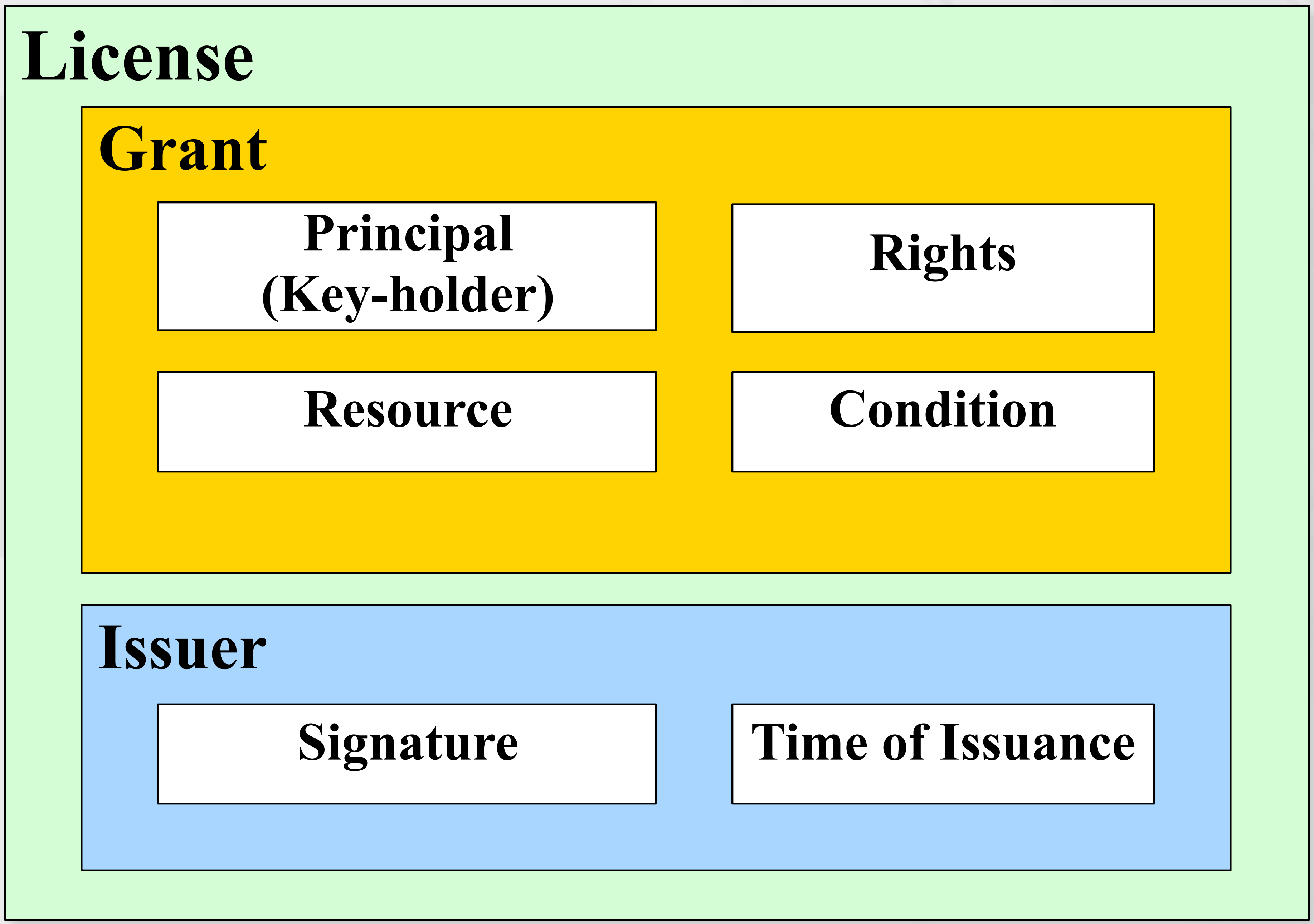
An example REL: XrML 2.X

- XrML, the *XML Rights Management Language*, is a standard currently under development

XrML introduction

- The only REL in working DRM systems.
- Specification language:
  – Programmers specify high-level rights in a license file.
  – An XrML interpreter parses the license file.
  – REL SDK for building an XrML interpreter.
- Data model:
  – License, grant, principal, right, resource and condition

XrML license



XrML 2.X

- In the RM context, XrML 2.X allows content owners a systematic way to express their intent for distribution and consumption.
- Like other policy languages, XrML 2.X **licenses** (statements) declare authorizations, but cannot enforce compliance.
  - Systems that consume XrML 2.X licenses must be trusted by the license issuer to properly enforce the grants specified within the license.
- Licenses are digitally signed by the issuer to protect their integrity.
- Licenses may be embedded within content or move independently.

Semantic of a Grant

- Every XrML 2.X grant has the following form:
  - Issuer authorizes principal to exercise a right with respect to a resource subject to conditions.
  - A license is a collection of one or more grants made by the same issuer.

- Grants may be **chained** together:
  - Bill's RM system trusts Tom and his delegates.
  - Tom delegates the right to license printing to John.
  - John issues a license: "Bill has the right to print the book."
  - Therefore Bill can print the book.

Sample XrML 2.X License

```
<?xml version="1.0" encoding="UTF-8" ?>
  <license>
  <grant>
    <keyHolder>  …  </keyHolder>
    <mx:play />
    <mx:diReference>
    <mx:identifier>
          urn:mpeg:example:2002:twotonshoe:album
     </mx:identifier>
     </mx:diReference>
  </grant>
  <issuer> … </issuer>
</license>
```

XrML authorization model

- Input
  - Principal
  - Right
  - Resource
  - Time interval
  - Licenses
  - Designated "root grants" (implicitly trusted)

- Output
  - "No"
  - "Yes," unconditionally
  - "Maybe," if a set of conditions are also met

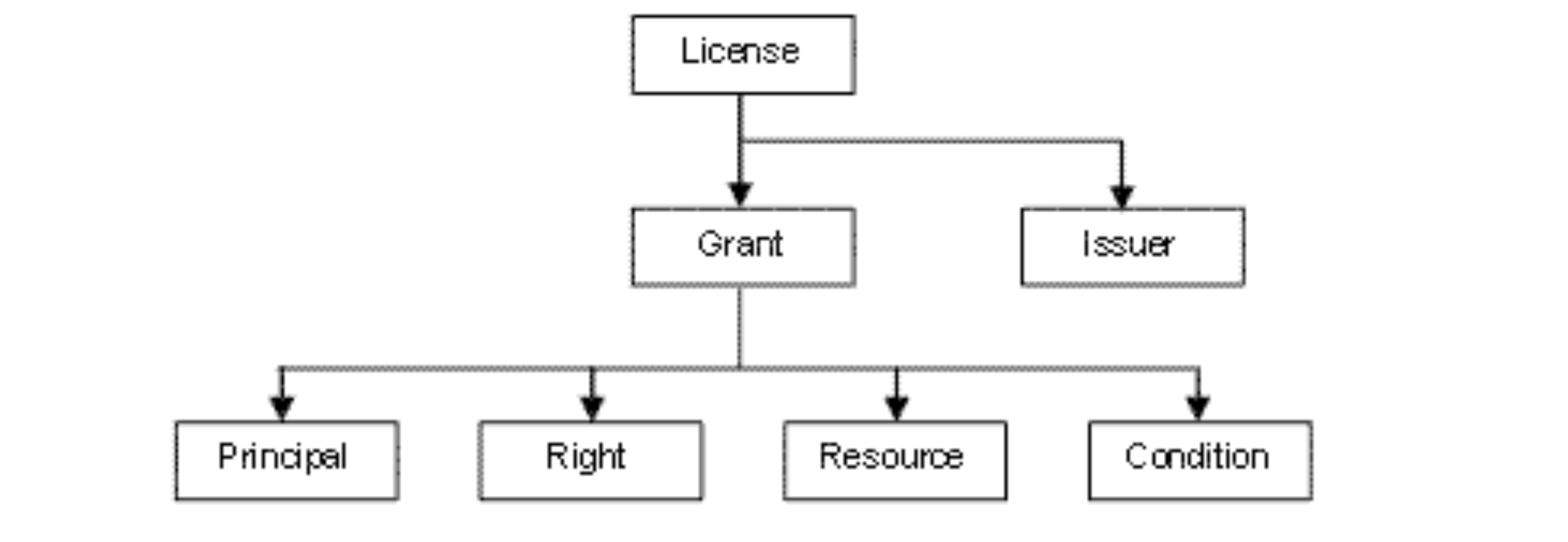
XrML Key Language Features

- Mechanisms for enhanced expressivity
  - Patterns, variables and quantifiers
  - Grouping grants
  - Delegation
- Meta-rights
  - Issue
  - Obtain
  - Revocation
  - PossessProperty
- Linking conditions
  - PrerequisiteRight

MPEG-21 REL

- Derived from XrML
- 3 Components:
  - –Kernel set
  - –Standard extension
  - –multimedia extension

structure of a simple license



浙江大学计算机学院
数字媒体与网络技术

XrML 2.X and Multiple Authorities

- XrML 2.X offers a new level of expressiveness
  - Enables representation of a wider range of scenarios.
- Example scenario: evaluating authorizations from multiple authorities for a resource.
  - Today, RM systems operate using a "closed-world assumption."
    - Any action not explicitly authorized by the content owner is prohibited.
  - Copyright doesn't work like this.
    - Copyright is a liability-based system.
    - Some actions are permitted by law even if they are not explicitly authorized by the copyright holder.
  - How might we use XrML 2.X to represent authorizations as well as limitations built into the law?

XrML 2.X and Multiple Authorities (cont'd)

- Content creators are given exclusive rights by law; these rights are then licensed to consumers.
- Limitations on the exclusive rights contained in a copyright can be thought of as independent grants of licenses by Congress to the consumer.
  - "Congress says every library has the right to make an archival copy of a work" (17 U.S.C. 108).
  - Variables allow us to write licenses that apply to (potentially undefined) sets of content and users.
  - Congressional grants can be conditioned on possession of a licensed copy of the work.
- RM systems would need to recognize both the content owner as well as Congress as authorities for a given work.

Case Studies

InterTrust

- Original DRM vendor (with IBM)
  - May have coined the term
  - Originally called Electronic Publishing Resources
  - First implementations in hardware
  - Major patent portfolio
- New technology: Rights|System
  - Framework for multiple devices
    - Rights|Desktop for PCs
    - Rights|TV for settop boxes
    - Rights|PDA for handheld devices
    - Rights|Phone for Symbian mobile phones
  - Public encryption algorithms

IBM EMMS

- Developed in IBM labs over period of 8 years
- Cross-device, like InterTrust
- Integration with IBM server components
  - –WebSphere
  - –DB2
  - –Service Provider Delivery Environment (SPDE)

Microsoft

- 1st generation: Windows Media Player
- 2nd generation: Digital Asset Server
  - Server for Microsoft Reader E-Books
  - Uses subset of XrML
- 3rd generation: “Unified DRM” (RMS)
  - One DRM for all devices & platforms
  - Open API for rendering app developers
  - XrML based

MacroVision (1985-)

- Copy protection technique for VHS tapes
- Inserts special signals into the vertical blanking interval of NTSC protocol
  - affects automatic gain control in most VCRs, but is ignored by most televisions
  - difficult to remove from the original signal
- Makes subsequent recordings shake and have periods of bright and dark frames

Apple's FairPlay Technology

- DRM for iTunes
  - –playing, recording, and sharing of files
- Moves beyond "protection only"
  - –allows media to be shared among devices
  - –allows others to listen to (but not copy) music
  - –allows music to be burned to an audio CD, which loses the DRM protection

How FairPlay Works

- iTunes uses encrypted MP4 audio files
- Acquire decryption key by trying to play song
  - player generates a unique ID
  - sends this ID to the iTunes server
  - if there are less than N authorizations in your account, the server responds with decryption key
- The decryption key itself is encrypted so cannot be given to another machine

Discussion

- Is FairPlay too lenient, too stringent, or just about right?
- What is your experience with this DRM?
- What happens if Apple decides to stop supporting FairPlay?

浙江大学计算机学院
数字媒体与网络技术

Digital Asset Management
数字媒体资源管理

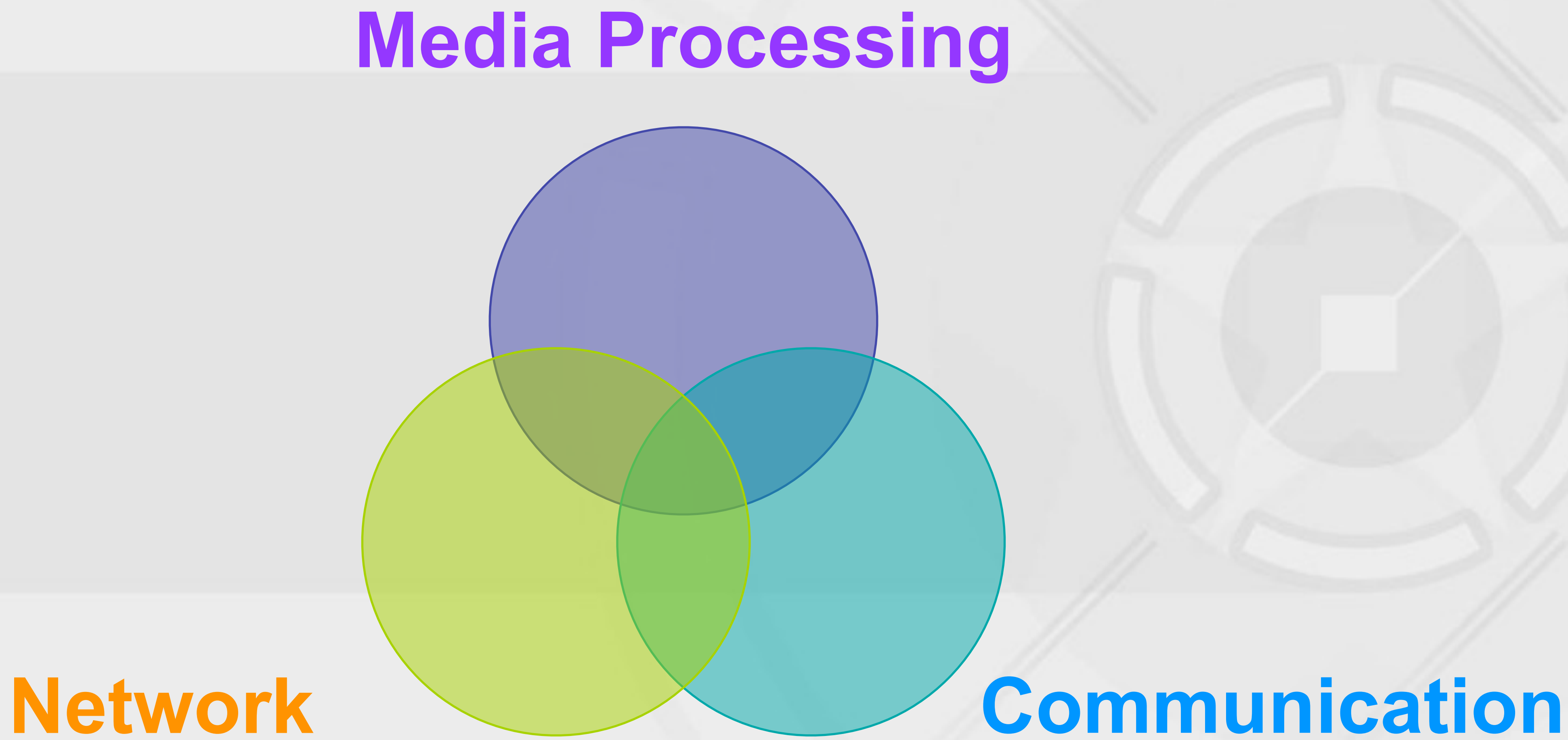
5. Streaming multimedia

任课老师：张宏鑫
2014-10-30

Keys of Streaming Media

- Algorithms (**)
- Standards (*****)
- Complete End-to-End systems (***)
- Research Frontiers(*)

Ubiquitous Multimedia



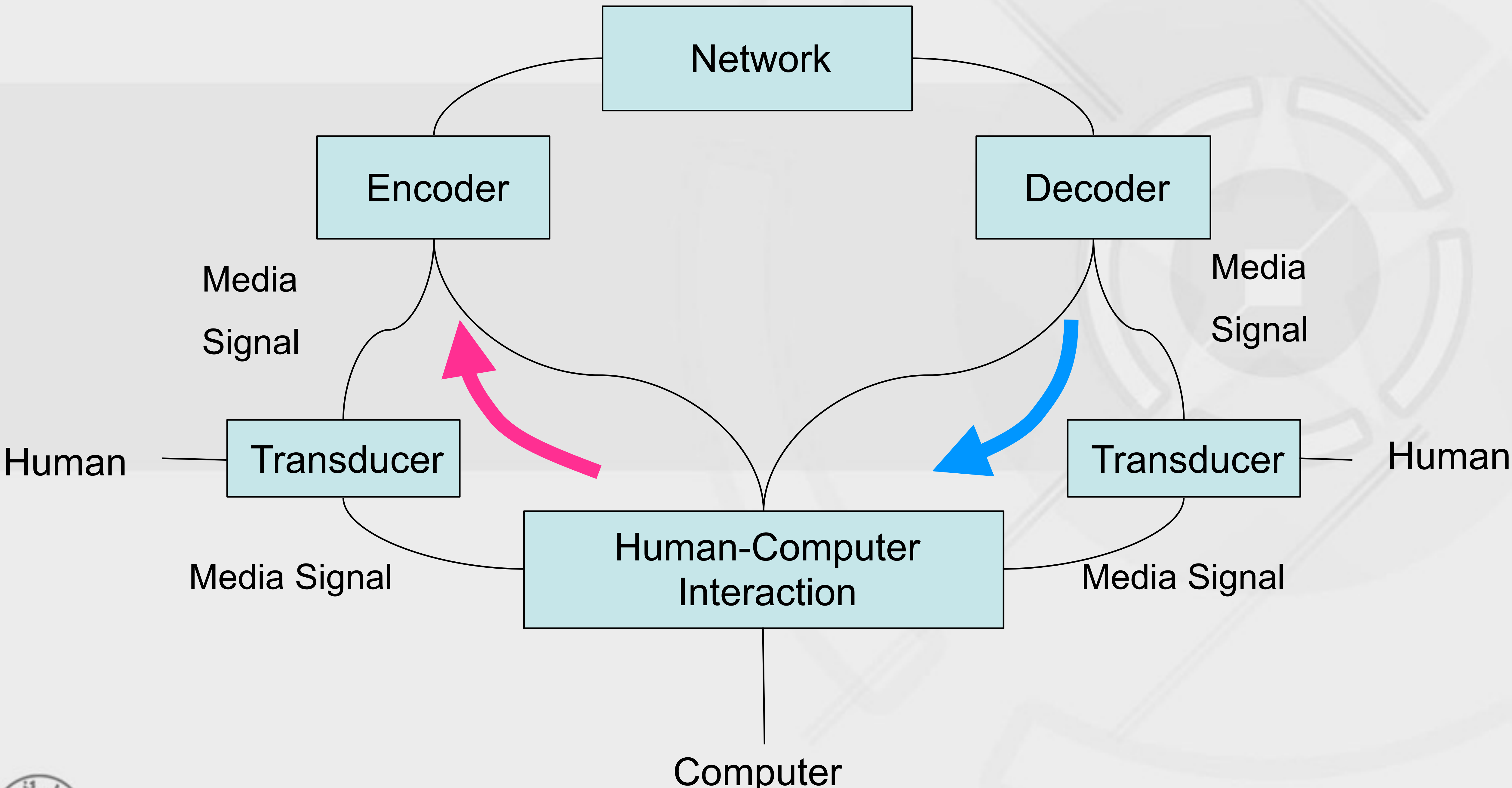
The dimensions of multimedia

- Audio, Video
- Images
- Graphics
- Animation
- Text, Data files, Speech, Handwriting

Key Dimensions of Communication Tech

- Antenna Diversity and Space-Time Processing
- Channel Equalization
- Bitstream Interleaving
- Multiple Access and Interference Cancellation
- Modulation and Error Protection

Multimedia Communications System



Key Dimensions of Networking

- Media Access Protocols
- Error Control: ARQ
- Admission Control, Scheduling
- Routing, Multicasting
- Mobility Management and User Tracking
- Protocols for ATM, IP and Hybrid Networks

Multimedia in networks (1): Characteristics

Fundamental characteristics:

- Typically **sensitive delay**.
- But **loss tolerant**: infrequent losses cause minor glitches that can be concealed.
- **Antithesis of data** (programs, banking info, etc.), which are loss intolerant but delay tolerant.
- Multimedia is also called "**continuous media**"

Classes of MM applications:

- Streaming stored audio and video
- Streaming live audio and video
- Real-time interactive video

Multimedia in networks (2): Applications

Streaming stored MM

- Clients request audio/video files from servers and pipeline reception over the network and display
- Interactive: user can control operation (similar to VCR: pause, resume, fast forward, rewind, etc.)
- Delay: from client request until display start can be 1 to 10 seconds

Unidirectional Real-Time:

- similar to existing TV and radio stations, but delivery over the Internet
- Non-interactive, just listen/view

Interactive Real-Time :

- Phone or video conference
- More stringent delay requirement than Streaming & Unidirectional because of real-time nature
- Video: < 150 msec acceptable
- Audio: < 150 msec good, <400 msec acceptable

Multimedia in networks (3): Challenges

- **TCP/UDP/IP** suite provides best-effort, no guarantees on delay or delay variation.
  - Streaming apps with initial delay of 5-10 seconds are now commonplace, but performance deteriorates if links are congested (transoceanic)
  - Real-Time Interactive apps have rigid requirements for **packet delay and jitter**.
  - **Jitter** is the variability of packet delays within the same packet stream.

- Design for multimedia apps would be easier if there were some 1st and 2nd class services (**QoS**).
  - But in the public Internet, all packets receive equal service.
  - Packets containing real-time interactive audio and video stand in line, like everyone else.
- There have been, and continue to be, efforts to provide differentiated service.

Multimedia in networks (4): making the best of best effort

To mitigate impact of "best-effort" Internet, we can:

- Use UDP to avoid TCP and its slow-start phase…
- Buffer content at client and control playback to remedy jitter
- We can timestamp packets, so that receiver knows when the packets should be played back.
- Adapt compression level to available bandwidth
- We can send redundant packets to mitigate the effects of packet loss.

We will discuss all these "tricks"

How should the Internet evolve to better support multimedia?

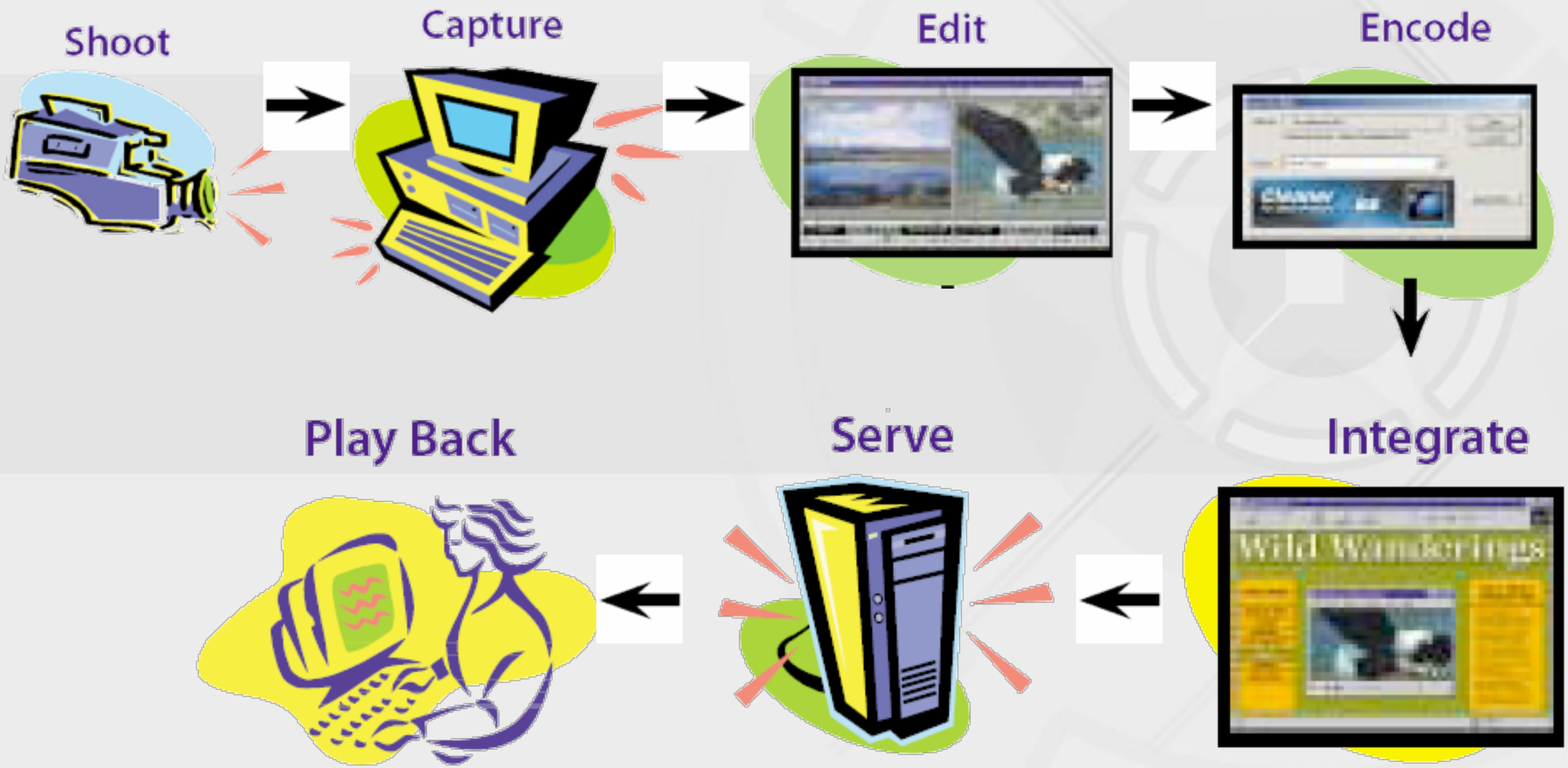
Integrated services philosophy:

- Change Internet **protocols** so that applications can reserve end-to-end bandwidth
  - Need to deploy protocol that reserves bandwidth
  - Must modify scheduling policies in routers to honor reservations
  - Application must provide the network with a description of its traffic, and must further abide to this description.
- Requires new, complex software in hosts & routers

Differentiated services philosophy:

- Fewer changes to Internet infrastructure, yet provide 1st and 2nd class service.
- Datagrams are marked.
- User pays more to send/receive 1st class packets.
- ISPs pay more to backbones to send/receive 1st class packets.

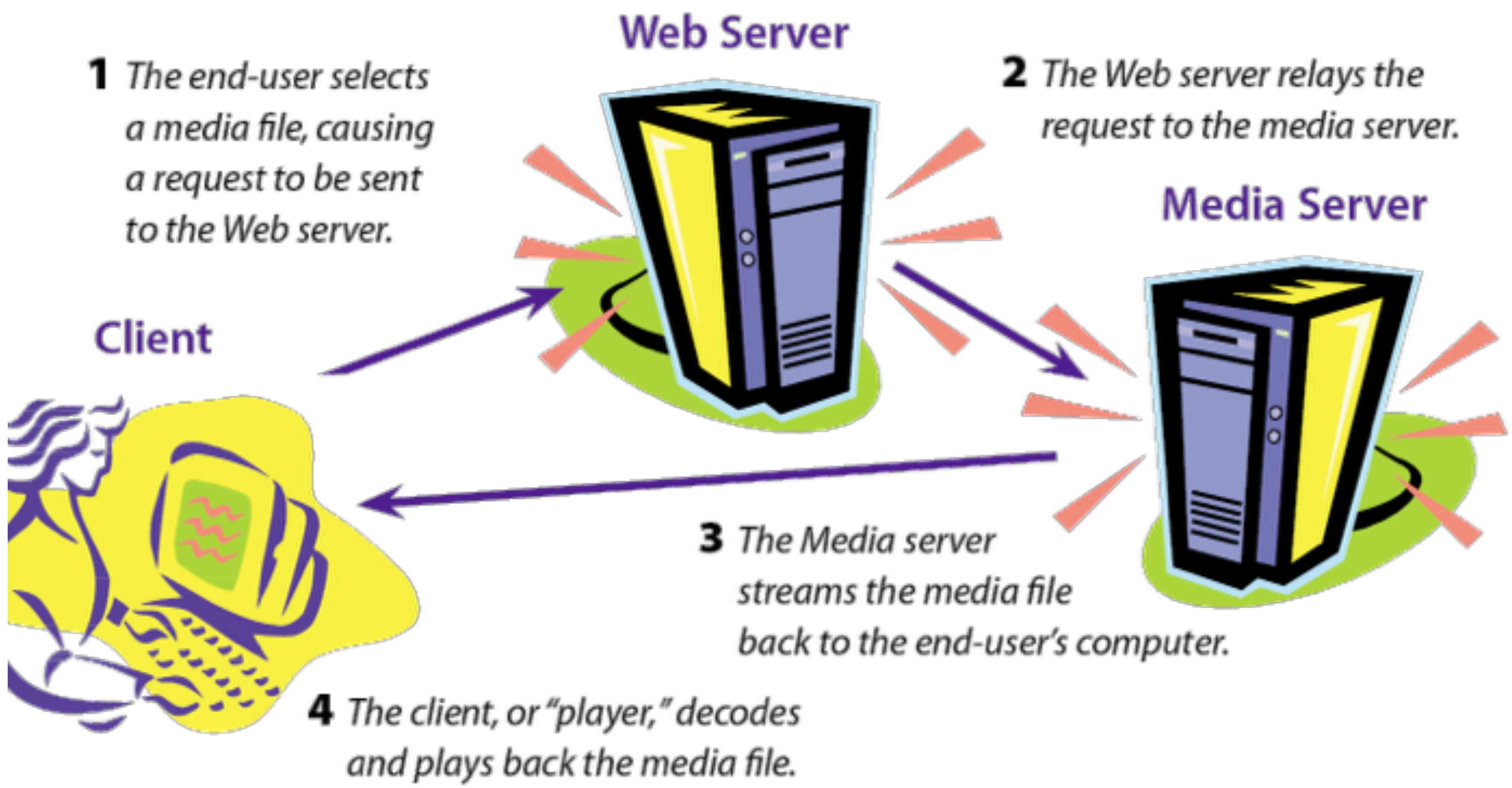
Workflow of Streaming Media



The primary characteristics of "streaming media"

- Three primary characteristics combine to define streaming media
  - –Streaming media technology enables real-time or on-demand access to multimedia content via the Internet or an intranet.
  - –Streaming media is transmitted by a media server application, and is processed and played back by a client player application, as it is received.
  - –A streamed file is received, processed, and played simultaneously and immediately, leaving behind no residual copy of the content on the receiving device.

HOW DOES STREAMING WORK?

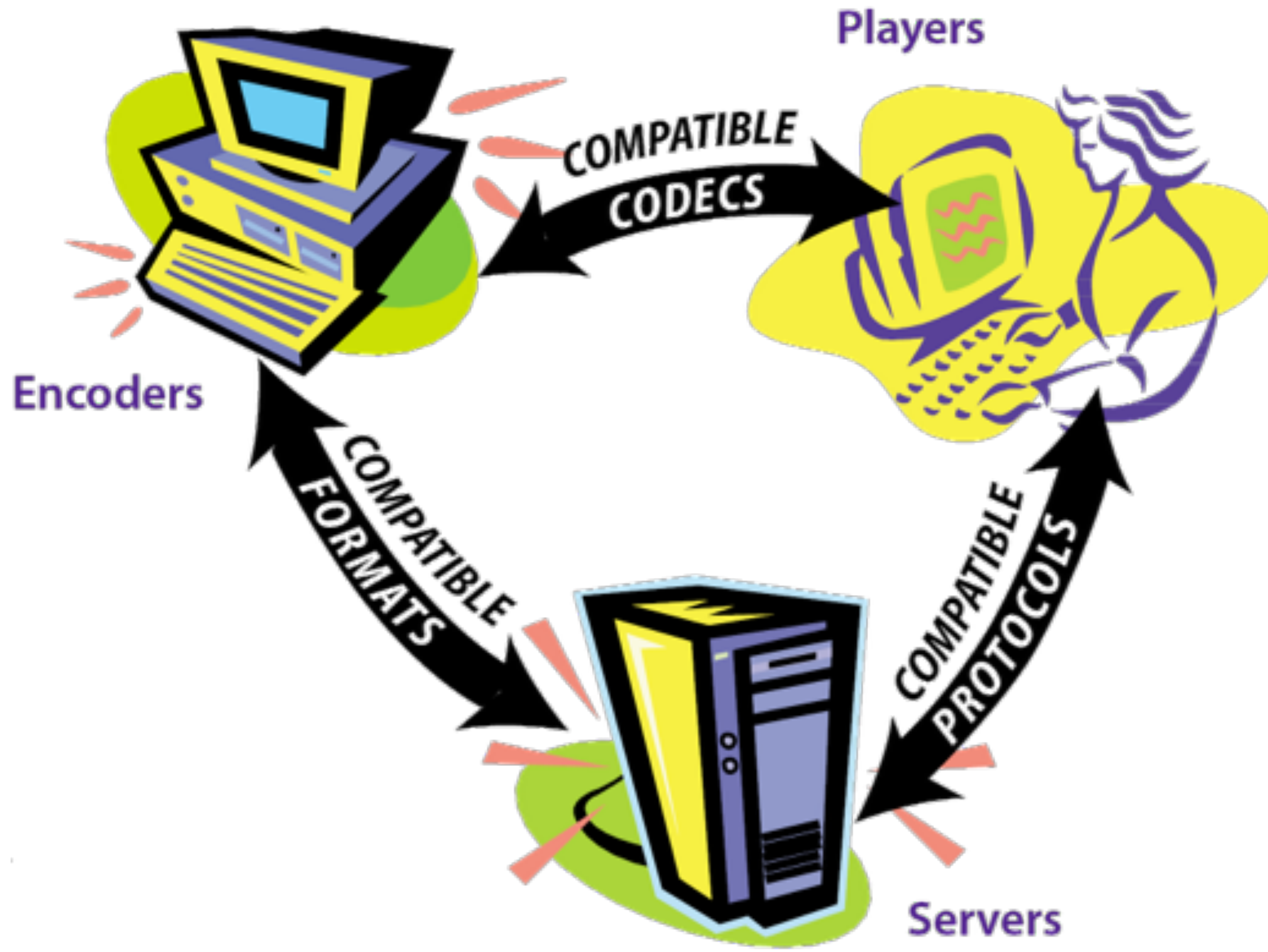


WHERE DO STREAMS COME FROM?

- Streaming media architectures.
  - –Streaming media architectures are comprised of
    - encoding and transmission methods,
    - server software, and
    - players (client software).
  - –The three most popular streaming media architectures
    - RealMedia,
    - Windows Media, and
    - QuickTime.

It is all interrelated

- In a streaming architecture, everything must be compatible.



Streaming media formats

Architecture	Native Formats	Streaming Media File Extensions
QuickTime	QuickTime Format	.mov (sometimes .qt or .qti)
RealMedia	RealMedia Format	.rm
Windows Media	Advanced Streaming Format or Windows Media Video/Audio	.asf, .wmv, .wma

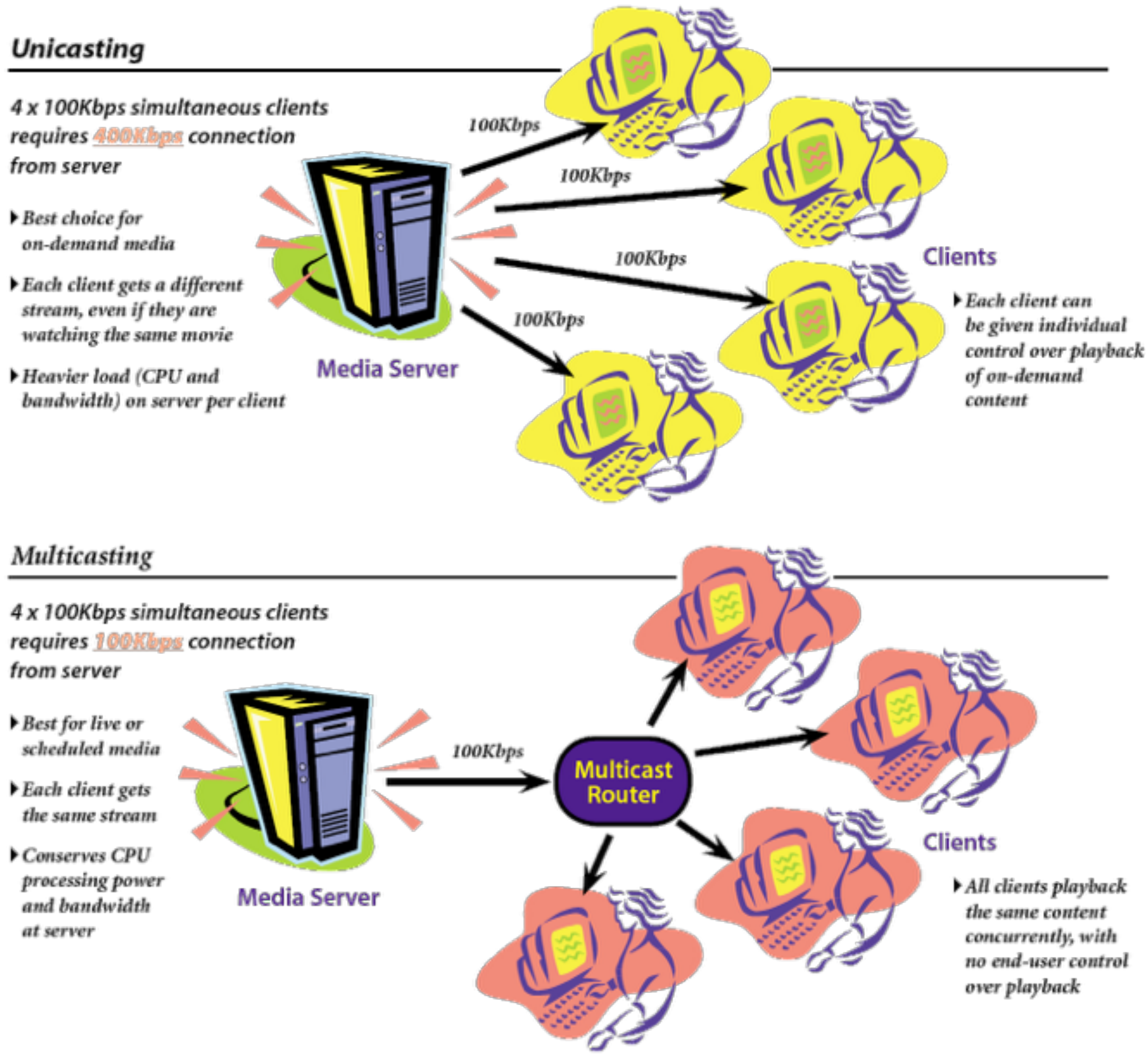
- **MPEG standard**
  - –Windows Media Video v1 is a derivative of the MPEG-4 codec, which has been renamed to avoid confusion.
  - –QuickTime 5 is the first full implementation of MPEG-4 for streaming media.

Streaming ...

- Progressive streaming transport (PST)
  - –use HTTP
  - –no jump

- Real-time streaming transport
  - –Real server (Real-time streaming protocol, RTSP)
  - –Windows Media server (M$ media server, MMS)
  - –Quicktime server

Unicast and Multicast

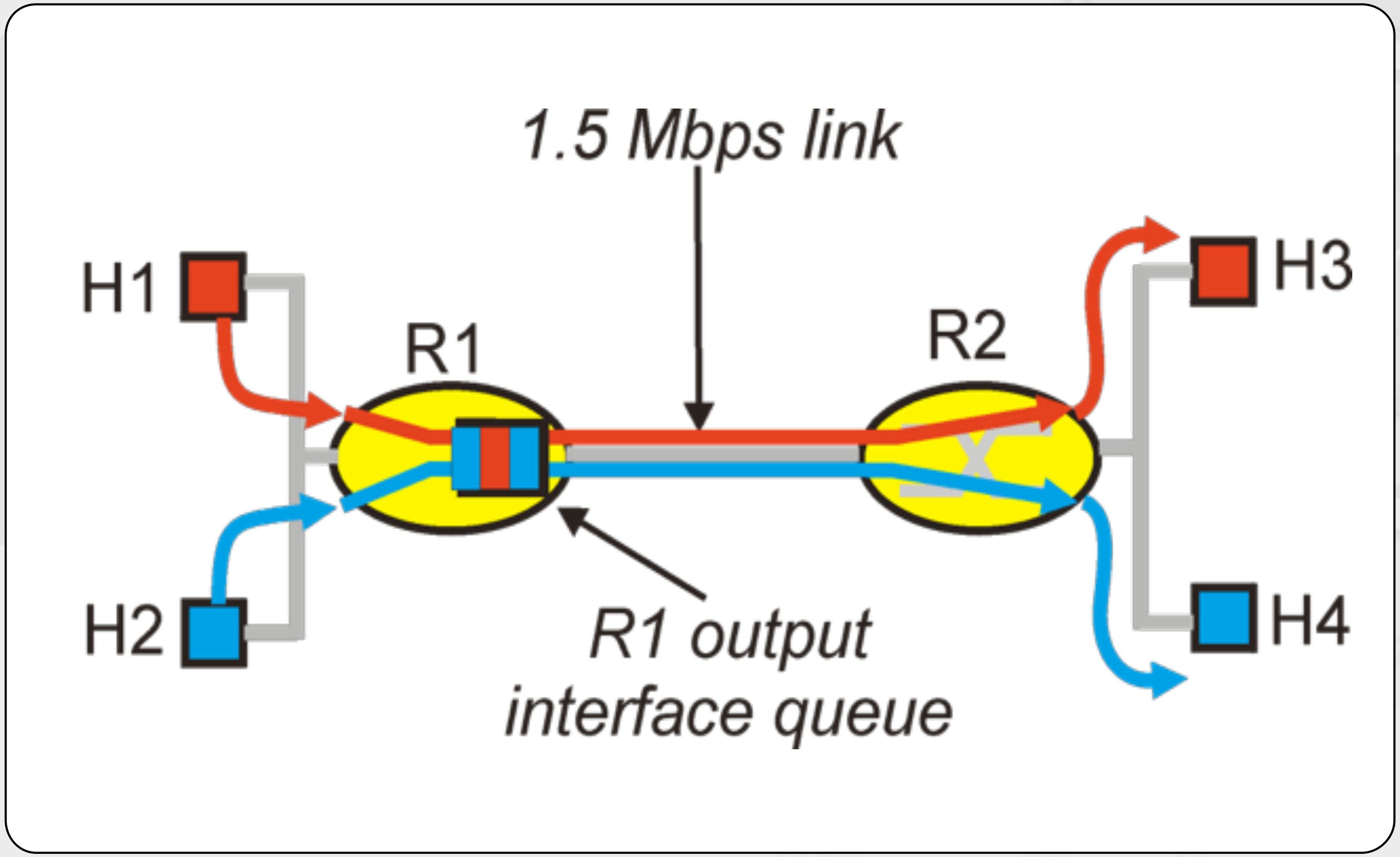


QoS: Quality of Service

- A defined measure of performance in a data communications system

- resource reservation control mechanisms
  - make the actual determination of which packets have priority
    1. provide different priority to different users or data flows, or
    2. guarantee a certain level of performance to a data flow in accordance with requests from the application program

QoS: a simple example

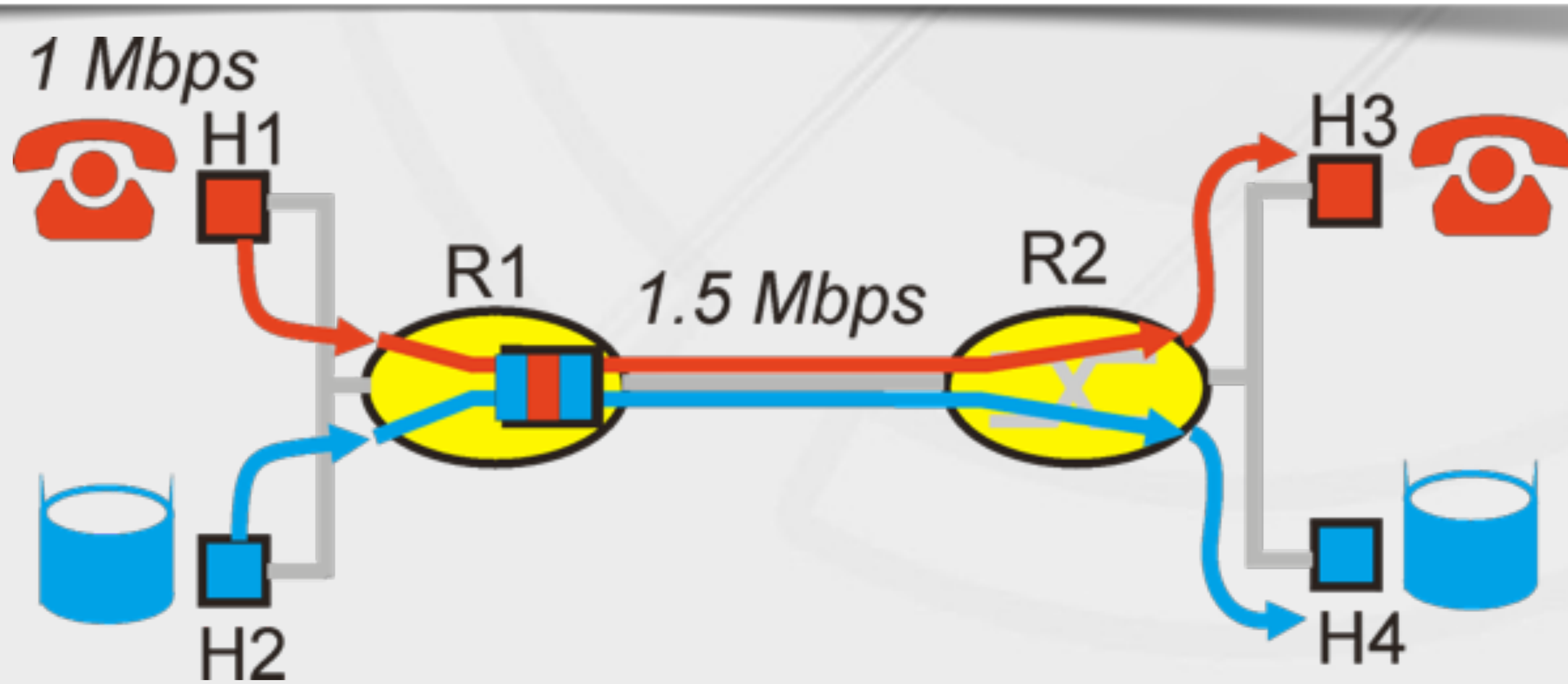


Improving QOS in IP Networks

Principles for QOS Guarantees

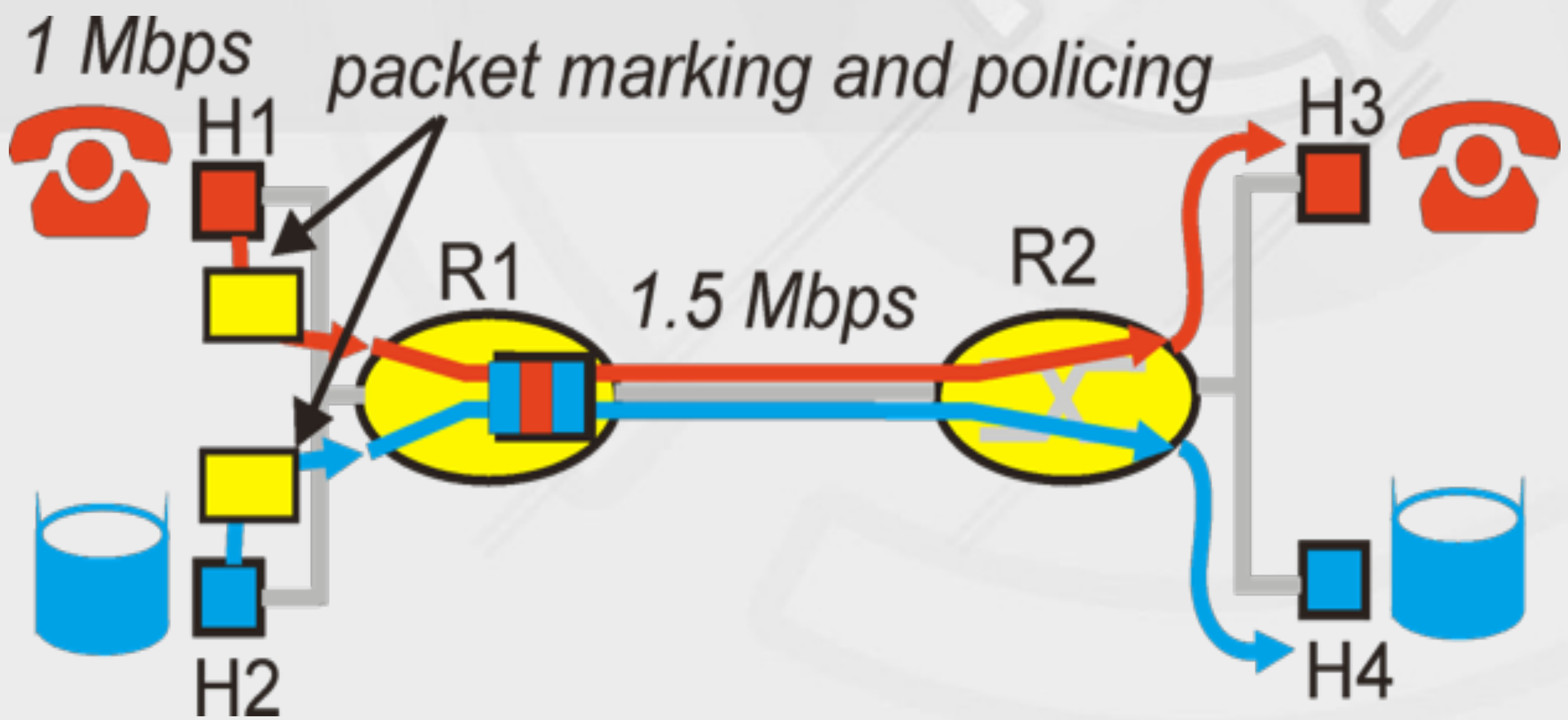
- **Consider:**
  - –a phone application at 1Mbps and
  - –an FTP application sharing a 1.5 Mbps link.
    - bursts of FTP can congest the router and cause audio packets to be dropped.
    - want to give priority to audio over FTP

- **PRINCIPLE 1: Marking of packets is needed for router to distinguish between different classes; and new router policy to treat packets accordingly**



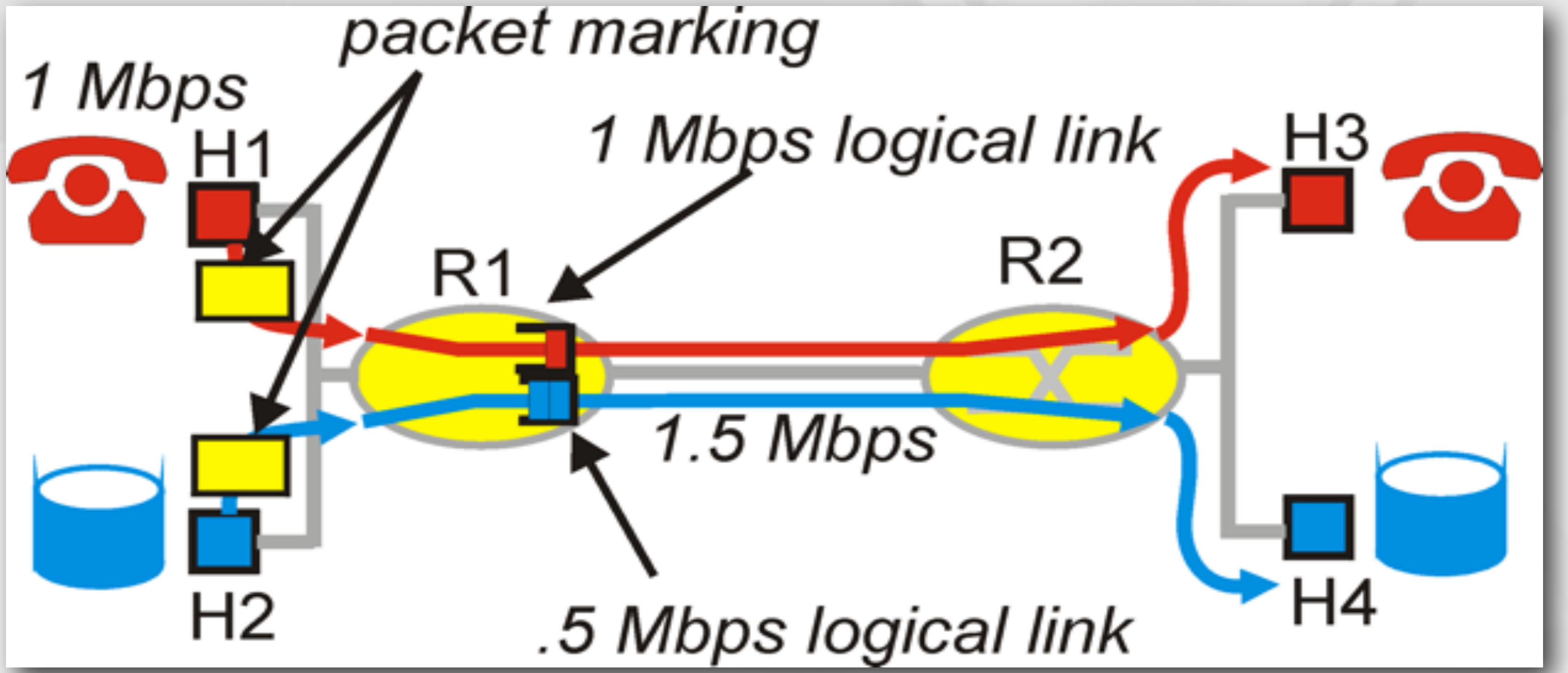
Principles for QOS Guarantees (more)

- Applications misbehave (audio sends packets at a rate higher than 1Mbps assumed above);
- **PRINCIPLE 2: provide protection (isolation) for one class from other classes**
- Require Policing Mechanisms to ensure sources adhere to bandwidth requirements; Marking and Policing need to be done at the edges:



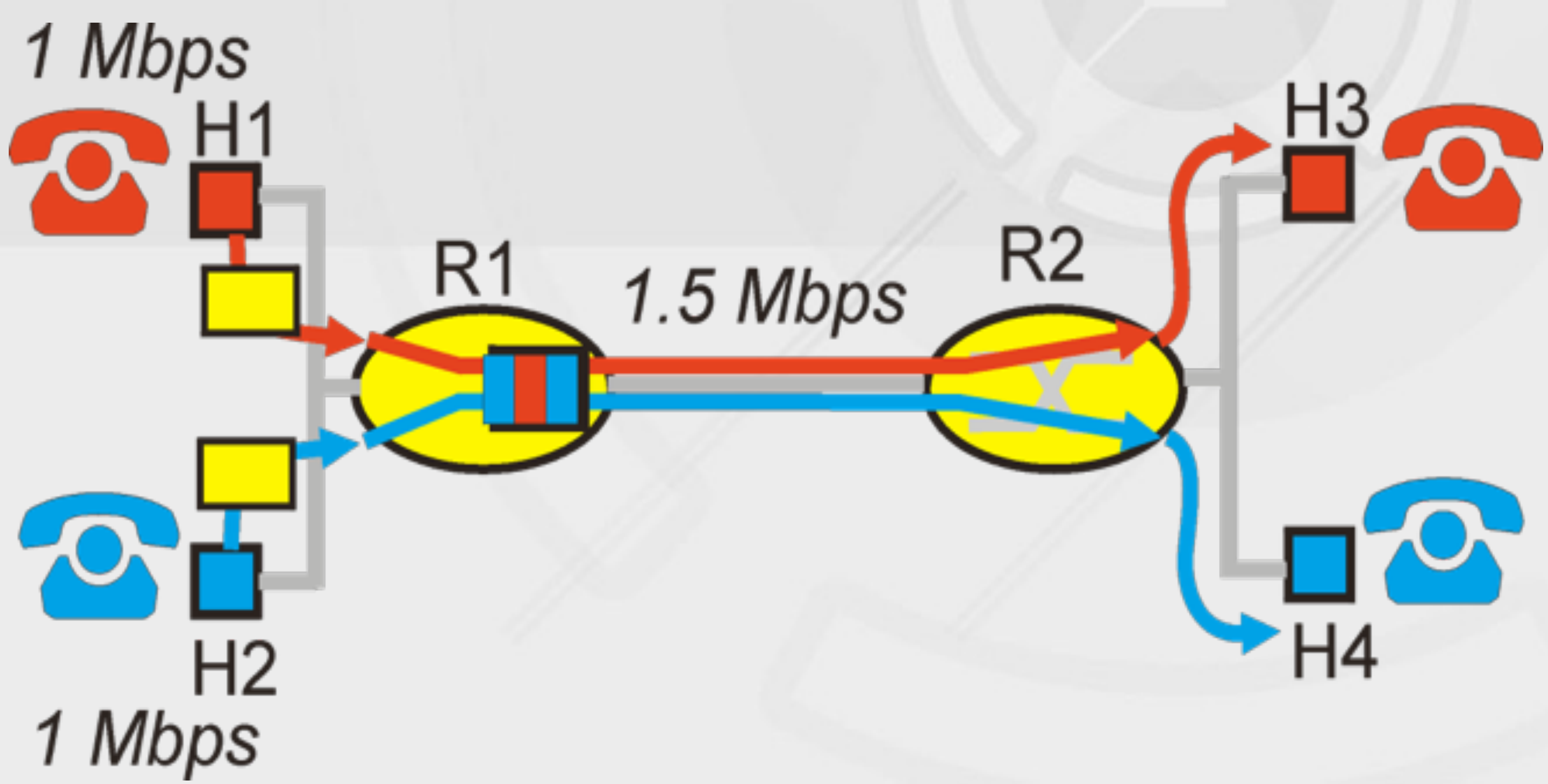
Principles for QOS Guarantees (more)

- Alternative to Marking and Policing: allocate a set portion of bandwidth to each application flow; can lead to inefficient use of bandwidth if one of the flows does not use its allocation
- **PRINCIPLE 3: While providing isolation, it is desirable to use resources as efficiently as possible**



Principles for QOS Guarantees (more)

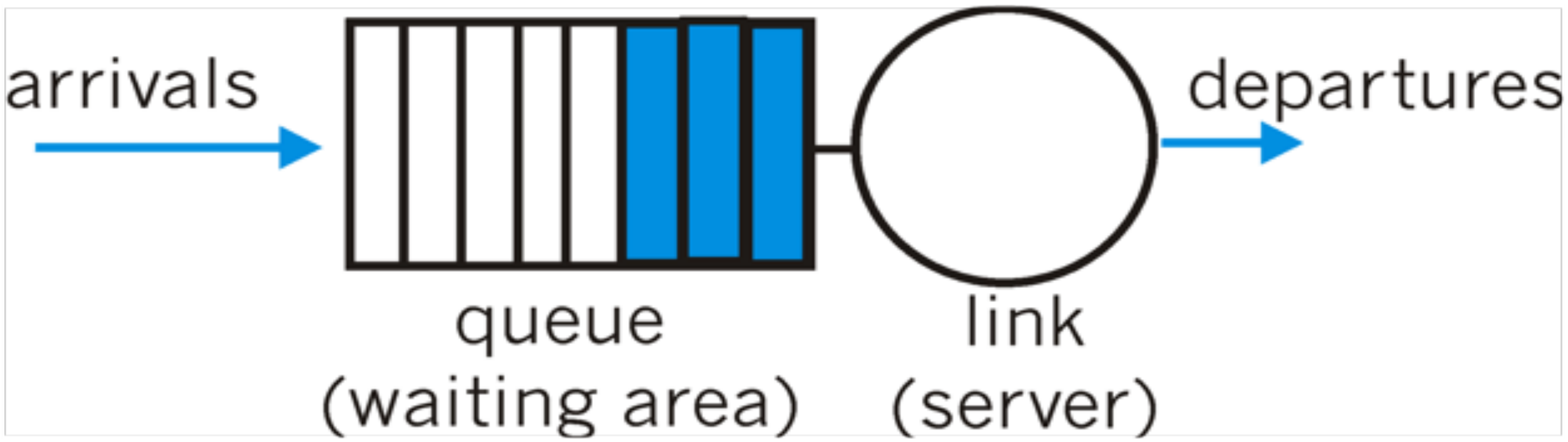
- Cannot support traffic beyond link capacity
- **PRINCIPLE 4: Need a Call Admission Process; application flow declares its needs, network may block call if it cannot satisfy the needs**



Scheduling And Policing Mechanisms

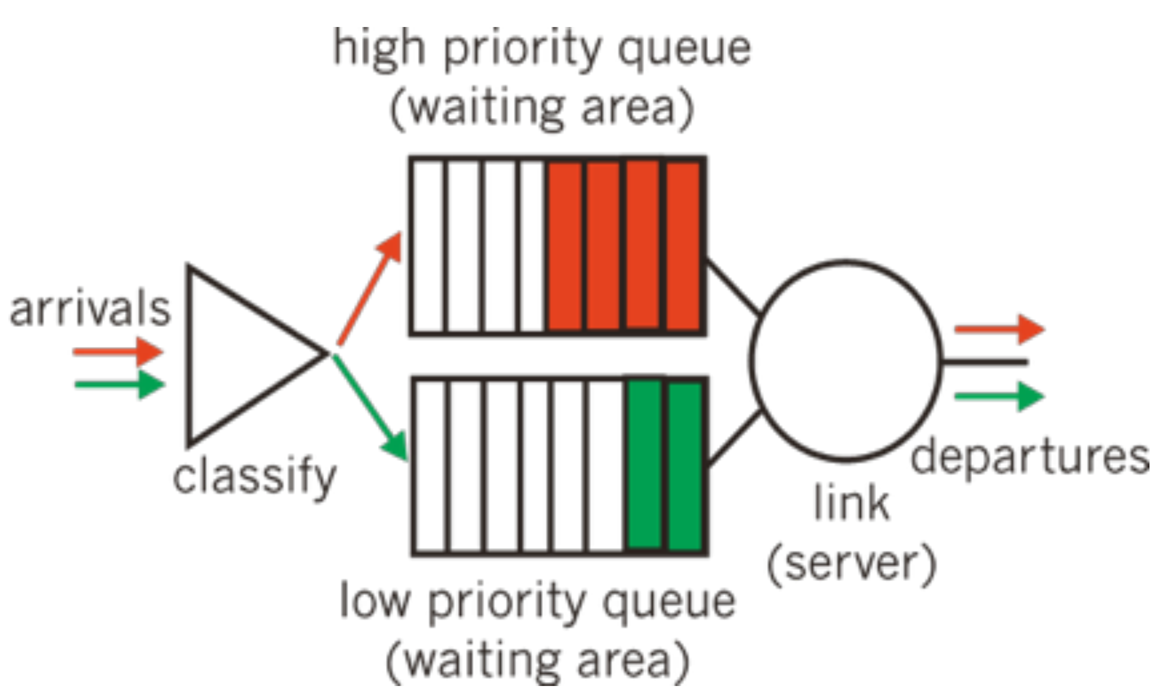
- **Scheduling:** choosing the next packet for transmission on a link can be done following a number of policies;

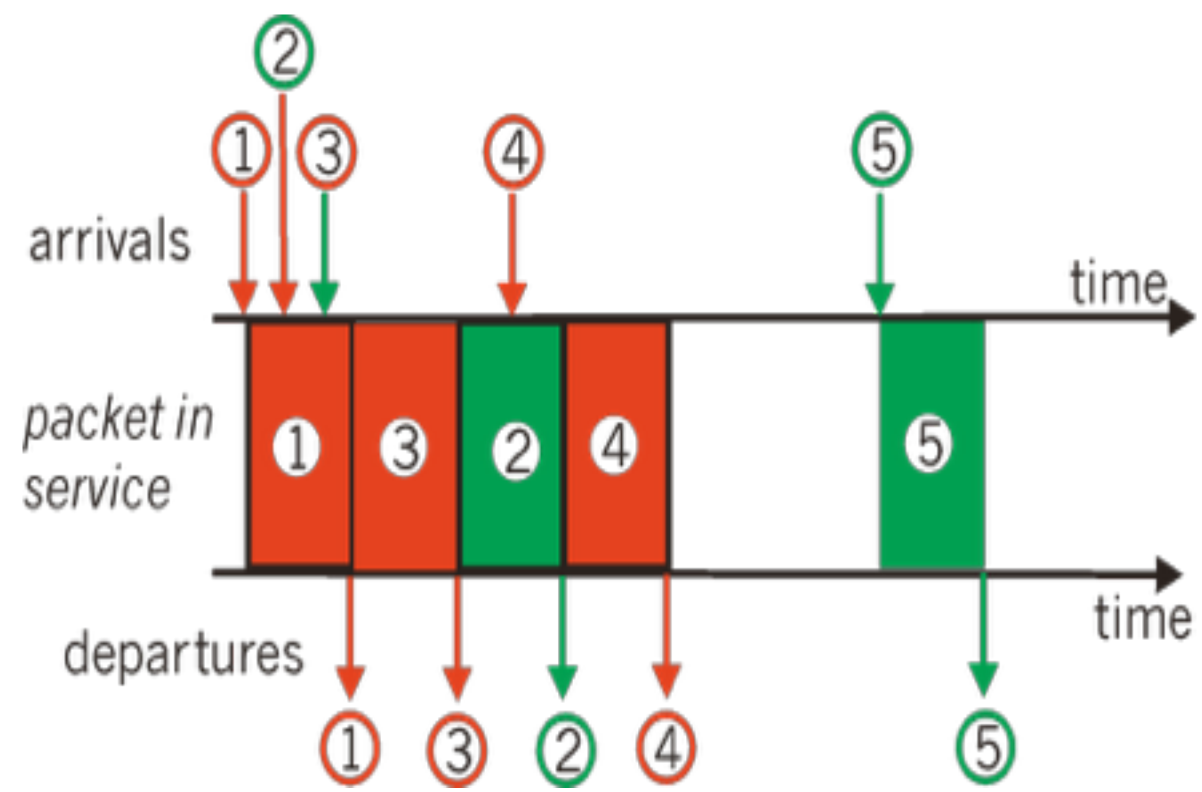
- **FIFO:** in order of arrival to the queue; packets that arrive to a full buffer are either discarded, or a discard policy is used to determine which packet to discard among the arrival and those already queued



Scheduling Policies

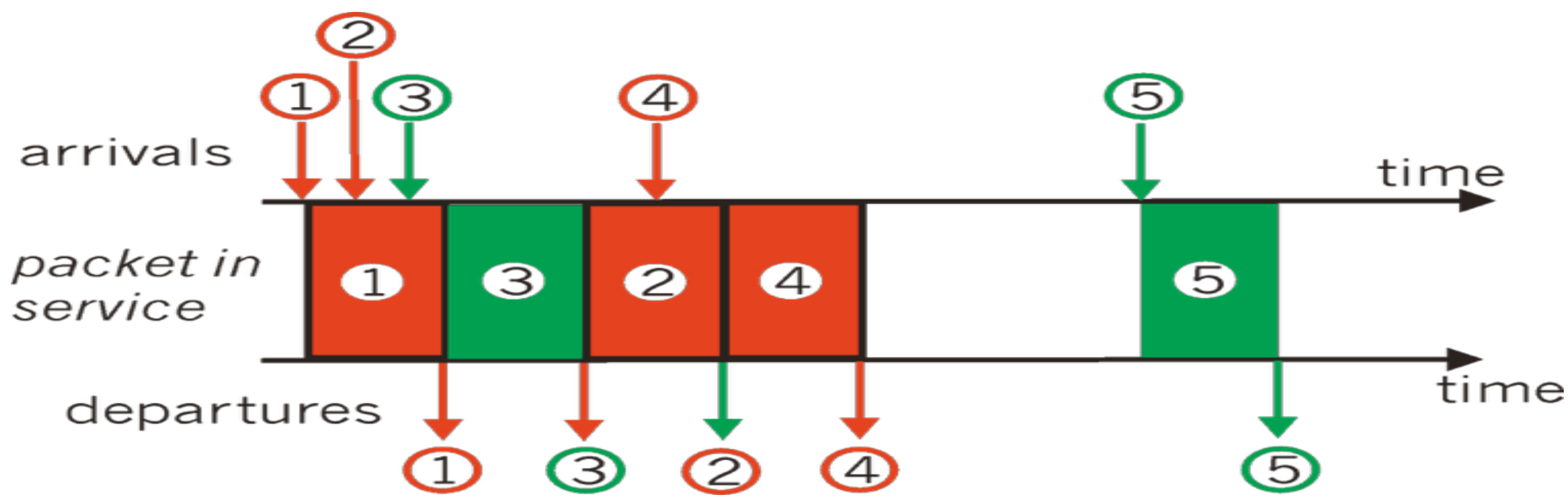
- Priority Queuing:
  classes have different priorities; class may depend on explicit marking or other header info, e.g. IP source or destination, TCP Port numbers, etc.
- Transmit a packet from the highest priority class with a non-empty queue
- Preemptive and non-preemptive versions





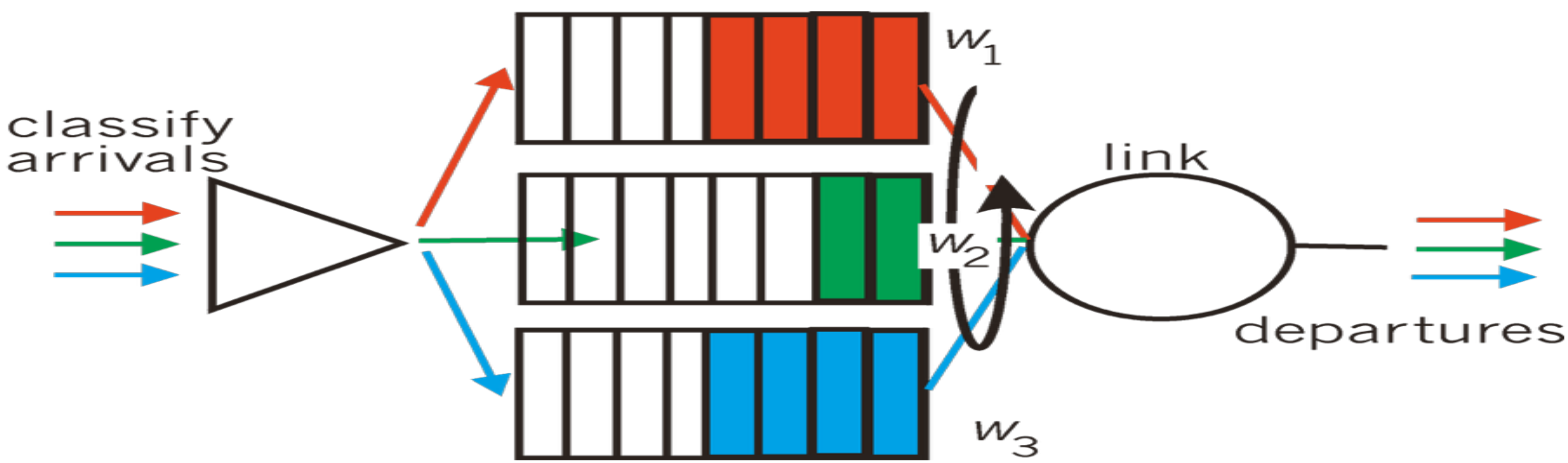
Scheduling Policies (more)

- Round Robin: scan class queues serving one from each class that has a non-empty queue



Scheduling Policies (more)

- Weighted Fair Queuing: is a generalized Round Robin in which an attempt is made to provide a class with a differentiated amount of service over a given period of time

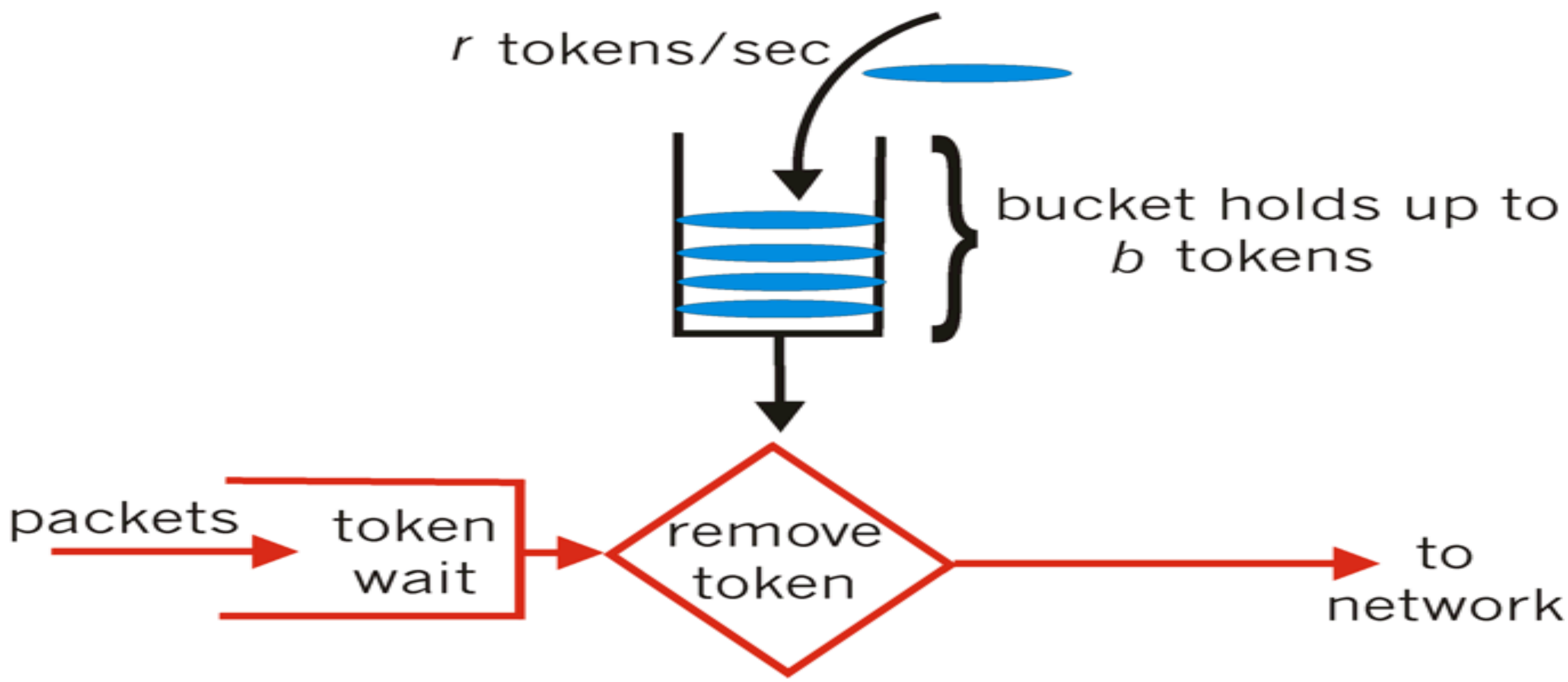


Policing Mechanisms

- Three criteria:
  - (Long term) **Average Rate** (100 packets/sec or 6000 packets/min), crucial aspect is the interval length
  - **Peak Rate**: e.g., 6000 p/min **Avg** and 1500 p/sec **Peak**
  - (Max.) **Burst Size**: Max. number of packets sent consecutively, i.e. over a short period of time

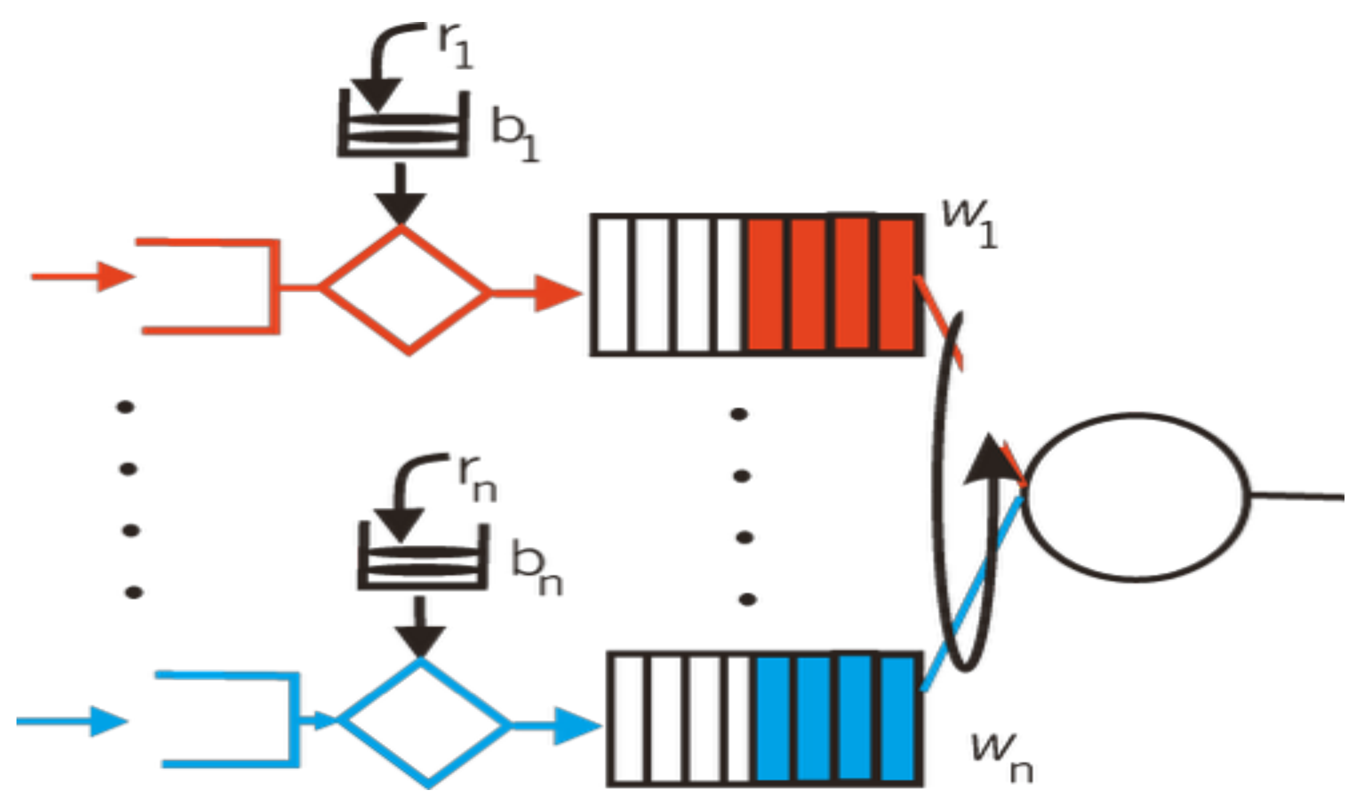
Policing Mechanisms

- **Token Bucket** mechanism, provides a means for limiting input to specified Burst Size and Average Rate.



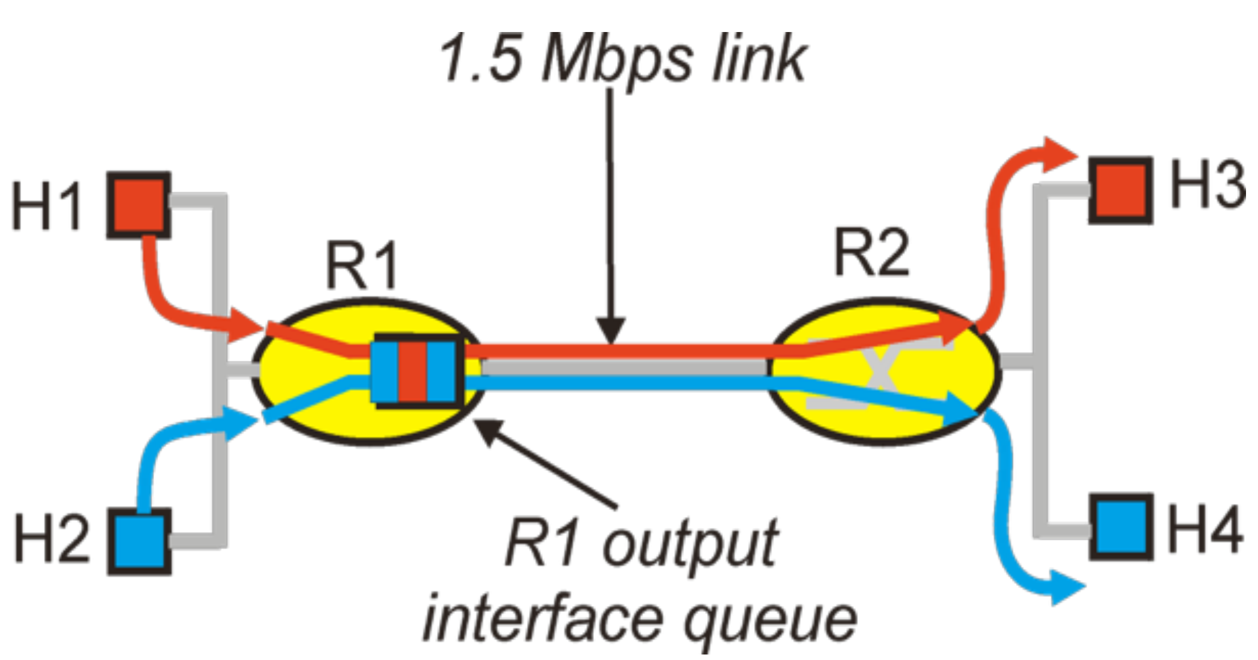
Policing Mechanisms (more)

- Bucket can hold b tokens; token are generated at a rate of r *token/sec* unless bucket is full of tokens.
- Over an interval of length t, the number of packets that are admitted is less than or equal to $(r\,t + b)$.
- Token bucket and WFQ can be combined to provide upper bound on delay.



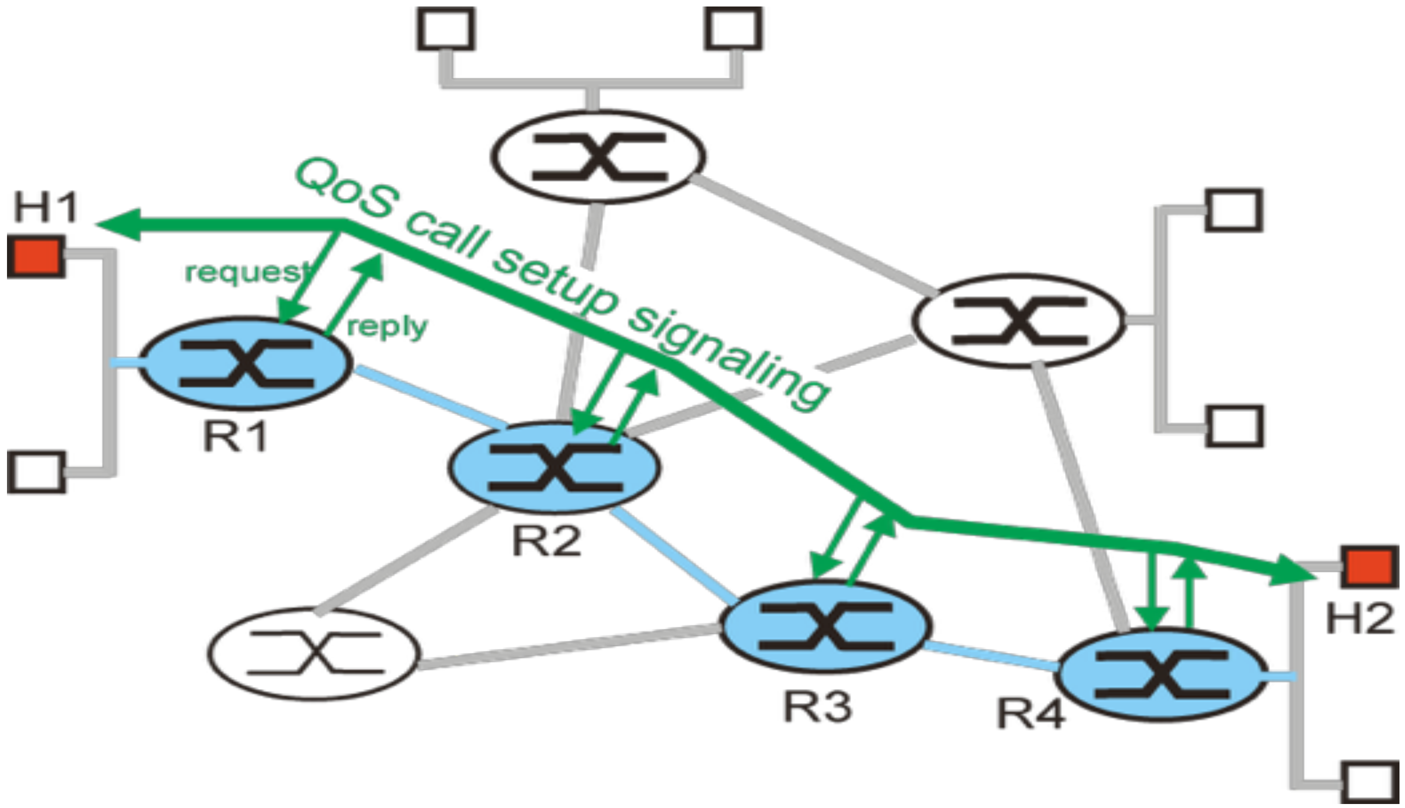
IETF IP QoS Efforts

- Policy based IP QoS Solutions
  - Integrated Services (RSVP protocol):
    - flow based
  - Differentiated Services (DiffServ byte settings):
    - packet based
  - Multi-Protocol Label Switching (MPLS):
    - flow+packet based
- IP Multicast and Anycast
- IPv6 QoS Support



Connection Oriented QoS

- **Integrated Services**: IETF RFC 1633
  - –Defined by RSVP requires resource reservation at each hop end-to-end for each IP packet flow, and end-to-end signaling along nodes in the path
  - –Reserve resources at the routers so as to provide QoS for specific user packet stream
  - –This architecture does not scale well (large amount of states)
  - –Many Internet flows are short lived, not worth setting up VC

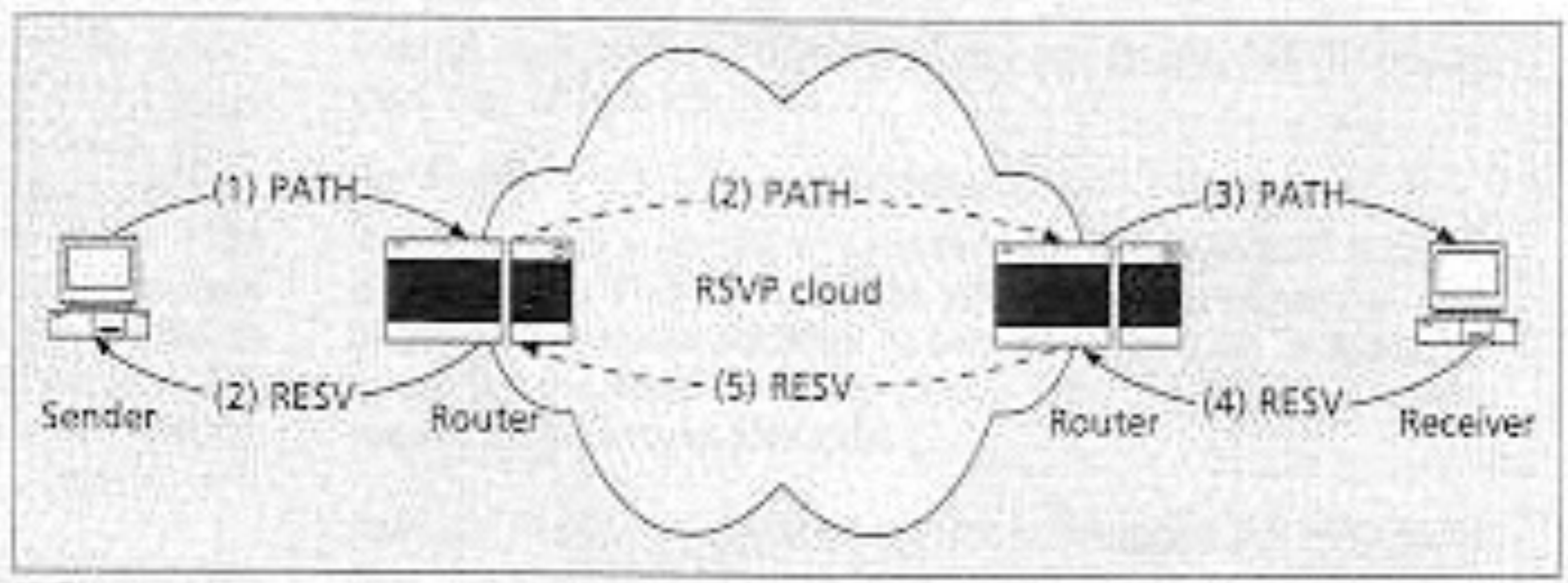


Integrated Services / RSVP (警车开道)

- Sender sends a "**PATH**" message to the receiver specifying characteristics of traffic
  - every intermediate router along the path forwards the "PATH" message to the next hop determined by the routing protocol

- Receiver responds with "RESV" message after receiving "PATH". "RESV" requests resources for flow

RSVP
=
Resource reservation protocol



浙江大学计算机学院
数字媒体与网络技术

Connectionless QoS: IP Diff Serv

- Mark IP packet to specify treatment: IETF RFC 2474,
  - e.g., first class, business class, coach, standby

通行证机制

- Per Hop Behaviors (PHBs) based on network-wide traffic classes
- Flows are classified at the edge router based on rules, and are aggregated into traffic classes, allowing scalability
- Diff Serv uses the **IP header TOS byte** (first 6 bits), which is renamed the DS field
- Diff Serv defines code points (DSCP) for the DS field, DE (default) = 000000 = best effort, and EF (Expedited Forwarding) = 101110 = low latency, etc.

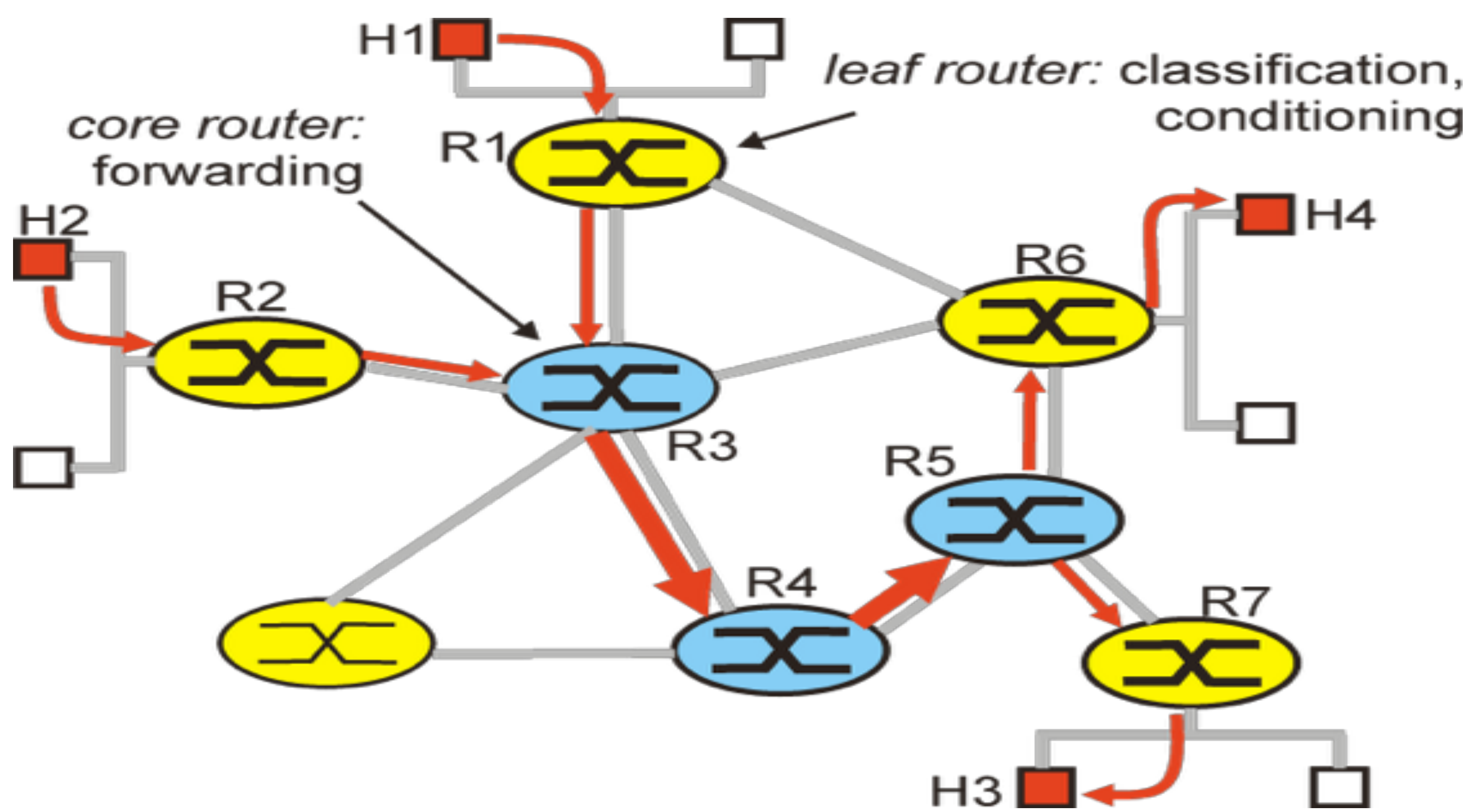
Differentiated Services

- Approach:
  - Only simple functions in the core, and relatively complex functions at edge routers (or hosts)
  - Do not define service classes, instead provides functional components with which service classes can be built

Edge Functions

- At DS-capable host or first DS-capable router

- **Classification**: edge node marks packets according to classification rules to be specified (manually by admin, or by some TBD protocol)
- **Traffic Conditioning**: edge node may delay and then forward or may discard



Core Functions

- **Forwarding**: according to “Per-Hop-Behavior” or PHB specified for the particular packet class; such PHB is strictly based on class marking (no other header fields can be used to influence PHB)

- BIG ADVANTAGE:

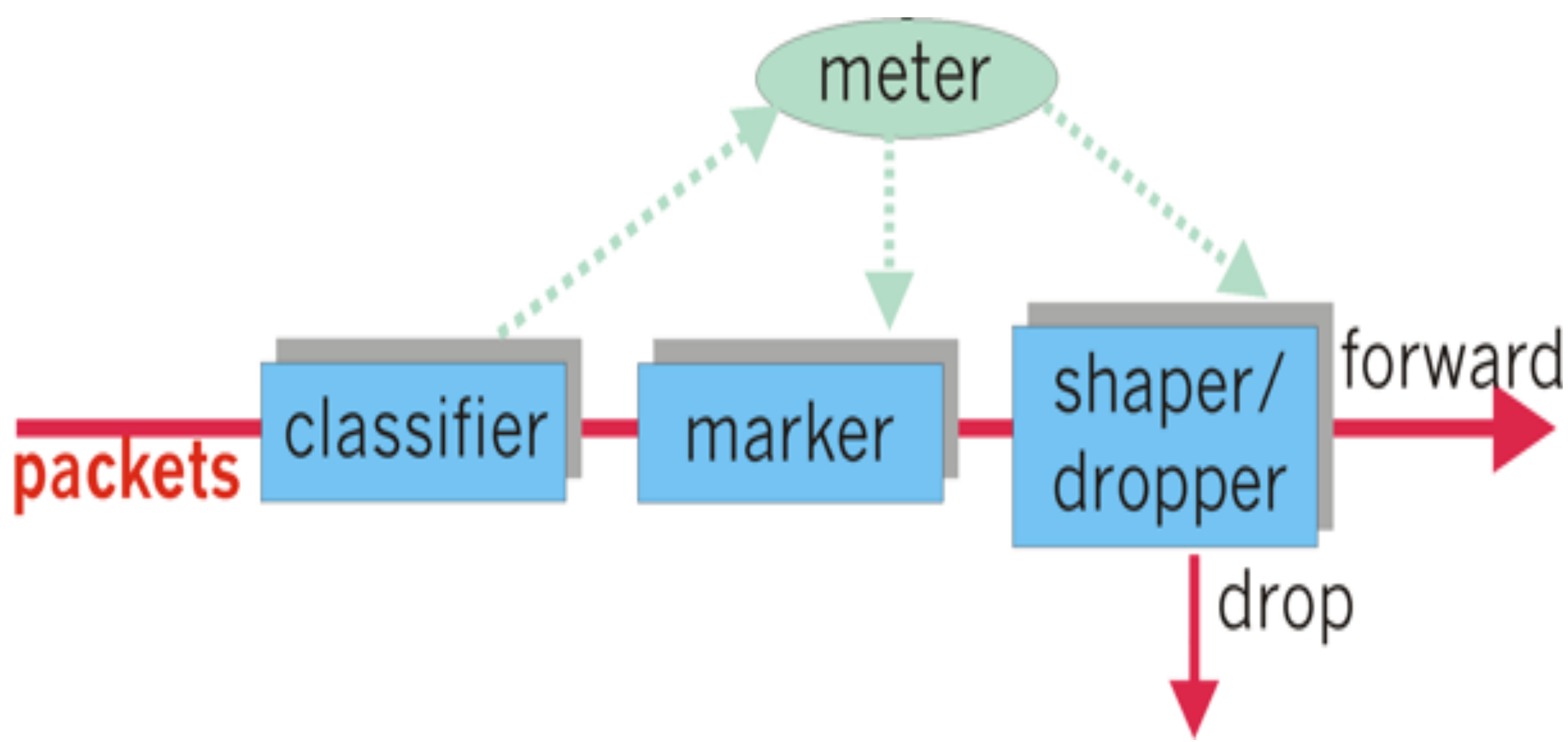
  No state info to be maintained by routers!

Classification and Conditioning

- Packet is marked in the Type of Service (TOS) in IPv4, and Traffic Class in IPv6
- 6 bits used for Differentiated Service Code Point (DSCP) and determine PHB that the packet will receive
- 2 bits are currently unused

Classification and Conditioning

- It may be desirable to limit traffic injection rate of some class; user declares traffic profile (e.g., rate and burst size);
- traffic is metered and shaped if non-conforming



QoS Priority Levels

Priority Level	Traffic Type
0	Best Effort
1	Background
2	Standard (Spare)
3	Excellent Load (Business Critical)
4	Controlled Load (Streaming Multimedia)
5	Voice and Video (Interactive Media and Voice) [Less than 100ms latency and jitter]
6	Layer 3 Network Control Reserved Traffic [Less than 10ms latency and jitter]
7	Layer 2 Network Control Reserved Traffic [Lowest latency and jitter]

IPv6 Support of QoS

- IPv6 Flow Labels provide support for Data Flows
  - Packet Prioritizing
    - sure that high priority traffic is not interrupted by less critical data

- IPv6 supports Multicast & Anycast
  - Multicast delivers data simultaneously to all hosts that sign up to receive it
  - Anycast allows one host initiate the efficient updating of routing tables for a group of hosts.

How should the Internet evolve to better support multimedia?

Integrated services philosophy:

- Change Internet protocols so that applications can reserve end-to-end bandwidth
  - Need to deploy protocol that reserves bandwidth
  - Must modify scheduling policies in routers to honor reservations
  - Application must provide the network with a description of its traffic, and must further abide to this description.
- Requires new, complex software in hosts & routers

Differentiated services philosophy:

- Fewer changes to Internet infrastructure, yet provide 1st and 2nd class service.
- Datagrams are marked.
- User pays more to send/receive 1st class packets.
- ISPs pay more to backbones to send/receive 1st class packets.

How should the Internet evolve to better support multimedia? (cont.)

Laissez-faire philosophy

- No reservations, no datagram marking
- As demand increases, provision more bandwidth
- Place stored content at edge of network:
  - ISPs & backbones add caches
  - Content providers put content in CDN nodes
  - P2P: choose nearby peer with content

Virtual private networks (VPNs)

- Reserve permanent blocks of bandwidth for enterprises.
- Routers distinguish VPN traffic using IP addresses
- Routers use special scheduling policies to provide reserved bandwidth.

Streaming Stored Audio & Video

Streaming stored media:

- Audio/video file is stored in a server
- Users request audio/video file on demand.
- Audio/video is rendered within, say, 10 s after request.
- Interactivity (pause, re-positioning, etc.) is allowed.

Media player:

- removes jitter
- decompresses
- error correction
- graphical user interface with controls for interactivity

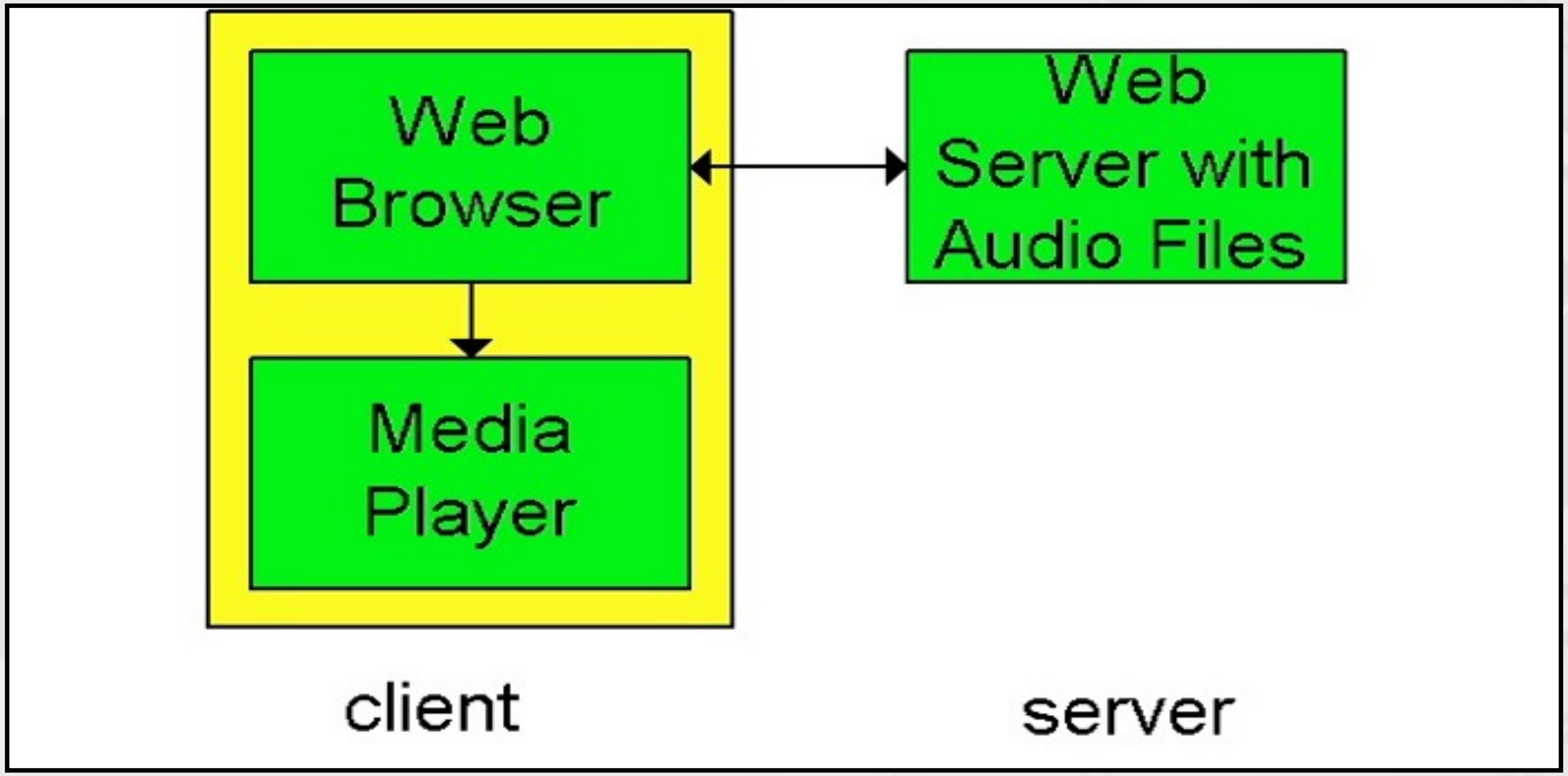
- Plug-ins may be used to imbed the media player into the browser window

Streaming ...

- Progressive streaming transport (PST)
  - –use HTTP
  - –no jump

- Real-time streaming transport
  - –Real server (Real-time streaming protocol, RTSP)
  - –Windows Media server (M$ media server, MMS)
  - –Quicktime server

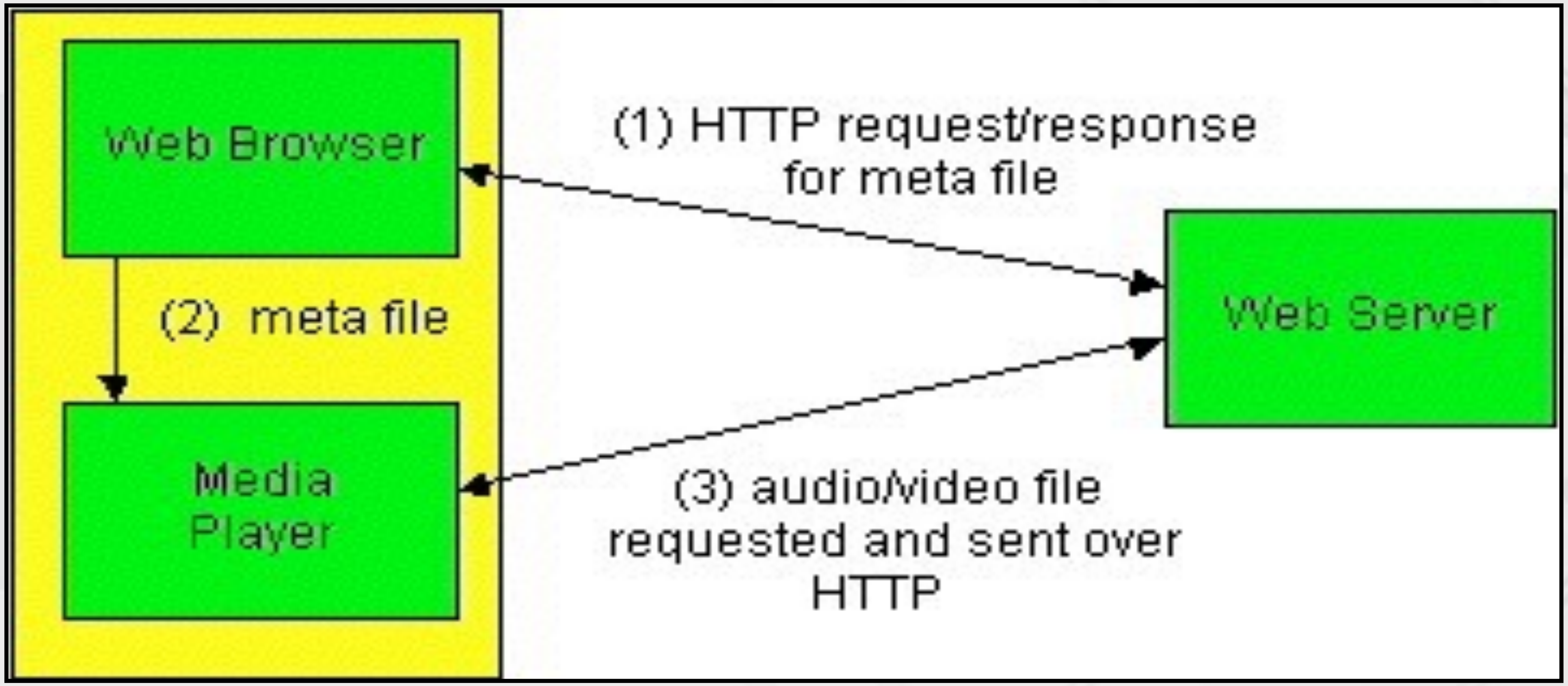
Streaming from Web server (1)



Major drawback:

media player interacts with server through intermediary of a Web browser

Streaming from Web server (2)

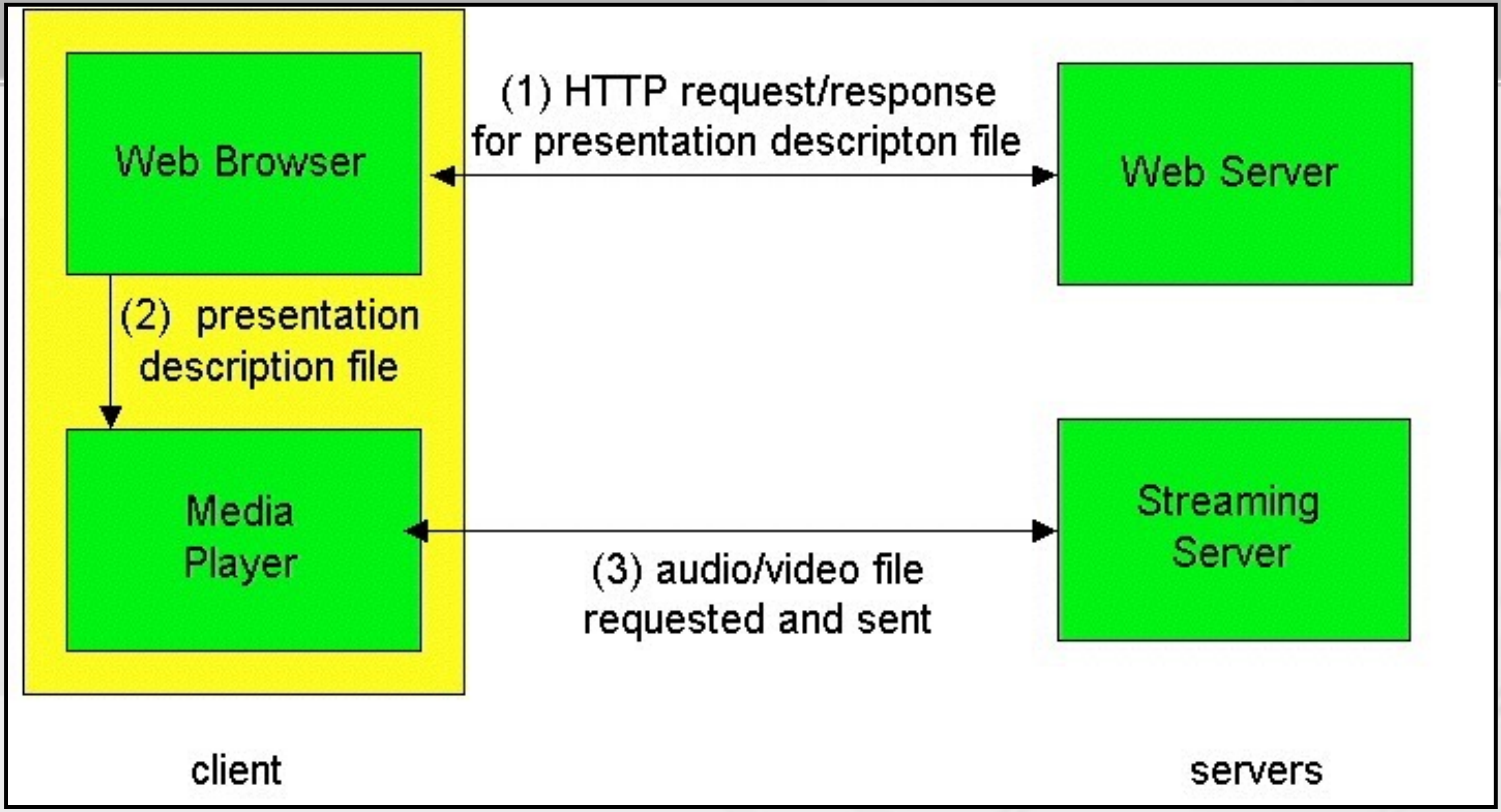


Some concerns:

•Media player communicates over HTTP, which is not designed with pause, ff, rwnd commands

•May want to stream over UDP

Streaming from a streaming server



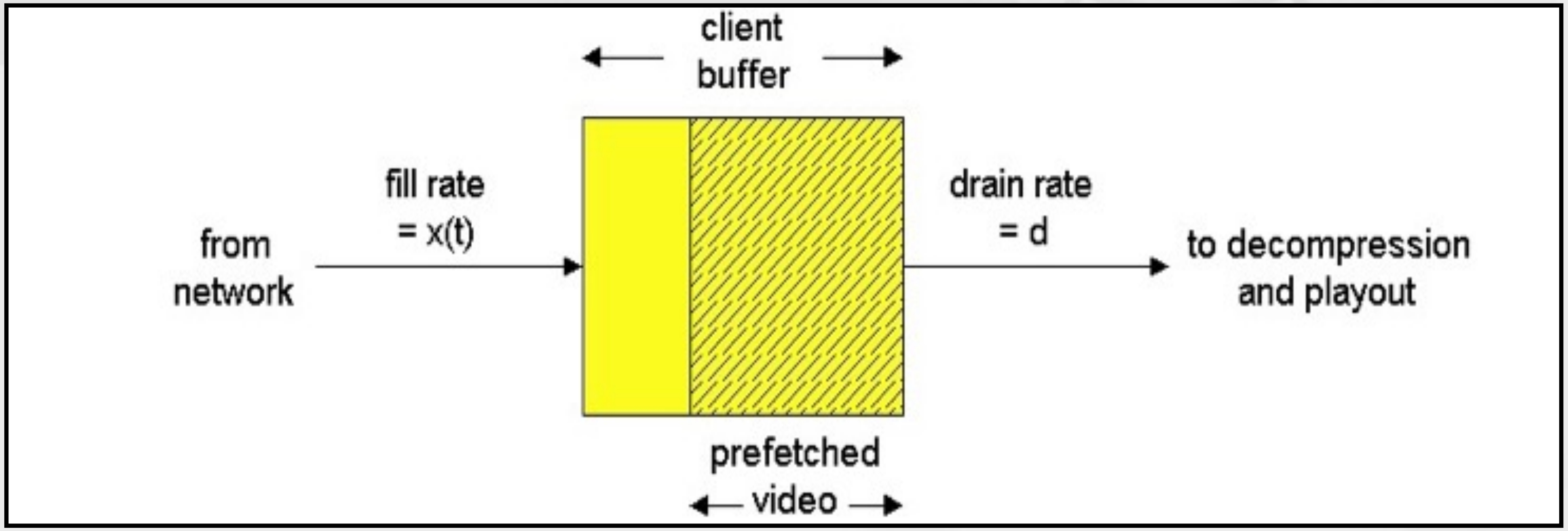
This architecture allows for non-HTTP protocol between server and media player

Can also use UDP instead of TCP.

Options when using a streaming server

- Send at constant rate over **UDP**. To mitigate the effects of jitter, buffer and delay playback for 1-10 s. Transmit rate = d, the encoded rate. Fill rate x(t) equals d except when there is loss.

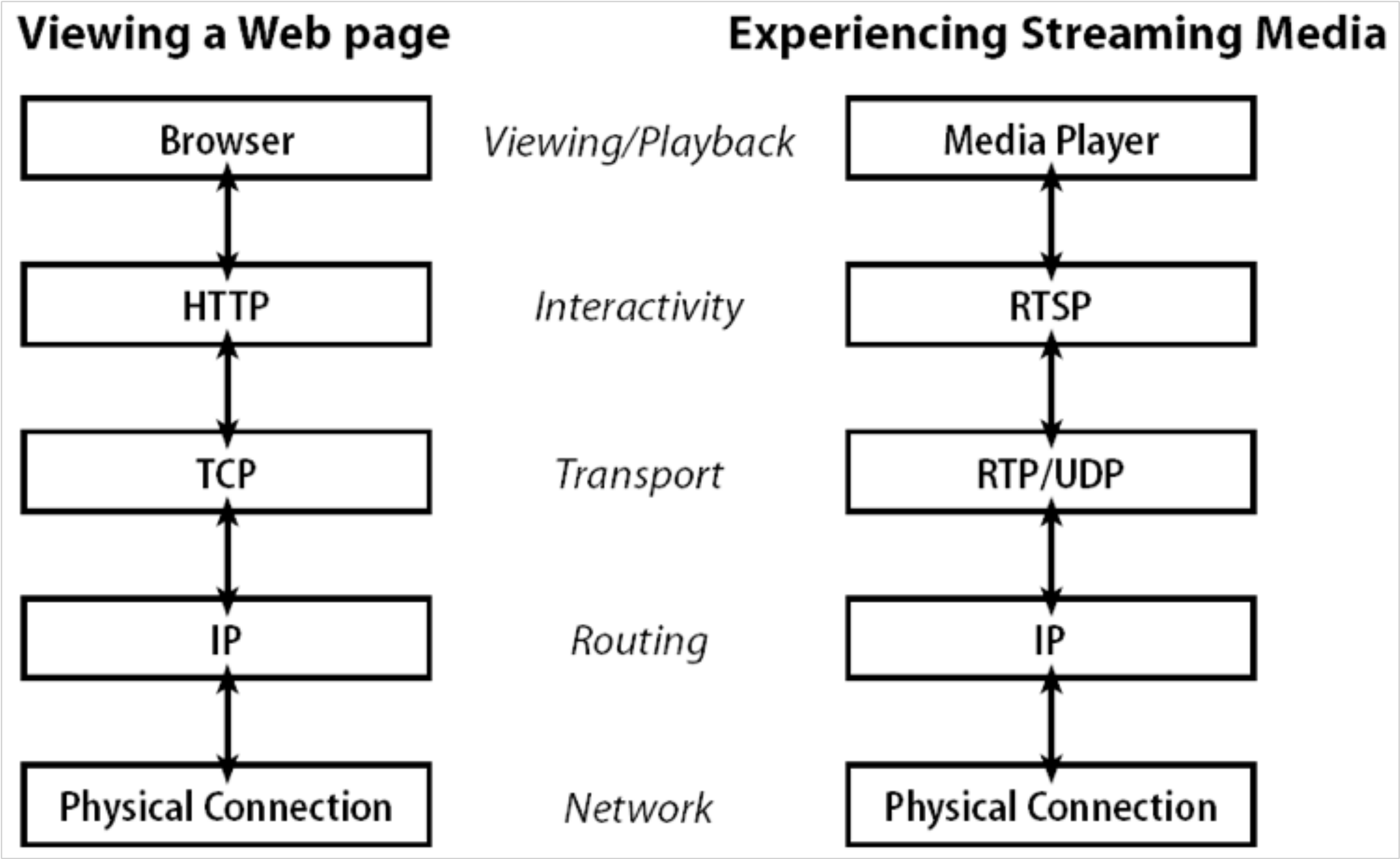
- Use TCP, and send at maximum possible rate under TCP; TCP retransmits when error is encountered; x(t) now fluctuates, and can become much larger than d. Player can use a much large buffer to smooth delivery rate of TCP.



Streaming Protocols

- **RTP (Real-time Transport Protocol)**
  - layered on top of **UDP**
  - RTP enables a one-way stream, transmitting media from the server to the client
- **RTSP (Real-time Streaming Protocol)**
  - similar to **HTTP** and to **FTP**—protocols used by Web servers
  - However, **HTTP** and **FTP** cannot be used for true streaming
    - both layered on top of **TCP**
  - a two-way protocol which uses TCP to communicate, and
  - usually layered on top of RTP
- **RTCP (Real-time Transport Control Protocol)**

Streaming Protocols



Real Time Streaming Protocol: RTSP

HTTP

- Designers of HTTP had fixed media in mind: HTML, images, applets, etc.
- HTTP does not target stored continuous media (i.e., audio, video, SMIL presentations, etc.)

RTSP: RFC 2326

- Client-server application layer protocol.
- For user to control display: rewind, fast forward, pause, resume, repositioning, etc…

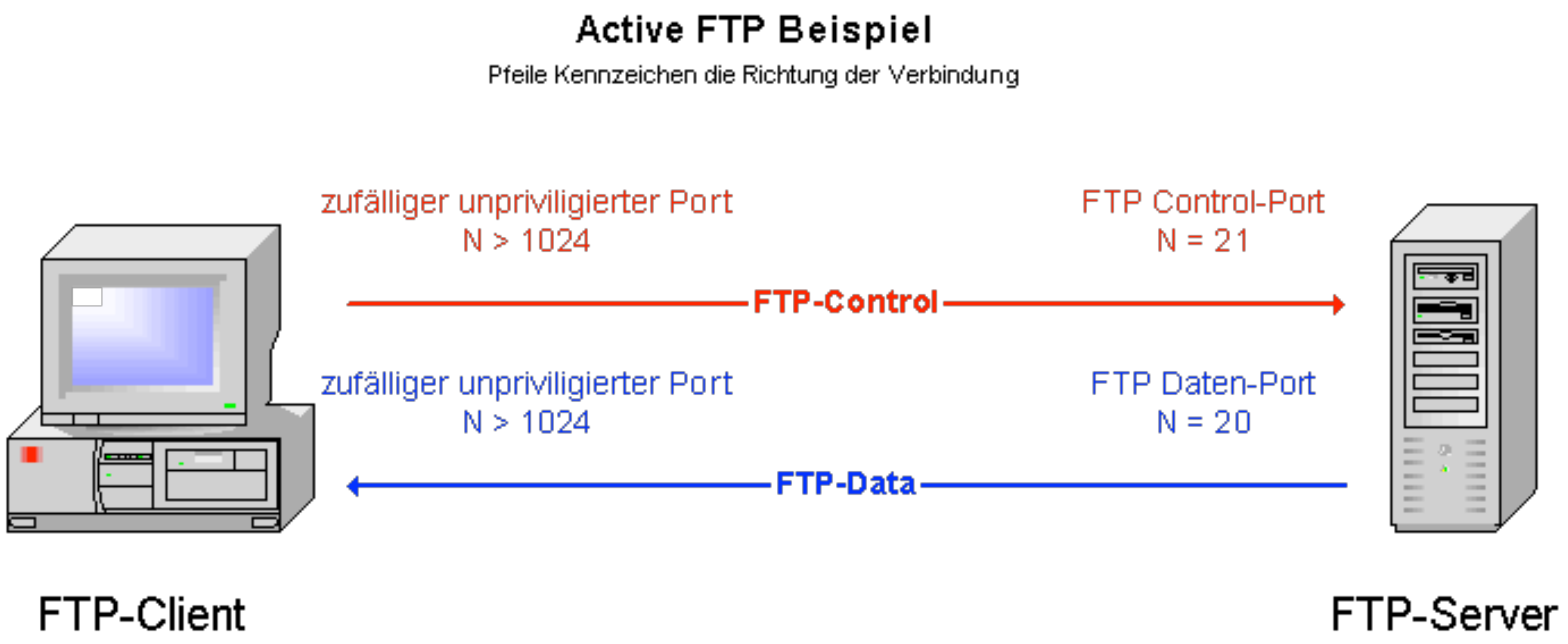
What it doesn't do:

- does not define how audio/video is encapsulated for streaming over network
- does not restrict how streamed media is transported; it can be transported over UDP or TCP
- does not specify how the media player buffers audio/video

RealNetworks

- Server and player use RTSP to send control info to each other

FTP?


Active FTP Beispiel
Pfeile Kennzeichen die Richtung der Verbindung
zufälliger unpriviligierter Port
N > 1024
FTP Control-Port
N = 21
FTP-Control
zufälliger unpriviligierter Port
N > 1024
FTP Daten-Port
N = 20
FTP-Data
FTP-Client
FTP-Server

RTSP: out of band control

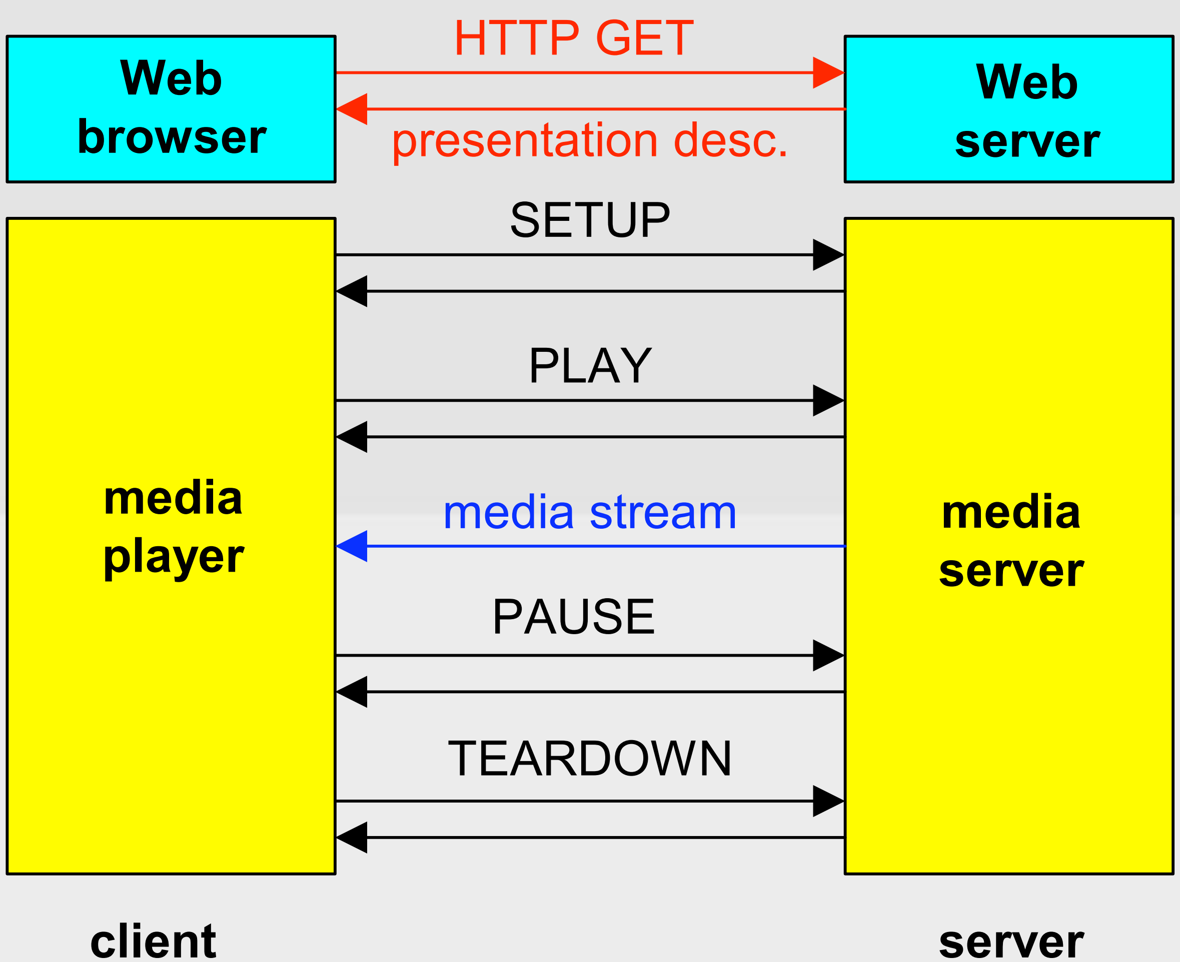
FTP uses an "out-of-band" control channel:

- A file is transferred over one channel.
- Control information (directory changes, file deletion, file renaming, etc.) is sent over a separate TCP connection.
- The "out-of-band" and "in-band" channels use different port numbers.

RTSP messages are also sent out-of-band:

- The RTSP control messages use different port numbers than the media stream, and are therefore sent out-of-band.
- The media stream, whose packet structure is not defined by RTSP, is considered "in-band".
- If the RTSP messages were to use the same port numbers as the media stream, then RTSP messages would be said to be "interleaved" with the media stream.

RTSP initiates and controls delivery



- Client obtains a description of the multimedia presentation, which can consist of several media streams.
- The browser invokes media player (helper application) based on the content type of the presentation description.
- Presentation description includes references to media streams, using the URL method **rtsp://**
- Player sends **RTSP SETUP** request; server sends RTSP SETUP response.
- Player sends **RTSP PLAY** request; server sends RTSP PLAY response.
- Media server pumps media stream.
- Player sends **RTSP PAUSE** request; server sends RTSP PAUSE response.
- Player sends **RTSP TEARDOWN** request; server sends **RTSP TEARDOWN** response.

RTSP session

- Each RTSP has a session identifier, which is chosen by the server.
- The client initiates the session with the SETUP request, and the server responds to the request with an identifier.
- The client repeats the session identifier for each request, until the client closes the session with the TEARDOWN request.
- RTSP port number is 554.
- RTSP can be sent over UDP or TCP. Each RTSP message can be sent over a separate TCP connection.

RTSP: interaction example

- C: SETUP rtsp://audio.example.com/twister/audio RTSP/1.0
- Transport: rtp/udp; compression; port=3056; mode=PLAY
- S: RTSP/1.0 200 1 OK
- Session 4231

- C: PLAY rtsp://audio.example.com/twister/audio.en/lofi RTSP/1.0
- Session: 4231
- Range: npt=0-

- C: PAUSE rtsp://audio.example.com/twister/audio.en/lofi RTSP/1.0
- Session: 4231
- Range: npt=37

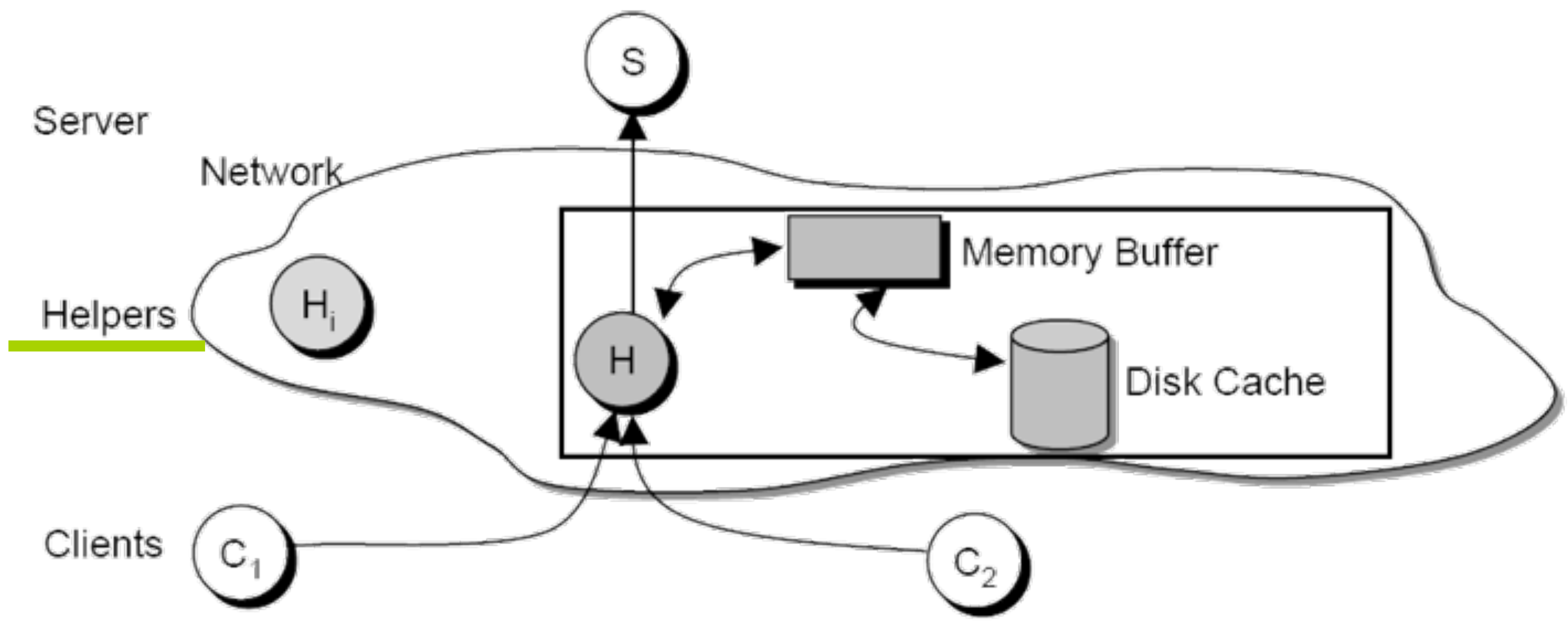
- C: TEARDOWN rtsp://audio.example.com/twister/audio.en/lofi RTSP/1.0
- Session: 4231
- S: 200 3 OK

RTSP: streaming caching

- Caching of RTSP response messages makes little sense.
- But desirable to cache media streams closer to client.

- Much of HTTP/1.1 cache control has been adopted by RTSP.
  - Cache control headers can be put in RTSP SETUP requests and responses:
    - If-modified-since: , Expires: , Via: , Cache-Control:

Design and Implementation of a Caching System for Streaming Media



Application layer aware helper in the network

RTSP: streaming caching

- **Proxy cache** may
  - hold only segments of a given media stream.
  - start serving a client from its local cache, and then have to connect to origin server and fill missing material, hopefully without introducing gaps at client.

- When origin server is sending a stream through client, and stream passes through a proxy, proxy can use TCP to obtain the stream; but proxy still sends RTSP control messages to origin server.

Real-time interactive applications

- **PC-2-PC phone**
- PC-2-phone
  - –Dialpad
  - –Net2phone
- videoconference
- Webcams

Internet phone over best-effort (1)

Best effort

- packet delay, loss and jitter

Internet phone example

- now examine how packet delay, loss and jitter are often handled in the context of an IP phone example.
- Internet phone applications generate packets during talk spurts
- bit rate is 64 kbps during talk spurt

- during talk spurt, every 20 msec app generates a chunk of 160 bytes = 8 kbytes/sec * 20 msec
- header is added to chunk; then chunk+header is encapsulated into a UDP packet and sent out
- some packets can be lost and packet delay will fluctuate.
- receiver must determine when to playback a chunk, and determine what do with missing chunk

Internet phone (2)

packet loss

- UDP segment is encapsulated in IP datagram
- datagram may overflow a router queue
- TCP can eliminate loss, but
  - retransmissions add delay
  - TCP congestion control limits transmission rate
- Redundant packets can help

end-to-end delay

- accumulation of transmission, propagation, and queuing delays
- more than 400 msec of end-to-end delay seriously hinders interactivity; the smaller the better

Internet phone (2)

delay jitter

- consider two consecutive packets in talk spurt
- initial spacing is 20 msec, but spacing at receiver can be more or less than 20 msec

removing jitter

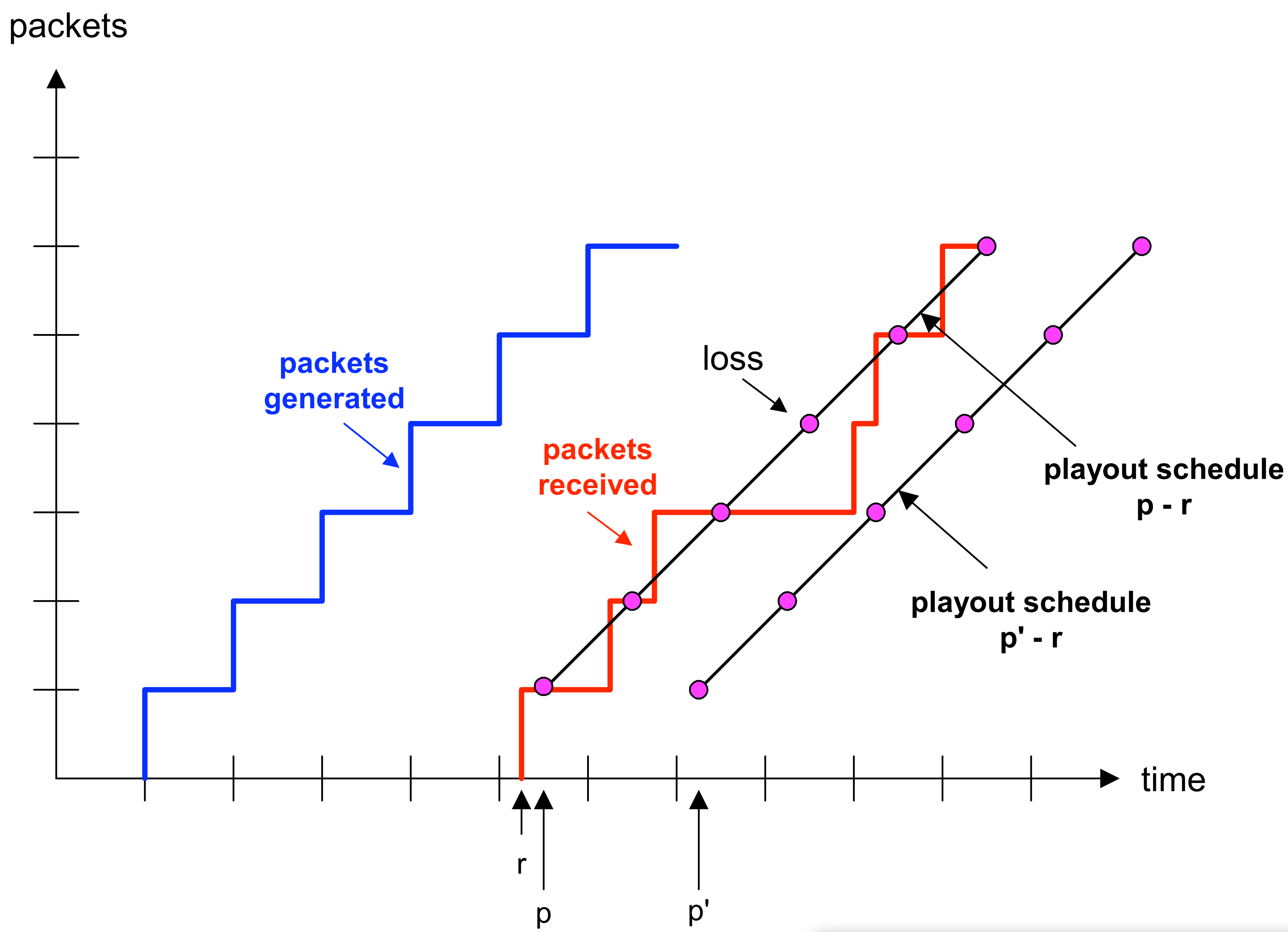
- sequence numbers
- timestamps
- delaying playout

Internet phone (3): fixed playout delay

- Receiver attempts to playout each chunk at exactly *q* msecs after the chunk is generated.
  - If chunk is time stamped *t*, receiver plays out chunk at *t+q*.
  - If chunk arrives after time *t+q*, receiver discards it.
- Sequence numbers are not necessary.
- Strategy allows for lost packets.

- Tradeoff for q:
  - larger q: less packet loss
  - smaller q: better interactive experience

Internet phone (4): fixed playout delay



- Sender generates packets every 20 msec during talk spurt.
- First packet received at time r
- First playout schedule: begins at p
- Second playout schedule: begins at p’

Adaptive playout delay (1)

- Estimate network delay and adjust playout delay at the beginning of each talk spurt.
- Silent periods are compressed and elongated.
- Chunks still played out every 20 msec during talk spurt.

t_i = timestamp of the ith packet

r_i = the time packet i is received by receiver

p_i = the time packet i is played at receiver

$r_i - t_i$ = network delay for ith packet

d_i = estimate of average network delay after receiving ith packet

Dynamic estimate of average delay at receiver:

$$d_i = (1-u)d_{i-1} + u(r_i - t_i)$$

where u is a fixed constant (e.g., u = .01).

Adaptive playout delay (2)

Also useful to estimate the average deviation of the delay, v_i :

$$v_i = (1-u)v_{i-1} + u\,|\,r_i - t_i - d_i\,|$$

The estimates d_i and v_i are calculated for every received packet, although they are only used at the beginning of a talk spurt.

For first packet in talk spurt, playout time is:

$$p_i = t_i + d_i + Kv_i$$

where K is a positive constant. For this same packet, the play out delay is:

$$q_i = p_i - t_i$$

For packet j in the same talk spurt, play packet out at

$$p_j = t_j + q_i$$

Adaptive playout (3)

How to determine whether a packet is the first in a talkspurt:

- If there were never loss, receiver could simply look at the successive time stamps.
  - Difference of successive stamps > 20 msec, talk spurt begins.

- But because loss is possible, receiver must look at both time stamps and sequence numbers.
  - Difference of successive stamps > 20 msec and sequence numbers without gaps, talk spurt begins.

Recovery from packet loss (1)

- Loss: packet never arrives or arrives later than its scheduled playout time

forward error correction (FEC): simple scheme

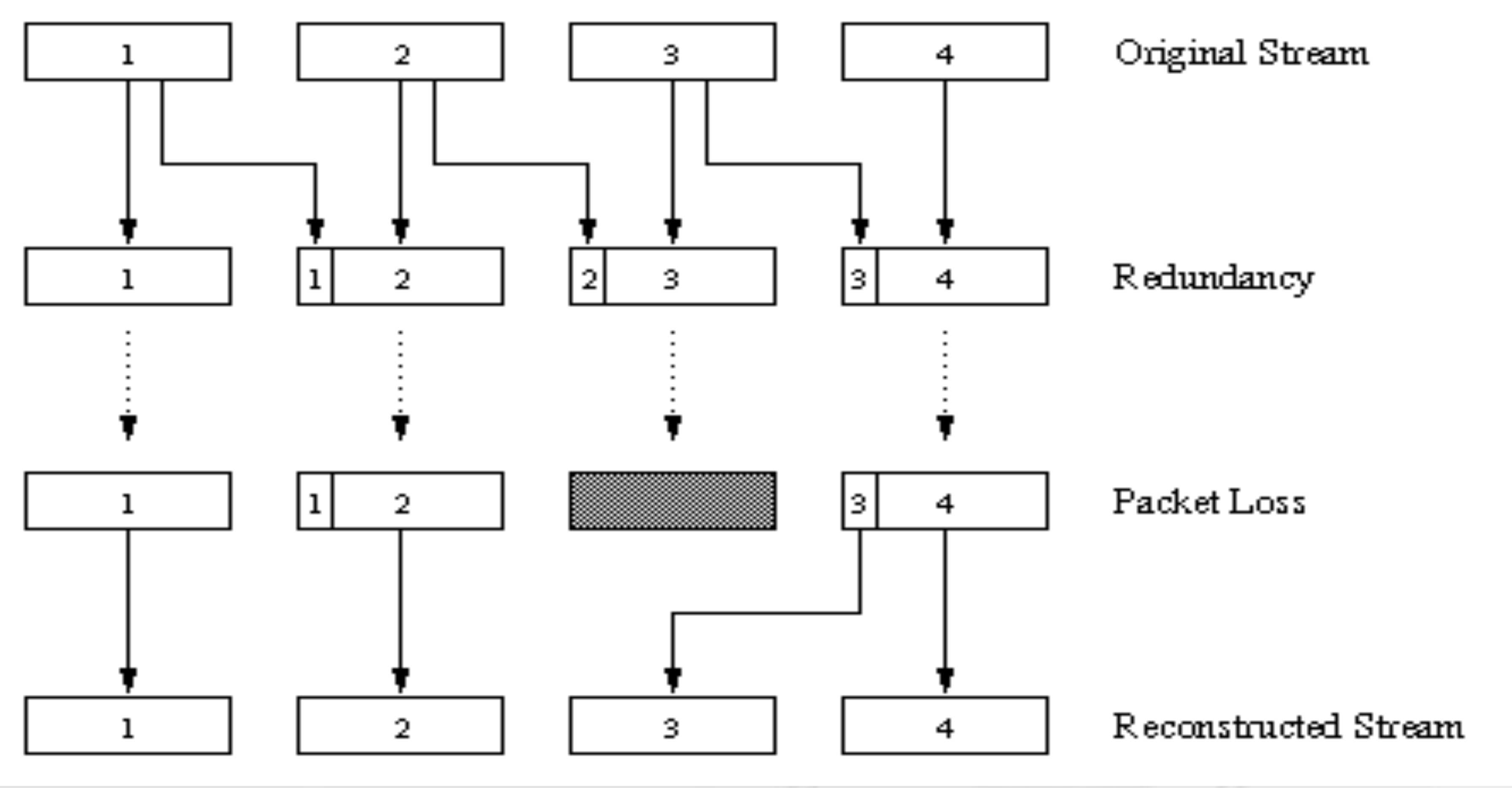
- for every group of n chunks create a redundant chunk by exclusive OR-ing the n original chunks
- send out n+1 chunks, increasing the bandwidth by factor 1/n.
- can reconstruct the original n chunks if there is at most one lost chunk from the n+1 chunks

- Playout delay needs to fixed to the time to receive all n+1 packets
- Tradeoff:
  - increase n, less bandwidth waste
  - increase n, longer playout delay
  - increase n, higher probability that 2 or more chunks will be lost

Recovery from packet loss (2)

2nd FEC scheme

- "piggyback lower quality stream"
- send lower resolution audio stream as the redundant information
- for example, nominal stream PCM at 64 kbps and redundant stream GSM at 13 kbps.
- Sender creates packet by taking the nth chunk from nominal stream and appending to it the (n-1)st chunk from redundant stream.

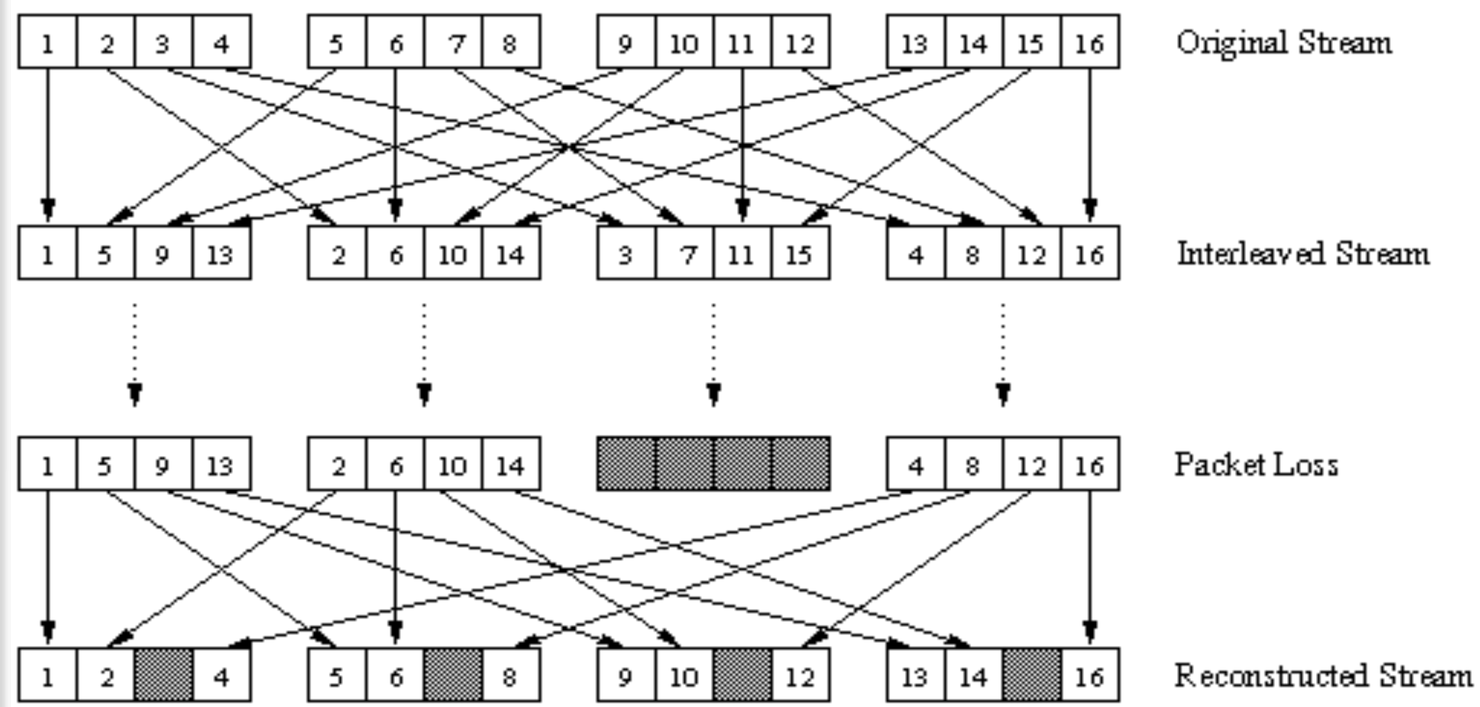


- Whenever there is non-consecutive loss, the receiver can conceal the loss.
- Only two packets need to be received before playback
- Can also append (n-1)st and (n-2)nd low-bit rate chunk

Recovery from packet loss (3)

<u>Interleaving</u>

- chunks are broken up into smaller units
- for example, 4 5 msec units per chunk
- interleave the chunks as shown in diagram
- packet now contains small units from different chunks



- Reassemble chunks at receiver
- if packet is lost, still have most of every chunk

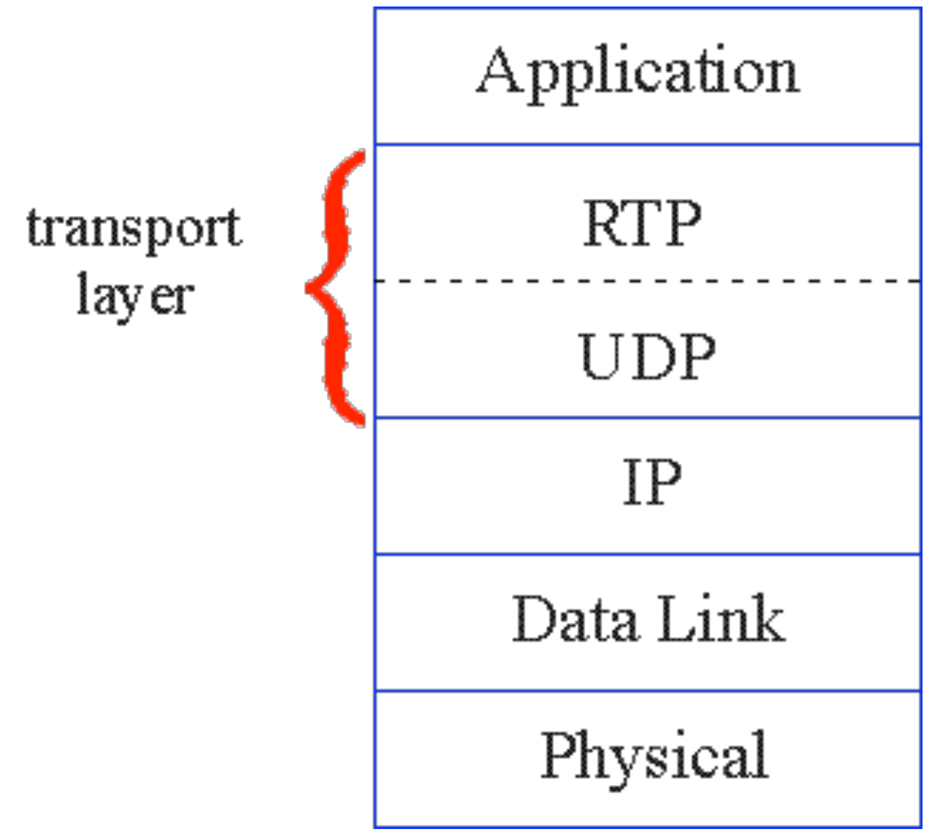
Recovery from packet loss (4)

Receiver-based repair of damaged audio streams

- produce a replacement for a lost packet that is similar to the original
- can give good performance for low loss rates and small packets (4-40 msec)
- simplest: repetition
- more complicated: interpolation

Real-Time Protocol (RTP)

- RTP specifies a packet structure for packets carrying audio and video data: RFC 1889.
- RTP packet provides
  - payload type identification
  - packet sequence numbering
  - timestamping



- RTP runs in the end systems.
- RTP packets are encapsulated in UDP segments
- Interoperability: If two Internet phone applications run RTP, then they may be able to work together

Real-Time Protocol (RTP)

- Provides standard packet format for real-time application
- Typically runs over UDP
- Specifies header fields below

RTP Header

- **Payload Type**: 7 bits, providing 128 possible different types of encoding; eg PCM, MPEG2 video, etc.
- **Sequence Number**: 16 bits; used to detect packet loss

RTP Header

Payload Type (7 bits): Used to indicate the type of encoding that is currently being used.

RTP Header

If a sender changes the encoding in the middle of a conference, the sender informs the receiver through this payload type field.

- Payload type 0: PCM mu-law, 64 Kbps
- Payload type 3, GSM, 13 Kbps
- Payload type 7, LPC, 2.4 Kbps
- Payload type 26, Motion JPEG
- Payload type 31. H.261
- Payload type 33, MPEG2 video

Sequence Number (16 bits): The sequence number increments by one for each RTP packet sent; may be used to detect packet loss and to restore packet sequence.

Real-Time Protocol (RTP)

- **Timestamp**: 32 bits; gives the sampling instant of the first audio/video byte in the packet; used to remove jitter introduced by the network

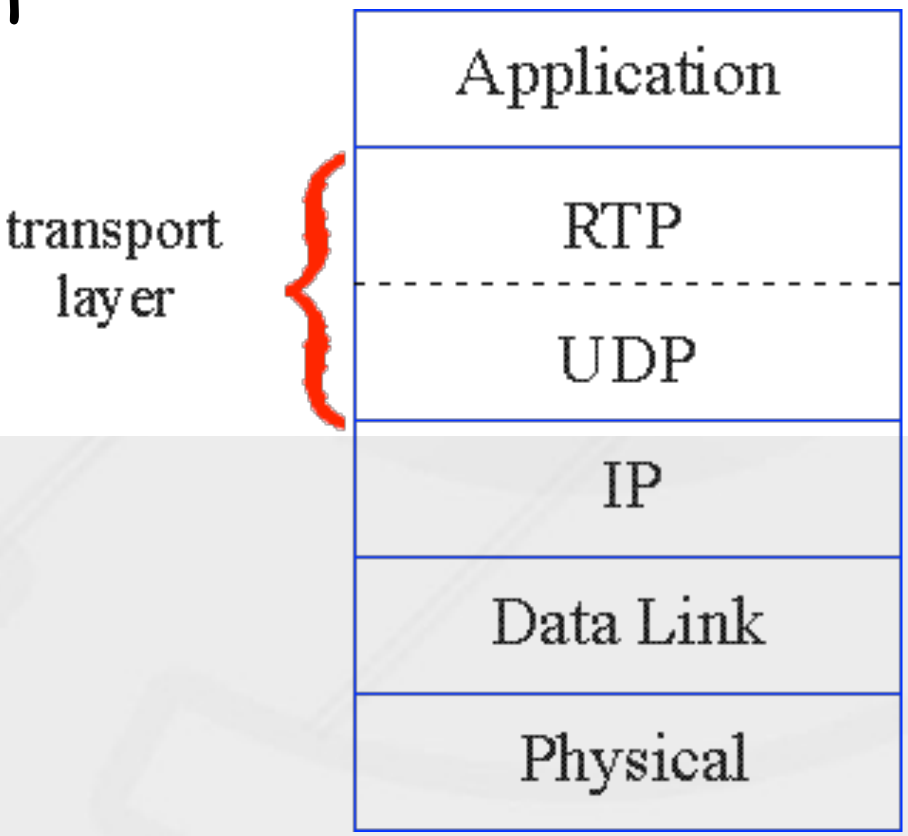
- **Synchronization Source identifier (SSRC)**: 32 bits; an id for the source of a stream; assigned randomly by the source

RTP Header

RTP runs on top of UDP

RTP libraries provide a transport-layer interface that extend UDP:

- port numbers, IP addresses
- error checking across segment
- payload type identification
- packet sequence numbering
- time-stamping



RTP Example

- Consider sending 64 kbps PCM-encoded voice over RTP.
- Application collects the encoded data in chunks, e.g., every 20 msec = 160 bytes in a chunk.
- The audio chunk along with the RTP header form the RTP packet, which is encapsulated into a UDP segment.

- RTP header indicates type of audio encoding in each packet; senders can change encoding during a conference. RTP header also contains sequence numbers and timestamps.

RTP and QoS

- RTP does **not** provide any mechanism to ensure timely delivery of data or provide other quality of service guarantees.
- RTP encapsulation is only seen at the end systems -- it is not seen by intermediate routers.
    - Routers providing the Internet's traditional best-effort service do not make any special effort to ensure that RTP packets arrive at the destination in a timely matter.
- In order to provide QoS to an application, the Internet most provide a mechanism, such as RSVP, for the application to reserve network resources.

RTP Streams

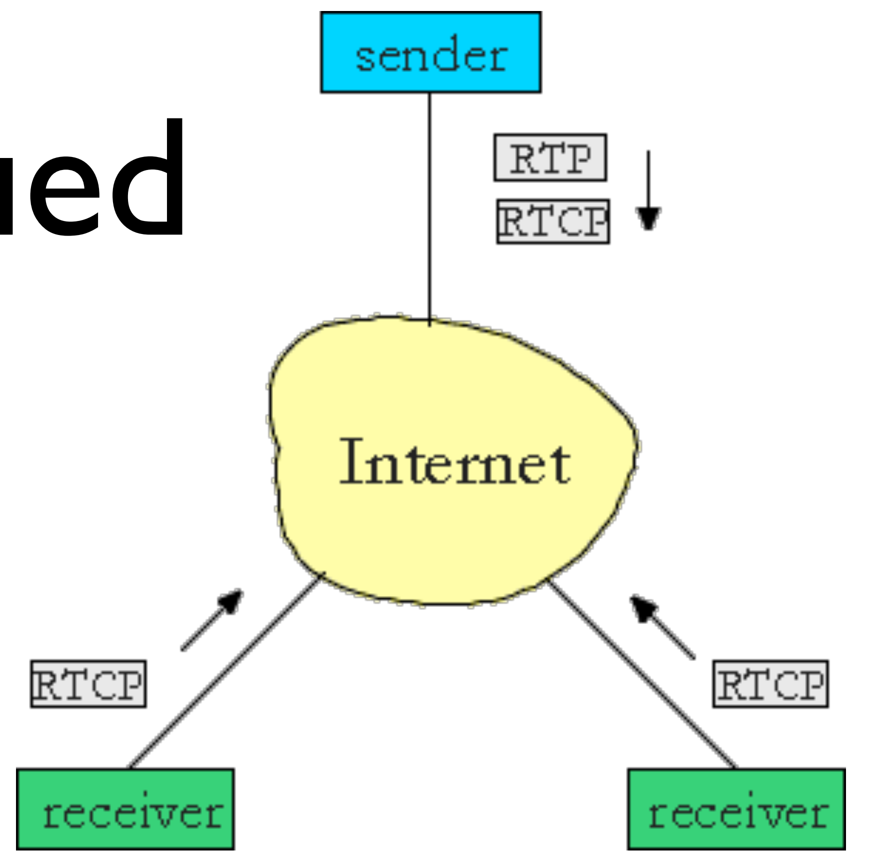
- RTP allows each source (for example, a camera or a microphone) to be assigned its own independent RTP stream of packets.
  - For example, for a videoconference between two participants, four RTP streams could be opened: two streams for transmitting the audio (one in each direction) and two streams for the video (again, one in each direction).

- However, some popular encoding techniques -- including MPEG1 and MPEG2 -- bundle the audio and video into a single stream during the encoding process. When the audio and video are bundled by the encoder, then only one RTP stream is generated in each direction.
- For a many-to-many multicast session, all of the senders and sources typically send their RTP streams into the same multicast tree with the same multicast address.

Real-Time Control Protocol (RTCP)

- Works in conjunction with RTP.
- Each participant in an RTP session periodically transmits RTCP control packets to all other participants. Each RTCP packet contains sender and/or receiver reports that report statistics useful to the application.
- Statistics include number of packets sent, number of packets lost, inter-arrival jitter, etc.
- This feedback of information to the application can be used to control performance and for diagnostic purposes.
- The sender may modify its transmissions based on the feedback.

RTCP - Continued



- For an RTP session there is typically a single multicast address: all RTP and RTCP packets belonging to the session use the multicast address.

- RTP and RTCP packets are distinguished from each other through the use of distinct port numbers.

- To limit traffic, each participant reduces his RTCP traffic as the number of conference participants increases.

RTCP Packets

Receiver report packets:

- fraction of packets lost, last sequence number, average interarrival jitter.

Sender report packets:

- SSRC of the RTP stream, the current time, the number of packets sent, and the number of bytes sent.

Source description packets:

- e-mail address of the sender, the sender's name, the SSRC of the associated RTP stream. Packets provide a mapping between the SSRC and the user/host name.

Synchronization of Streams

- RTCP can be used to synchronize different media streams within a RTP session.
- Consider a videoconferencing application for which each sender generates one RTP stream for video and one for audio.
- The timestamps in these RTP packets are tied to the video and audio sampling clocks, and are not tied to the wall-clock time (i.e., to real time).
- Each RTCP sender-report packet contains, for the most recently generated packet in the associated RTP stream, the timestamp of the RTP packet and the wall-clock time for when the packet was created. Thus the RTCP sender-report packets associate the sampling clock to the real-time clock.
- Receivers can use this association to synchronize the playout of audio and video.

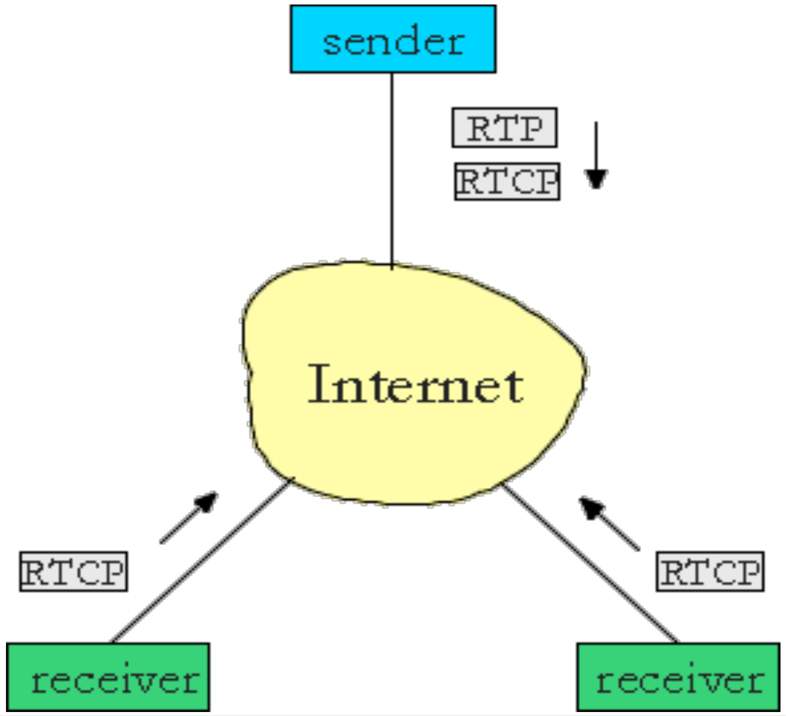
RTCP Bandwidth Scaling

- RTCP attempts to limit its traffic to 5% of the session bandwidth.
- For example, suppose there is one sender, sending video at a rate of 2 Mbps. Then RTCP attempts to limit its traffic to 100 Kbps.
- The protocol gives 75% of this rate, or 75 kbps, to the receivers; it gives the remaining 25% of the rate, or 25 kbps, to the sender.

- The 75 kbps devoted to the receivers is equally shared among the receivers. Thus, if there are R receivers, then each receiver gets to send RTCP traffic at a rate of 75/R kbps and the sender gets to send RTCP traffic at a rate of 25 kbps.
- A participant (a sender or receiver) determines the RTCP packet transmission period by dynamically calculating the the average RTCP packet size (across the entire session) and dividing the average RTCP packet size by its allocated rate.

RTP Control Protocol (RTCP)

- Protocol specifies report packets exchanged between sources and destinations of multimedia information
- Three reports are defined: Receiver reception, Sender, and Source description
- Reports contain statistics such as the number of packets sent, number of packets lost, inter-arrival jitter
- Used to modify sender transmission rates and for diagnostics purposes



RTCP Bandwidth Scaling

- If each receiver sends RTCP packets to all other receivers, the traffic load resulting can be large
- RTCP adjusts the interval between reports based on the number of participating receivers
- Typically, limit the RTCP bandwidth to 5% of the session bandwidth, divided between the sender reports (25%) and the receivers reports (75%)

小结RTCP的四种功能

- 提供传送端信息传输的品质回报
- 展现层等级的识别信息，即CNAME
- 成员传送回报 => 传送的速率控制
- 提供少量session控制信息

Streaming media servers (softwares)

- Audio
- Video
- P2P ...

Audio streaming

- 校园广播站:
  - SHOUTcast + Winamp （2003）
- 豆瓣FM

Commercial solutions

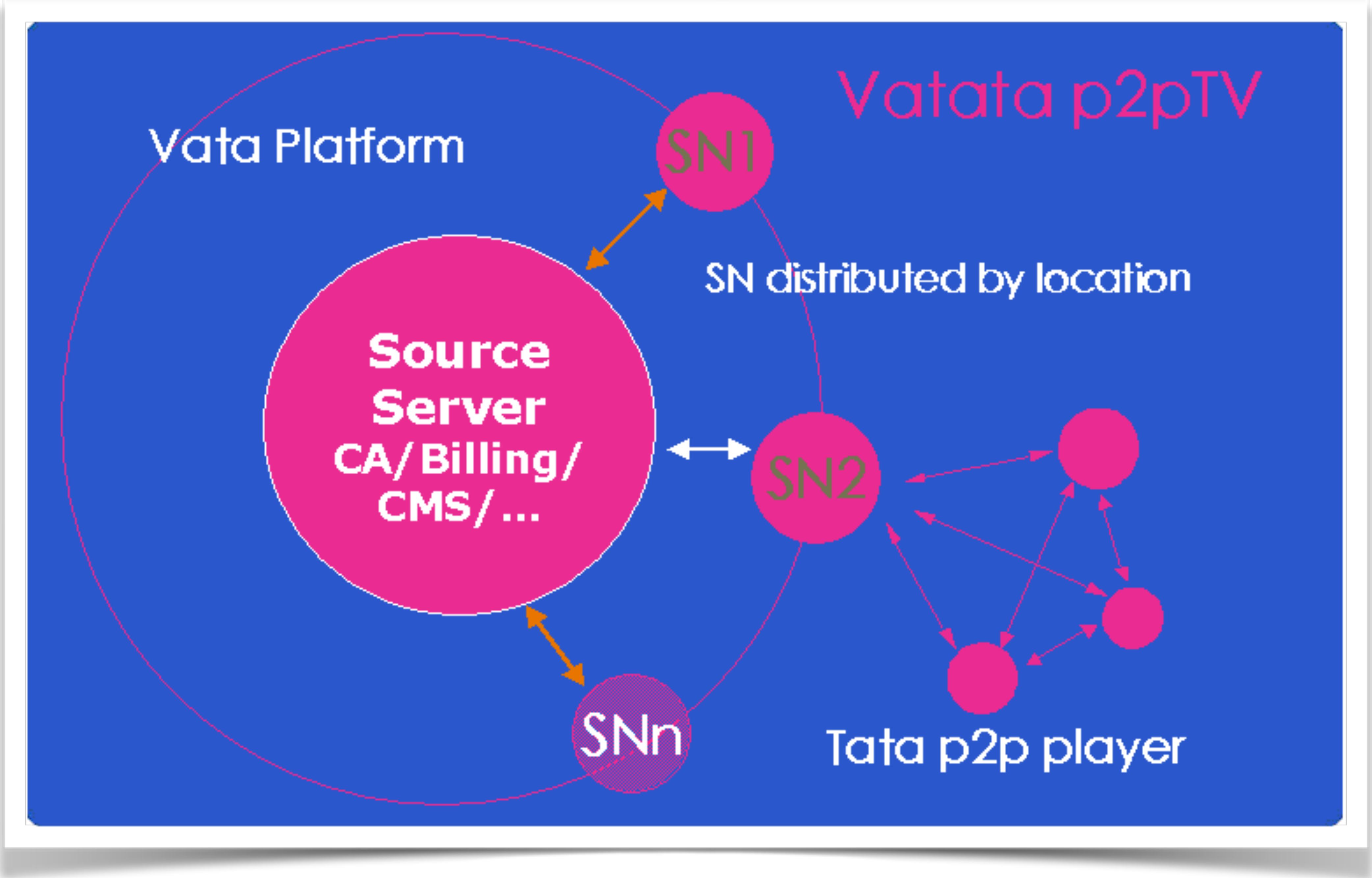
- Real Producer
- Windows Media Encoder
- QuickTime Streaming Server

P2P solutions

http://clin003.com/ideas/p2p-streaming-media-server-1318/

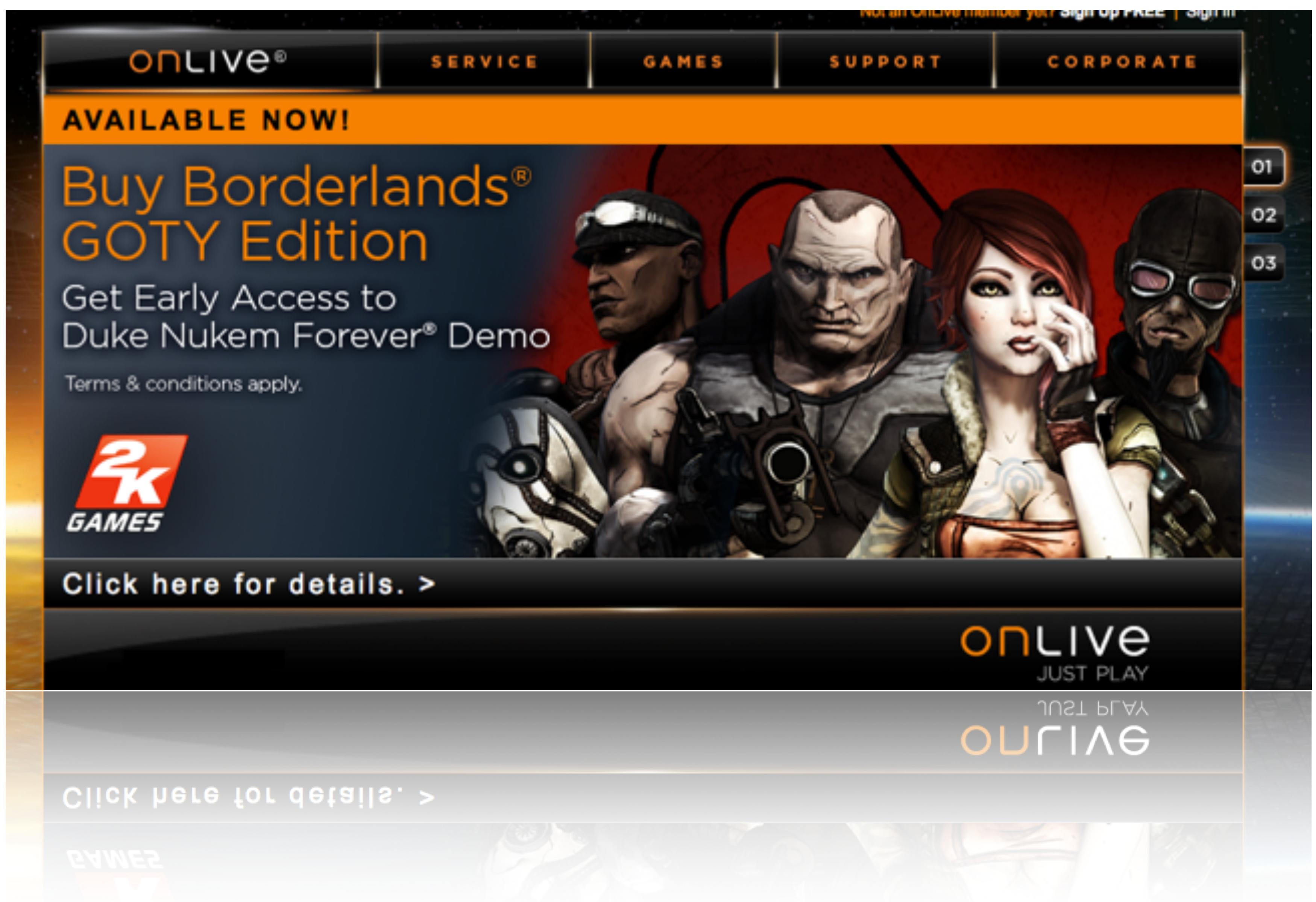
- CCIPTV Live Server
- PPS流媒体服务器
- MSMStream P2P流媒体
- 蓝天PPD下载点播系统
- 超限组播视像服务器
- Vatata流媒体系统目录服务器
- QVOD 流媒体点播系统 (BT, 免费试用)

Vatata


Vatata p2pTV
Vata Platform
SN1
SN distributed by location
Source Server
CA/Billing/
CMS/...
SN2
SNn
Tata p2p player

Future of steaming media

- Play game through streaming!



Reference

- Fundamentals of Multimedia, Chapter 16
  - 机械工业出版社

Digital Asset Management
数字媒体资源管理

6. Digital Media Retrieval

任课老师：张宏鑫
2014-11-06

Origin of digital media retrieval

- IR (Information Retrieval)
  - –To retrieve information that users want based on some keys or hints
  - –Support:
    - daily life use
    - authoring
    - thinking and designing

Main methods of digital media retrieval

- **Text-based** digital media retrieval
  - –Boolean model
  - –Clustering model
  - –Vector model
  - –Probability model
- **Content-based** digital media retrieval
  - **–Query By Examples**
- **Semantic-based** digital media retrieval

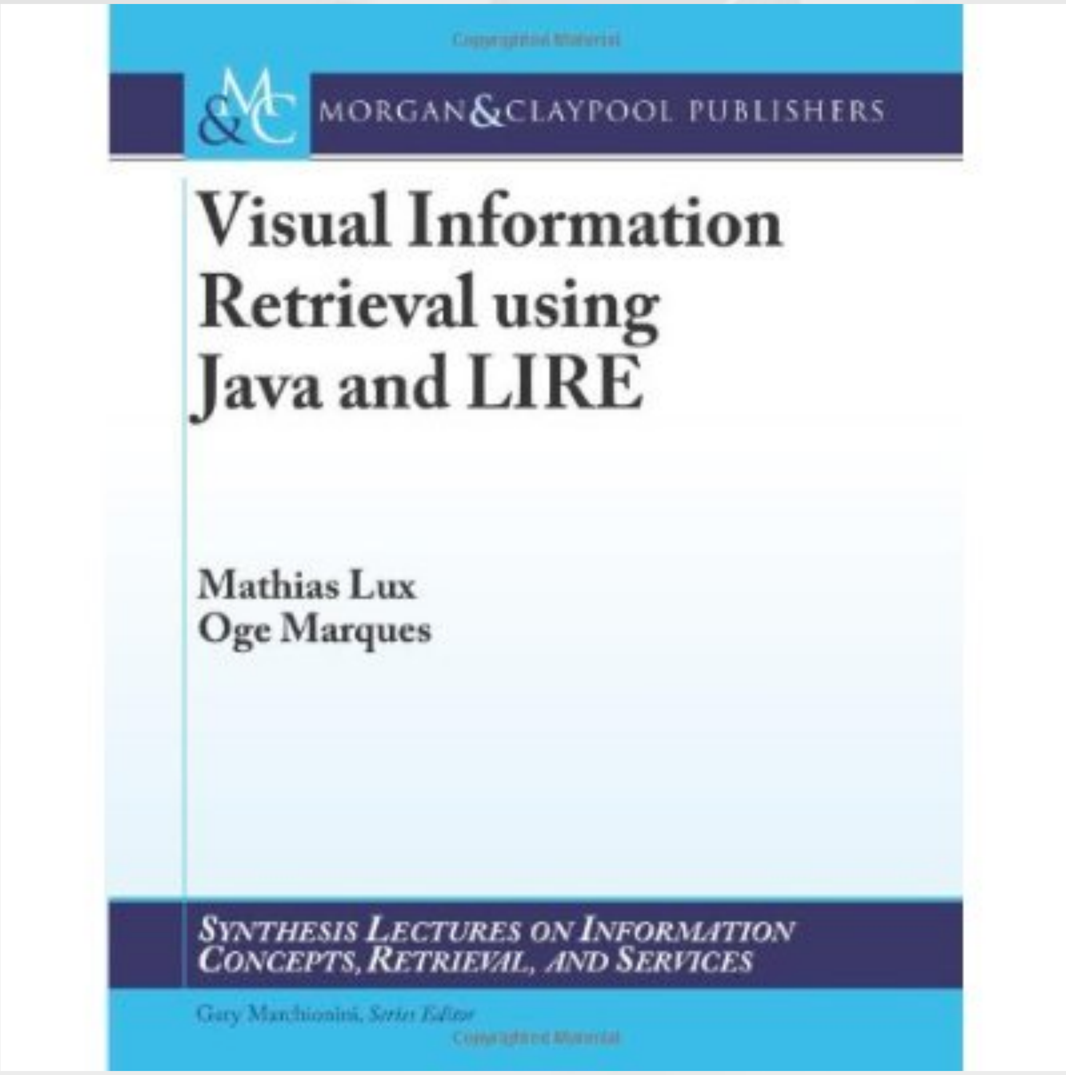
Content based digital media retrieval

- Query by example on multimedia-data

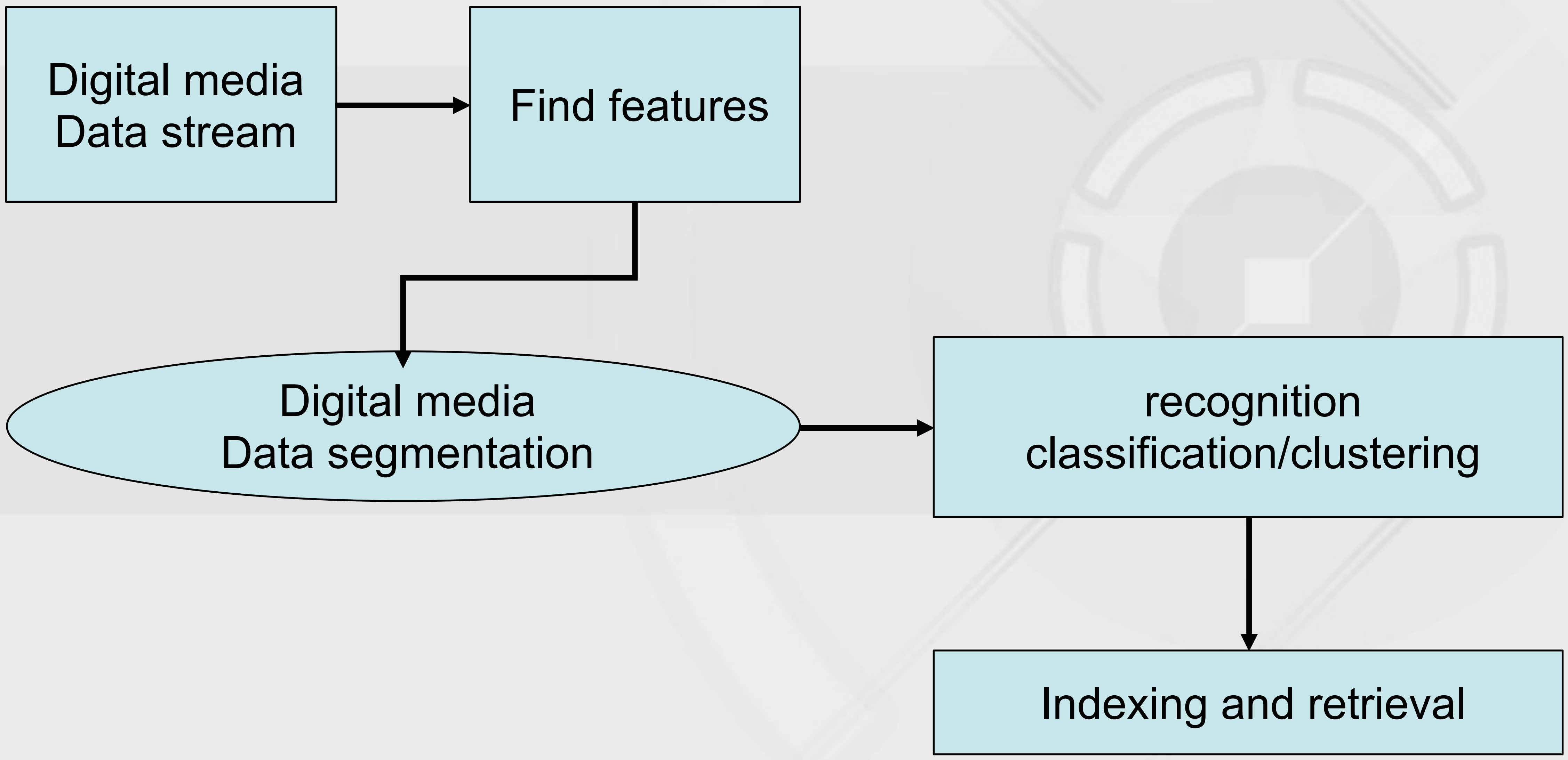
- Demo:
  - **The GNU Image-Finding Tool**
  - http://www.gnu.org/software/gift/

LIRE
Image Search Engine with Lucene

- https://code.google.com/p/lire/



The workflow of digital media analysis and retrieval



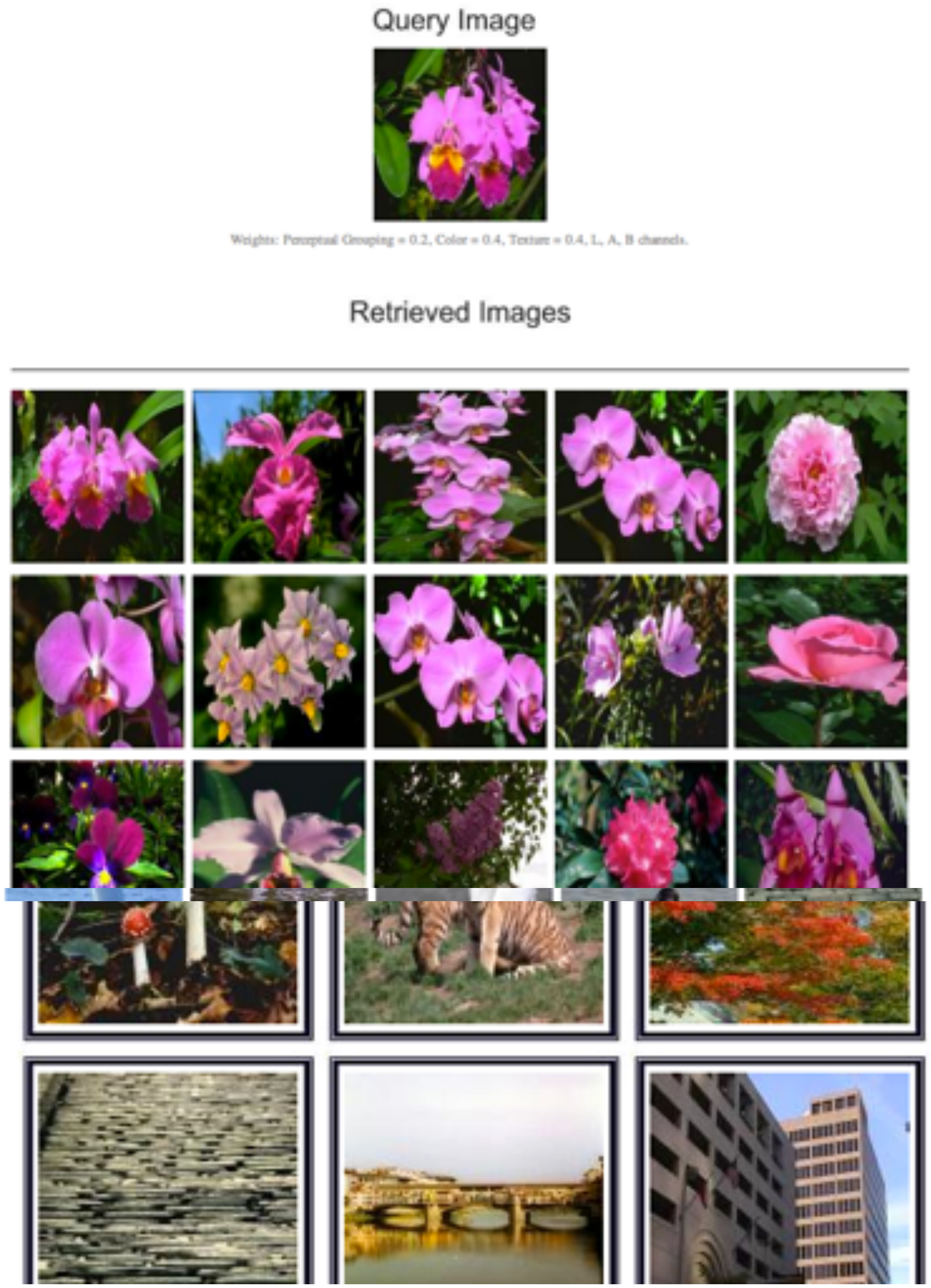
Content-based digital media retrieval

- In this lesson, we will know …
  - –Content-based image retrieval
  - –Content-based video retrieval
  - –Content-based audio retrieval
  - –Content-based graphics retrieval

  - –Merging and analysis of multiple media
  - –Development and challenging

1. Content-based image retrieval

CBIR



CIRES

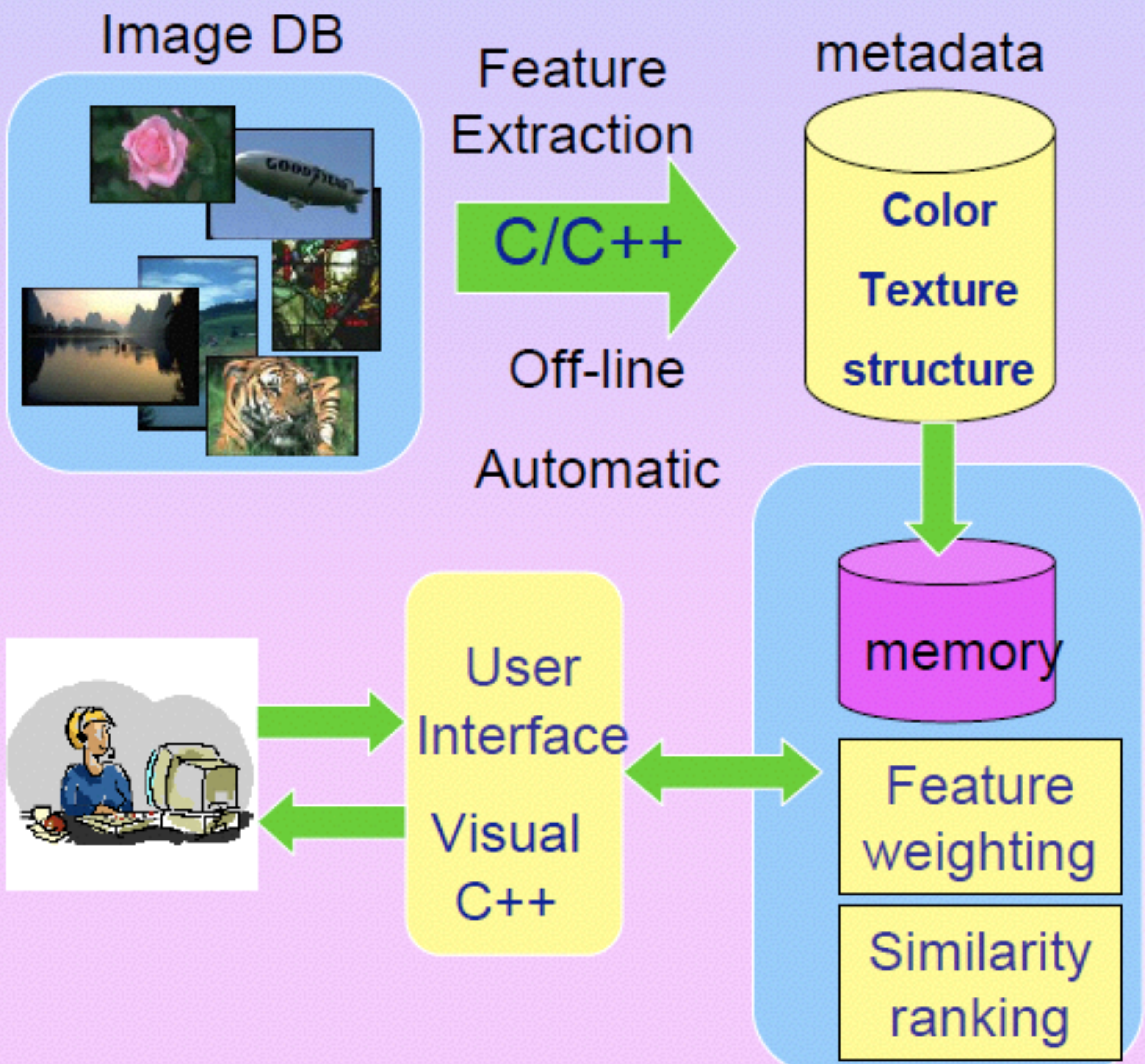
http://amazon.ece.utexas.edu/~qasim/sample_queries.htm

DEMO from the RGB group

iPad APP：”服饰绘”

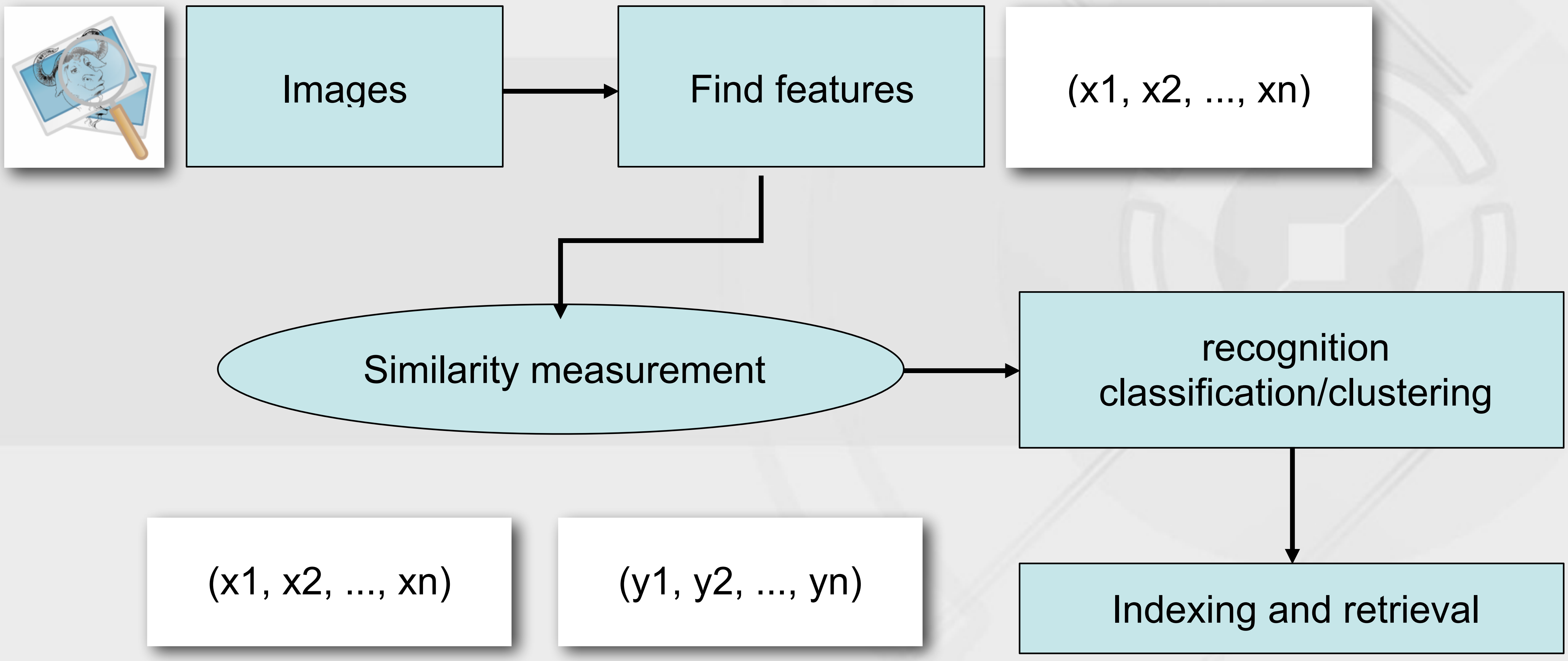
Multimedia Information Retrieval

Content-based Image Retrieval



- **Color**
  - ✓Color histogram
  - ✓Color moments
  - ✓Color correlogram
- **Texture**
  - ✓Tamura texture
  - ✓Co-occurrence matrices
  - ✓Gabor features
  - ✓Wavelet moments
- **Shape**
  - ✓Fourier descriptor
- **Structure**
  - ✓Edge-based features

Workflow of CBIR



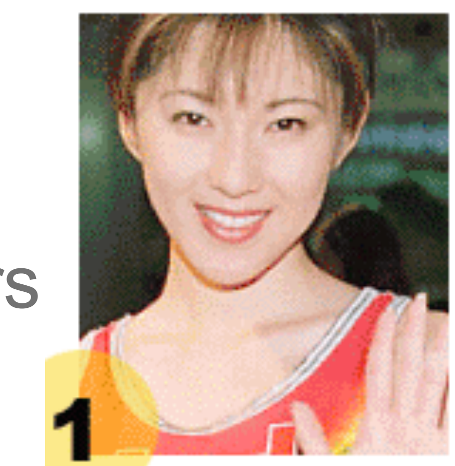
Features of image

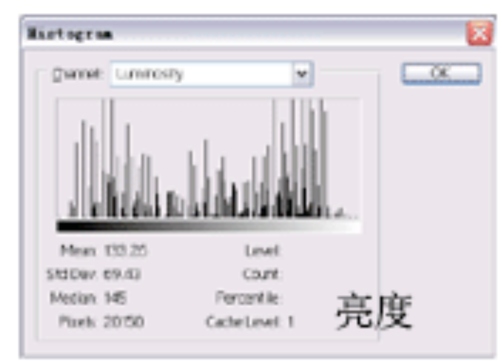
- Finding out features of image is a key step of image retrieval
  - Image-based retrieval usually need to pre-construct feature database of images for retrieval

- Major image features:
  - Color features
  - Texture features
  - Shape features
  - Space relation features

Color features of image

- Color feature is a most widely used vision feature. It is mainly used to analyze color distributions in an image, including:
  - Color histogram
  - Color moments
  - Color set
  - Color clustering vectors
  - Color relation graph





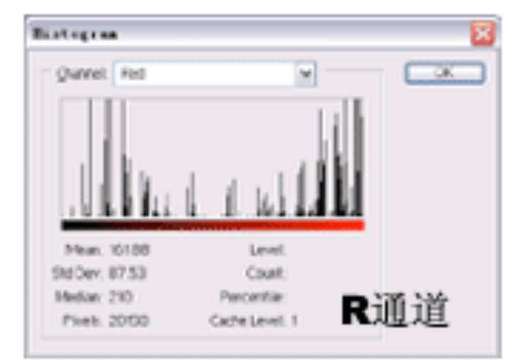
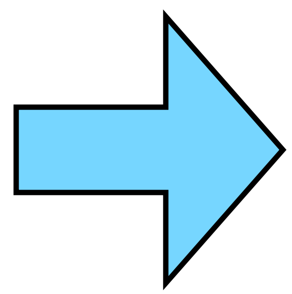


Image histogram



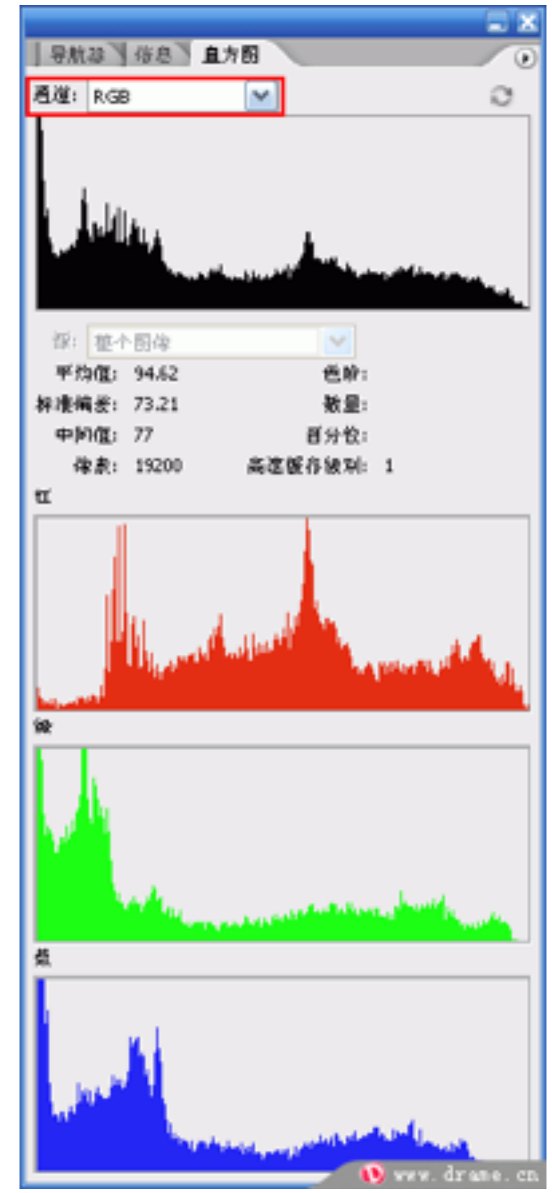
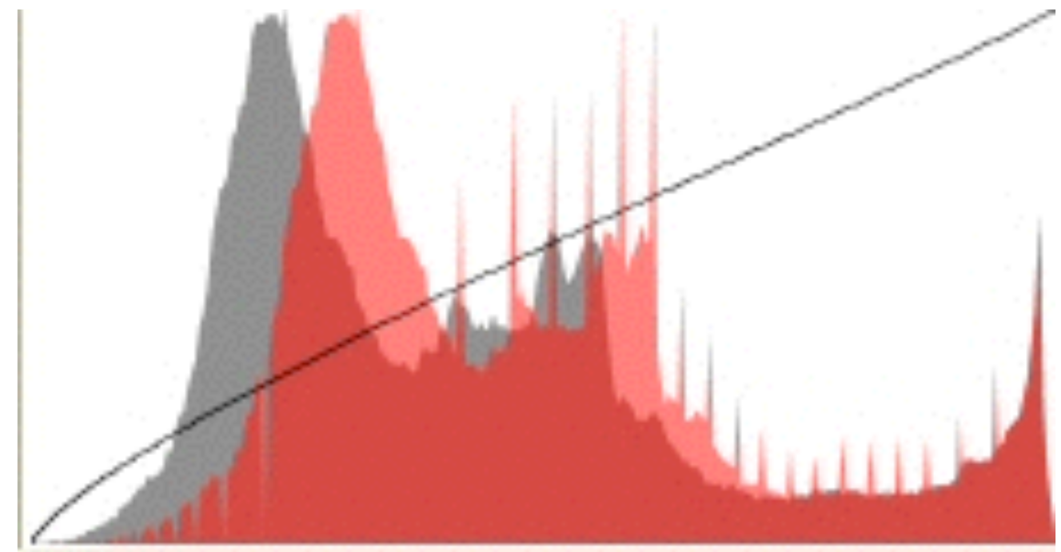


Image histogram



图像的颜色矩 (color moments)

- Color moments are global statistical features of an image, which are proposed by Stricker and Orengo.
  - First order moment (mean) $$\mu_i = \frac{1}{n}\sum_{j=1}^{n} I_{ij}$$
  - Second order moment (variance) $$\sigma_i^2 = \frac{1}{n}\sum_{j=1}^{n} (I_{ij} - \mu_i)^2$$
  - Third order moment (skewness) $$s_i^3 = \frac{1}{n}\sum_{j=1}^{n} (I_{ij} - \mu_i)^3$$
- Color moments are always applied with other image features for efficiently shrinking seeking ranges.

color moments: example

1	3	6	3	1
3	6	8	6	3
6	8	10	8	6
3	6	8	6	3
1	3	6	3	1

mean =4.72

variance =6.52

skewness =2.34

Image texture features

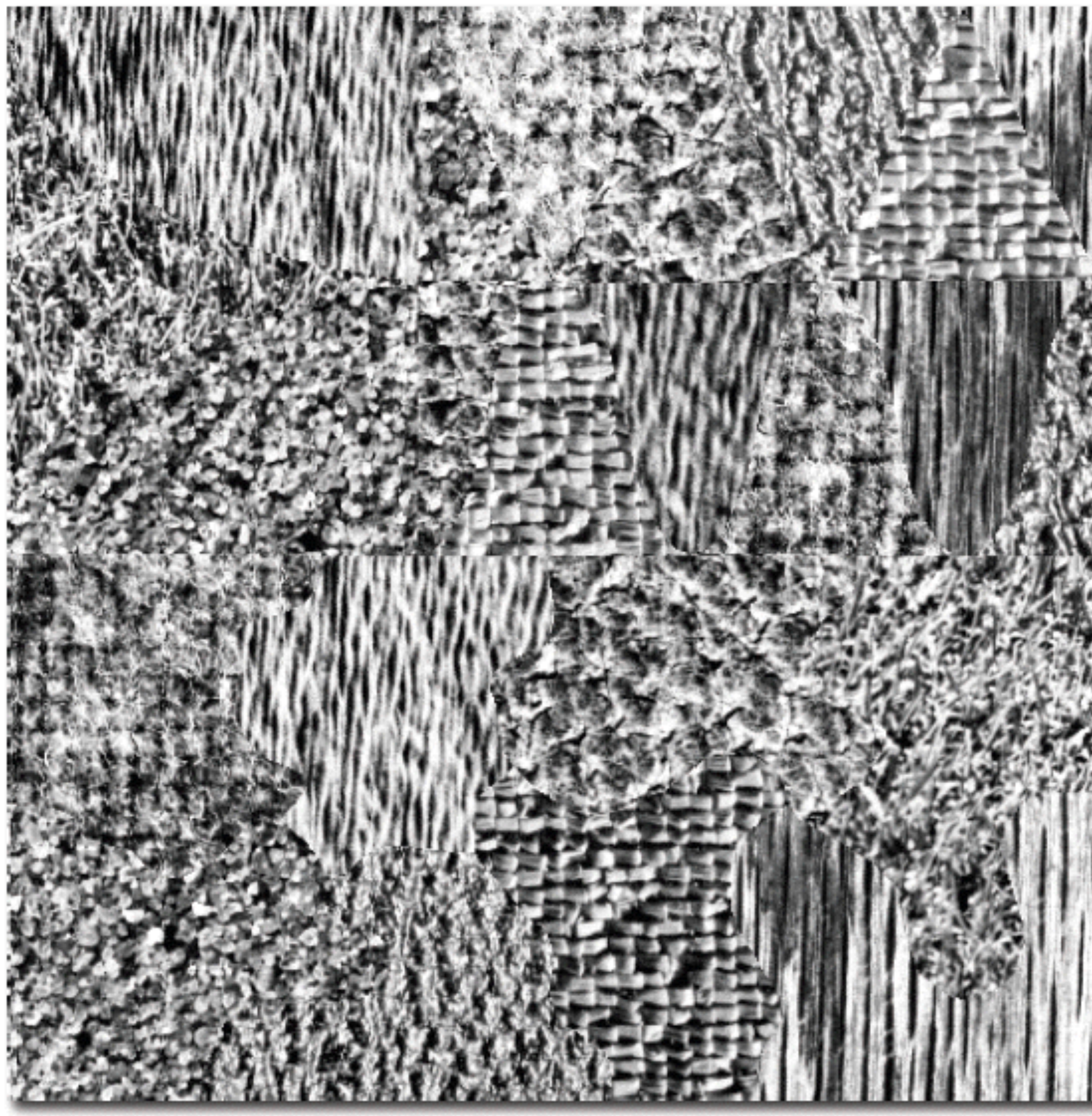


Image texture features

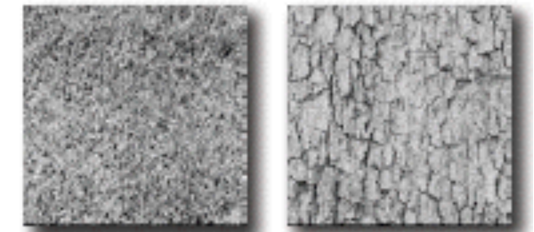
- Texture features are such vision features employed to measure homogeneous phenomenon in images. They are
  - independent to color or illuminance,
  - and are intrinsic features of object surfaces.

- Major texture features
  - Tamura texture features
  - Self-regression texture model
  - Transform based texture features
    - DWT，DFT，Garbor filter bank
  - others

Tamura texture features

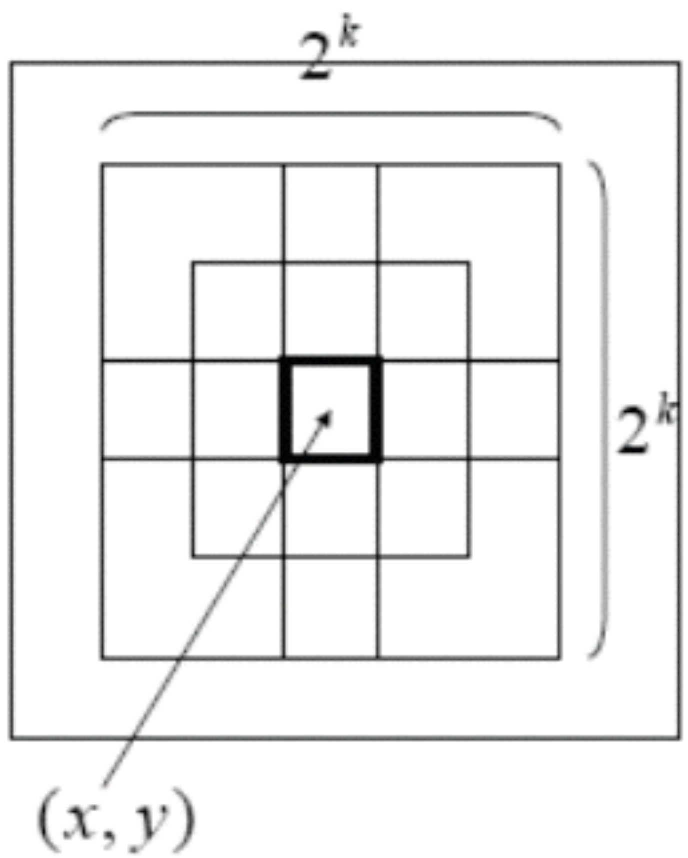
- a set of texture feature representation based on the psychology research results on human vision cognition of textures:
  - coarseness (粗糙度)
  - contrast (对比度)
  - directionality (方向度)
  - line-likeness (线相似度)
  - regularity (规整度)
  - roughness (粗略度)

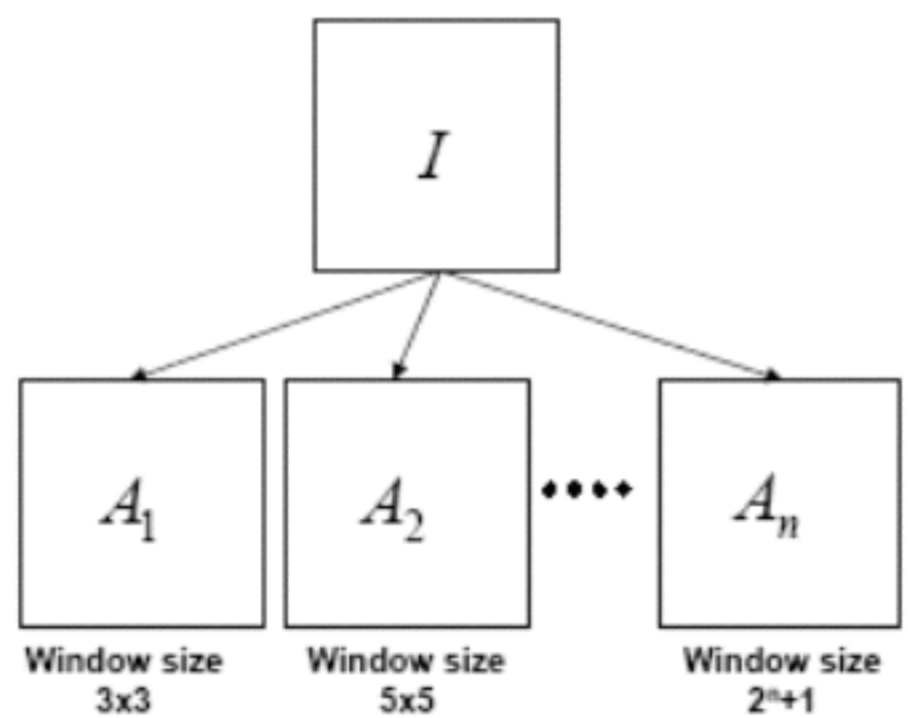
Tamura – Coarseness



- Goal
  - Pick a large size as best when coarse texture is present, or a small size when only fine texture
- Step 1: Compute averages at different scales at every points

$$A_k(x,y) = \sum_{i=x-2^{k-1}}^{x+2^{k-1}-1} \sum_{j=y-2^{k-1}}^{y+2^{k-1}-1} I(i,j)/2^{2k}$$

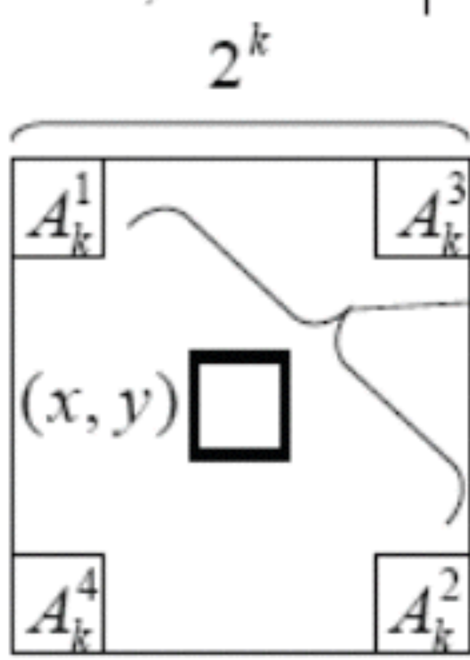




Tamura – Coarseness (cont.)

- Step 2: compute neighborhood difference at each scale on opposite sides of different directions

$$E_{k,h}(x,y) = \left| A_k(x-2^{k-1},y) - A_k(x+2^{k-1},y) \right|$$



$$E_{k,a}(x,y) = \left| A_k^{\,1} - A_k^{\,2} \right|$$

$$E_{k,b}(x,y) = \left| A_k^{\,3} - A_k^{\,4} \right|$$

$$(x,y) \longrightarrow \{E_{1,a}, E_{1,b}, E_{2,a}, E_{2,b}, \ldots, E_{n,a}, E_{n,b}\}$$

Tamura – Coarseness (cont.)

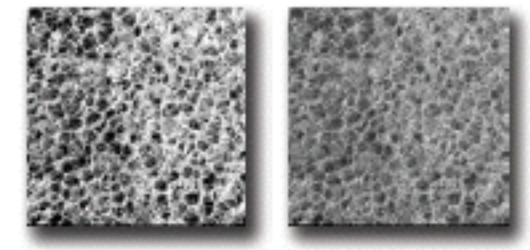
- Step 3: select the scale with the largest variation

$$S_{max}(x,y) = 2^k \quad / \quad E_k = max\{E_1, E_2, \ldots, E_L\}$$

- Step 4: compute the coarseness

$$M_{crs} = \frac{1}{n \times m} \sum_{i}^{n} \sum_{j}^{m} S_{max}(i,j)$$

Tamura – Contrast



- Gaussian-like histogram distribution → low contrast

- Histogram polarization. Is it Gaussian? How many peaks it has? Where they are?

- Polarization can be estimated by the kurtosis (曲率度)

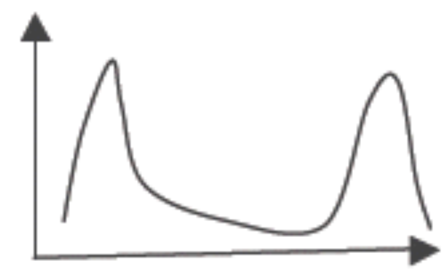
$$\alpha_4 = \frac{\mu_4}{\sigma^4}$$

$$\mu_4 = E[I^4(x,y)]$$

$$\sigma^4 = E[(I(x,y) - \mu)^4]$$

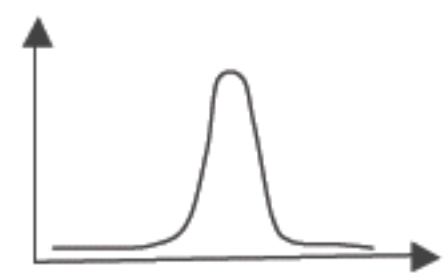
Tamura – Contrast (cont.)

$$\alpha_4 = \frac{\mu_4}{\sigma^4} \downarrow$$



distribution with two separate peaks

$$\alpha_4 = \frac{\mu_4}{\sigma^4} \uparrow$$

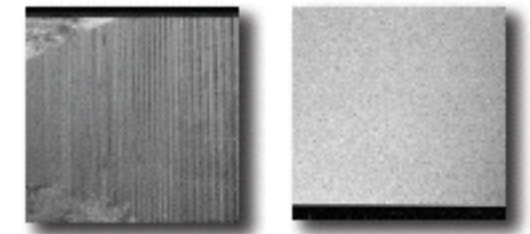


unimodal distribution

- Contrast estimate is given by:

$$M_{contrast} = \frac{\sigma}{(\alpha_4)^{\frac{1}{4}}}$$

Tamura – Orientation

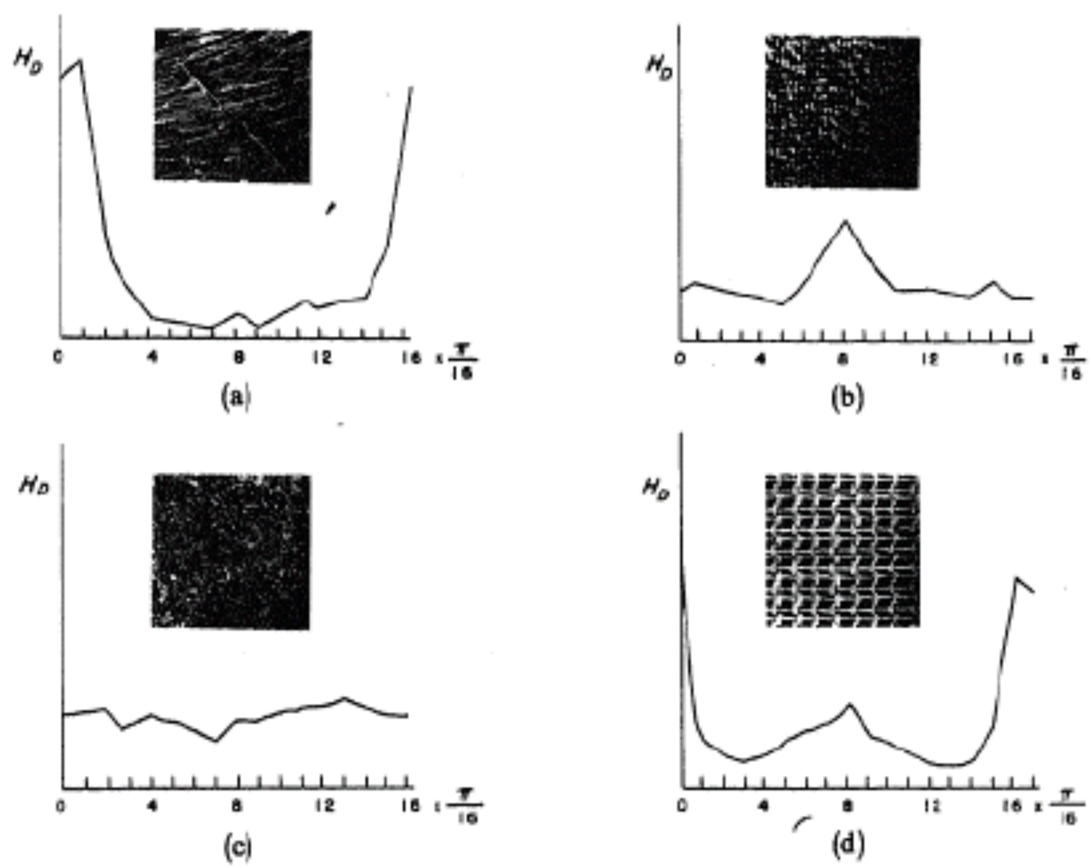


- Building the histogram of local edges at different orientations $H_D(k)$
  - By deriving the edge magnitude at X and Y directions

$$\theta = \mathrm{tg}^{-1}(\nabla_V / \nabla_H) + \frac{\pi}{2}$$

$$|\nabla G| = (|\nabla_V| + |\nabla_H|)/2$$

$$\nabla_V \quad \begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix} \qquad \nabla_H \quad \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix}$$

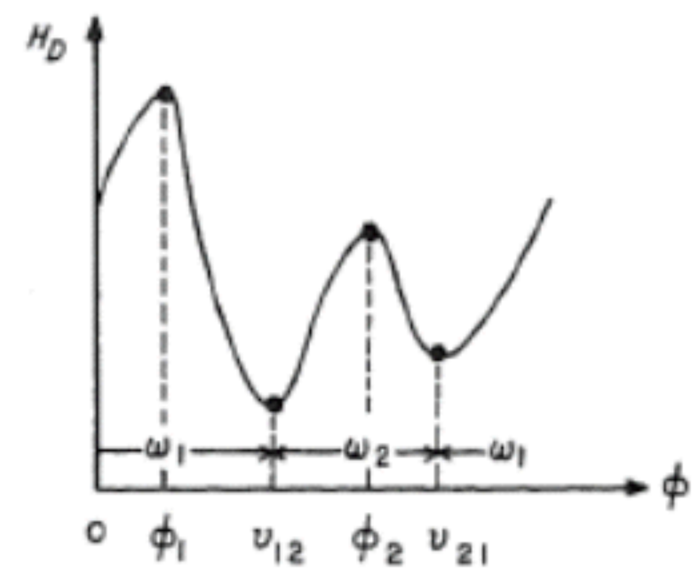


Tamura – Orientation (cont.)

- Compute the estimate from the sharpness of the peaks
  - By summing the second moments around each peak
    e.g., flat histogram
    → large 2nd moment (variance)
    → small orientation

$$M_{orient} = 1 - r \cdot n_p \cdot \sum_{p}^{n_p} \sum_{\phi \in w_p} (\phi - \phi_p)^2 \cdot H_D(\phi)$$

$n_p =$	*Number of peaks*
$\phi_p =$	*Position of peak, p, in* H_D
$w_p =$	*Points in peak p*
$r =$	*Normalisation factor*

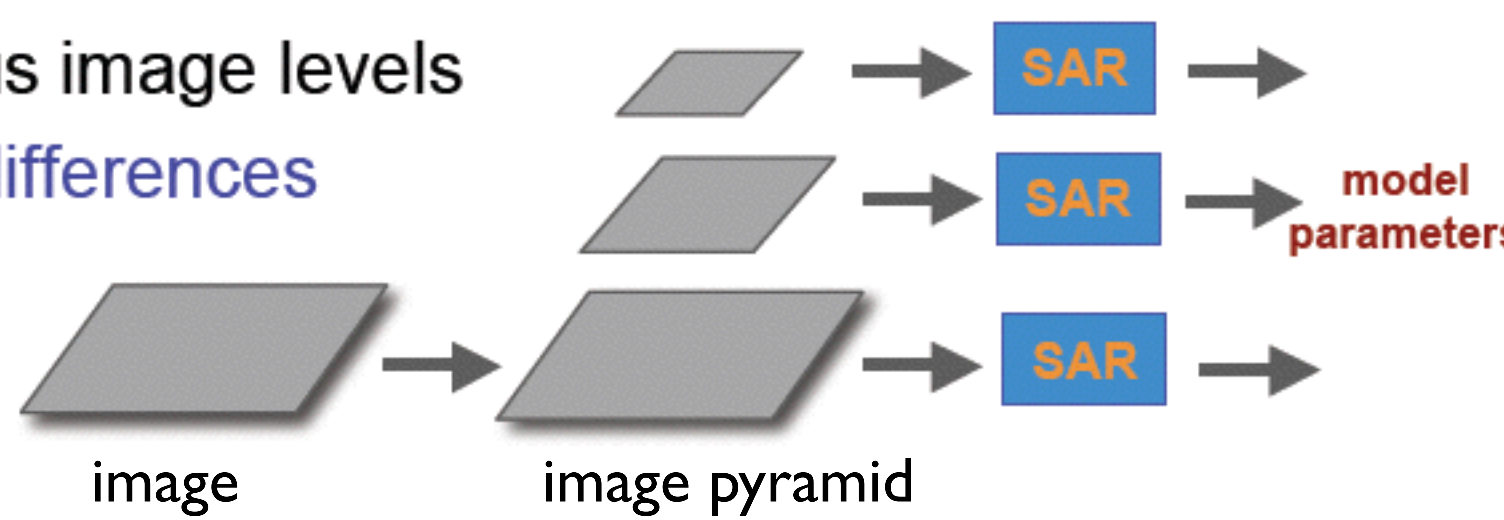


(MR)SAR

[Mao'92]

- Each pixel is a random variable whose value is estimated from its neighboring pixels + noise
  - A kid of Markov Random Field model

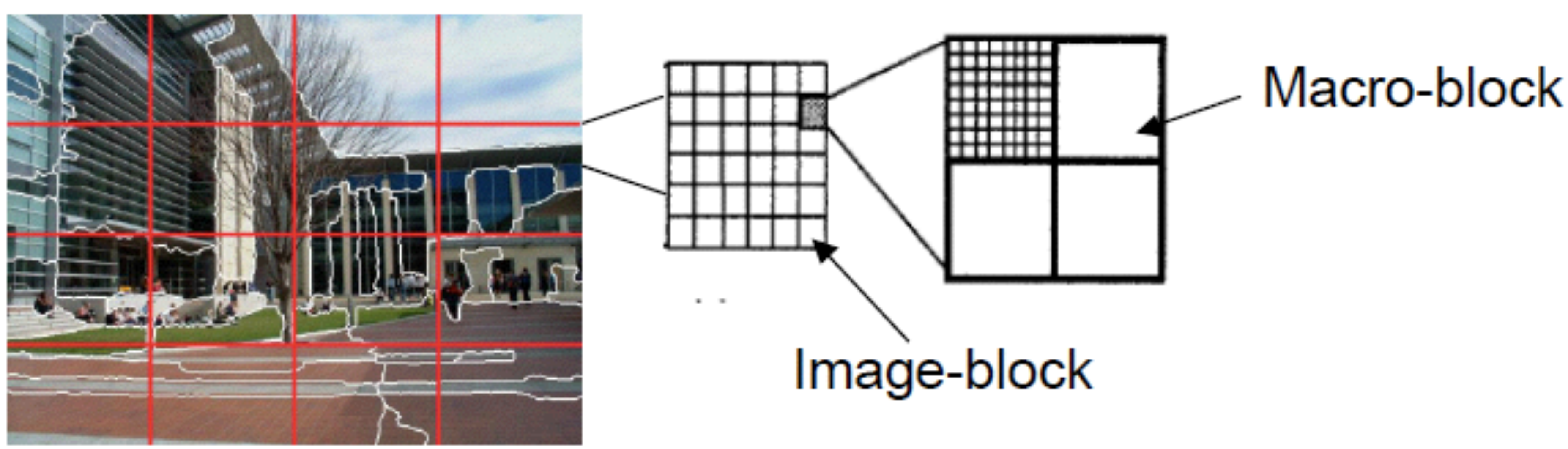
- **SAR** Model (Simultaneous Autoregressive)
  - Describes each pixel in terms of its neighboring pixels.
- **MRSAR** Model (MultiResolution SAR)
  - Describing granularities by representing textures at variety of resolutions
  - SAR applied at various image levels
  - Metric → parameter differences



Edge Histogram

- Edge histogram (EHD)
- Captures the spatial distribution of the edge in six statues: 0º, 45º, 90º, 135º, non direction and no edge.
  - Utilizing the filters

  - Global EHD of an image: Concatenating 16 sub EHDs into a 96 bins
  - Local EHD of a segment
    - Grouping the edge histogram of the image-blocks fallen into the segment

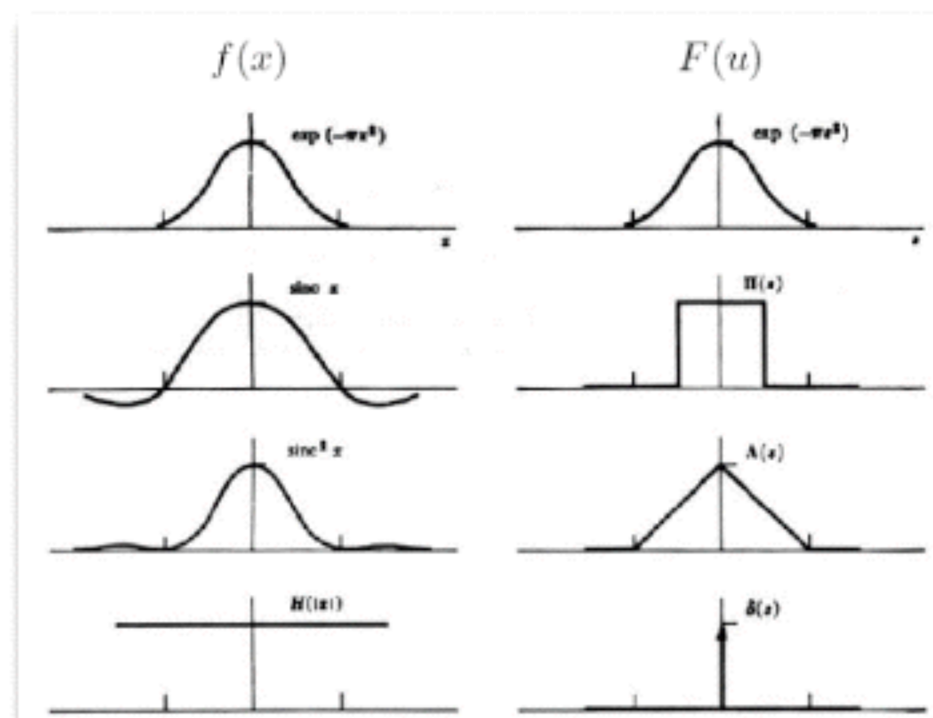


The Fourier Transform

- Represent function on a new basis
  - Think of functions as vectors, with many components
  - We now apply a linear transformation to transform the basis
    - dot product with each basis element

$$F\big(g(x,y)\big)(u,v)=\iint_{\mathrm{R}^2} g(x,y)e^{-i2\pi(ux+vy)}dxdy$$

- In the expression, u and v select the basis element, so a function of x and y becomes a function of u and v
- basis elements have the form $e^{-i2\pi(ux+vy)}$



Discrete Fourier Transform

- 2D DFT

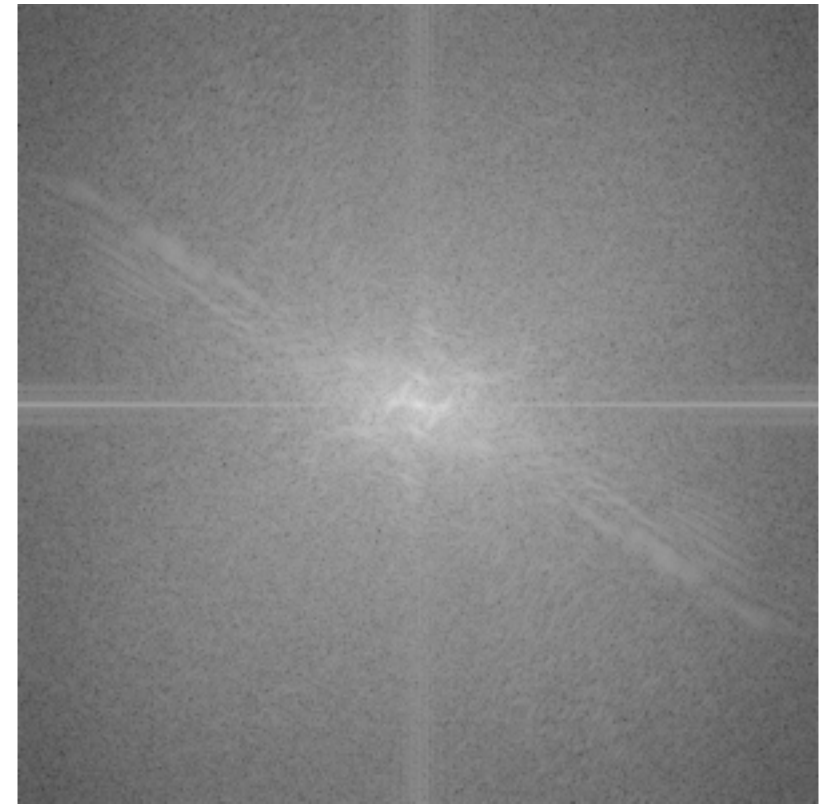
$$F(k,l) = \frac{1}{N^2} \sum_{a=0}^{N-1} \sum_{b=0}^{N-1} f(a,b)\, e^{-\iota 2\pi \left(\frac{ka}{N} + \frac{lb}{N}\right)}$$

- 2D IDFT

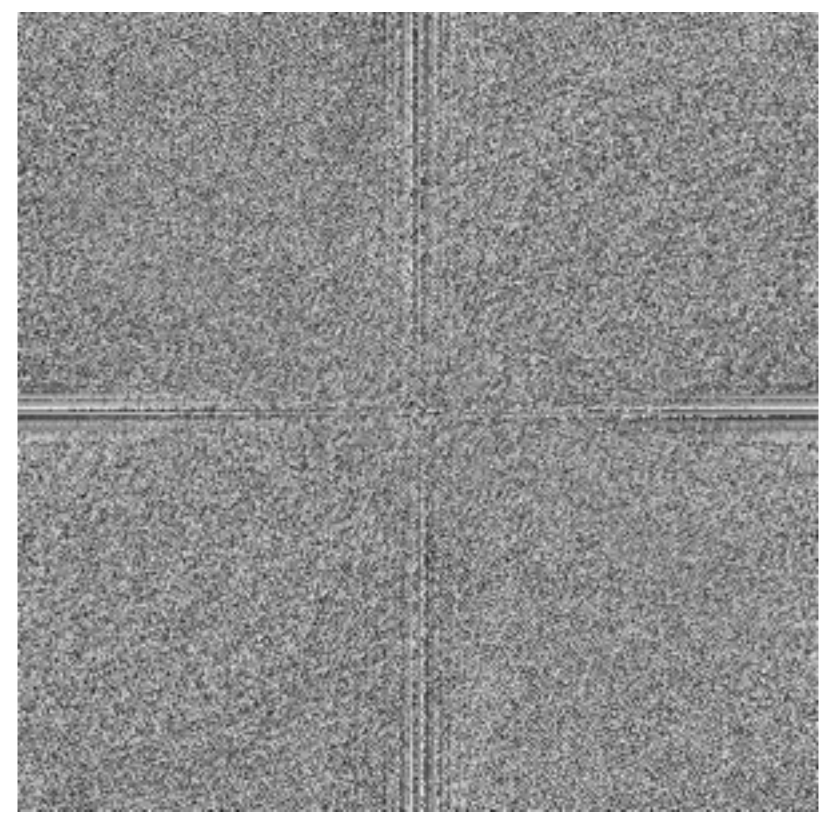
$$f(a,b) = \frac{1}{N^2} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} F(k,l)\, e^{\iota 2\pi \left(\frac{ka}{N} + \frac{lb}{N}\right)}$$

Zebra

Fourier Transform



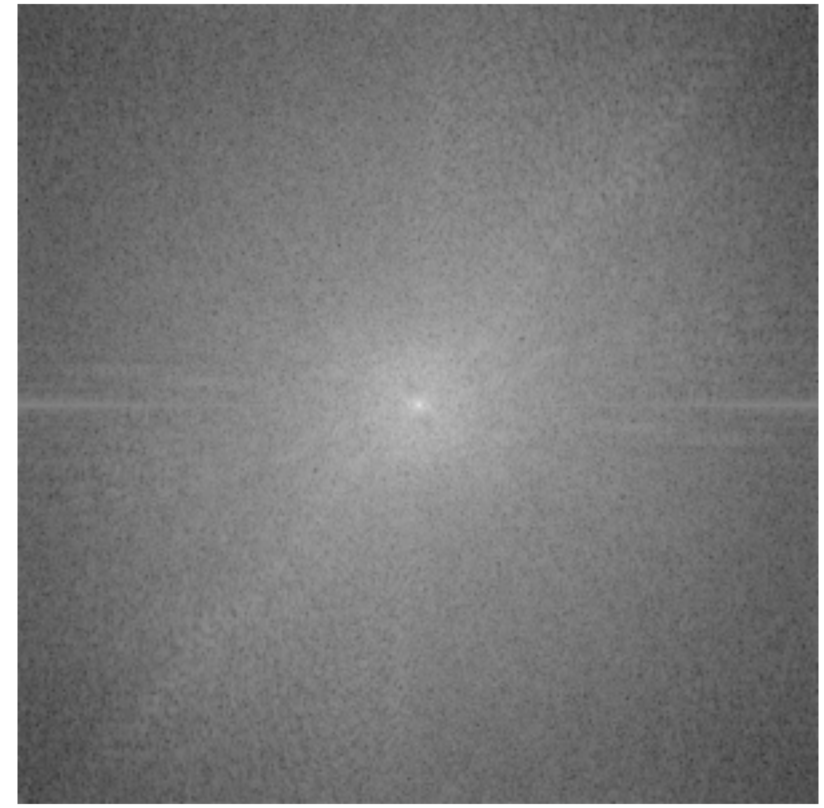
magnitude transform



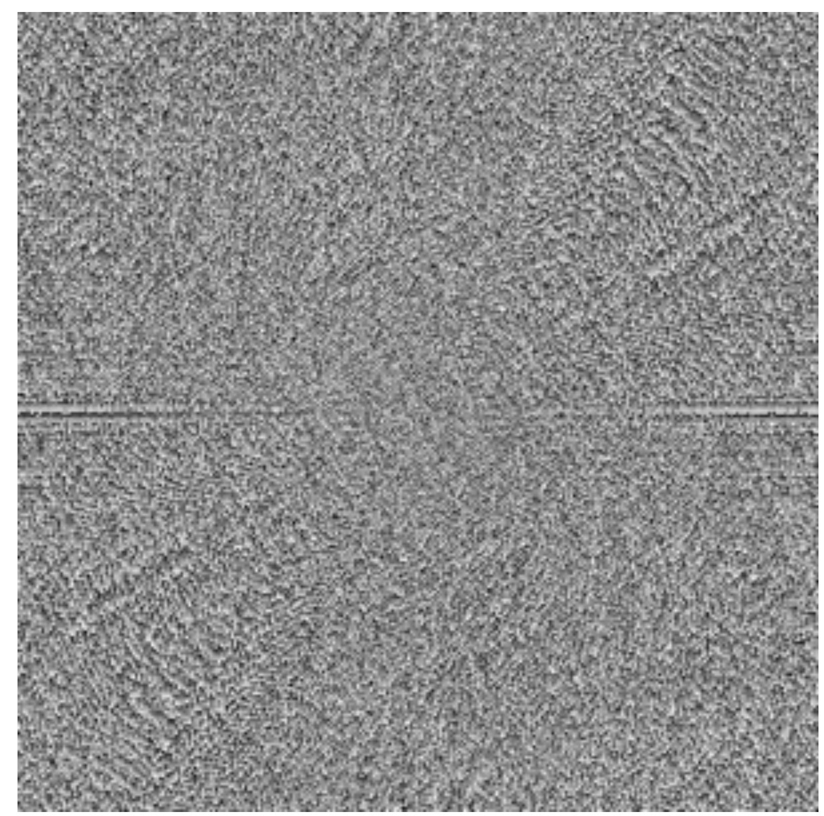
phase transform

Leopard

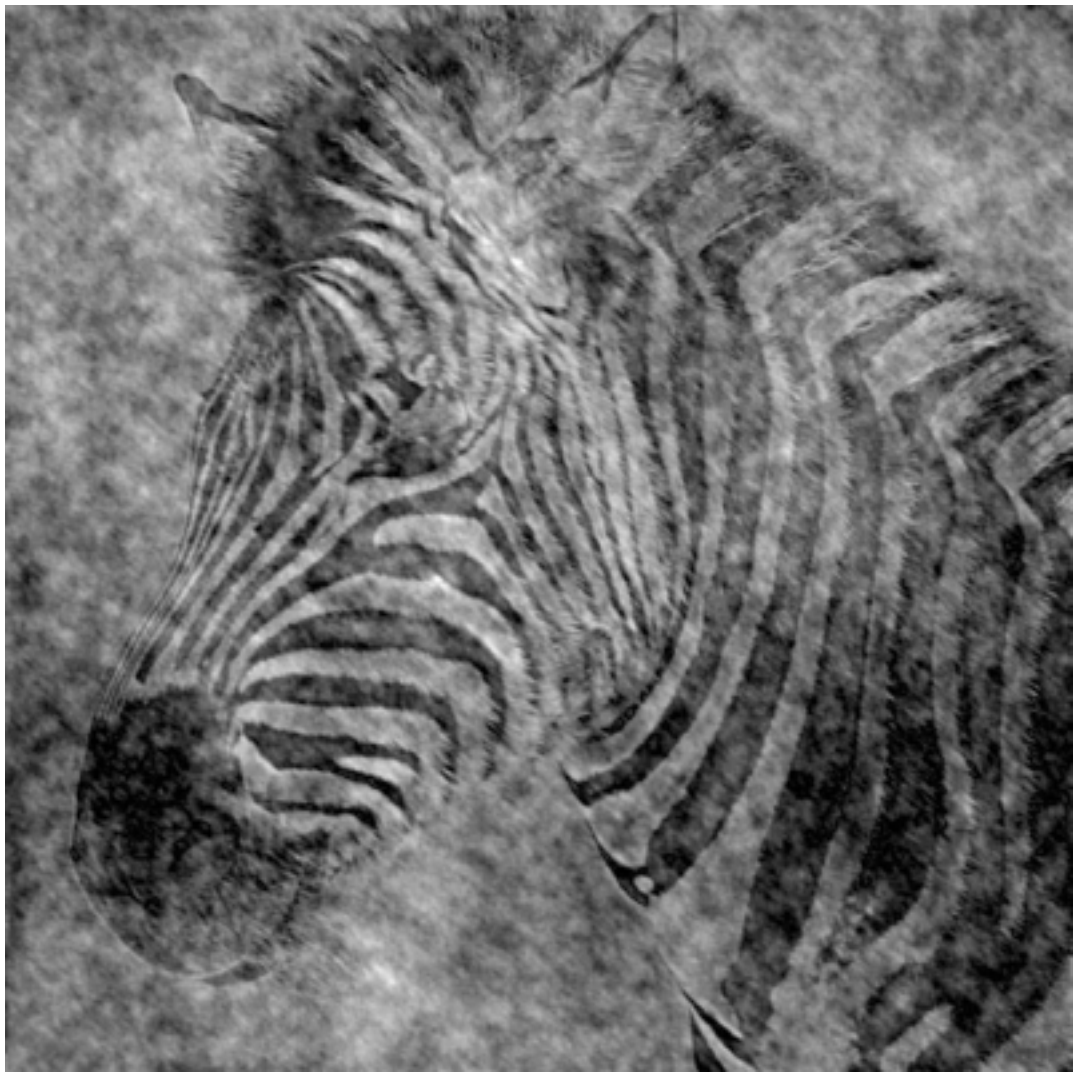
Fourier Transform



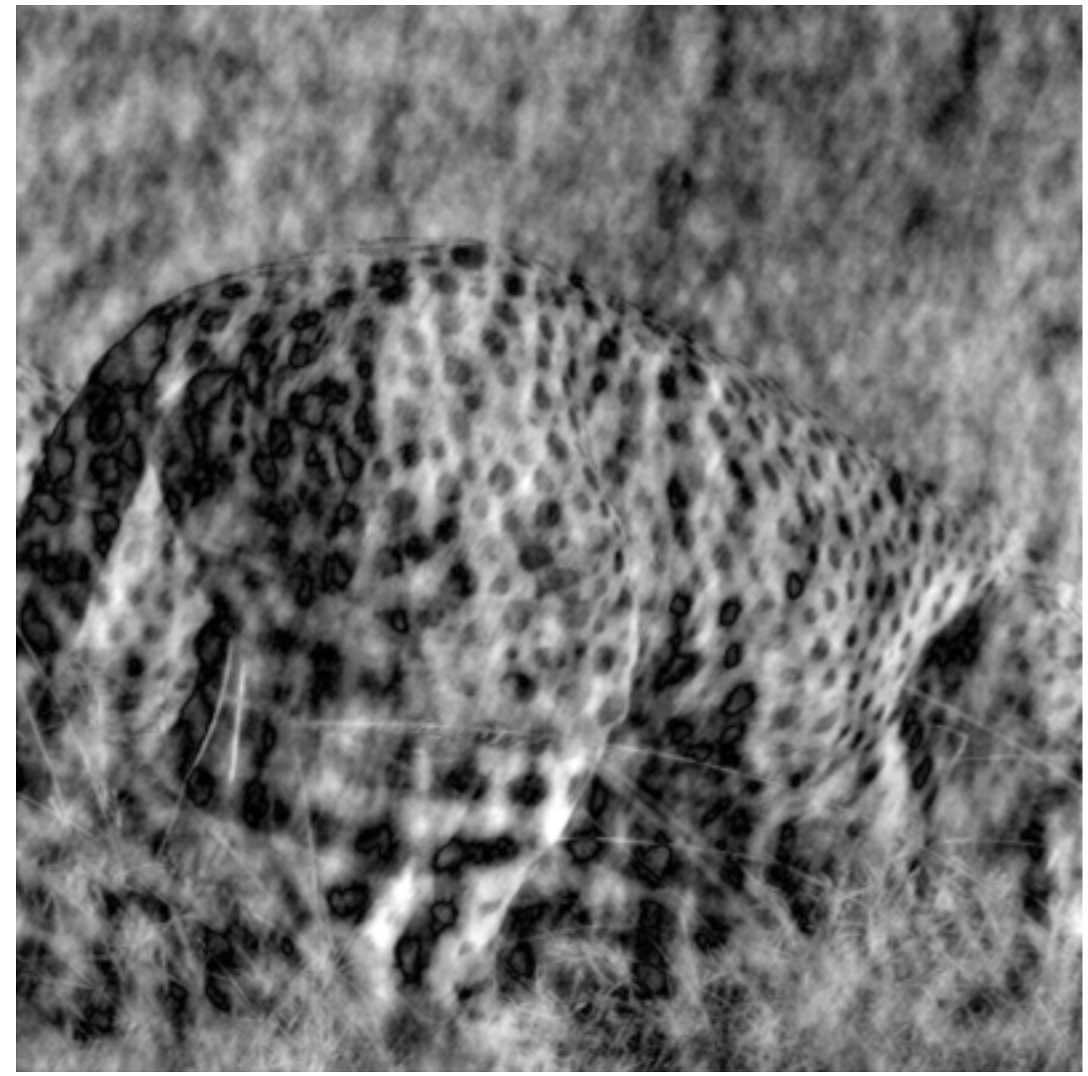
magnitude transform



phase transform

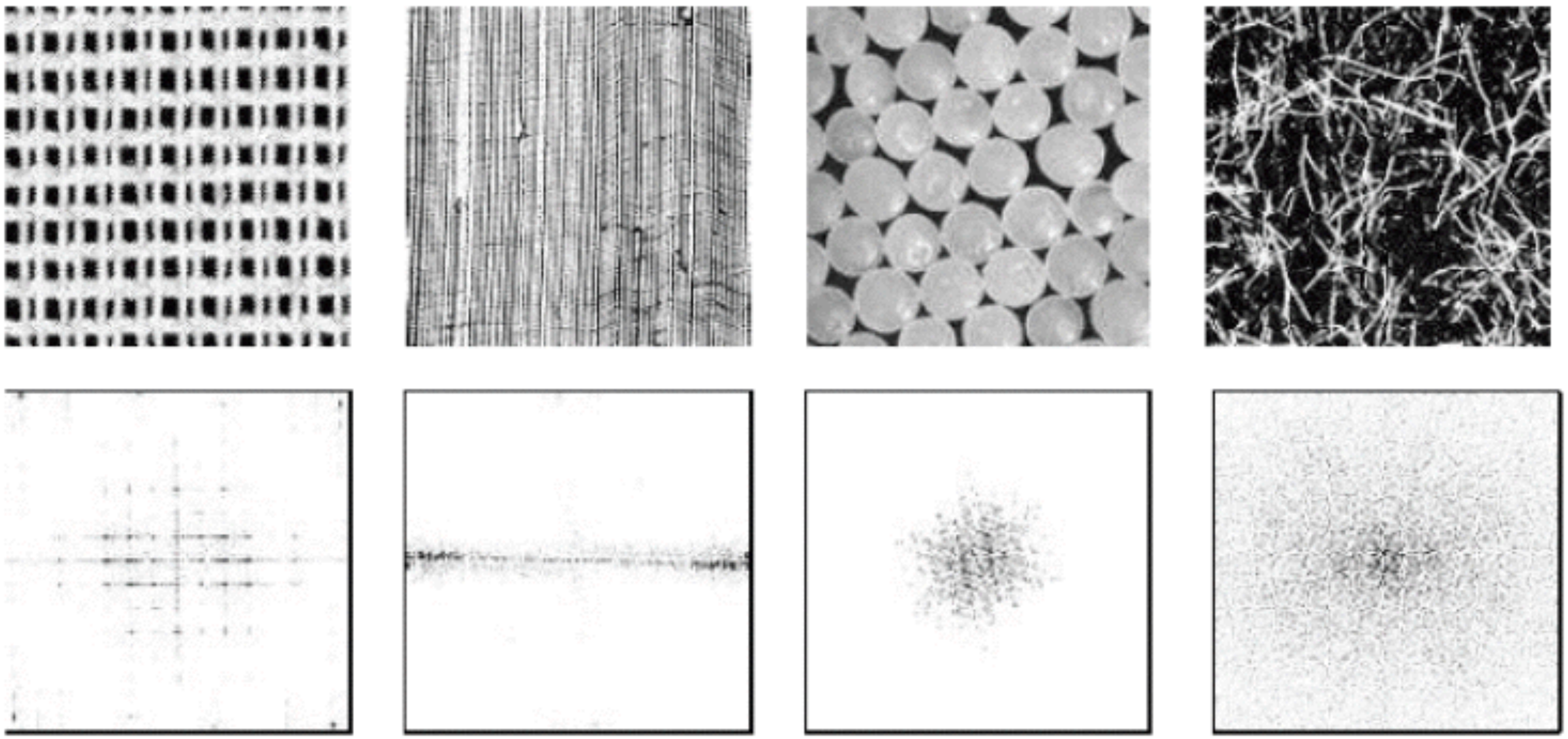


Zebra's phase
+ Leo's mag



Leo's phase
+ Zebra's mag

Natural Images and Their FT



(a) structured (b) oriented (c) granular (d) random

- What happened to the FT patterns when the texture scale and orientation are changed?

Frequency Domain Features

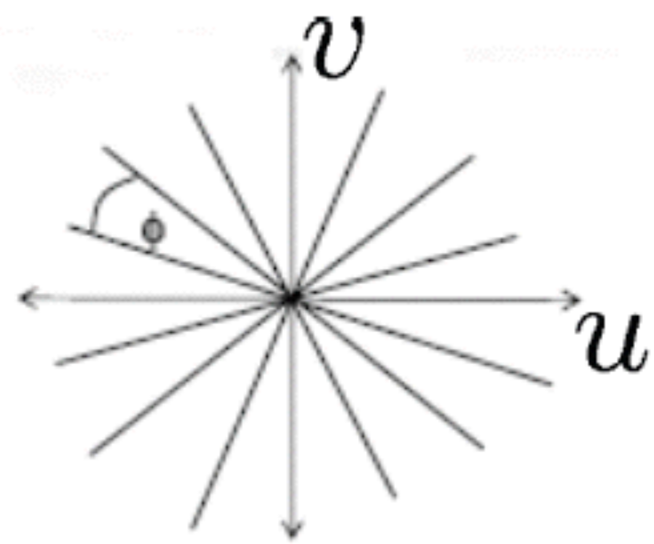
Fourier domain energy distribution

- Angular features (directionality)

$$V^{(a)}_{\theta_1\theta_2} = \int\int |F(u,v)|^2 du dv$$

where,

$$\theta_1 \leq \tan^{-1}[\frac{v}{u}] \leq \theta_2$$

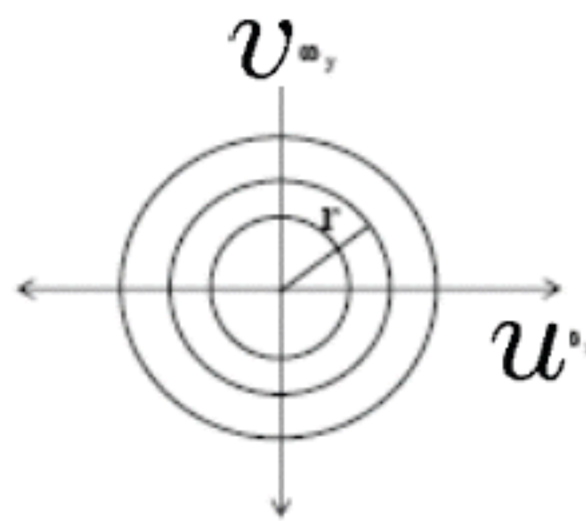


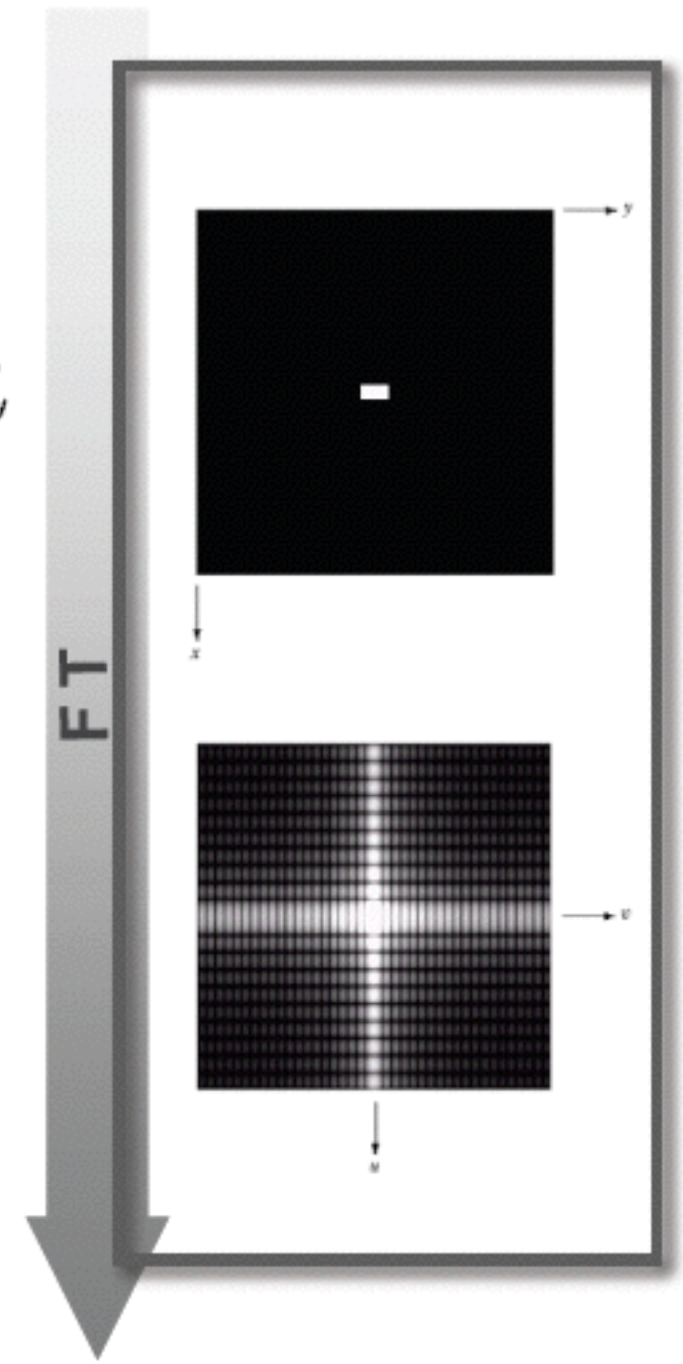
- Radial features (coarseness)

$$V^{(r)}_{r_1 r_2} = \int\int |F(u,v)|^2 du dv$$

where,

$$r_1 \leq u^2 + v^2 < r_2$$

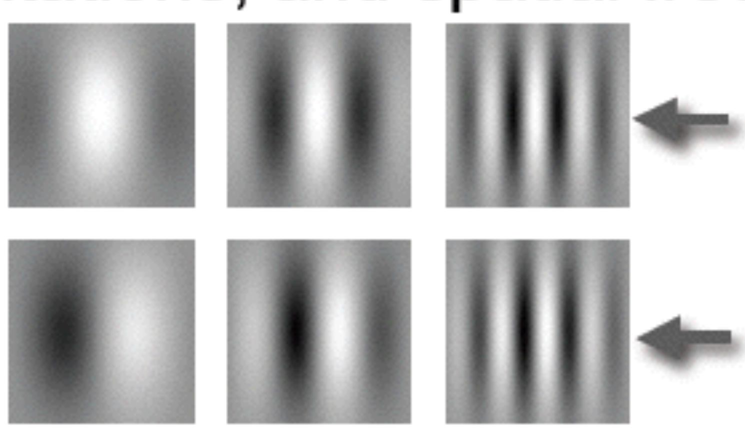




Uniform division may not be the best!!

Gabor Texture

- Fourier coefficients depend on the entire image (Global) → we lose spatial information
- Objective: local spatial frequency analysis
- Gabor kernels: looks like Fourier basis multiplied by a Gaussian
  - The product of a symmetric (even) Gaussian with an oriented sinusoid
  - Gabor filters come in pairs: symmetric and anti-symmetric (odd)
  - Each pair recover symmetric and anti-symmetric components in a particular direction
  - (k_x, k_y): the spatial frequency to which the filter responds strongly
  - σ : the scale of the filter. When σ = infinity, similar to FT
- We need to apply a number of Gabor filters are different scales, orientations, and spatial frequencies

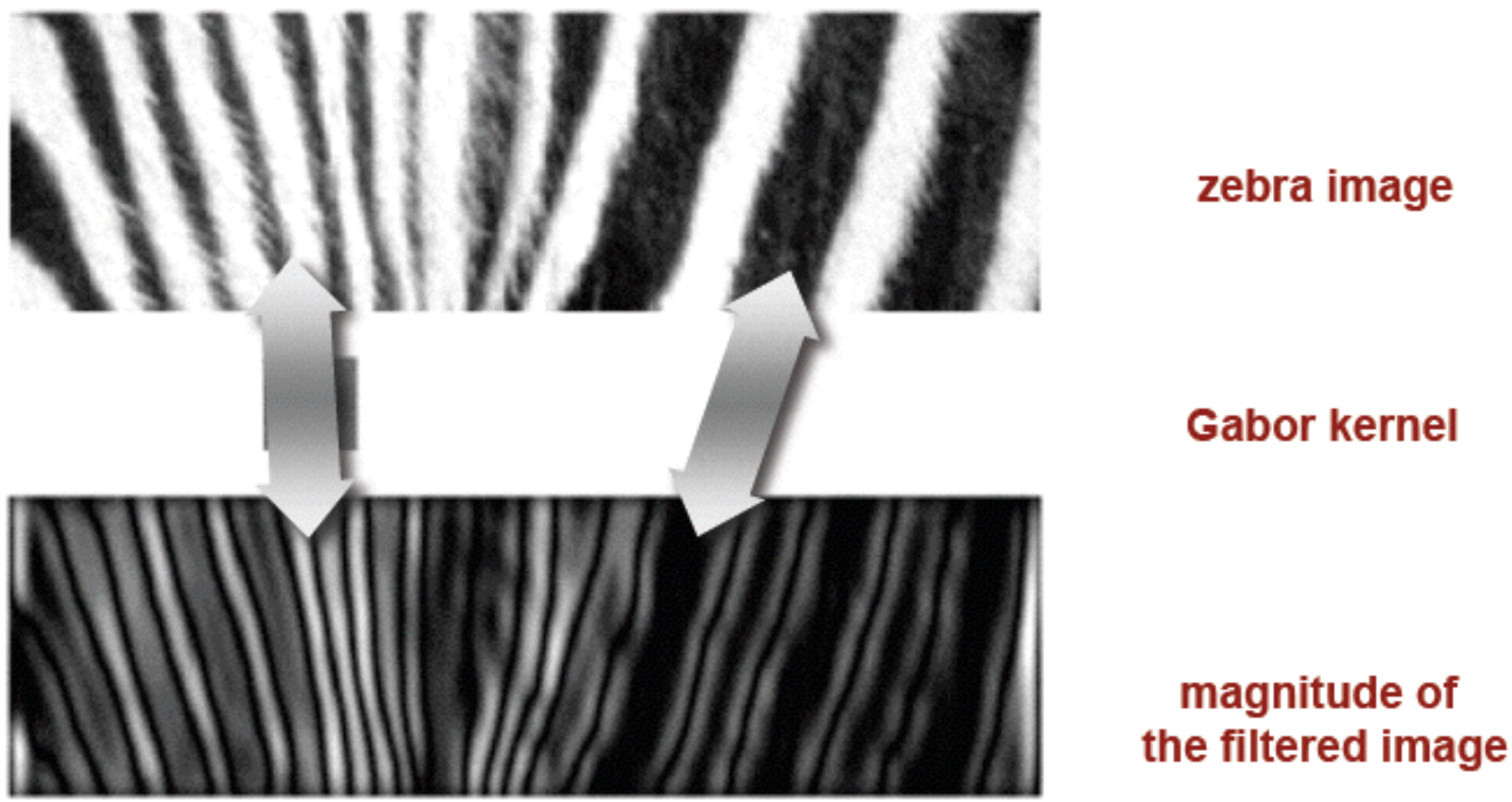


$$G_{symmetric}(x,y) = \cos(k_x x + k_y y) \exp -\frac{x^2+y^2}{2\sigma^2}$$

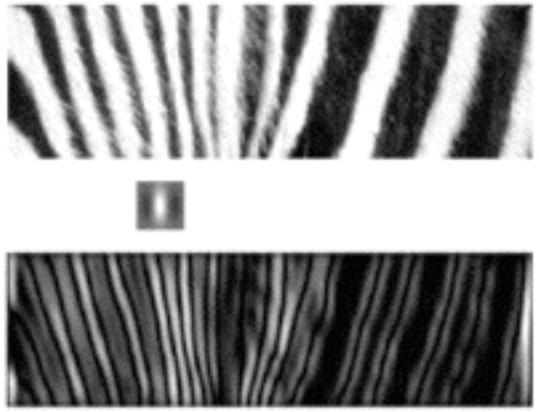
$$G_{anti-symmetric}(x,y) = \sin(k_x x + k_y y) \exp -\frac{x^2+y^2}{2\sigma^2}$$

Example – Gabor Kernel

- Zebra stripes at different scales and orientations and convolved with the Gabor kernel
- The response falls off when the stripes are larger or smaller
- The response is large when the spatial frequency of the bars roughly matches the windowed by the Gaussian in the Gabor kernel
- Local spatial frequency analysis



Gabor Texture (cont.)



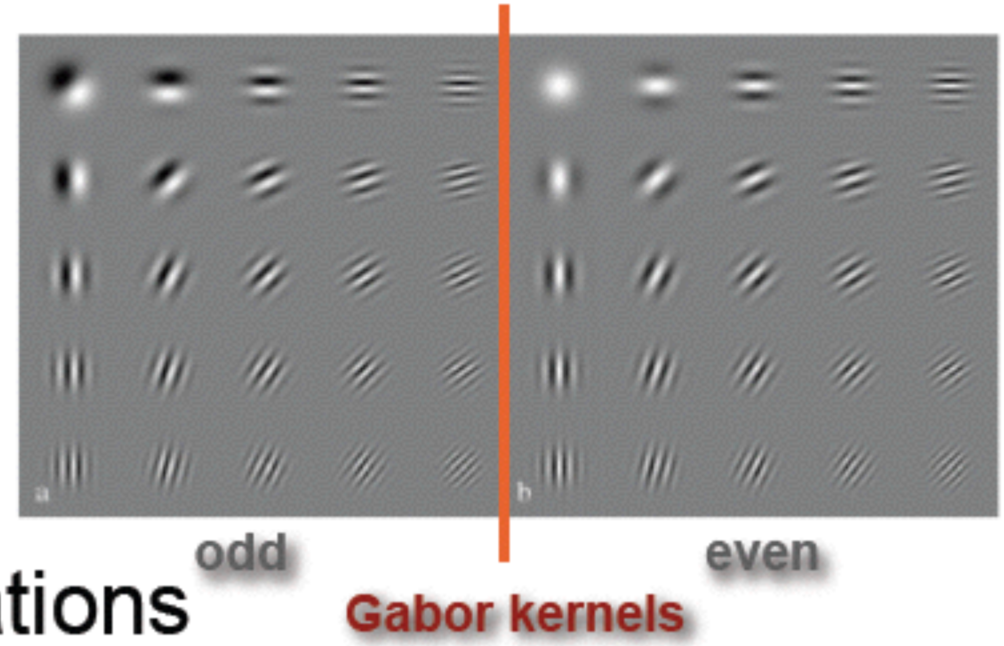
- Image *I(x,y)* convoluted with Gabor filters h_{mn} (totally M x N)

$$W_{mn}(x,y) = \int I(x_1,y_1) h_{mn}(x-x_1, y-y_1) dx_1 dy_1$$

- Using first and 2nd moments for each scale and orientations

$$\mu_{mn} = \int\int |W_{mn}(x,y)| dx dy$$

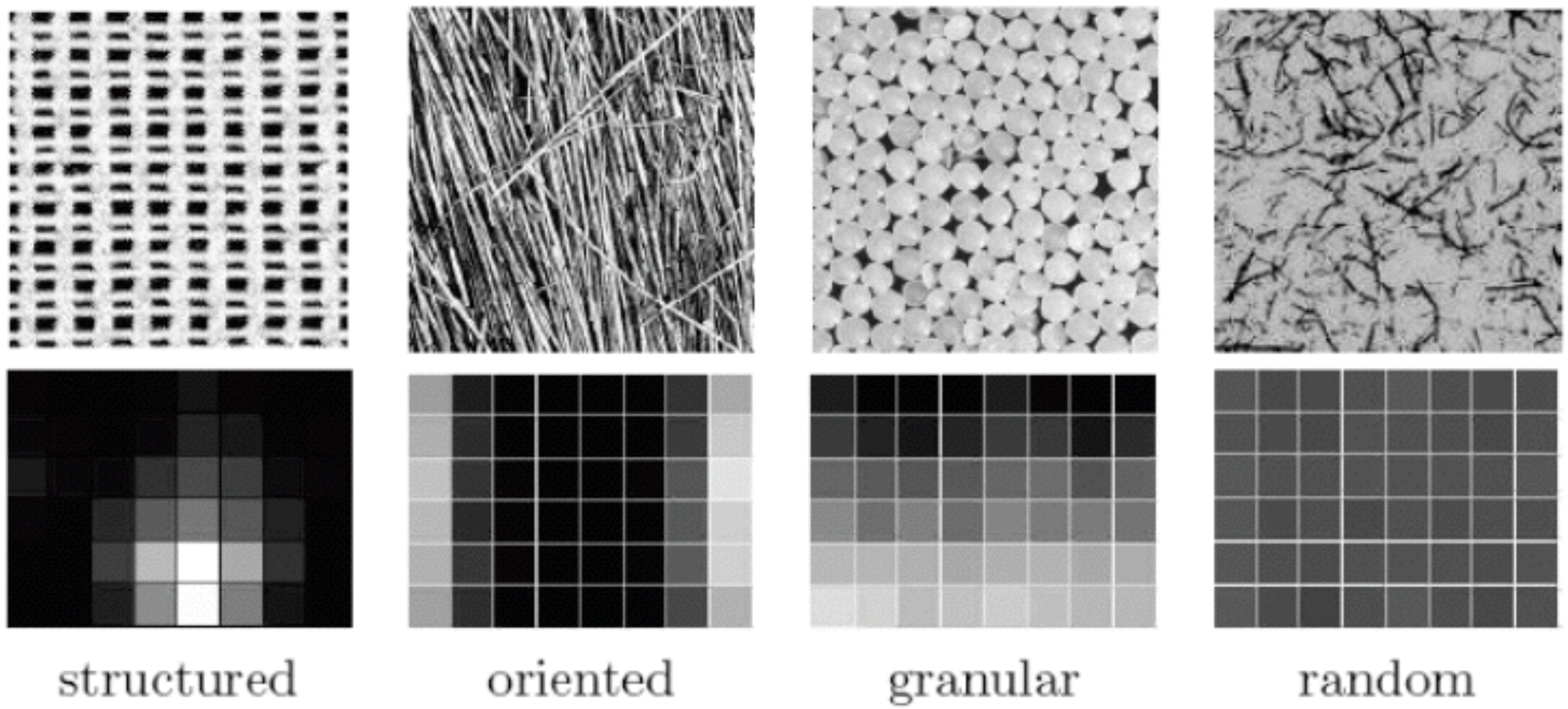
$$\sigma_{mn} = \sqrt{\int\int (|W_{mn}(x,y)| - \mu_{mn})^2 dx dy}$$



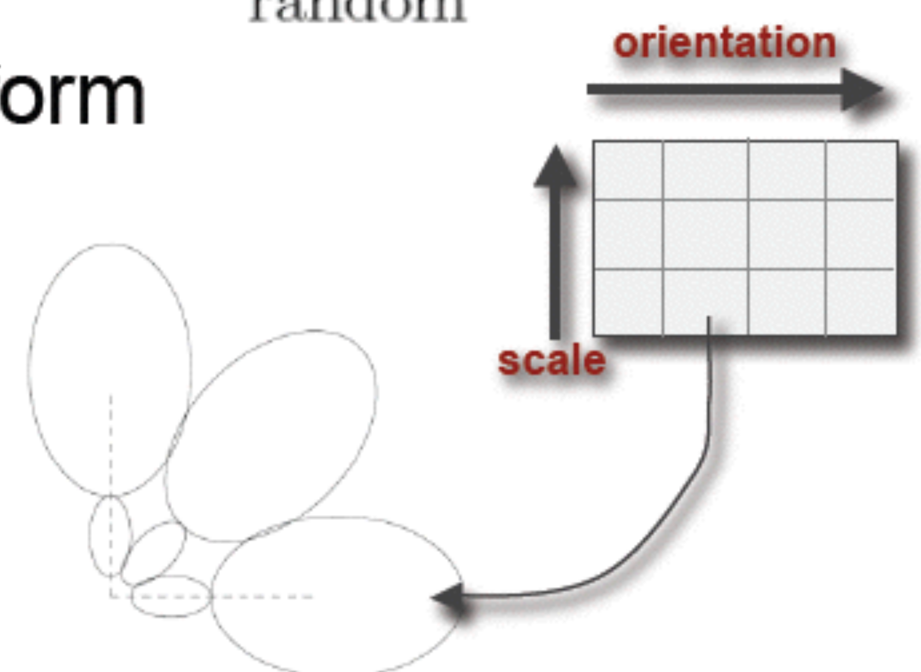
- Features: e.g., 4 scales, 6 orientations → 48 dimensions

$$\bar{v} = [\mu_{00}, \sigma_{00}, \mu_{01}, \ldots, \mu_{35}, \sigma_{35}]$$

Gabor Texture (cont.)

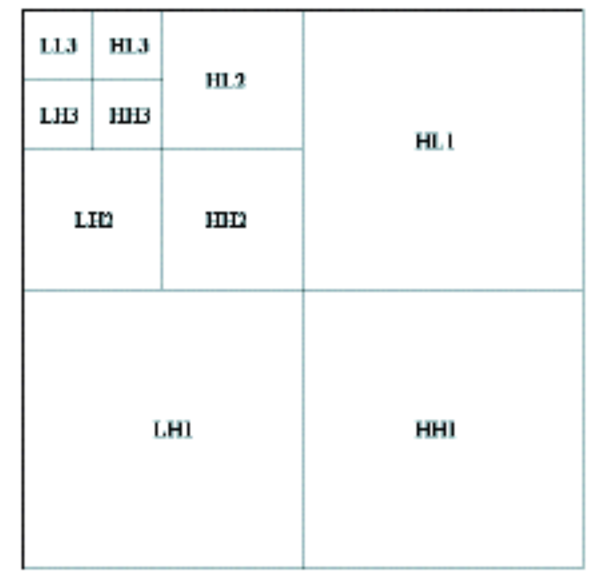


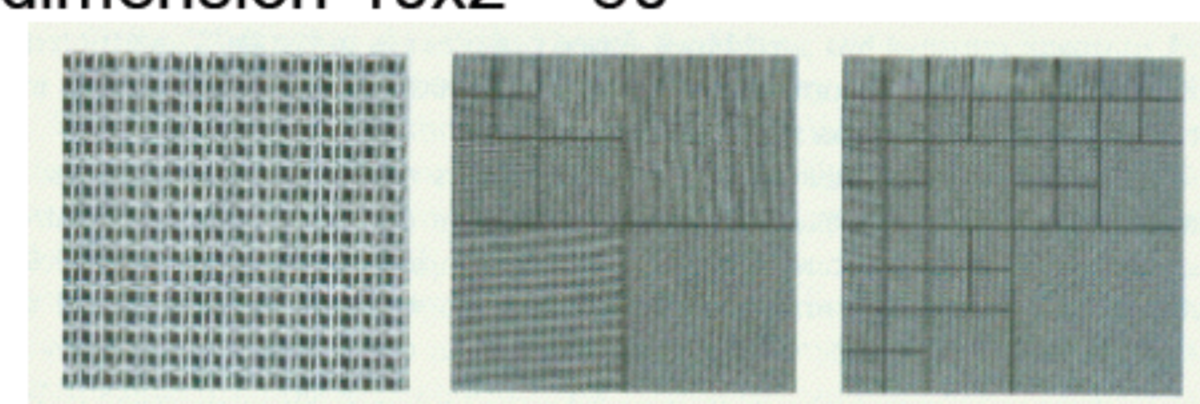
- Arranging the mean energy in a 2D form
  - structured: localized pattern
  - oriented (or directional): column pattern
  - granular: row pattern
  - random: random pattern



Wavelet Features (PWT, TWT)

- Wavelet
  - Decomposition of signal with a family of basis functions with recursive filtering and sub-sampling
  - Each level, decomposes 2D signal into 4 subbands, LL, LH, HL, HH (L=low, H=high)
- PWT: pyramid-structured wavelet transform
  - Recursively decomposes the LL band
  - Feature dimension (3x3x1+1)x2 = 20
- TWT: pyramid-structured wavelet transform
  - Some information in the middle frequency channels
  - Feature dimension 40x2 = 80



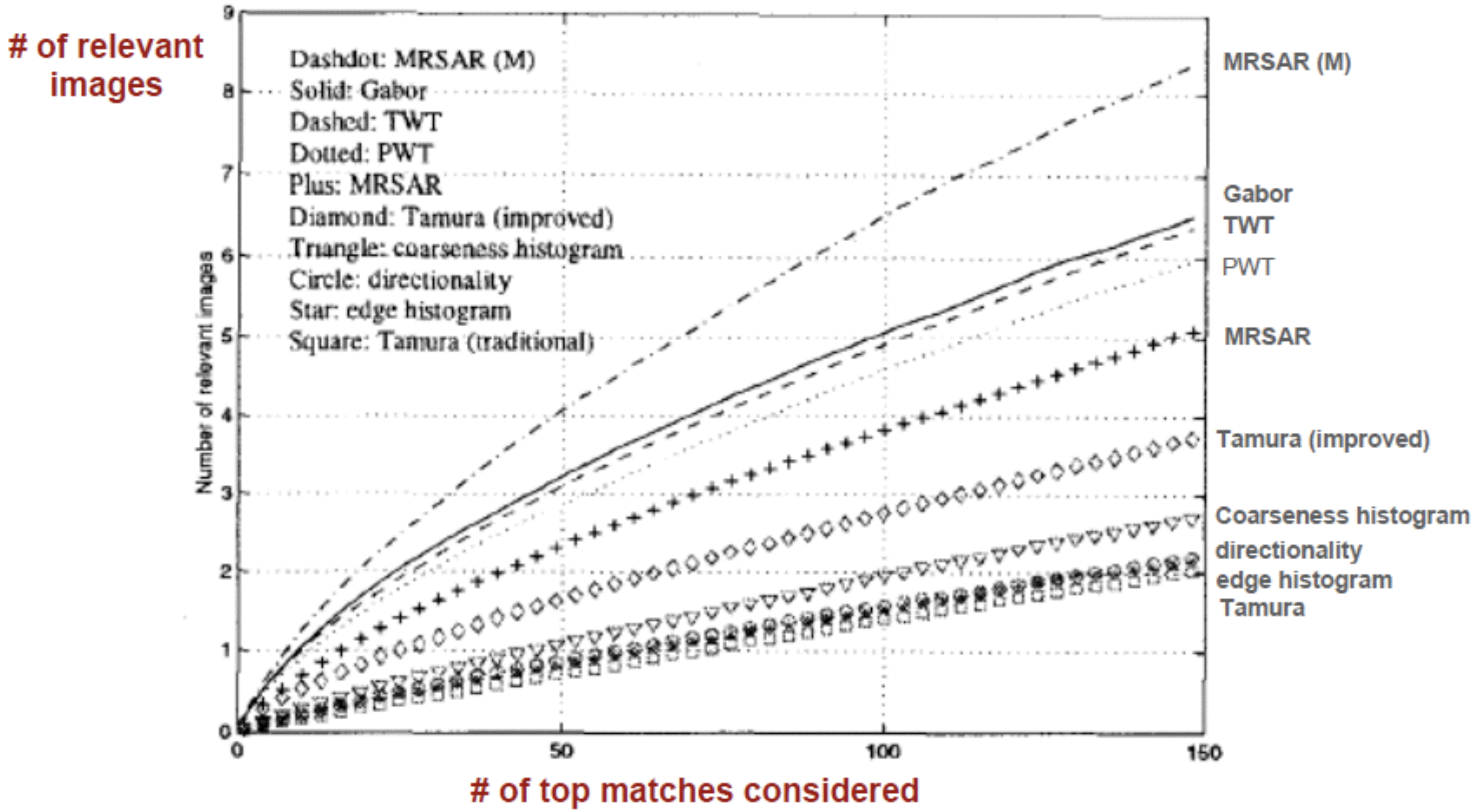


original image PWT TWT

Texture Comparisons

[Ma'98]

- Retrieval performance of different texture features according to the number of relevant images retrieved at various scopes using Corel Photo galleries



Texture directionality

- Gradient:

-1	0	1
-1	0	1
-1	0	1

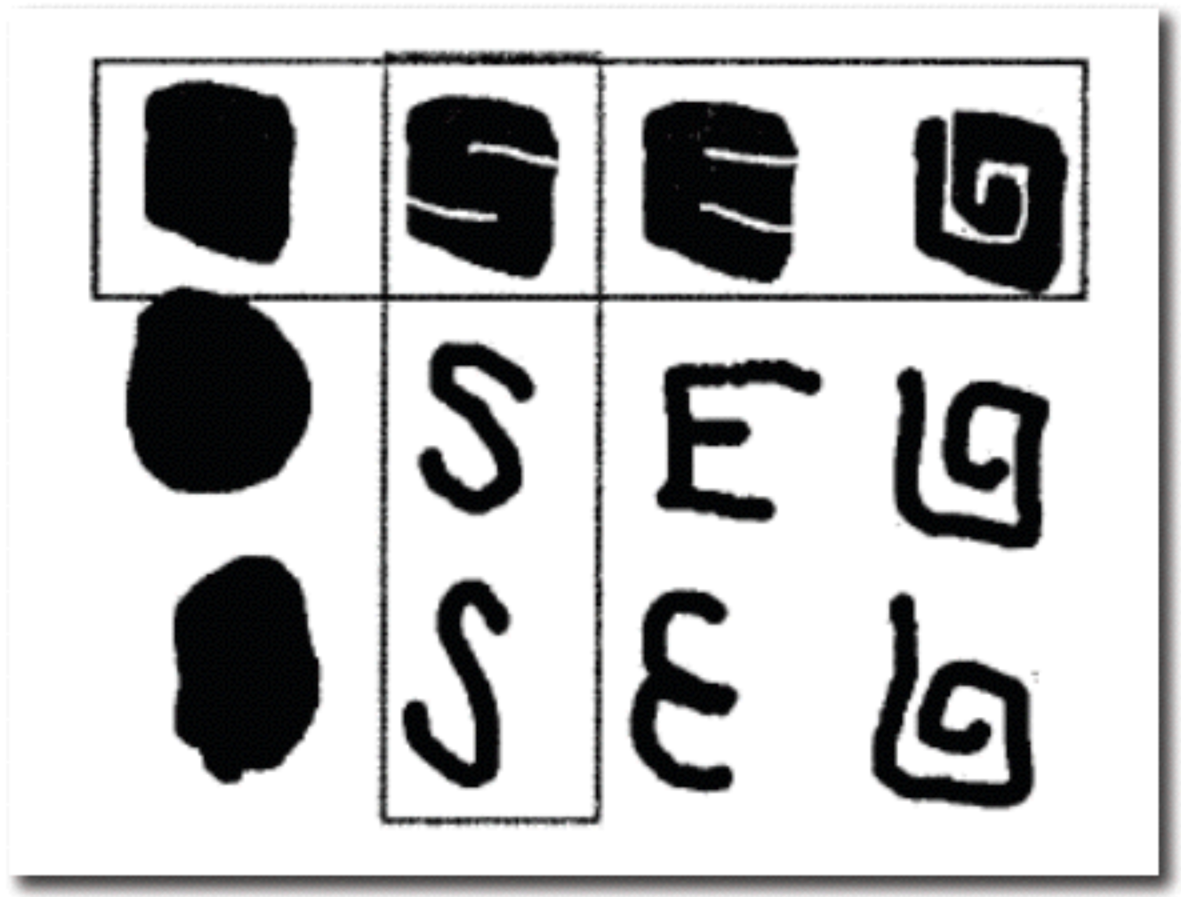
1	1	1
0	0	0
-1	-1	-1

Image shape features

- Shape features are computed out based on object segments or regions, mainly including
  - contour features
  - and regions features.

- Typical approaches include
  - Fourier shape description
  - Moment invariants

Region-based vs. Contour-based Descriptor



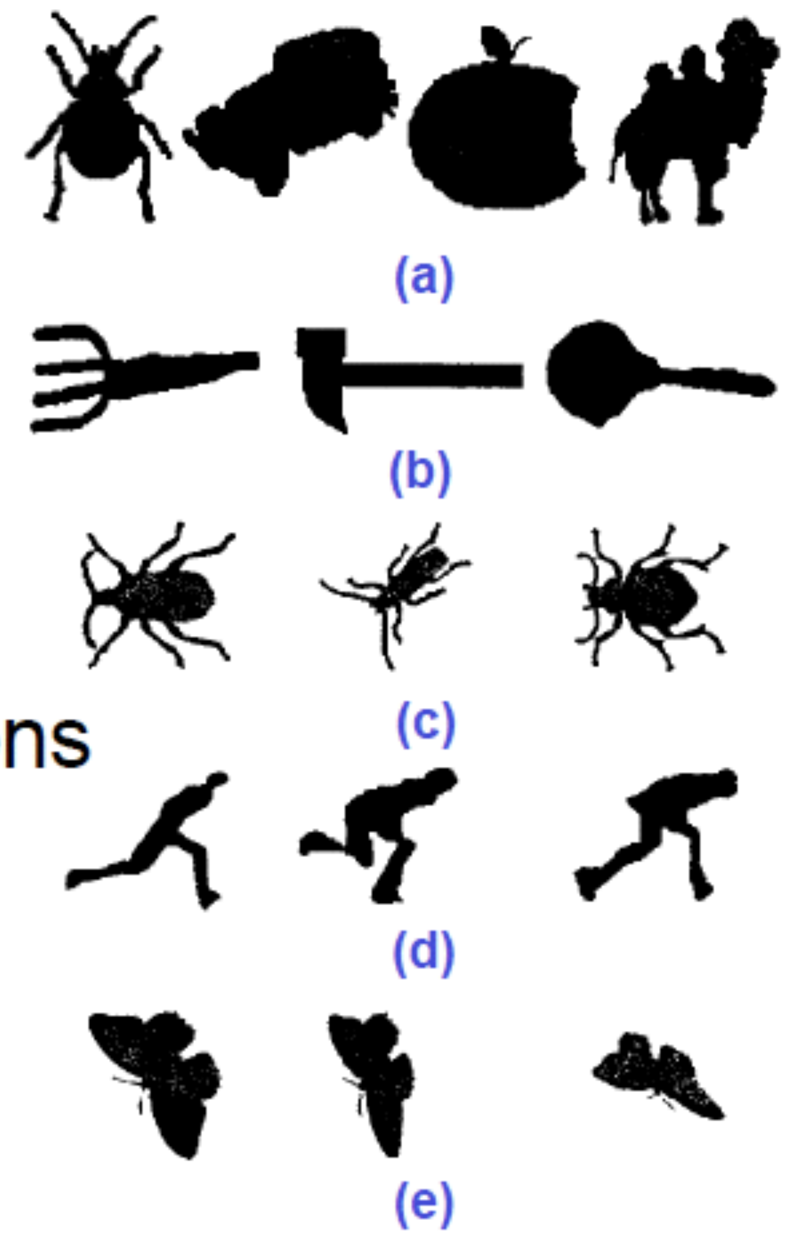
- Columns indicate contour similarity
  - Outline of contours
- Rows indicate region similarity
  - Distribution of pixels

Region-based Descriptor

- Express pixel distribution within a 2D object region
- Employs a complex 2D Angular Radial Transformation (ART)
  - 35 fields each of 4 bits
- Rotational and scale invariance
- Robust to some non-rigid transformation
- L_1 metric on transformed coefficients
- Advantages
  - Describing complex shapes with disconnected regions
  - Robust to segmentation noise
  - Small size
  - Fast extraction and matching

Contour-based Descriptor

- It's based on Curvature (曲率) Scale-Space (CSS) representation
- Found to be superior to
  - Zernike moments
  - ART
  - Fourier-based
  - Turning angles
  - Wavelets
- Rotational and scale invariance
- Robust to some non-rigid transformations
- For example
  - Applicable to (a)
  - Discriminating differences in (b)
  - Finding similarities in (c)-(e)



Problems in Shape-based Indexing

Many existing approaches assume

- Segmentation is given
- Human operator circle object of interest
- Lack of clutter and shadows
- Objects are rigid
- Planar (2-D) shape models
- Models are known in advance

Dimensional reduction for image features

In image retrieval system, increasing feature dimension can enhance precision of retrieval greatly. However, high feature dimension will lead to high computation cost. Hence it is important to reduce the redundant in feature data.

- Image feature space reduction
  - Linear dimensional reduction techniques: PCA …
  - Nonlinear dimensional reduction techniques: Isomap, LLE …
  - Clustering based feature reduction methods
- High-dimensional feature indexing
  - Database oriented high-dimensional data indexing
    - Bucketing grouping searching techniques，K-d tree, R tree …
  - Clustering methods
  - SOM

Image similarities

- How to measure similarity of different images base on features?
  - Image features always form into a fixed-length feature vector.
  - The similarity therefore can be measure by
    - Euclidian distance
    - Histogram intersection
    - Quadratic distance
    - Mahalanobis distance (马氏距离)
    - Non-geometrical similarity

Similarity and distance

- Similarity: ↑

- distance: ↓

Practical image retrieval systems

- QBIC (Query By Image Content)
  - http://www.qbic.almaden.ibm.com/
- Virage
  - http://wwwvirage.com/cgi-bin/query-e
- RetrievalWare
  - http://vrw.excalib.com/cgi-bin/sdk/cst/cst2.bat
- Photobook
- MARS
  - http://jadzia.ifp.uiuc.edu:8000

Practical image retrieval systems (cont.)

- Most existing image retrieval systems have one or more of following functions features:
  - –Random browsing
  - –Classified browsing
  - –Example based retrieval
  - –Sketch based retrieval
  - –Texture based retrieval

Future of image retrieval

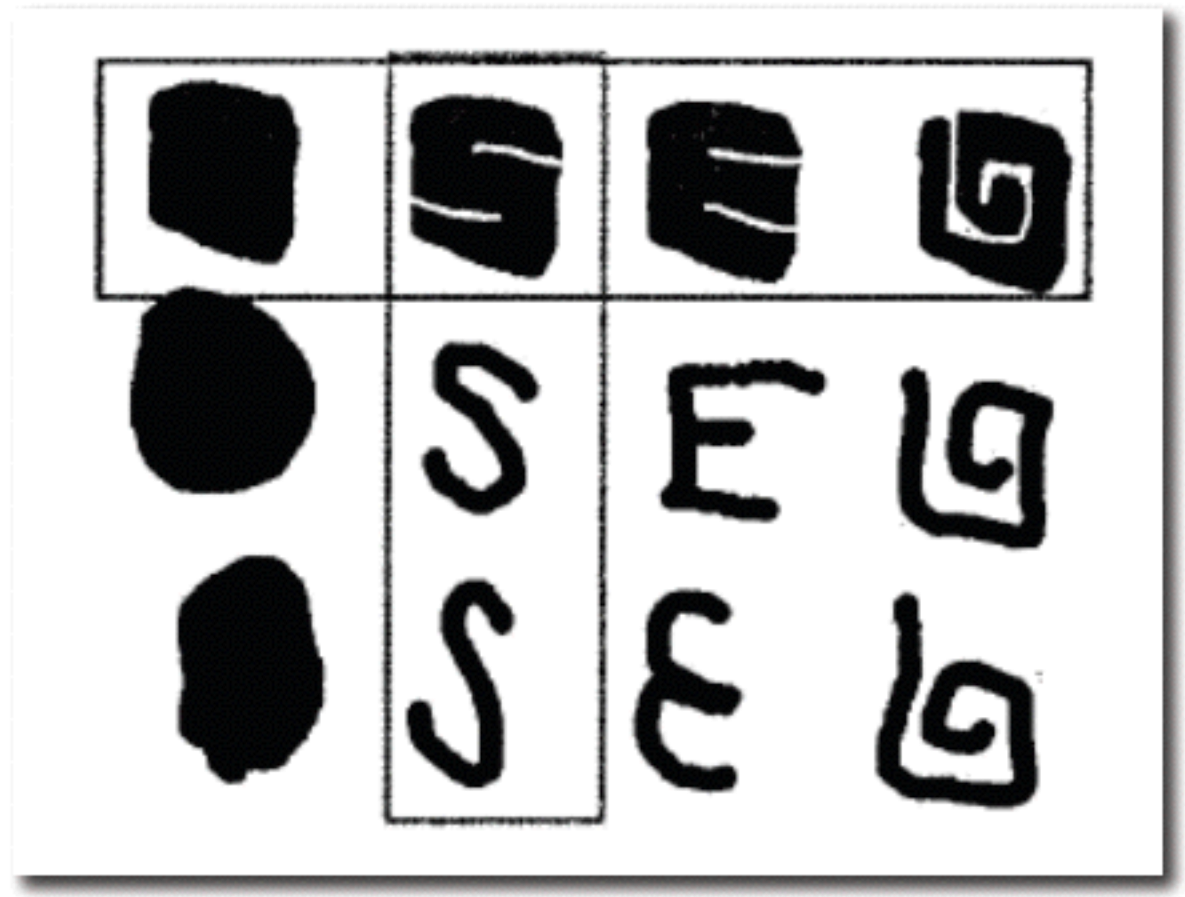
- Human-computer interaction
- Semantic speech
- Web-oriented
- High dimensional data
- Perspective
- Multiple media channels
- Image feature mapping
- Standards of performance measurements
- Construction of test sets

Image shape features

- Shape features are computed out based on object segments or regions, mainly including
  - contour features
  - and regions features.

- Typical approaches include
  - Fourier shape description
  - Moment invariants

Region-based vs. Contour-based Descriptor



- Columns indicate contour similarity
  - Outline of contours
- Rows indicate region similarity
  - Distribution of pixels

Region-based Descriptor

- Express pixel distribution within a 2D object region
- Employs a complex 2D Angular Radial Transformation (ART)
  - 35 fields each of 4 bits
- Rotational and scale invariance
- Robust to some non-rigid transformation
- L_1 metric on transformed coefficients
- Advantages
  - Describing complex shapes with disconnected regions
  - Robust to segmentation noise
  - Small size
  - Fast extraction and matching

Contour-based Descriptor

- It's based on Curvature (曲率) Scale-Space (CSS) representation
- Found to be superior to
  - Zernike moments
  - ART
  - Fourier-based
  - Turning angles
  - Wavelets
- Rotational and scale invariance
- Robust to some non-rigid transformations
- For example
  - Applicable to (a)
  - Discriminating differences in (b)
  - Finding similarities in (c)-(e)

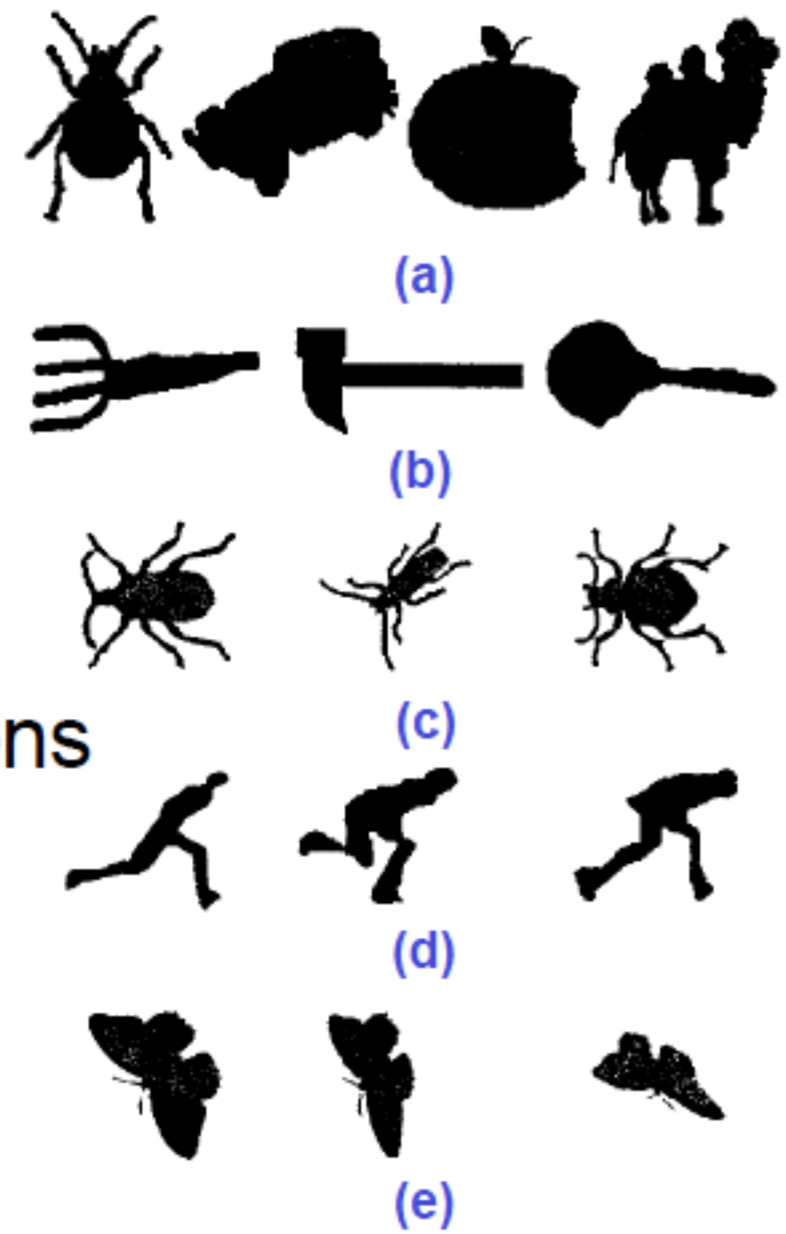
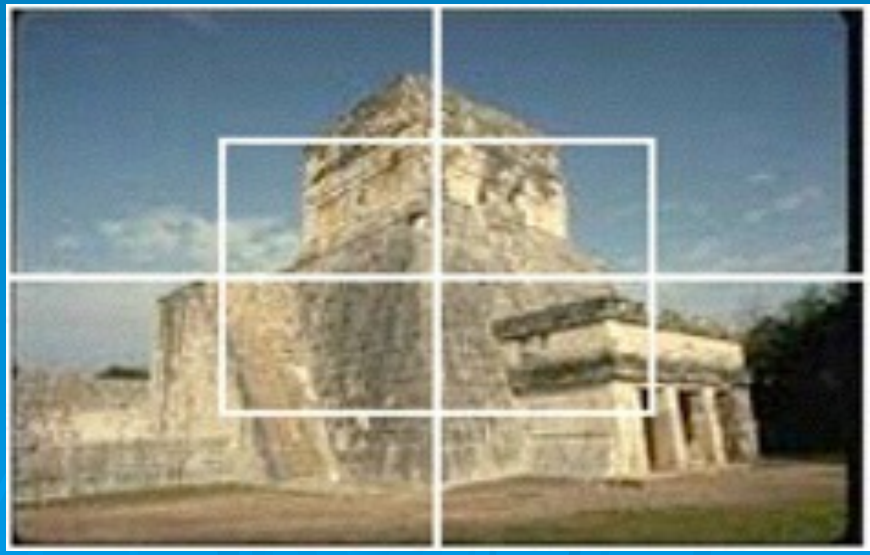


Image Retrieval Phase (cont.)

- Query by color anglogram (cont.)
  - Convert RGB to HSV [wikipedia]

  - Global and sub-image histogram forms LSI matrix.



[Zhao & Grosky 2002]

$\longrightarrow (x_1, x_2, ..., x_n)$

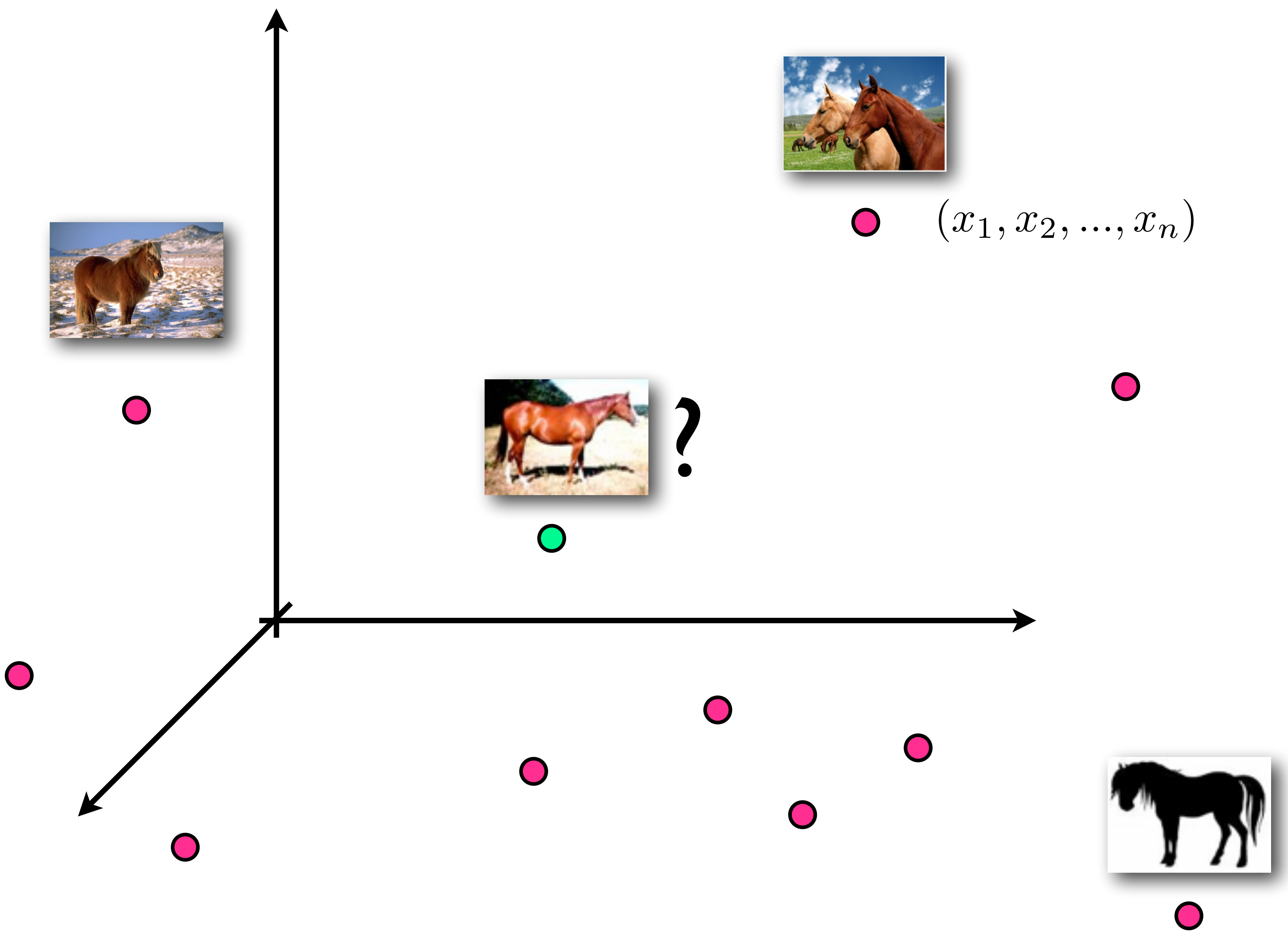
Image A

Feature
extraction

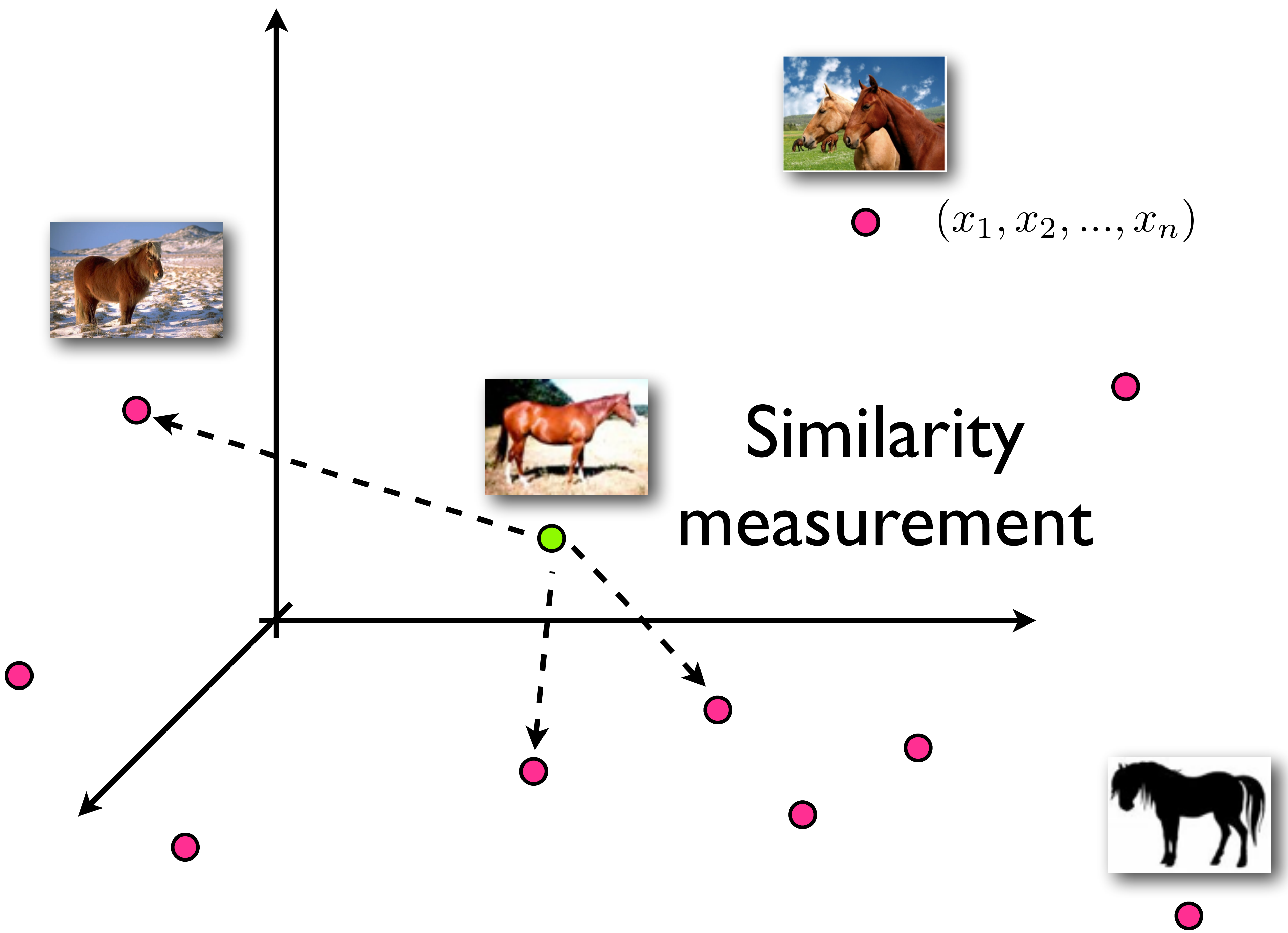
$\longrightarrow (y_1, y_2, ..., y_n)$

Image B

A geometrical view of CBIR



A geometrical view of CBIR



A geometrical view of CBIR

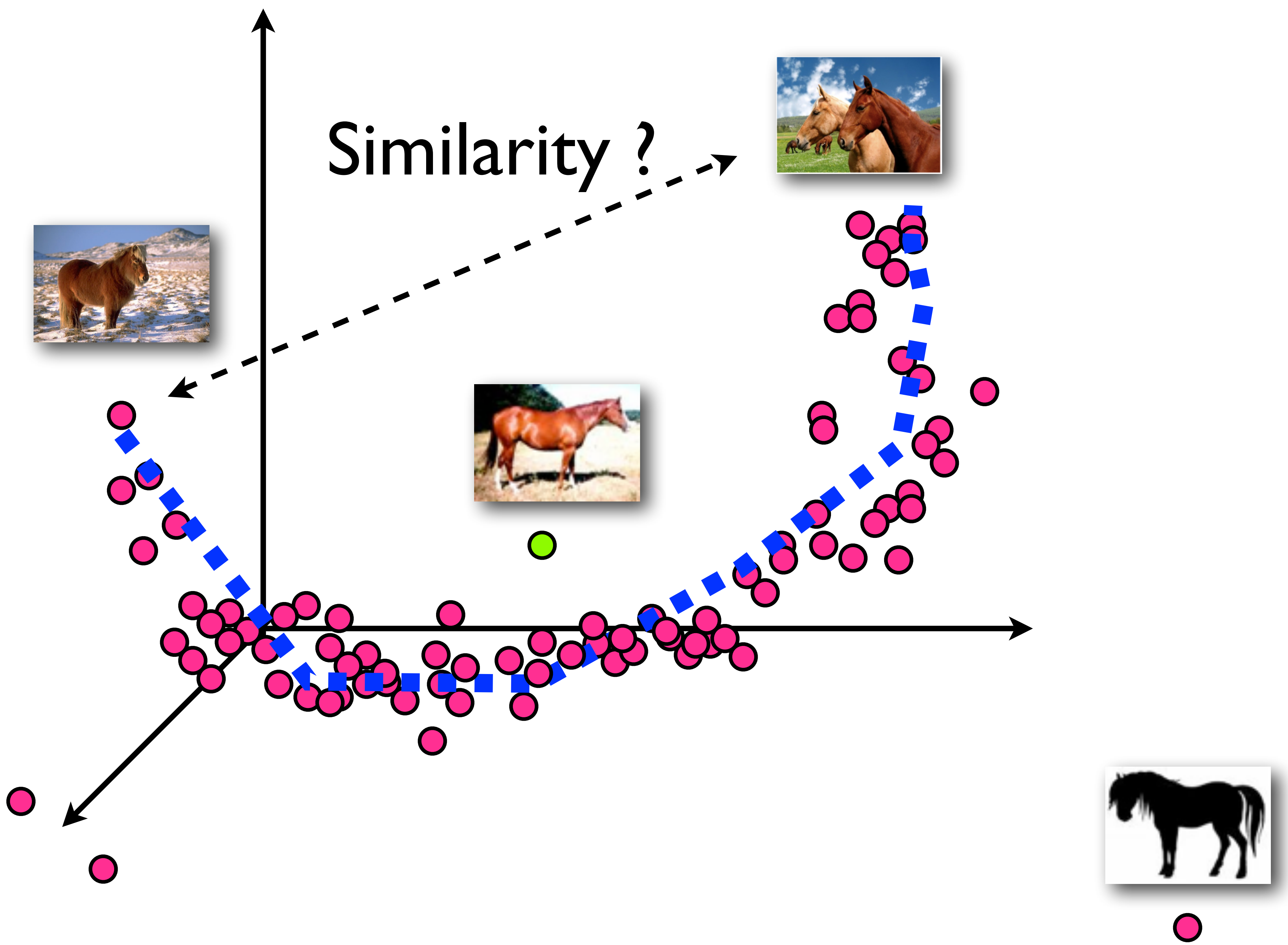


Image similarities

- How to measure similarity of different images base on features?
  - Image features always form into a fixed-length feature vector.
  - The similarity therefore can be measure by
    - Euclidian distance
    - Histogram intersection
    - Quadratic distance
    - Mahalanobis distance (马氏距离)
    - Non-geometrical similarity

Practical image retrieval systems

- QBIC (Query By Image Content)
  - http://www.qbic.almaden.ibm.com/
- Virage
  - http://wwwvirage.com/cgi-bin/query-e
- RetrievalWare
  - http://vrw.excalib.com/cgi-bin/sdk/cst/cst2.bat
- Photobook
- MARS
  - http://jadzia.ifp.uiuc.edu:8000

Practical image retrieval systems (cont.)

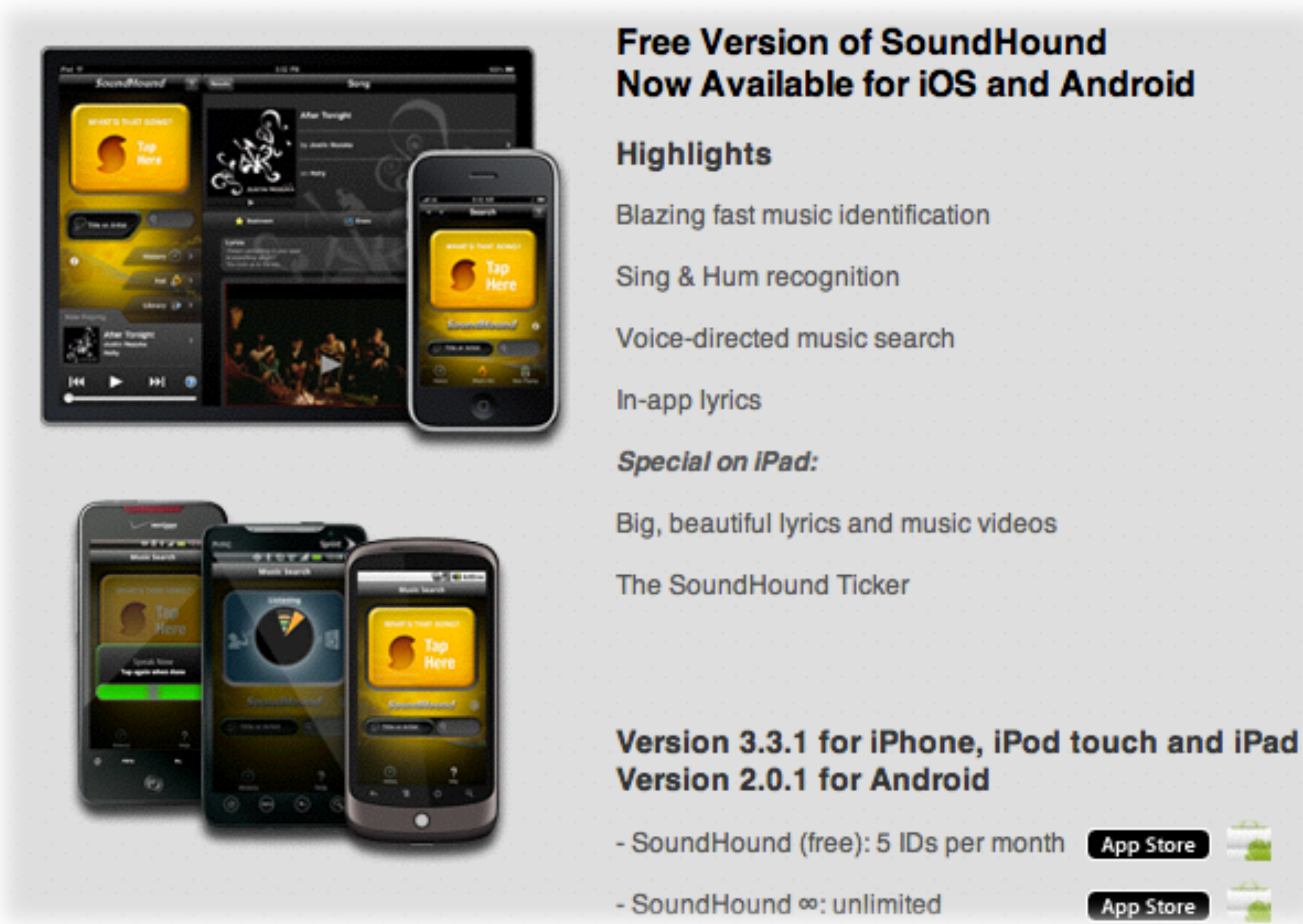
- Most existing image retrieval systems have one or more of following functions features:
  - –Random browsing
  - –Classified browsing
  - –Example based retrieval
  - –Sketch based retrieval
  - –Texture based retrieval

2. music retrieval techniques

Content based music retrieval

- 說明
  - 用聲音的內容為根據，做音樂的檢索
- 目的
  - 讓使用者可以用自然的方式點選歌曲

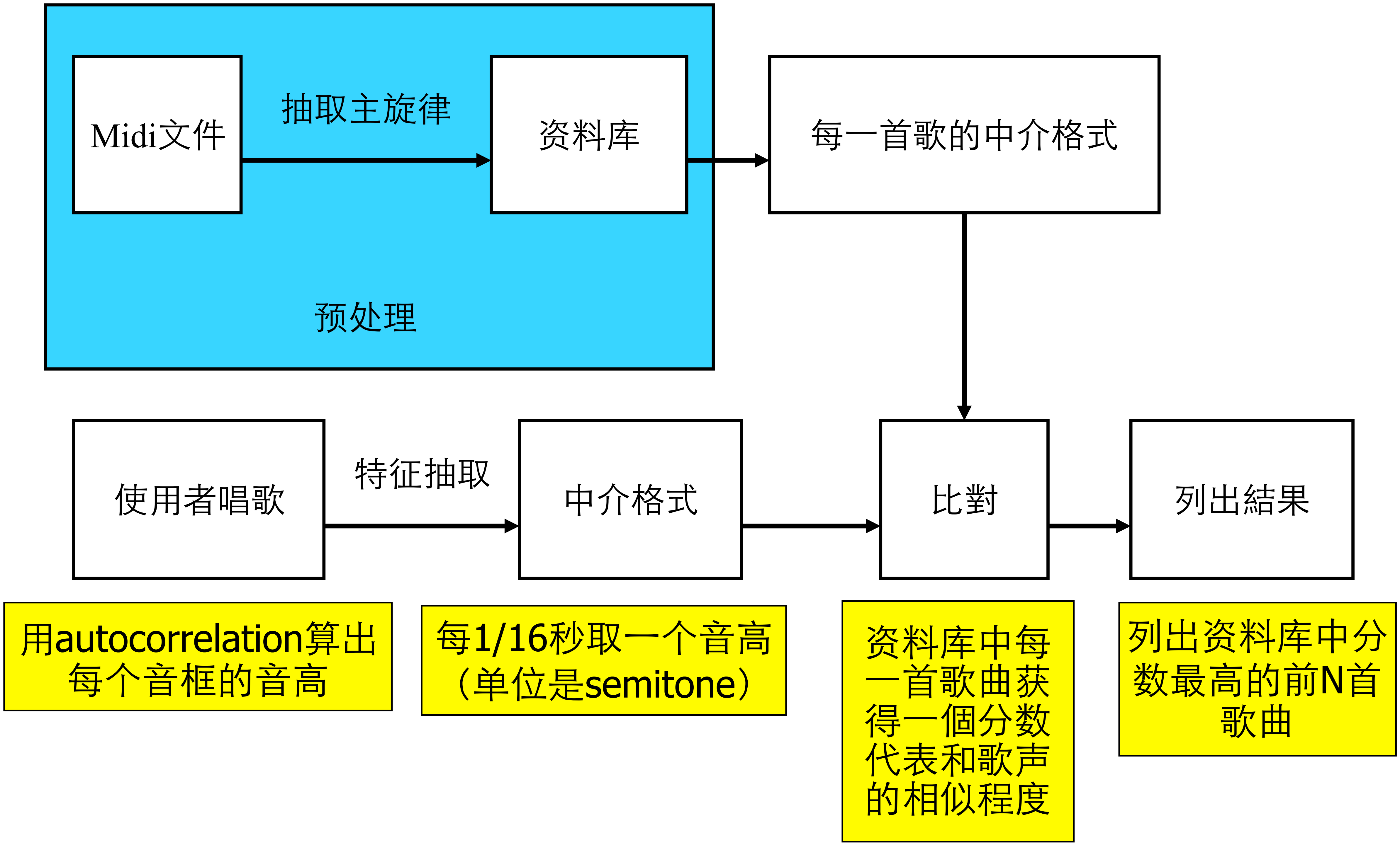
http://www.soundhound.com/



Content based music retrieval

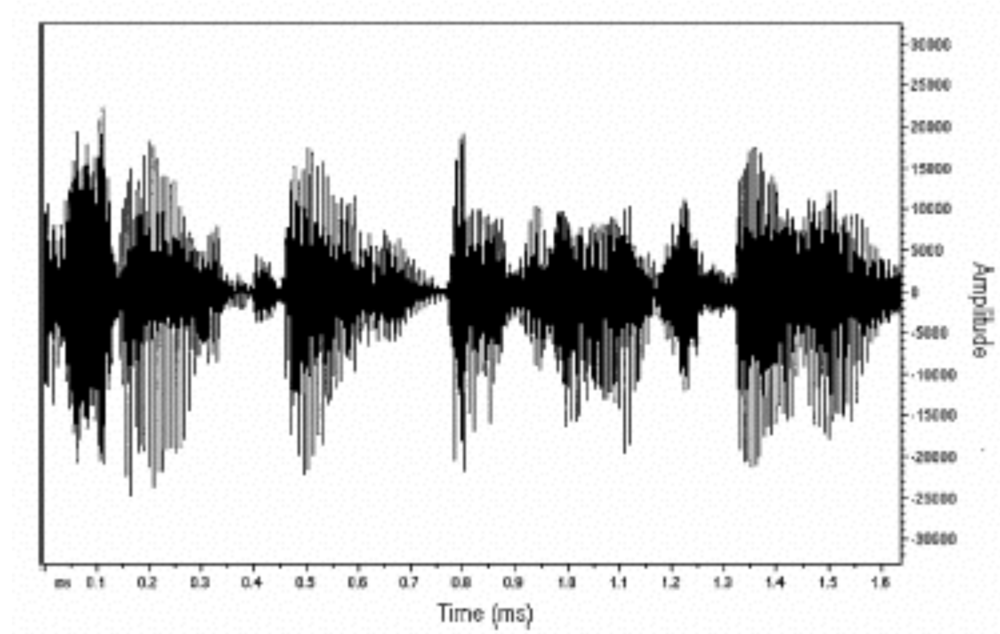
- 困難
  - 使用者的節奏（tempo）快慢不同、拍子不準、音調（key）高低不同
  - 若允許使用者從歌的任意處唱，計算量很大

CBMR系统流程图


Midi文件
抽取主旋律
资料库
每一首歌的中介格式
预处理
使用者唱歌
特征抽取
中介格式
比對
列出結果
用autocorrelation算出每个音框的音高
每1/16秒取一个音高（单位是semitone）
资料库中每一首歌曲获得一個分数代表和歌声的相似程度
列出资料库中分数最高的前N首歌曲

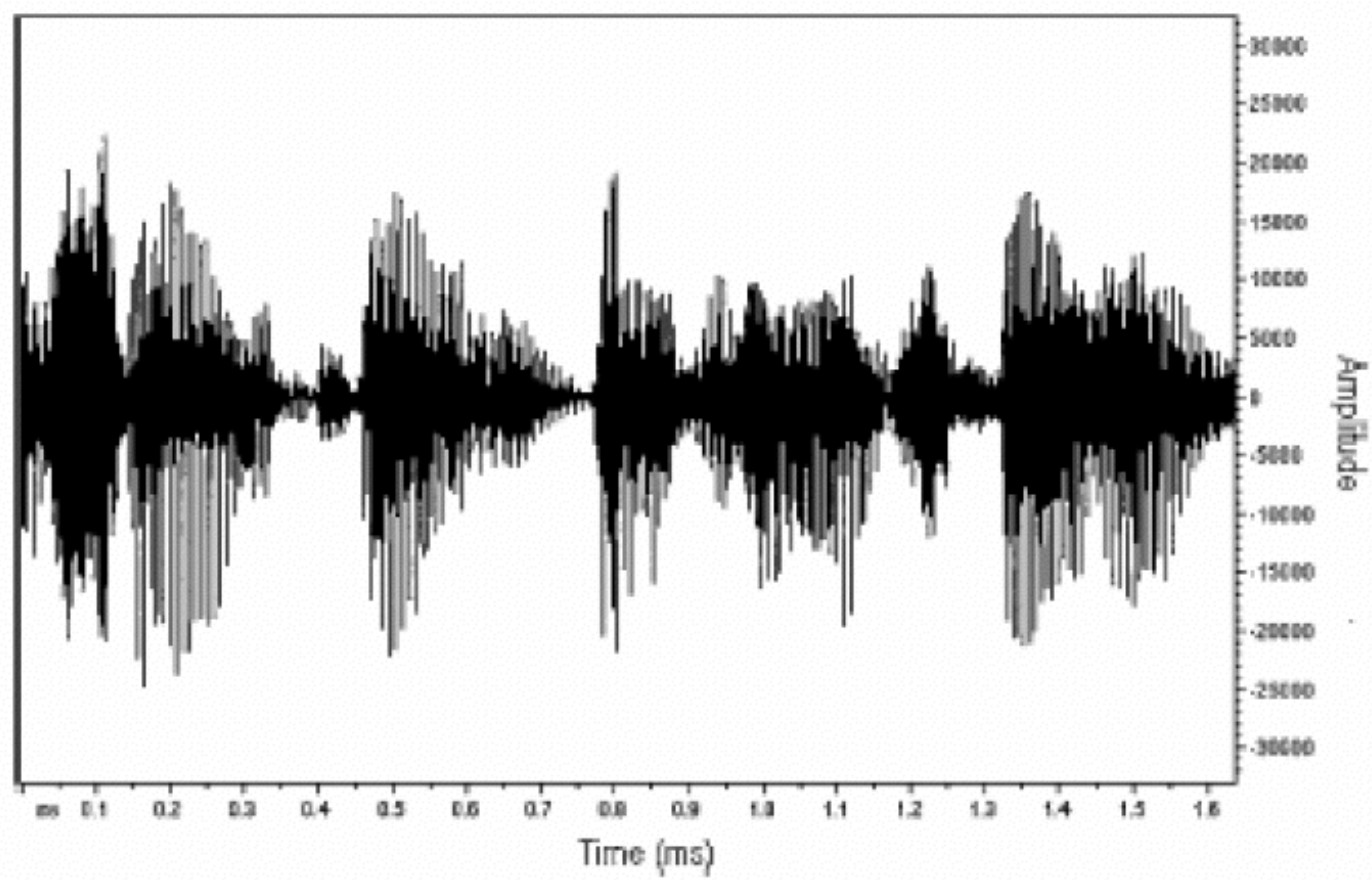
Main Audio Features

- Time-Domain Features
  - Average Energy
  - Zero Crossing Rate
  - Silence Ratio
- Frequency-Domain Features
  - Sound Spectrum
  - Bandwidth
  - Energy Distribution
  - Harmonicity
  - Pitch
- Spectrogram



Time-Domain Features

- Amplitude-time representation of an audio signal



Time-Domain Features (2)

- Average Energy
  - Indicates the loudness of the audio signal

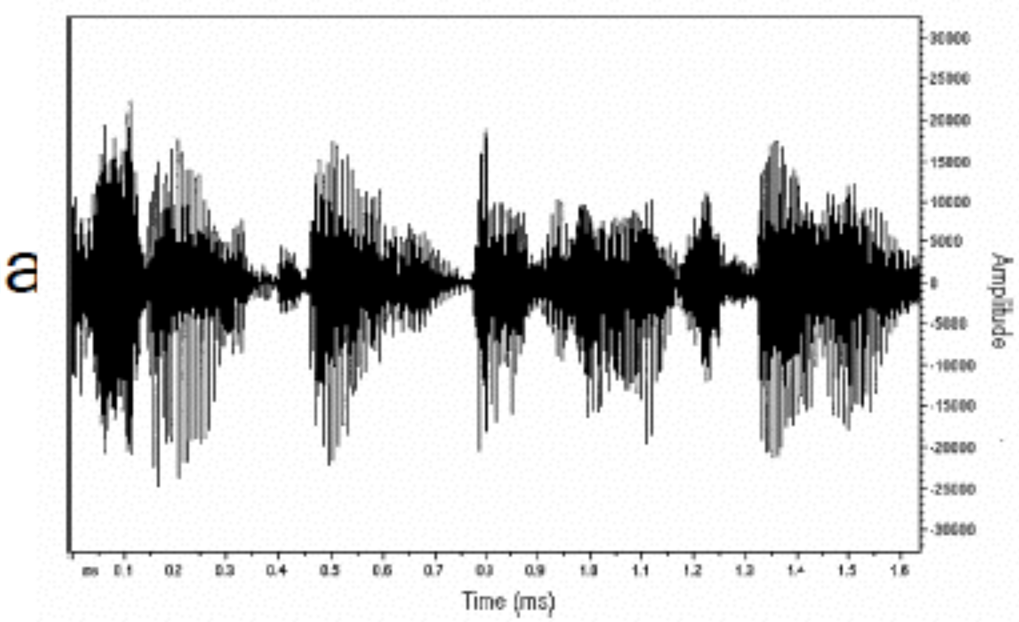
$$E = \frac{\sum_{n=1}^{N-1} x(n)^2}{N}$$

- Zero Crossing Rate
  - Indicates the frequency of signa amplitude sign change

$$ZC = \frac{\sum_{n=1}^{N-1} \operatorname{sgn}[x(n)] - \operatorname{sgn}[x(n-1)]}{2N}$$

$$\operatorname{sgn}(a) = \begin{cases} 1 & a > 0 \\ 0 & a = 0 \\ -1 & a < 0 \end{cases}$$

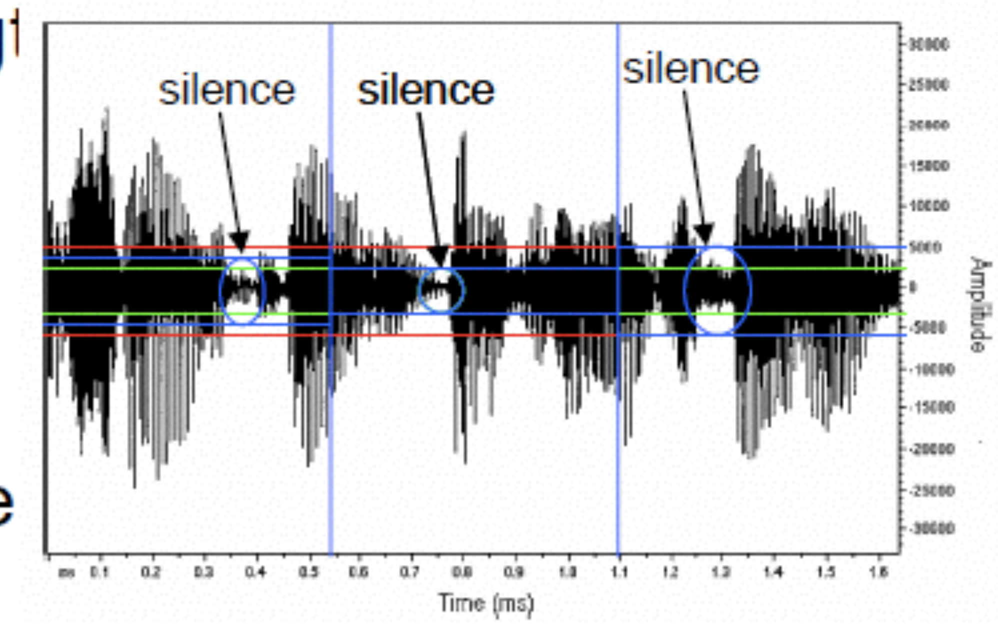
$x(n)$



n

Time-Domain Features (3)

- Silence Ratio
  - Indicates the proportion of the sound piece that is silent
  - Silence is a period within which the absolute amplitude values of a certain number of samples are below a certain threshold
  - Silence ratio is calculated as the ratio between the sum of silent periods and the total lengt

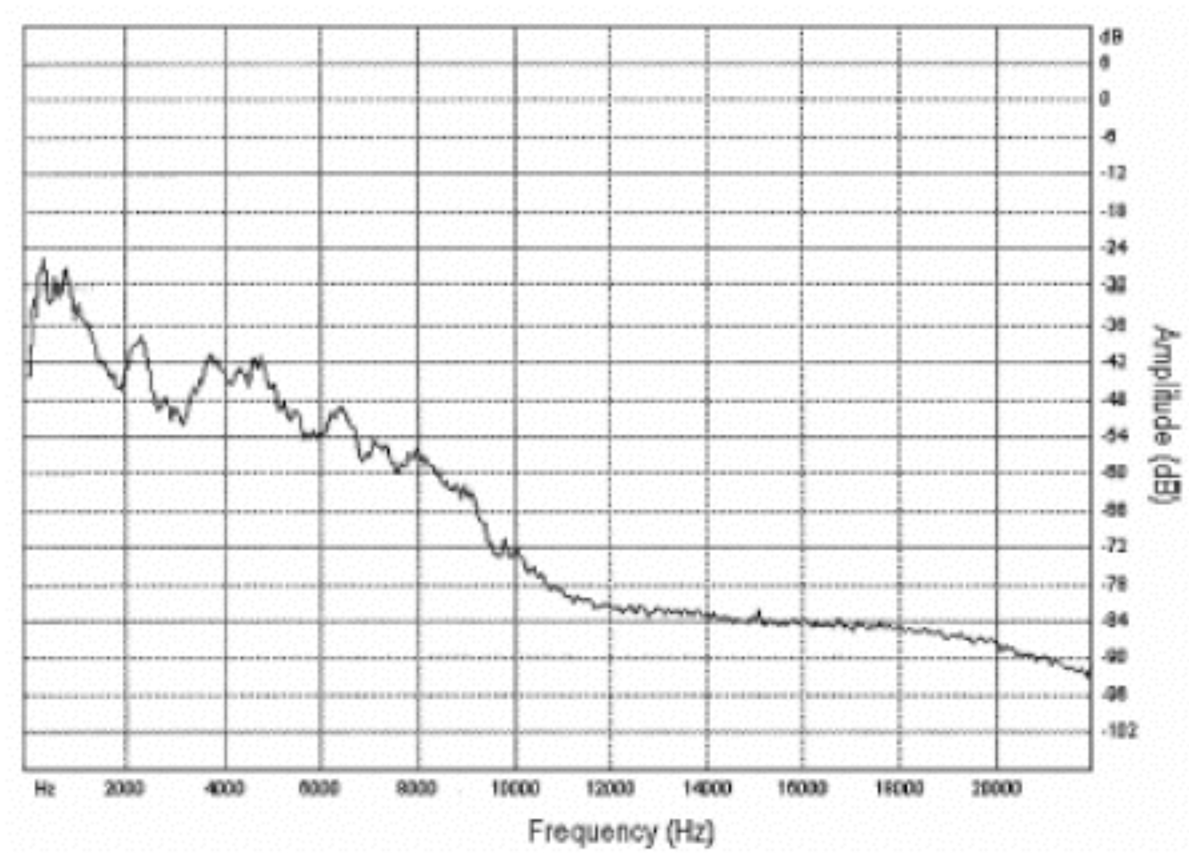


Approaches:

1. Fixed Threshold
2. Select Reference Silence Value
3. Adaptive Silence Thresholds

Frequency-Domain Features

- Sound Spectrum



Discrete Fourier Transform (DFT)

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-\frac{j2\pi nk}{N}}$$

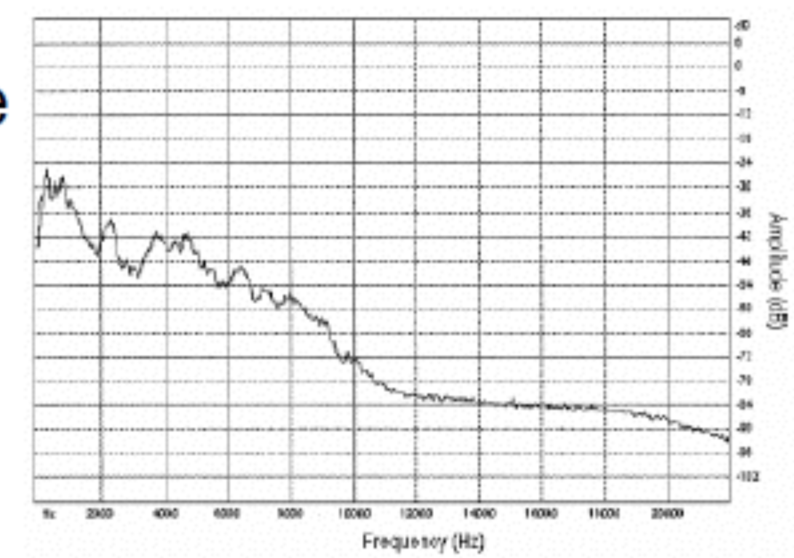
Inverse Discrete Fourier Transform (IDFT)

$$x(n) = \frac{1}{N} \sum_{n=0}^{N-1} X(k) e^{\frac{j2\pi nk}{N}}$$

- For large value of N, the signal is often broken into blocks called frames and DFT is applied to each of the frames.

Frequency-Domain Features (2)

- Bandwidth
  - indicated the frequency range of a sound
  - can be taken as the difference between the highest frequency and lowest frequency of non-zero spectrum components
  - “non-zero” may be defined as at least 3dB above the silence level



- Energy distribution
  - Signal distribution across frequency components
  - One important feature derived from the energy distribution is the *centroid*, which is the mid-point of the spectral energy distribution of a sound. Centroid is also called *brightness*

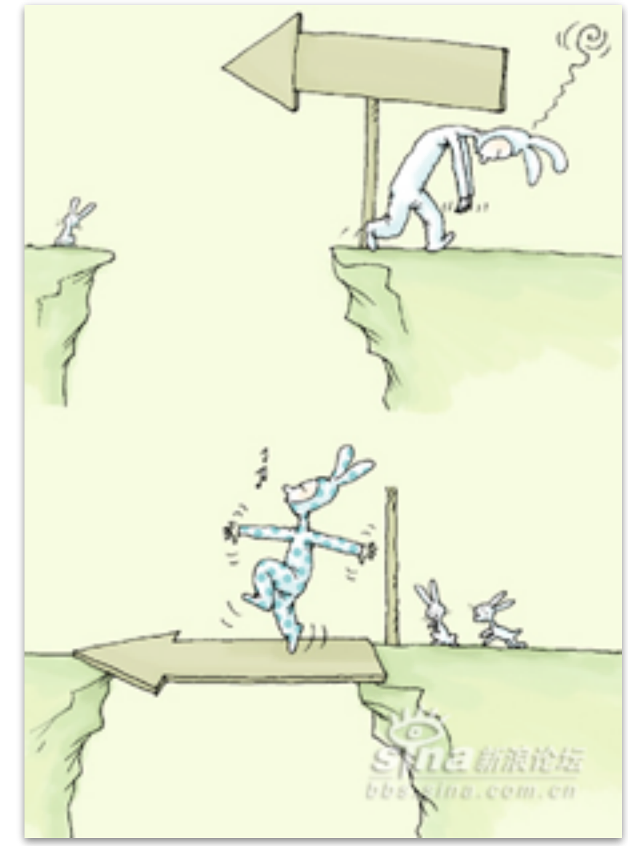
Frequency-Domain Features (3)

- Harmonicity
  - In harmonic sound, the spectral components are mostly whole number multiples of the lowest and most often loudest frequency
  - Lowest frequency is called *fundamental frequency*
  - Music is normally more harmonic than other sounds
- Pitch
  - the distinctive quality of a sound, dependent primarily on the frequency of the sound waves produced by its source
  - only period sounds, such as those produced by musical instruments and the voice, give rise to a sensation of a pitch
  - In practice, we use the fundamental frequency as the approximation of the pitch

相關研究

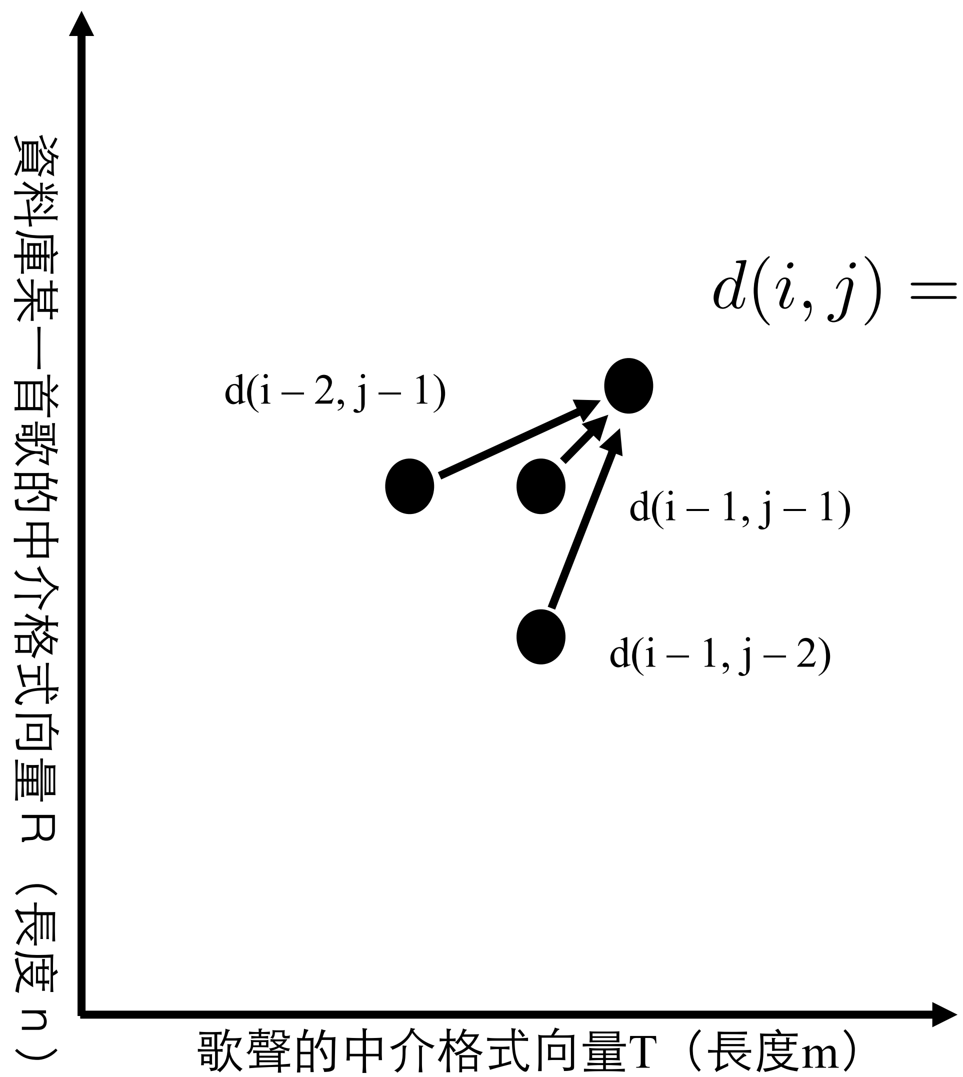
名稱	使用限制	比對方法	資料庫歌曲數目
Query by Humming	必須唱 ta 或 da 斷音	Baeza-Yates & Perleberg (92)	183
MELDEX (Melody Indexing)	必須唱 ta 或 da 斷音	Dynamic programming	9400
SoundCompass	必須照節拍器唱	Pitch transitions & histograms for weighted average	10086
MELODISCOV	無	FlExPat, Rolland (99)	未知

前人的方法



- 克服節奏快慢不同的問題
  - Dynamic time warping
- 克服音調高低不同的問題
  - Key transposition
    - 將使用者的歌聲和資料庫中的歌轉換後的中介格式的平均值都平移到0，做一次dtw比對
    - 將資料庫中的歌的中介格式上下平移再做四次dtw比對，以找出最短的距離
- 全曲比對費時很久且準確率低
- 使用浮點數運算

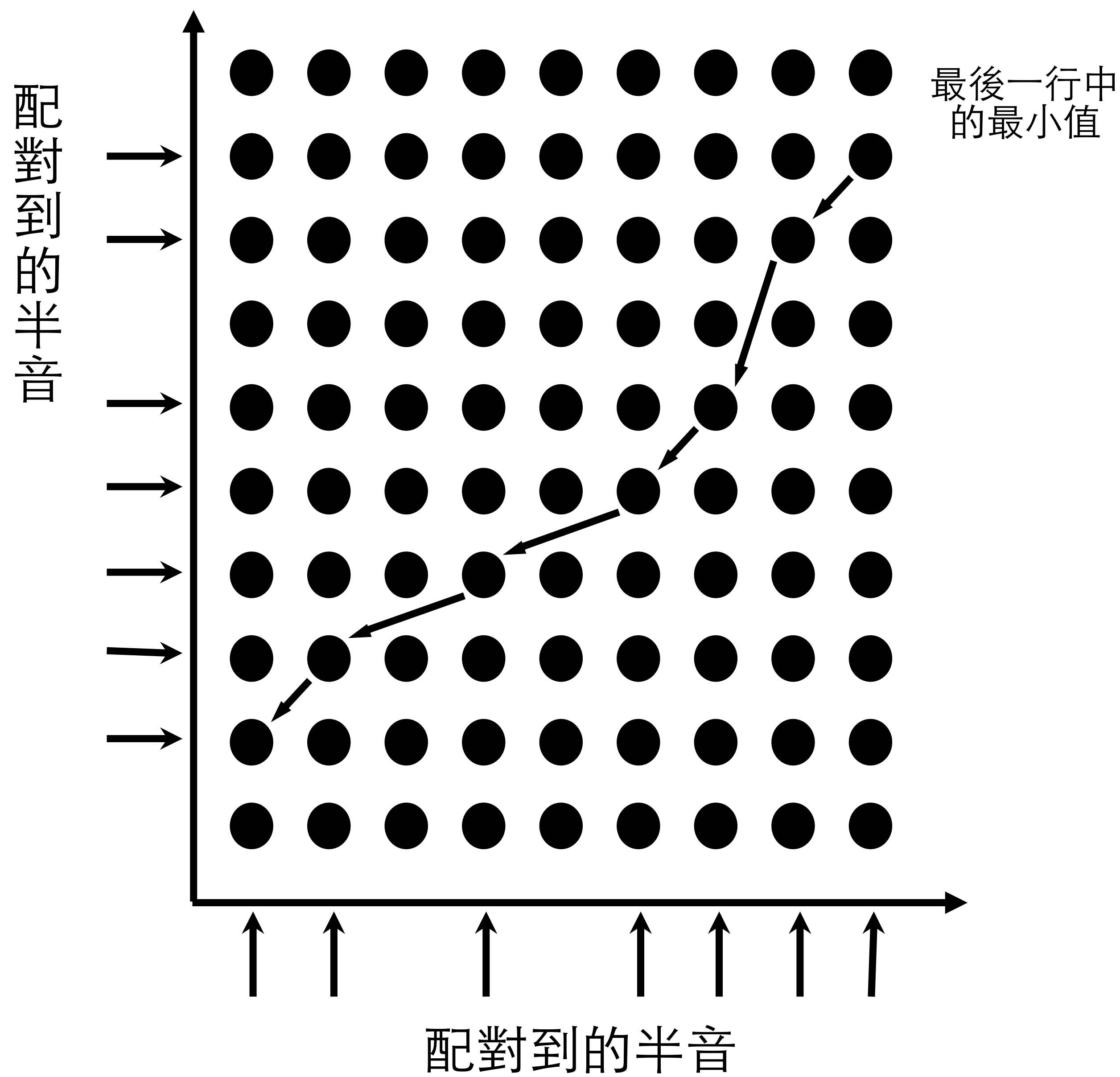
Dynamic Time Warping



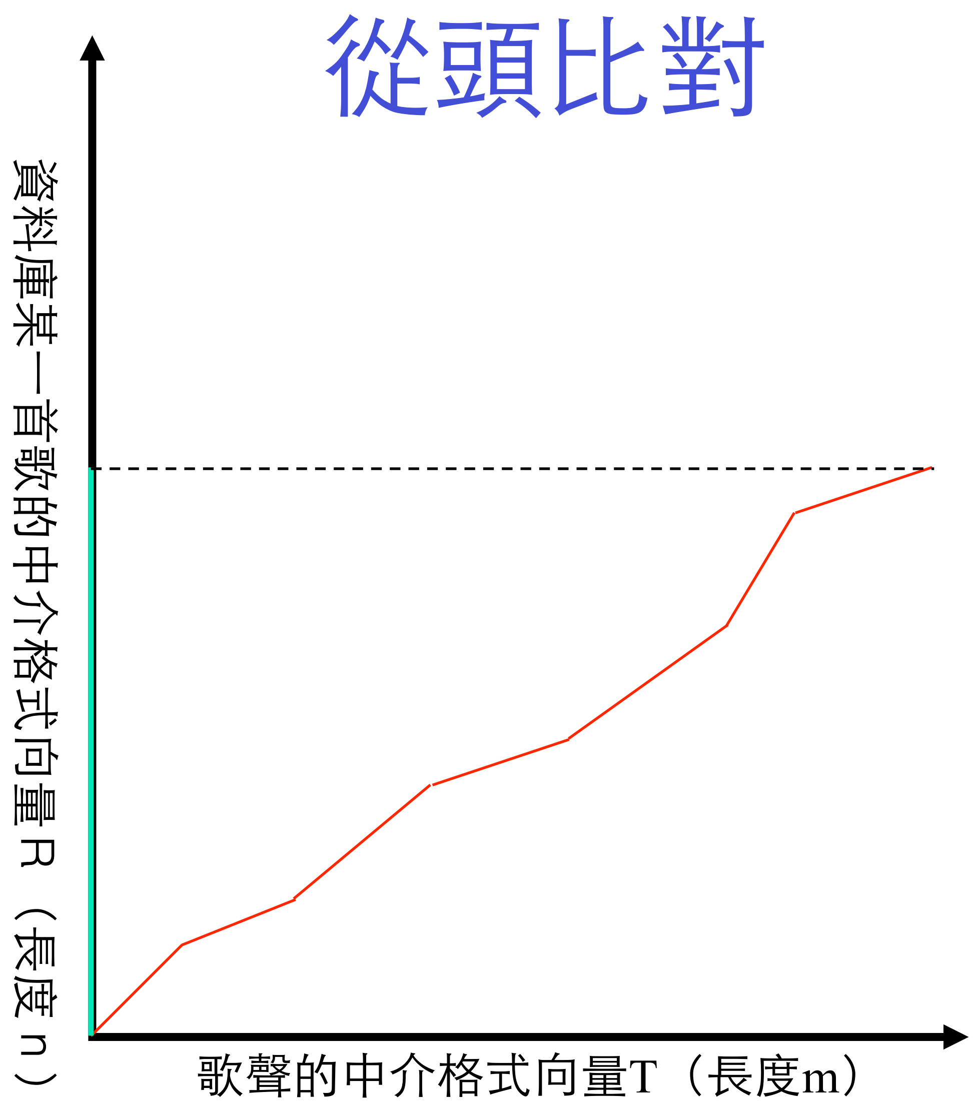
$$d(i,j) = \min \begin{cases} d(i-1, j-1) \\ d(i-2, j-1) + |T(i) - R(j)| \\ d(i-1, j-2) \end{cases}$$

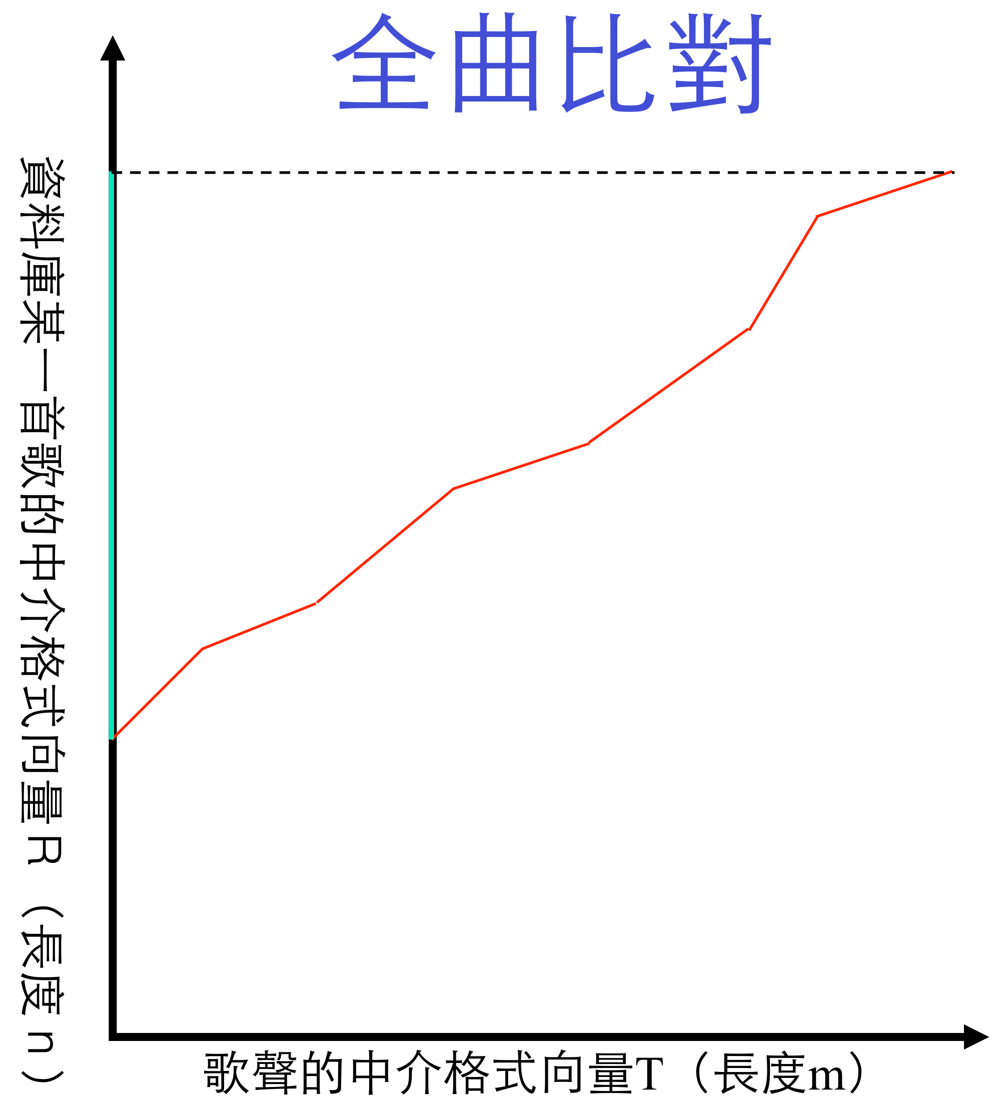
$$dtw(T,R) = \min_{1 \le j \le n} d(m, j)$$

Dynamic Time Warping

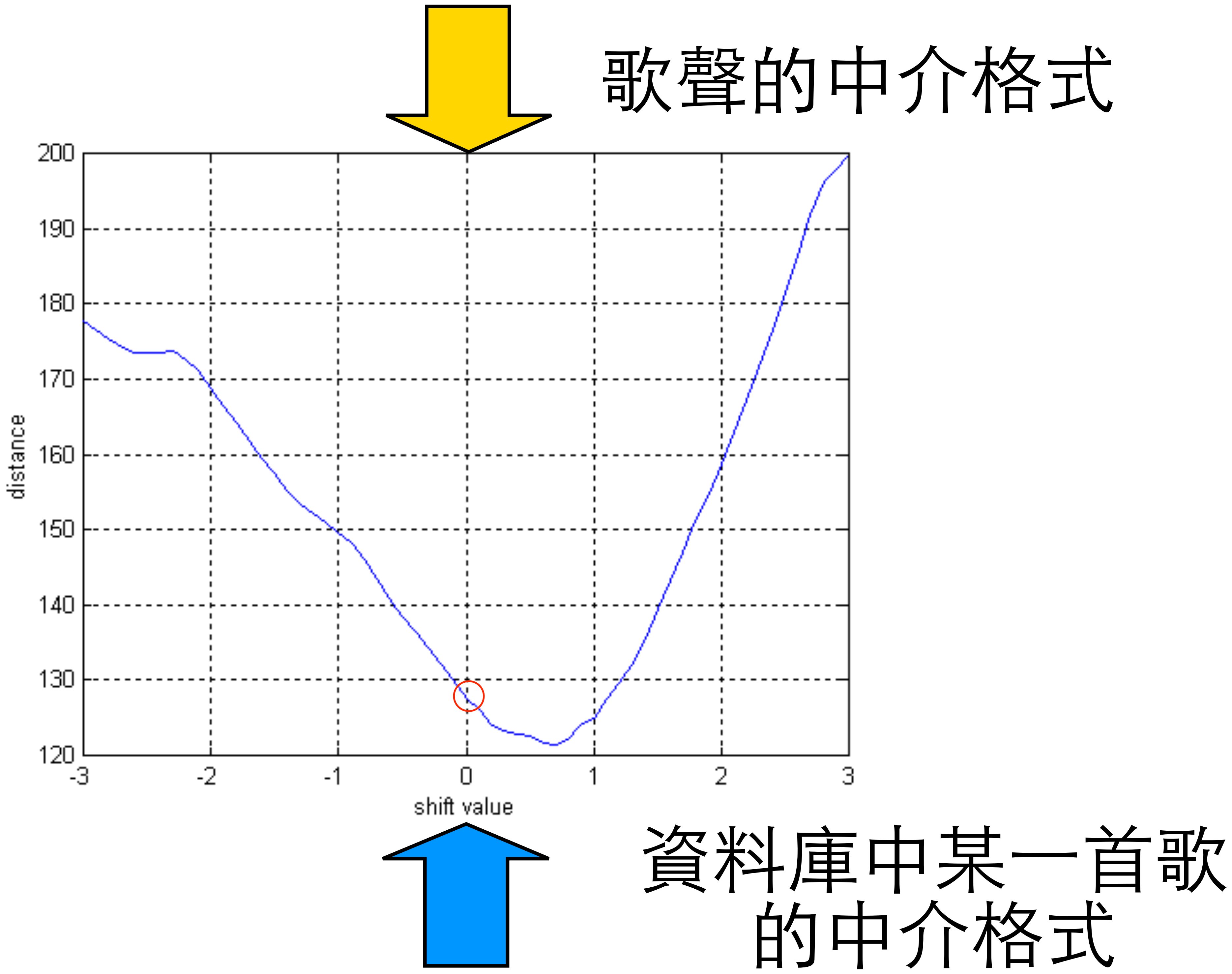


Dynamic Time Warping



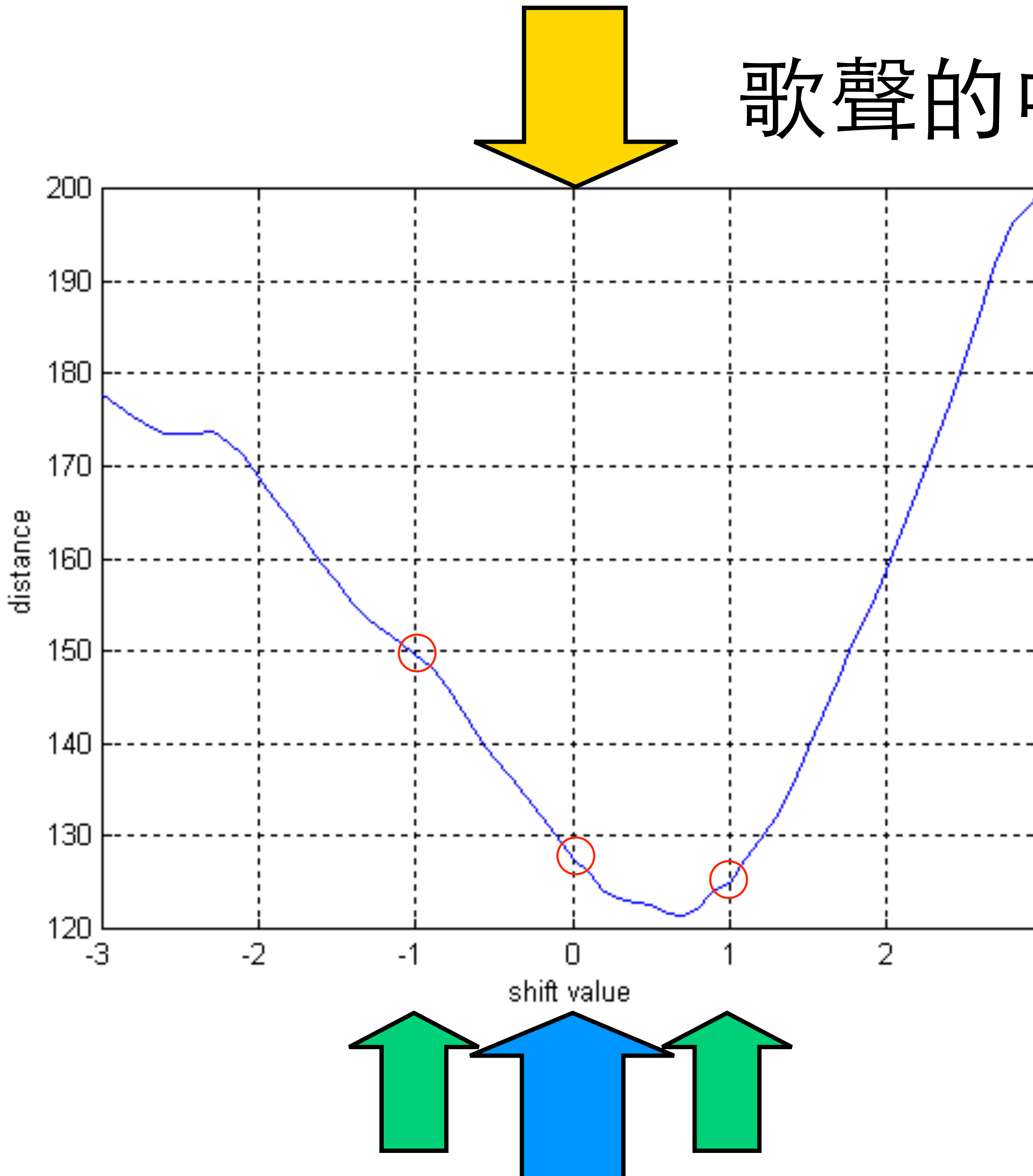


Key Transposition



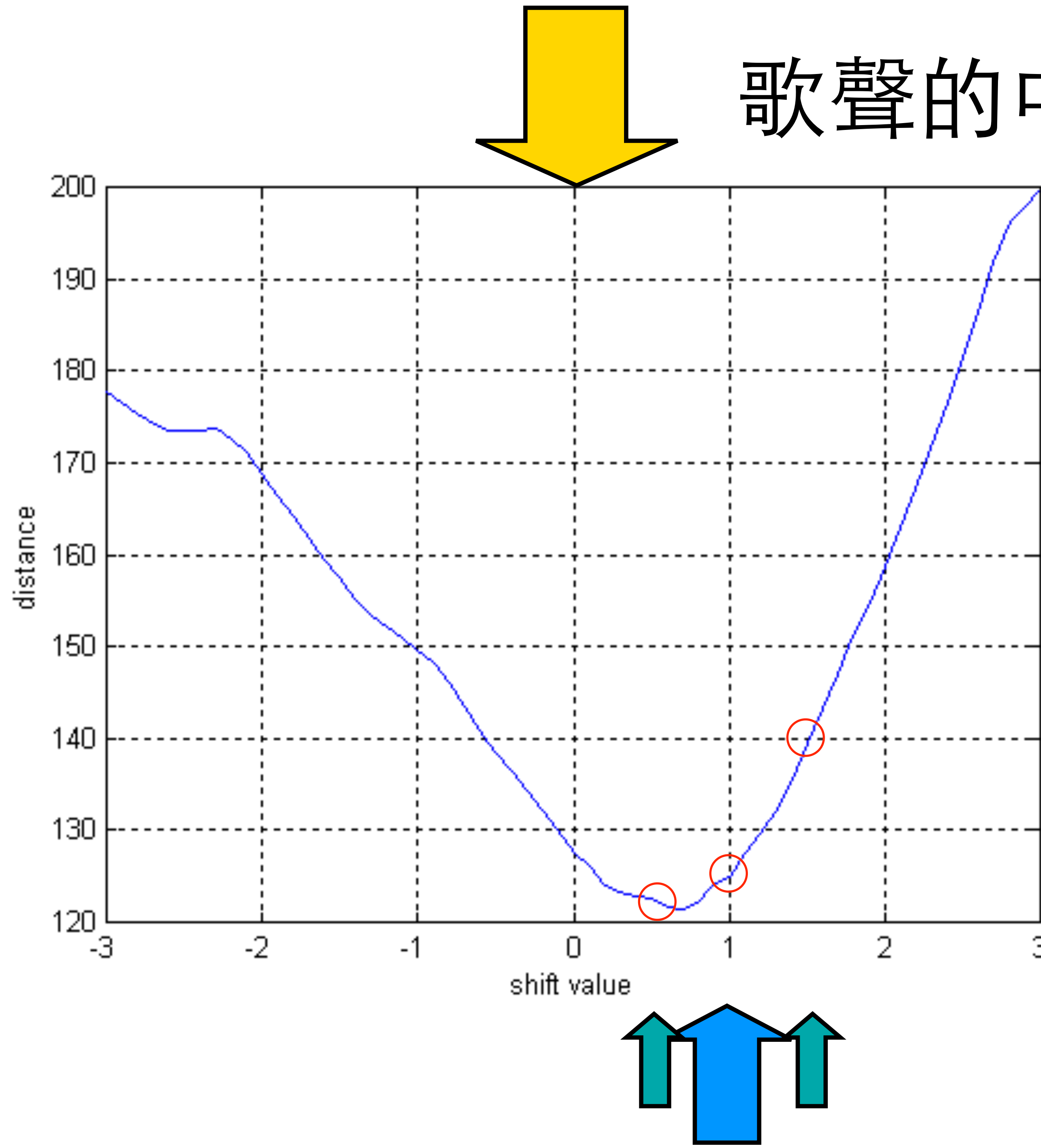
Key Transposition

歌聲的中介格式



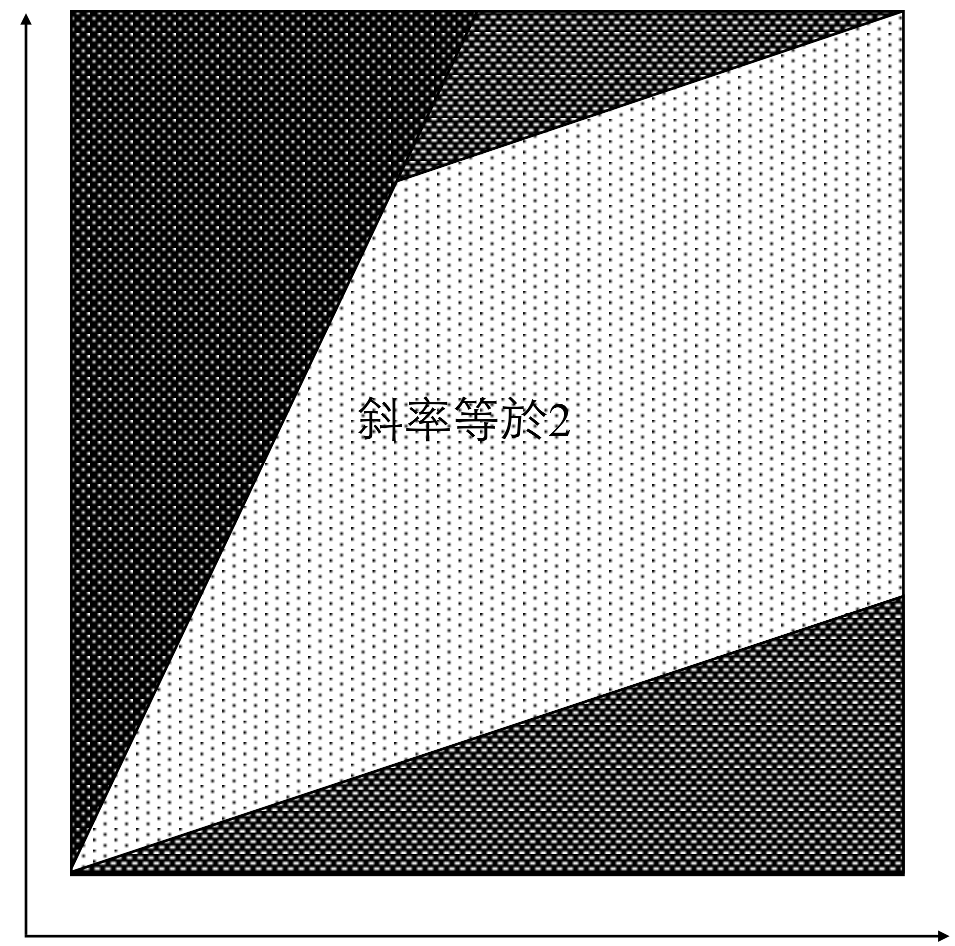
Key Transposition

歌聲的中介格式



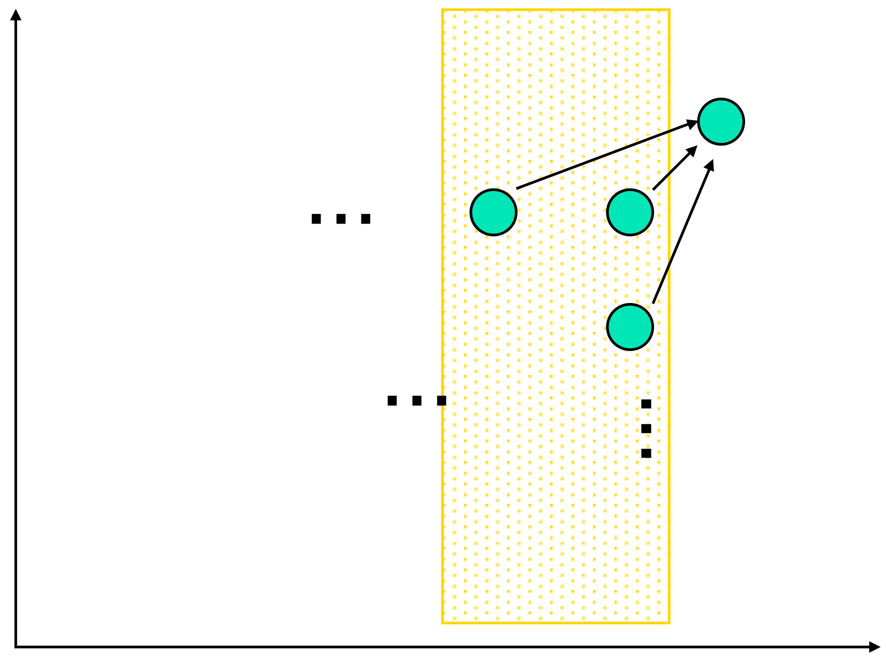
改進方法一

- 改用整數運算
  - 犧牲部分准确率
- 改良式dtw
  - 因為T的頭尾必須match到R的某一段，所以顏色較深的地方不必計算



改進方法一

- 改良式dtw
  - 用dtw計算每一首歌的距離時，若發現dtw table最近兩行(column)的最小值超過之前最短距離的前k名，則停止dtw



改進方法二

- 將資料庫中每一首歌的中介格式，從每一個音符為起點切成數個長度為 D=72 的片段

中介格式　(69, 69, 67, 67, 67, 71, 72, ……)

→　片段1　(69, 69, 67, 67, 67, 71, 72, …)

　　片段2　(67, 67, 67, 71, 72, …)

　　片段3　(71, 72, …)

　　……

改進方法二

- 用兩階段的方法比對
  - 第一階段：線性伸縮比對（linear scaling）
    - 將歌聲的中介格式伸縮11次（長度為原來的0.75倍到1.25倍），分別取出前 D 點後為 T_i（$1 \leqq i \leqq 11$），假設資料庫中的第 j 首歌的中介格式有 n_j 個片段為 R_{jk}（$1 \leqq j \leqq$ 資料庫中的歌曲數目，$1 \leqq k \leqq n_j$），令 T_i 和 R_{jk} 的距離為

$$dist(T_i, R_{jk}) = \sqrt{\sum_{t=1}^{D} \left(\left(T_{it} - \overline{T_i}\right) - \left(R_{jkt} - \overline{R_{jk}}\right) \right)^2}$$

改進方法二

- 令資料庫第 j 首歌的分數為 $100 - \min_{\substack{1 \le i \le 11 \\ 1 \le k \le n_j}} \frac{dist(T_i, R_{jk})}{D}$ 令

  $$k' = \arg \min_{\substack{1 \le i \le 11 \\ 1 \le k \le n_j}} dist(T_i, R_{jk})$$

- 篩選出前 n=200 首分數最高者做第二階段的比對
- 缺點：每和資料庫中第 j 首歌做比對就要計算n_j*11次距離

- 第二階段：dynamic time warping
  - 將篩選後的每一首歌的最接近片段平移音調4次，總共和歌聲原本的中介格式計算5次距離，找出最小值並轉換成分數，當成資料庫該首歌最後的分數

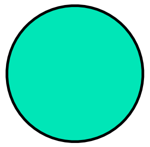
針對全曲比對的加速方法

- 用兩階段式比對
- 在第一階段將資料庫歌曲的每一個片段看成一個 D 度空間中的資料點，歌聲伸縮後的中介格式則是空間中的查詢點，利用快速找最近鄰居的方法找出每首歌最接近的片段
  - Vantage-point tree
  - Branch-and-bound tree
  - Equal-average hyperplane partitioning method

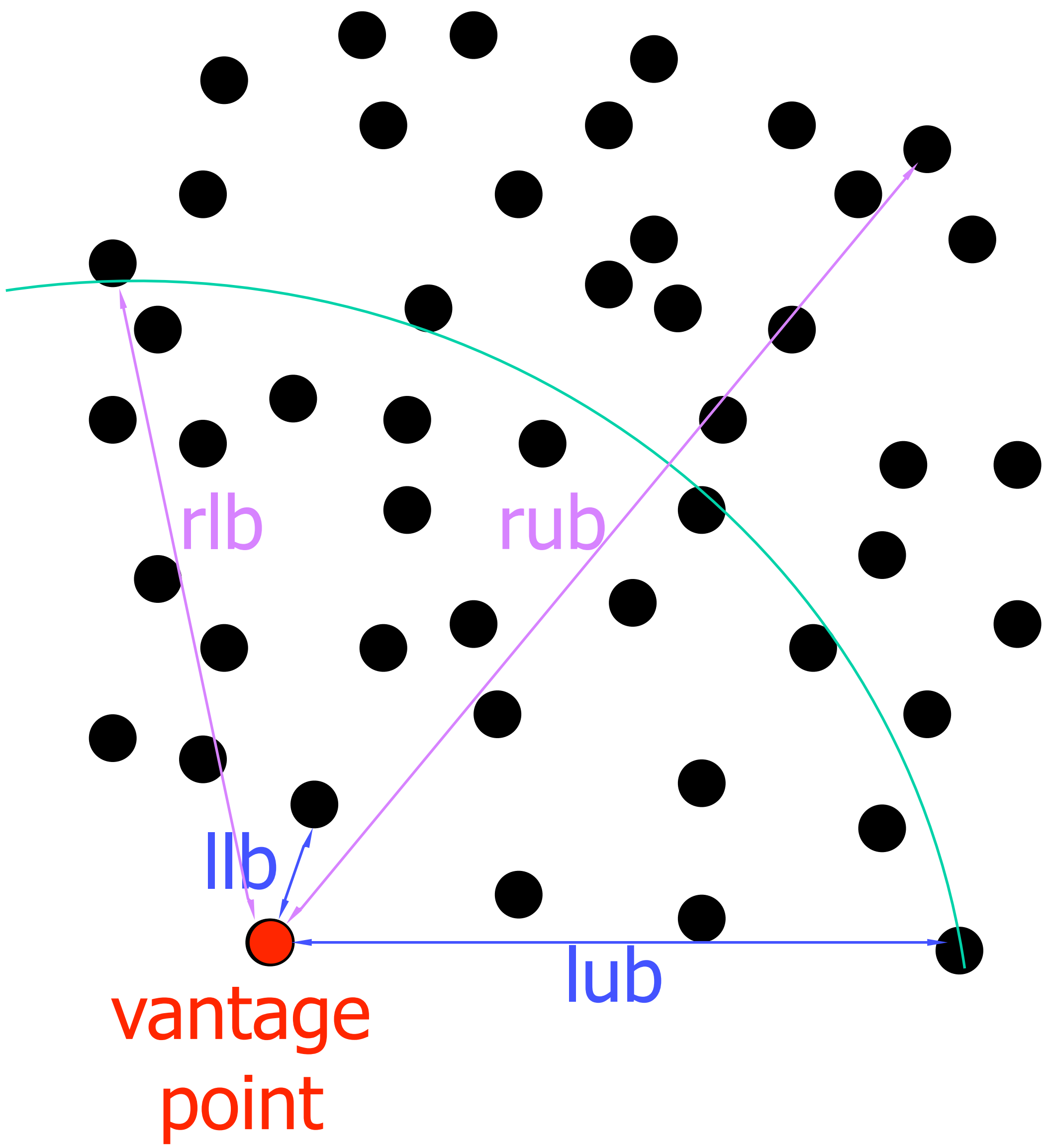
Vantage-Point Tree

- 將資料點建立成一個平衡的二叉树，利用距離滿足三角不等式的關係，可以減少計算查詢點到資料點的距離的次數
  - 缺點：建樹麻煩，使用時耗記憶體
  - 困難：資料點少，查詢點到所有資料點的距離都差不多，加速效果不顯著

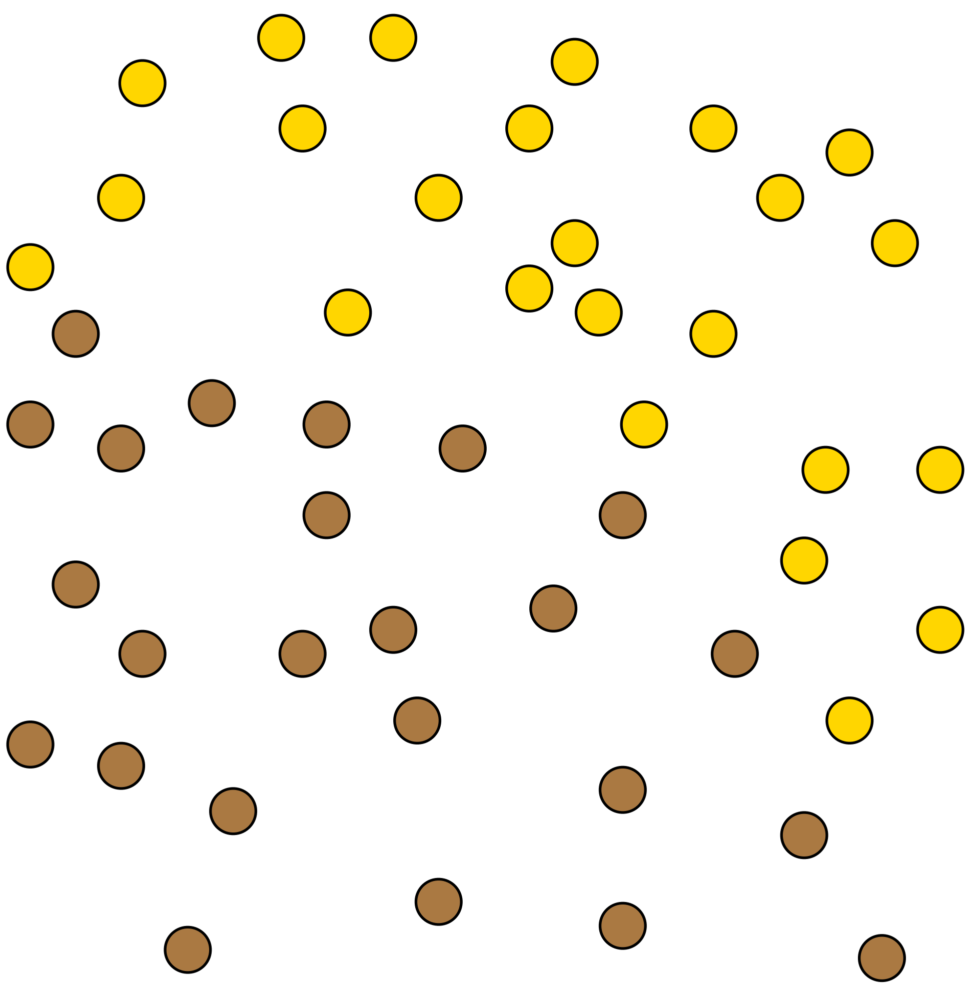
Vantage-Point Tree

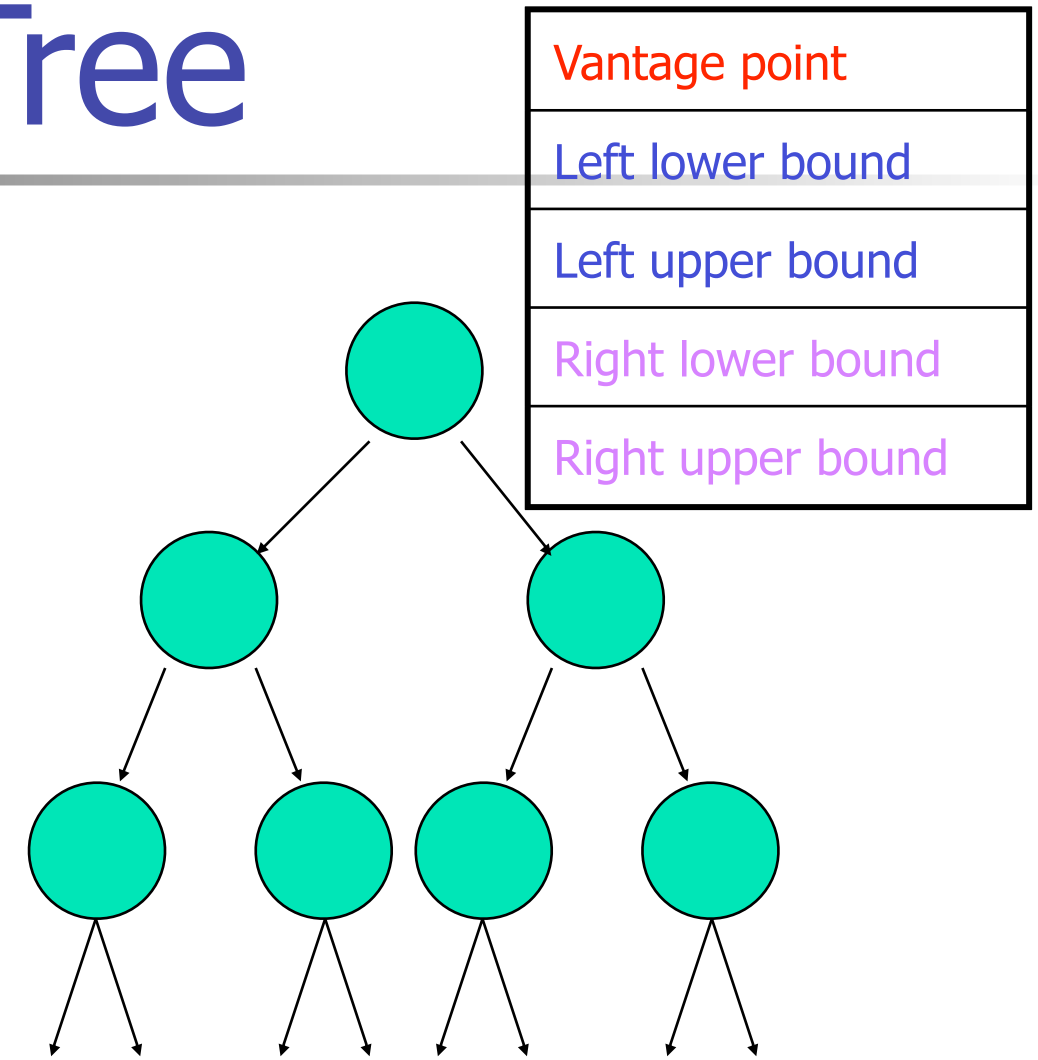


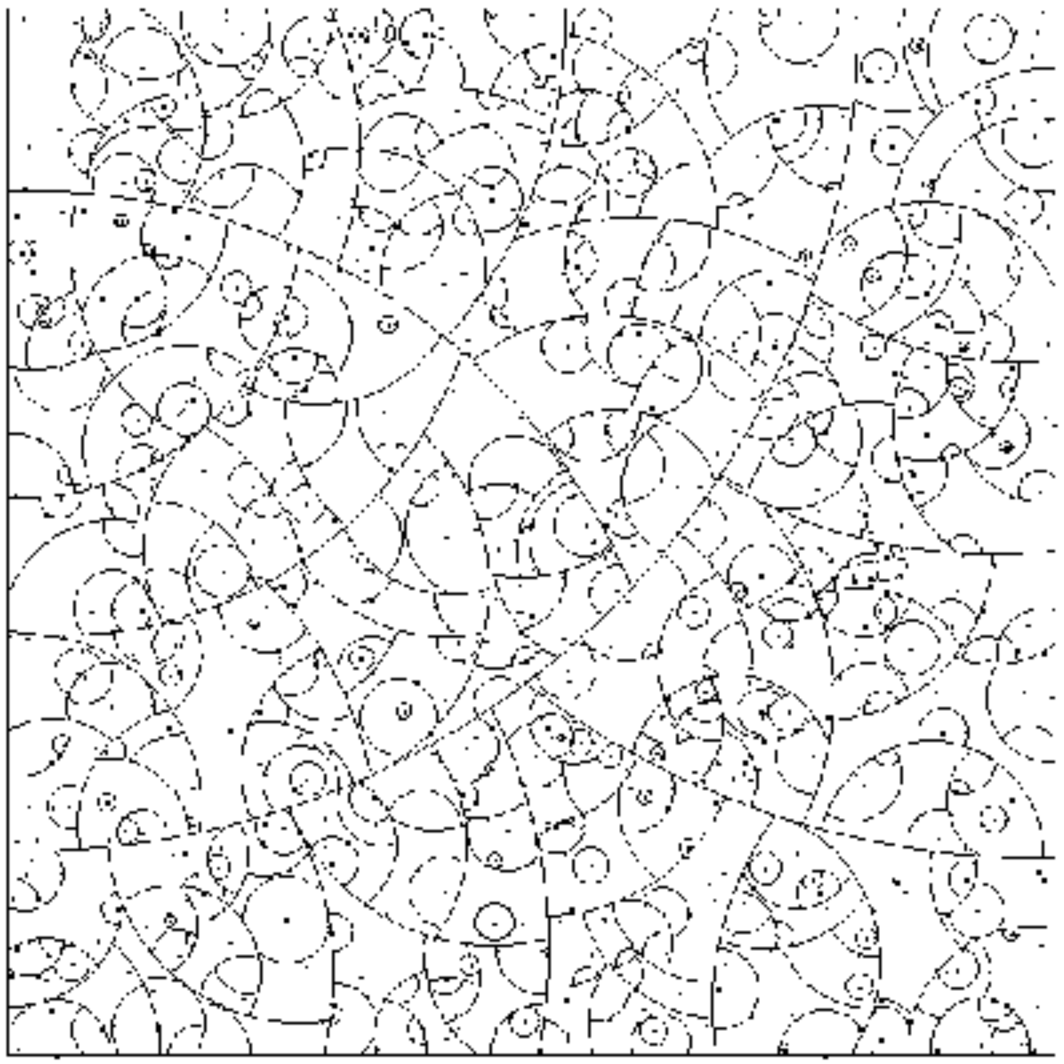
Vantage point
Left lower bound
Left upper bound
Right lower bound
Right upper bound



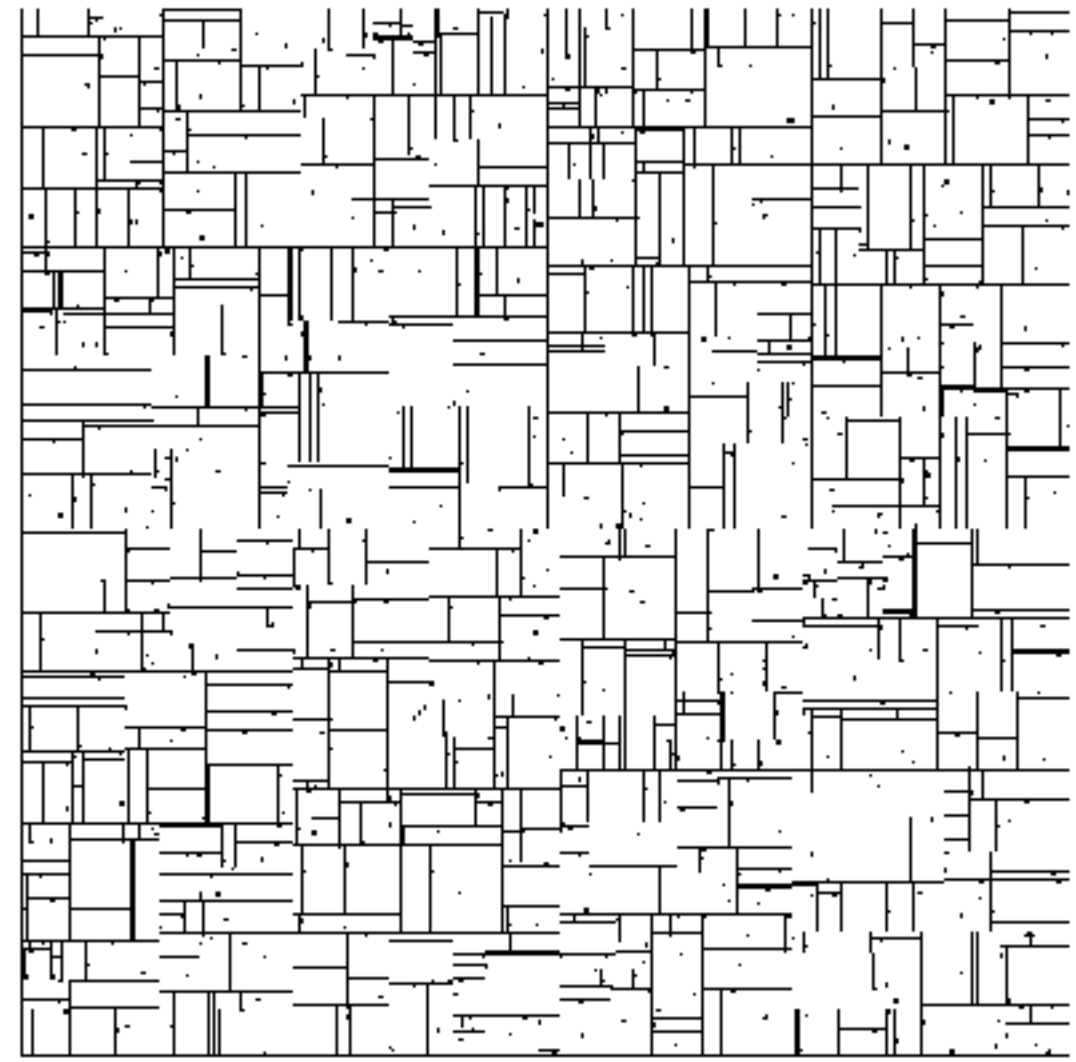
Vantage-Point Tree







VP-tree decomposition

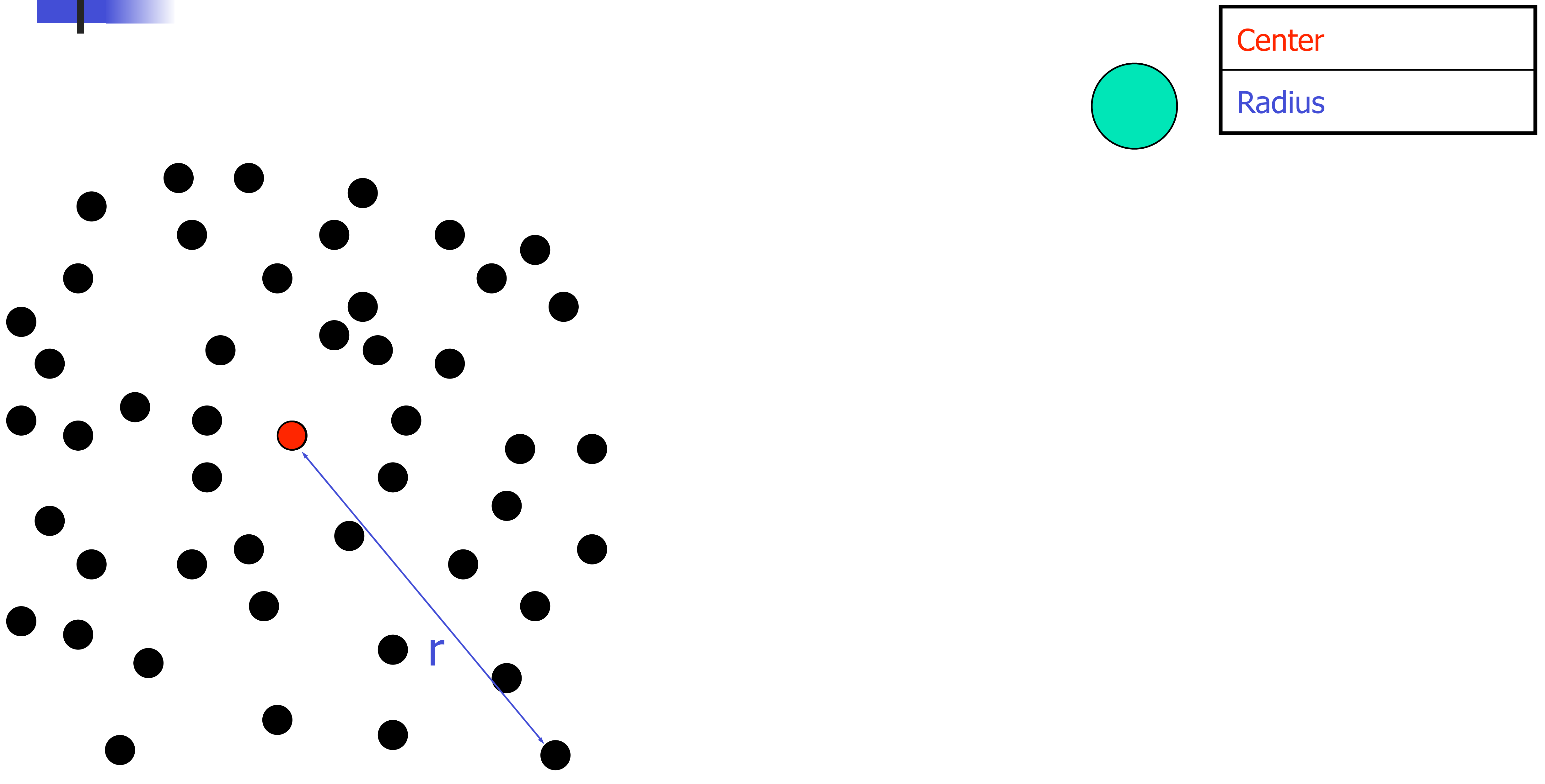


KD-tree decomposition

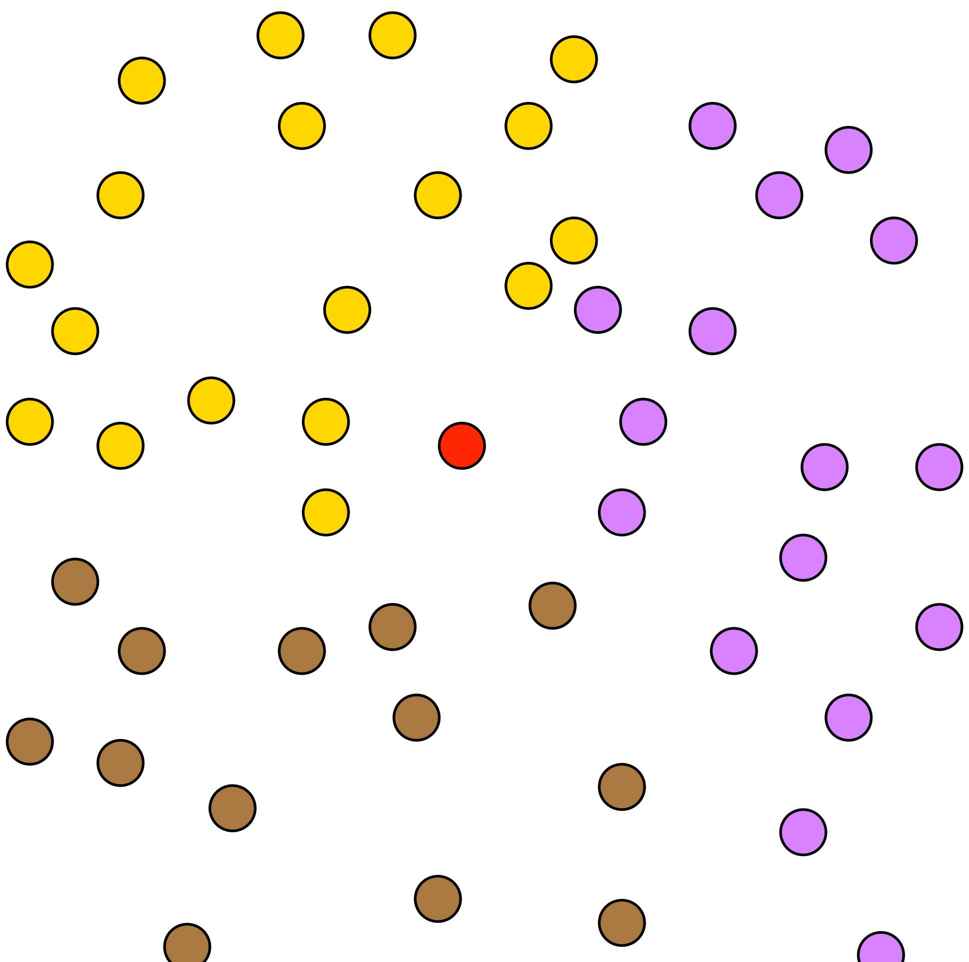
Branch-and-Bound Tree

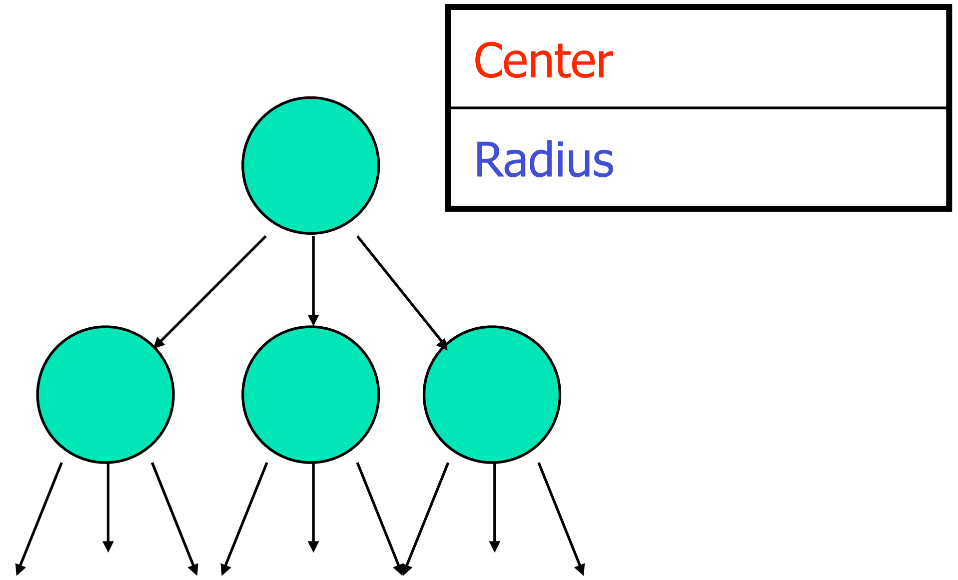
- 原理類似vantage-point tree，將資料點建立成一個多元樹，利用距離滿足三角不等式的關係，可以減少計算查詢點到資料點的距離的次數
  - 缺點：建樹時使用K-means，麻煩耗時，使用時耗記憶體
  - 困難：資料點少，查詢點到所有資料點的距離都差不多，加速效果不顯著

Branch-and-Bound Tree



Branch-and-Bound Tree



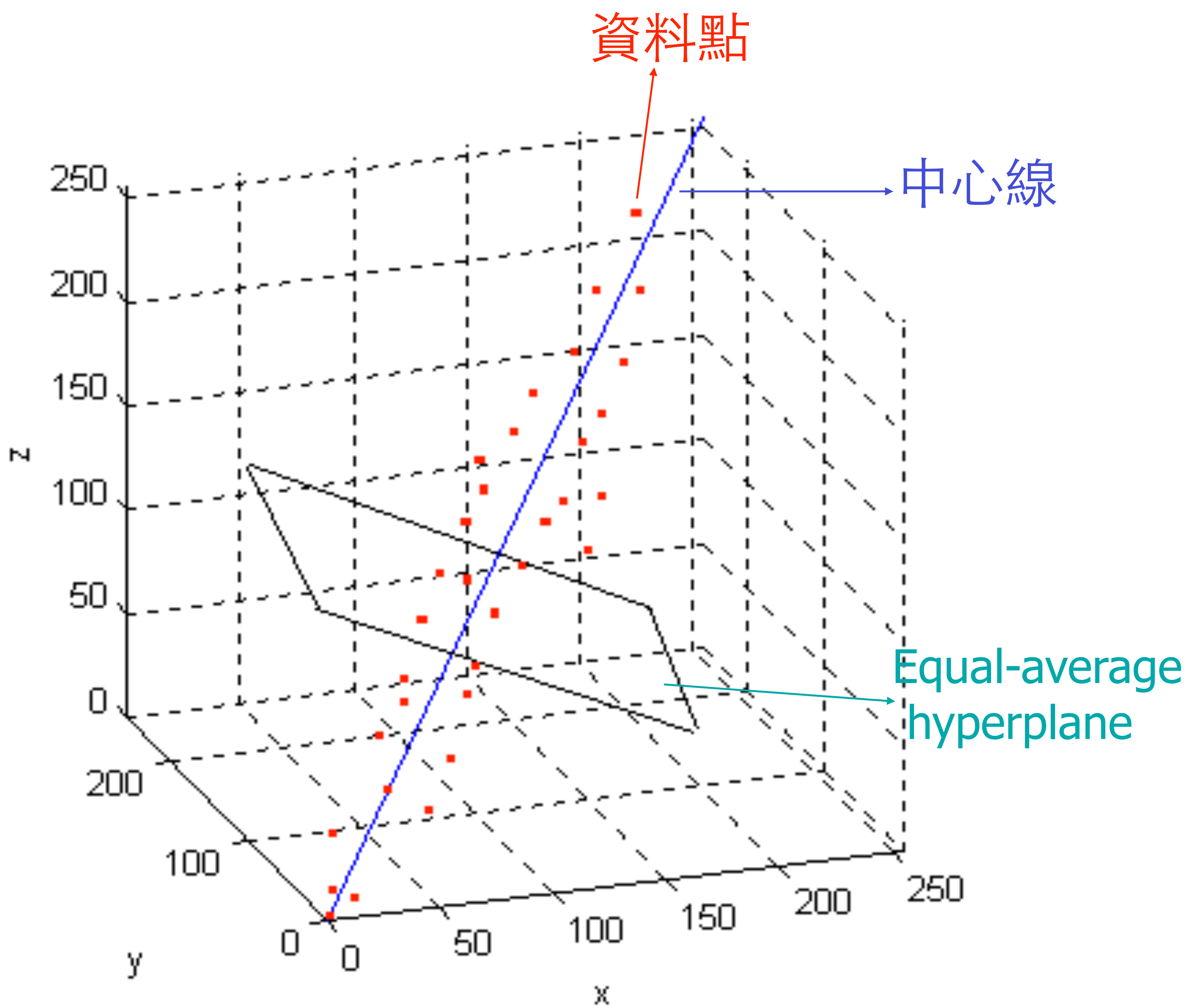


Equal-average Hyperplane Partitioning Method

- 讓資料點沿一直線分散開來，事先算出每一個資料點投影到直線上的位置。利用兩資料點間的距離會大於或等於這兩點投影到該直線上的距離的關係，可以減少計算查詢點到資料點的距離的次數
  - 優點：實作容易，省記憶體
  - 缺點：使用時機較狹隘

Equal-average Hyperplane Partitioning Method

- 中心線 L 是一條通過原點和 (1, 1, 1, …, 1) 的直線
- Equal-average hyperplane是和 L 垂直的面



Equal-average Hyperplane Partitioning Method

- 假設 a=(s, s, …, s) 為 L 上的一個點
- 則和 L 相交於 a 點的 equal-average hyperplane 其方程式為

$$H(x) = s\sum_{i=1}^{D} x_i - Ds^2 = 0$$

$$\Rightarrow s = \frac{1}{D}\sum_{i=1}^{D} x_i \quad \leftarrow \text{座標平均值}$$

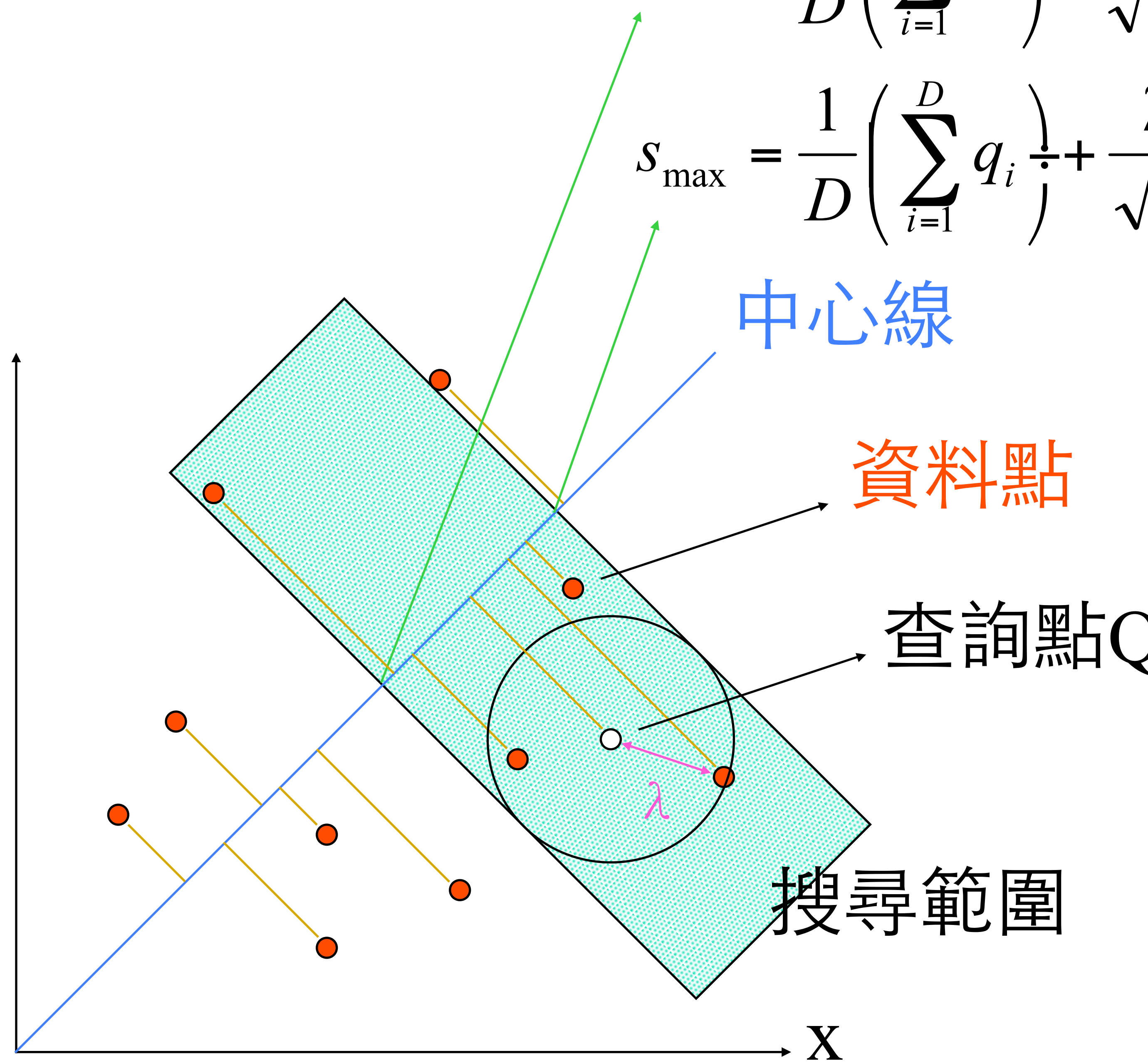
- 座標平均值相同的點會落在同一個equal-average hyperplane

Equal-average Hyperplane Partitioning Method

- 座標平均值落在 $[S_{min}, S_{max}]$ 之外的點不可能是最近鄰居

$$s_{\min} = \frac{1}{D}\left(\sum_{i=1}^{D} q_i\right) - \frac{\lambda}{\sqrt{D}}$$

$$s_{\max} = \frac{1}{D}\left(\sum_{i=1}^{D} q_i\right) + \frac{\lambda}{\sqrt{D}}$$

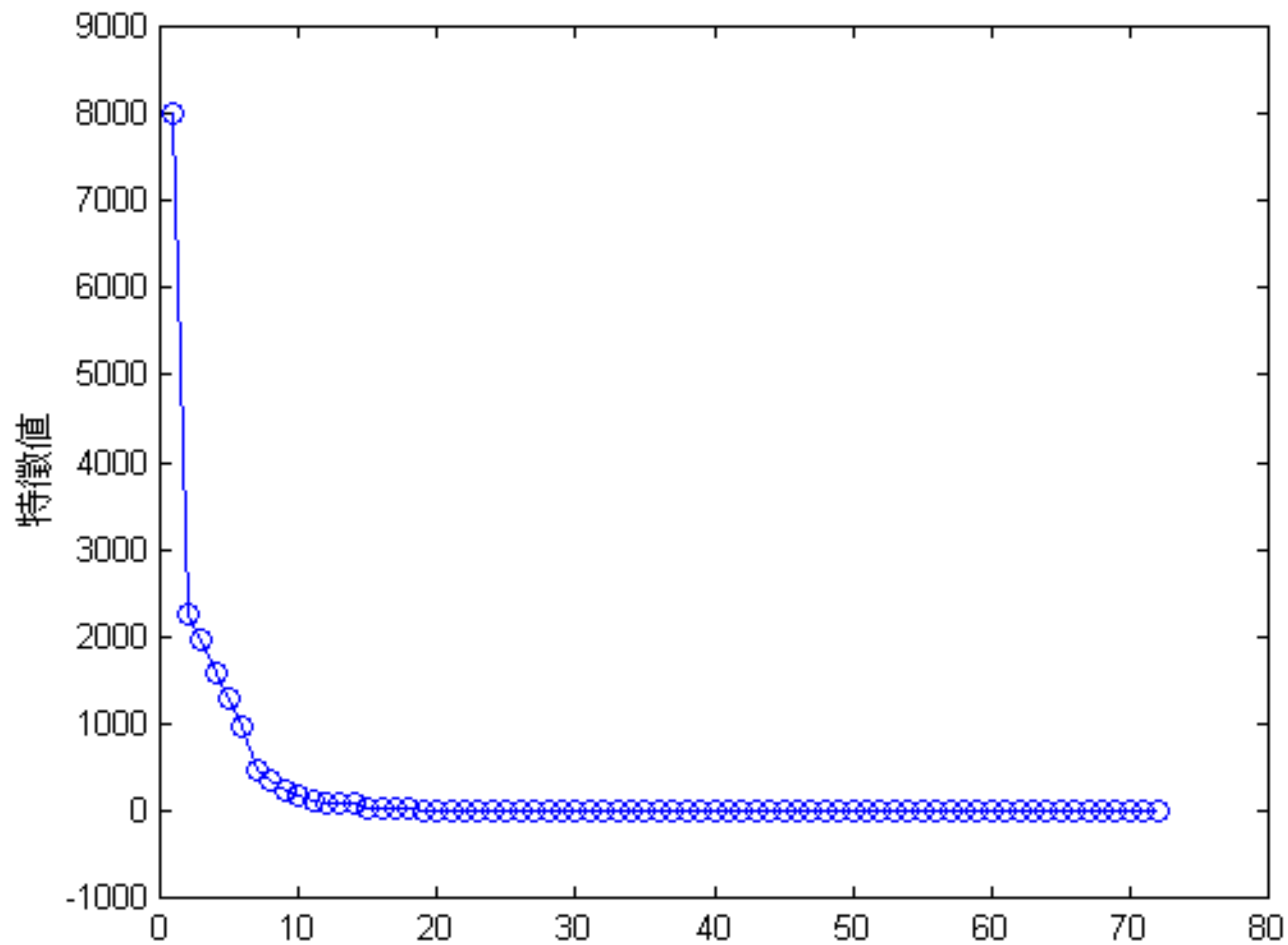


Equal-average Hyperplane Partitioning Method

- 因為我們將每個資料點平移到平均值為0，所以必須另外找一條直線 L 代替原本的中心線
  - 用 principal component analysis 找出 L
    - 令 M 矩陣的每一列為資料點R_{jk}的座標
    - 求出 M^TM 的eigenvalue和eigenvector
    - 對應到最大eigenvalue的eigenvector即為所求
  - 用 $\frac{R_{jk} \times L}{\sqrt{D}}$ 代替座標平均值

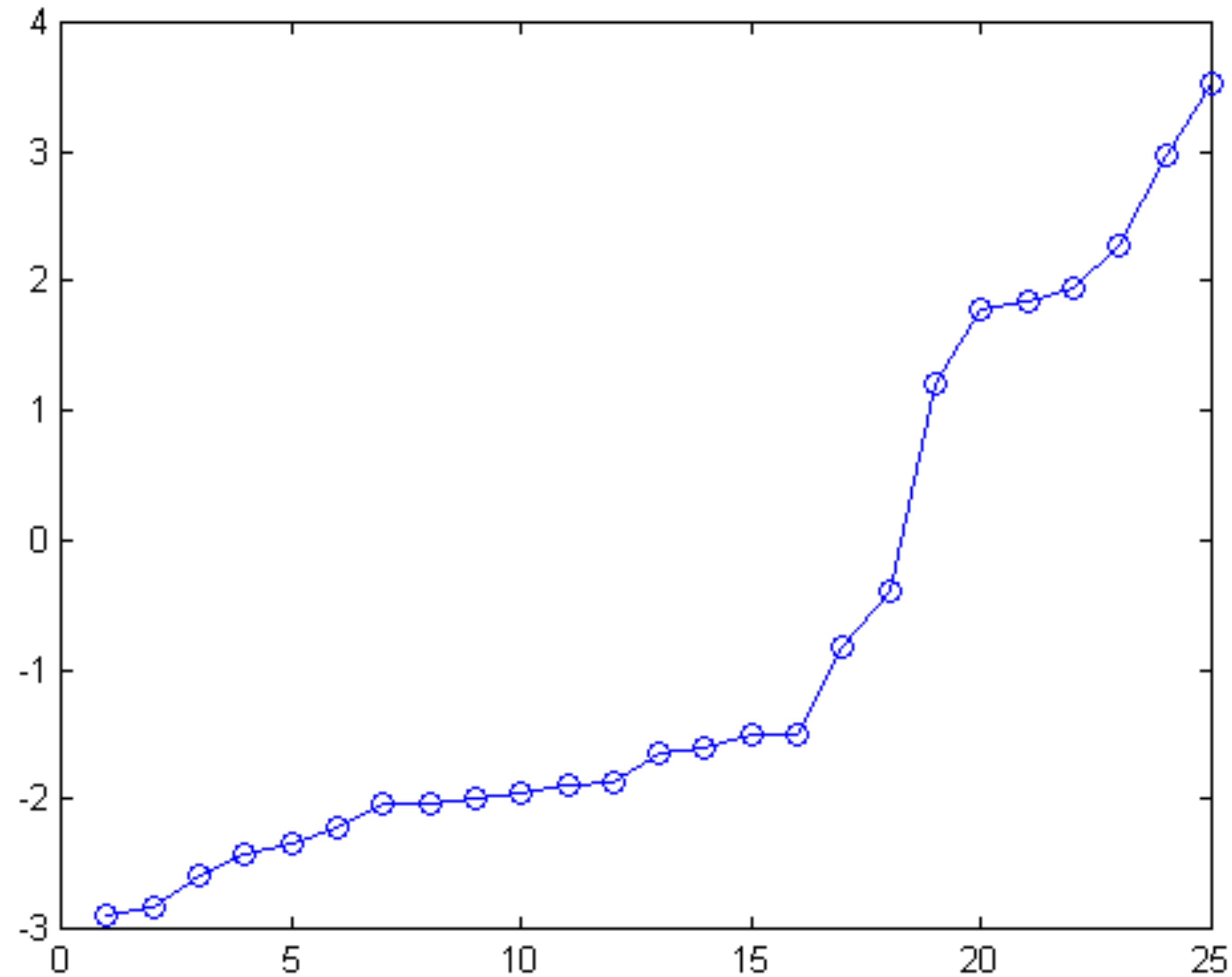
Equal-average Hyperplane Partitioning Method

- 資料庫中某一首歌曲有25個片段，每個片段長度為72
- 用PCA找出來的72個特徵值從大到小排序



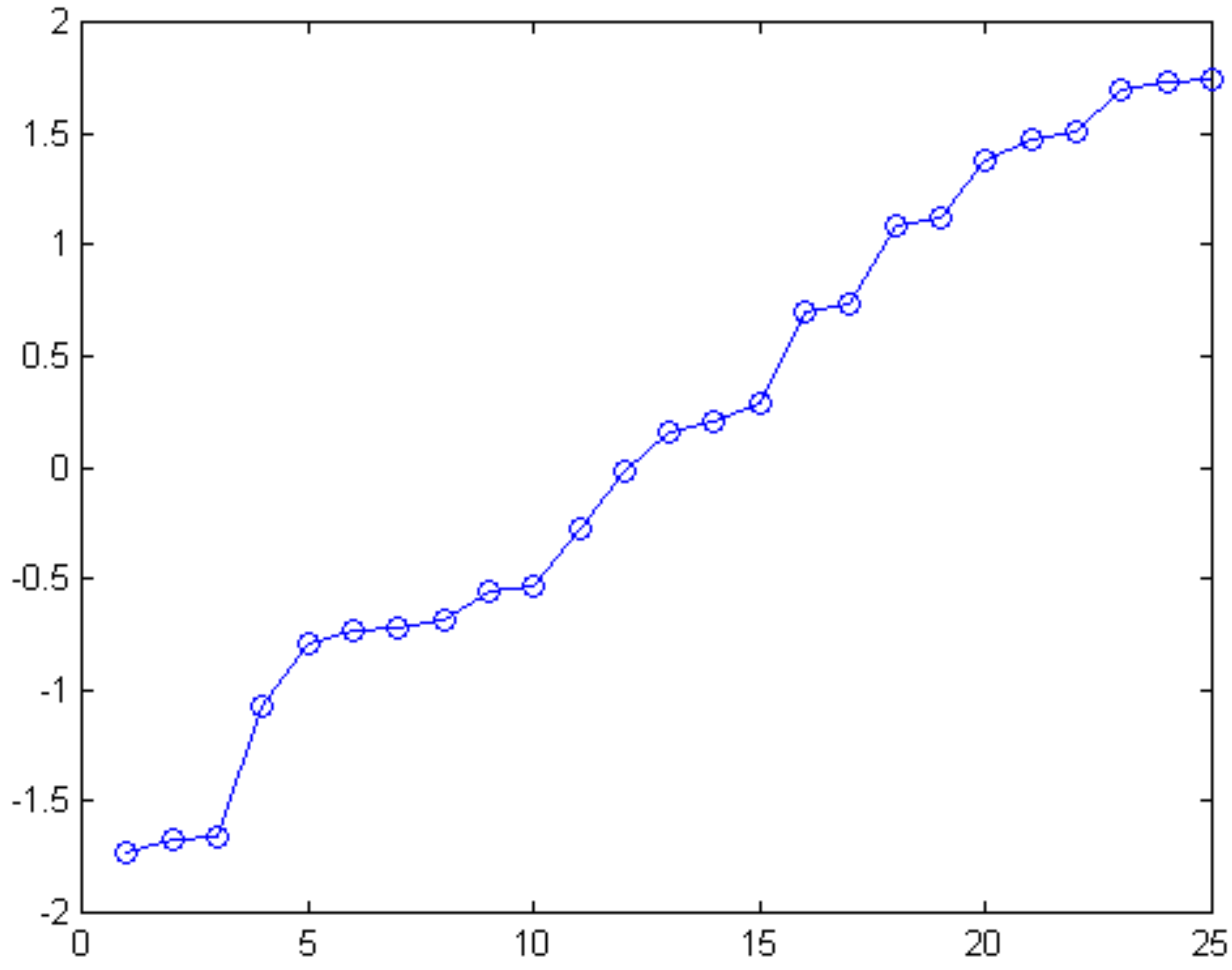
Equal-average Hyperplane Partitioning Method

- 25個片段投影在對應到最大特徵值的特徵向量後的座標平均值從小到大排序



Equal-average Hyperplane Partitioning Method

- 25個片段投影在對應到第二大特徵值的特徵向量後的座標平均值從小到大排序

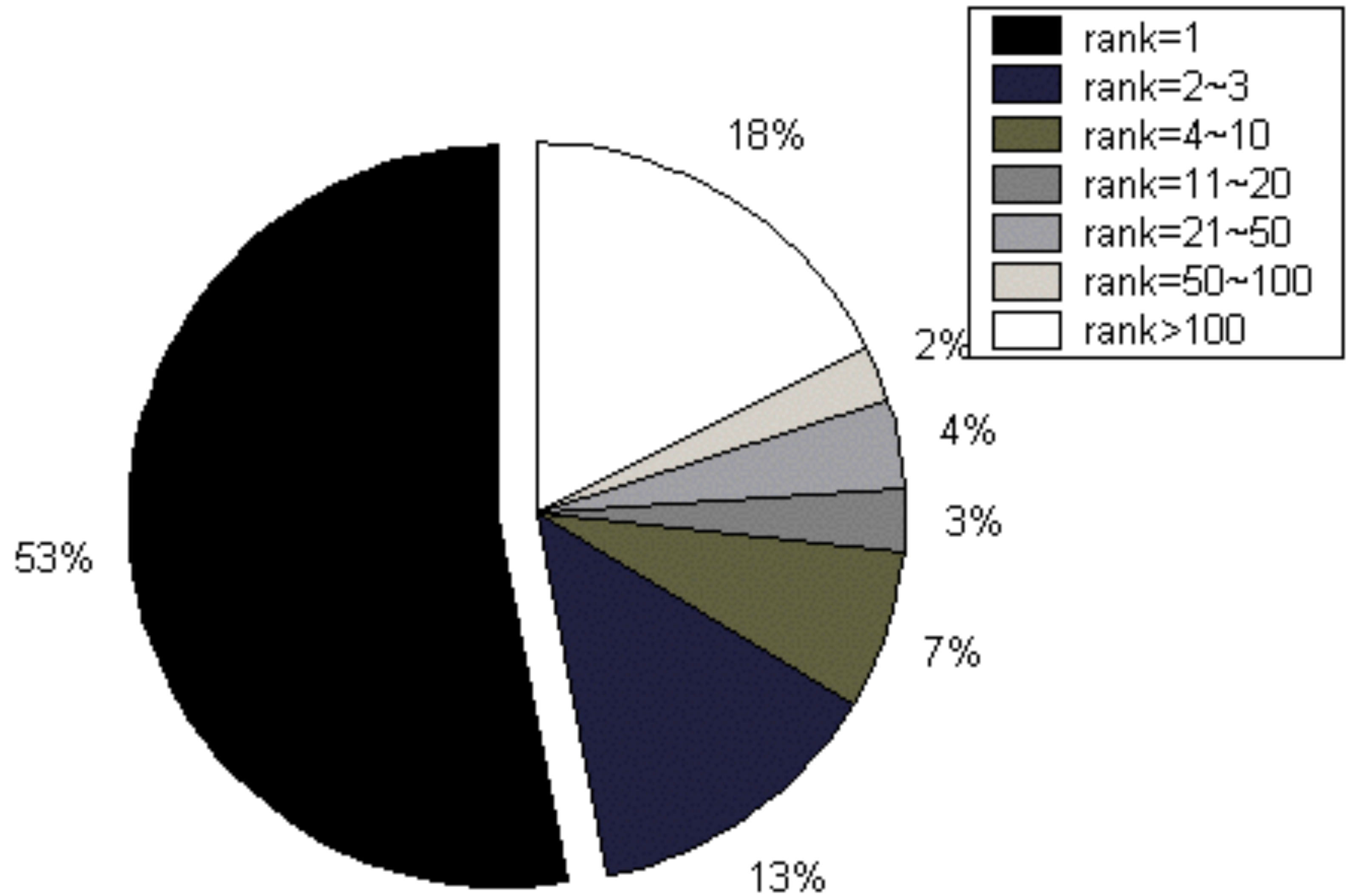


實驗結果

- 實驗環境
  - 電腦 PIII 800，256MB RAM，Windows 2000
  - 資料庫
    - 8552首，包括中文、台語、英文和日文歌曲
  - 測試歌聲wav檔
    - 從頭唱 1054 首
    - 從任意處唱 1650 首
    - 每一首長 8 秒鐘

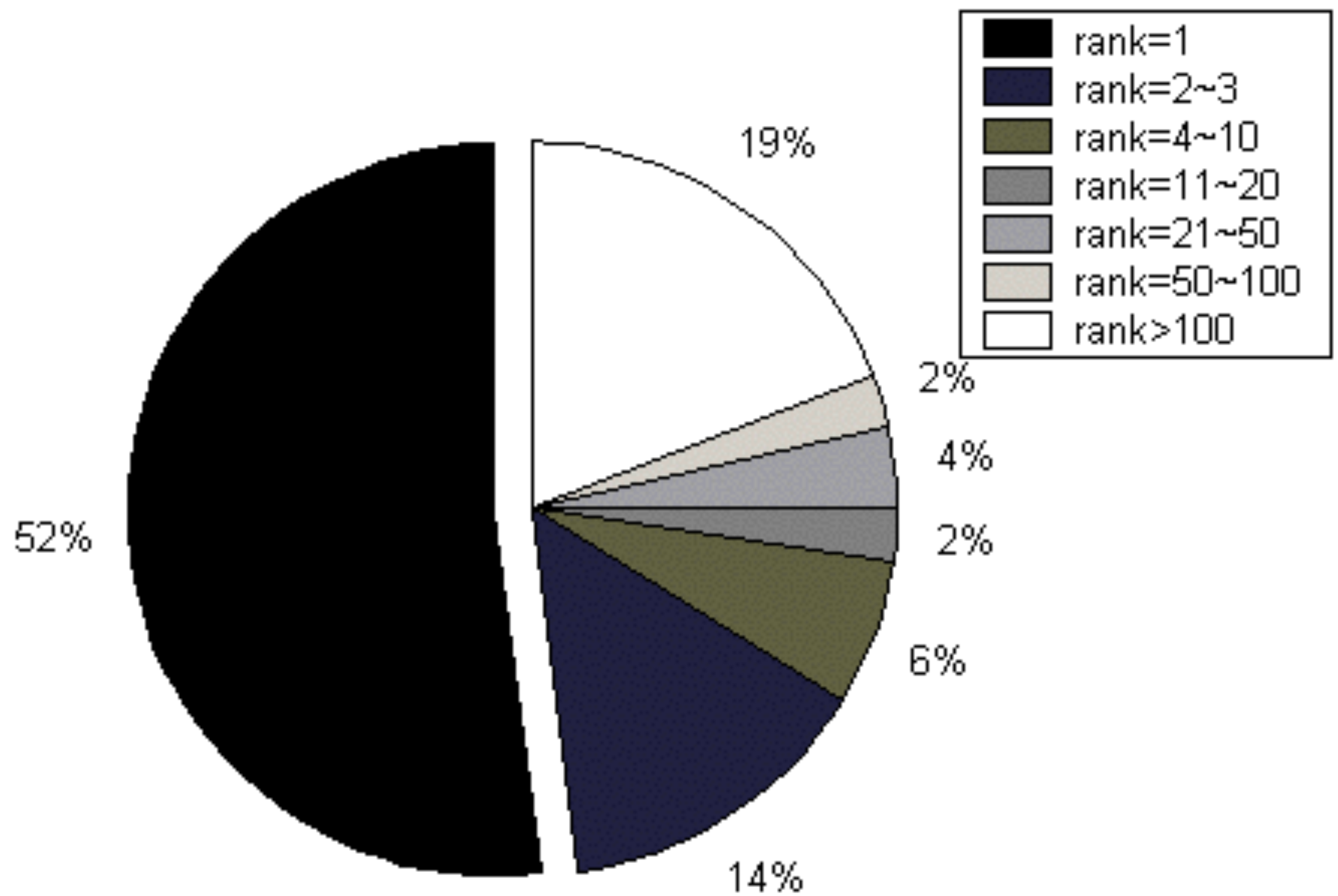
實驗結果—從頭比對

- 只用dtw
- 前三名66.41%
- 平均搜尋時間 16.71秒



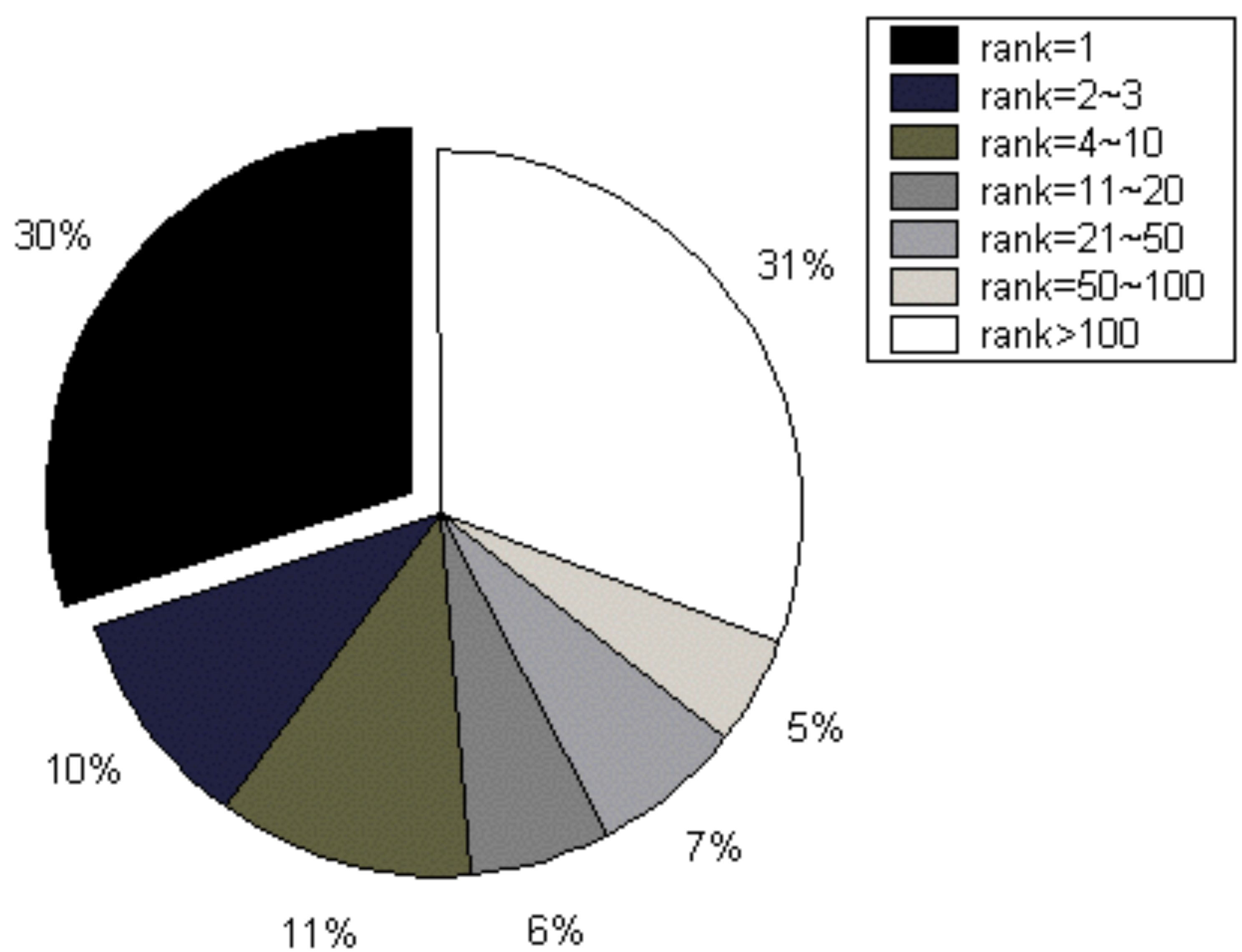
實驗結果—從頭比對

- 只用改良式dtw，保留100名距離
- 前三名66.13%
- 平均搜尋時間6.96秒



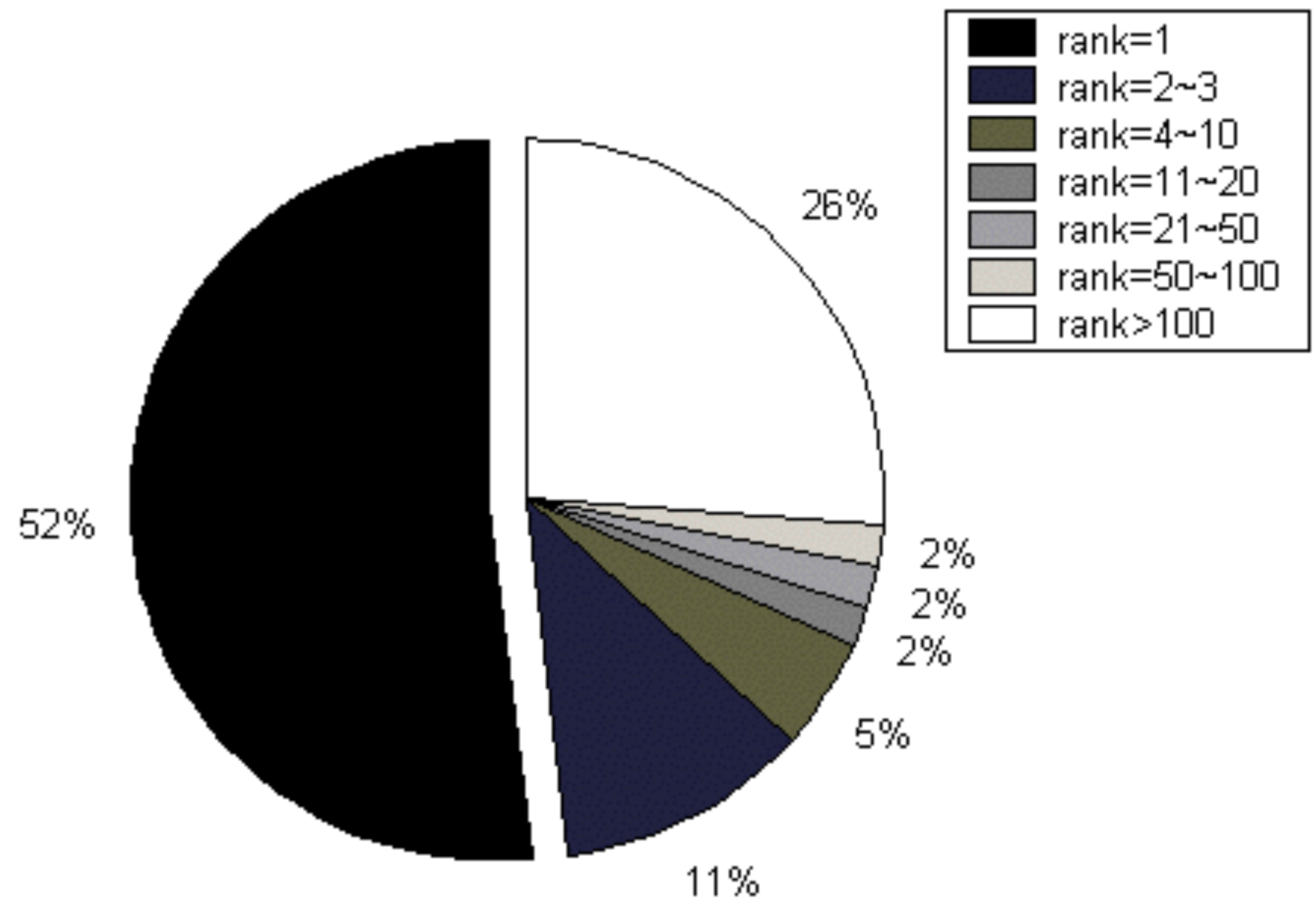
實驗結果─從頭比對

- 只用linear scaling
- 前三名39.94%
- 平均搜尋時間0.1秒

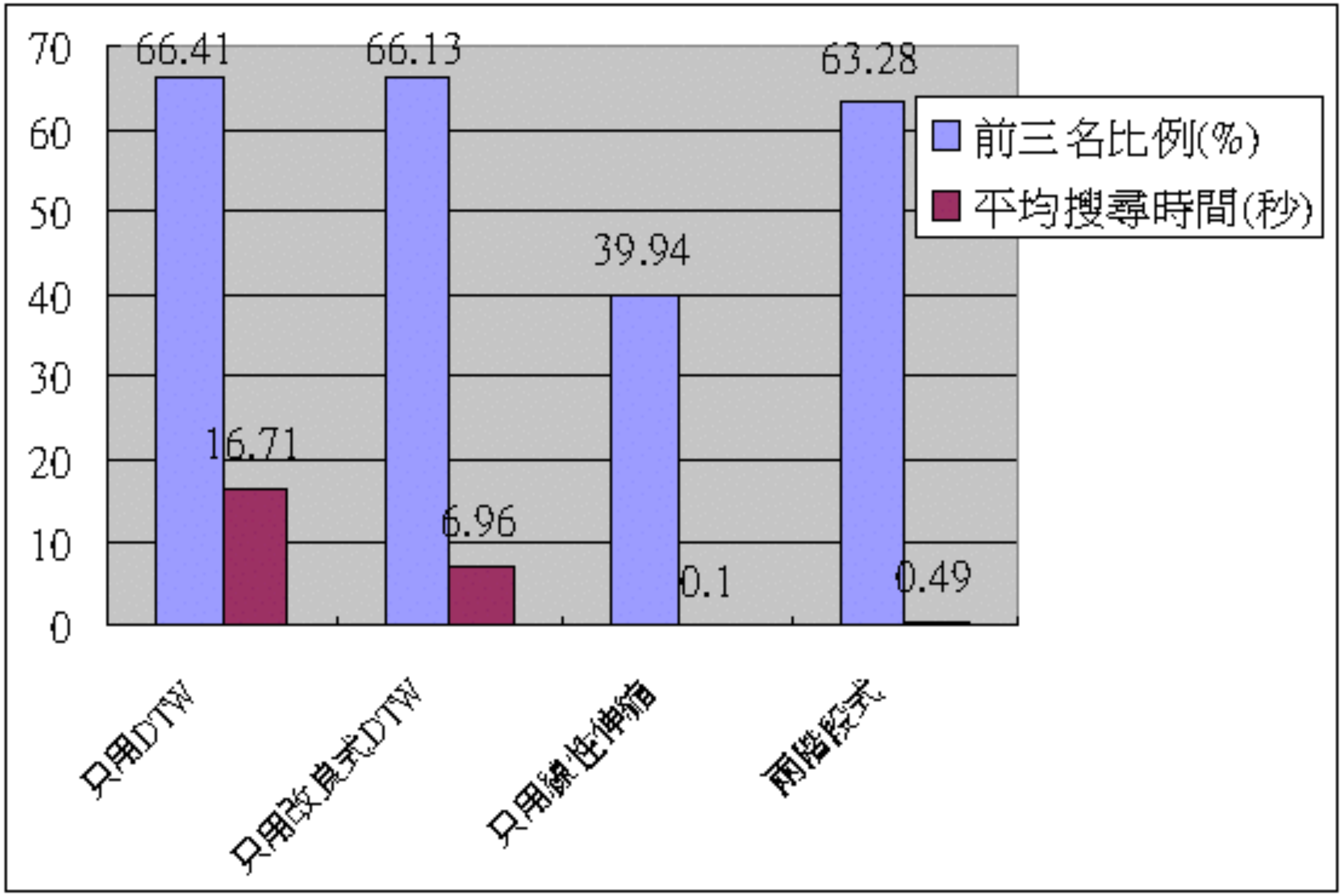


實驗結果─從頭比對

- 用兩階段式比對
- 第一階段保留200首歌進入第二階段
- 前三名63.28%
- 平均搜尋時間0.49秒

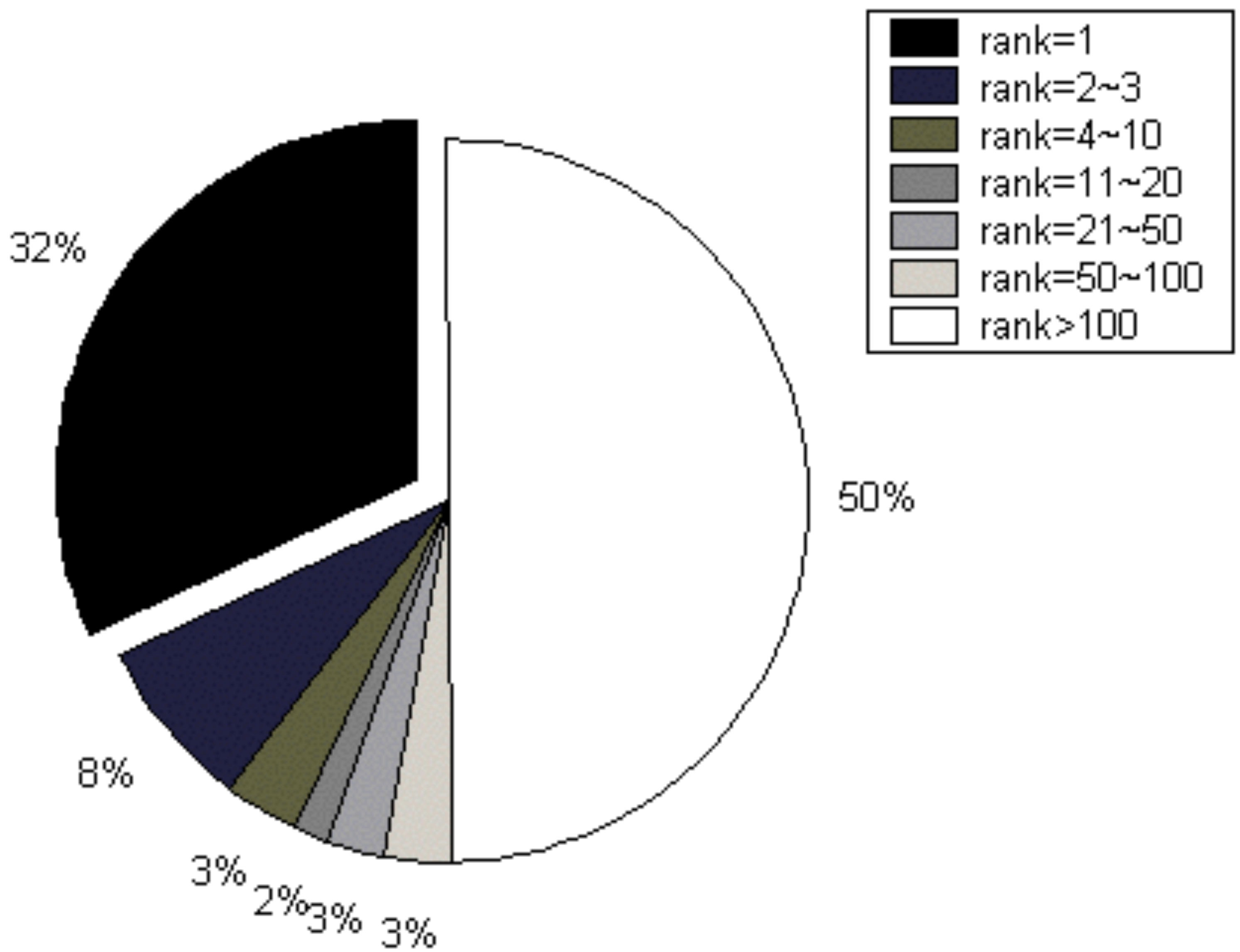


實驗結果—從頭比對比較圖

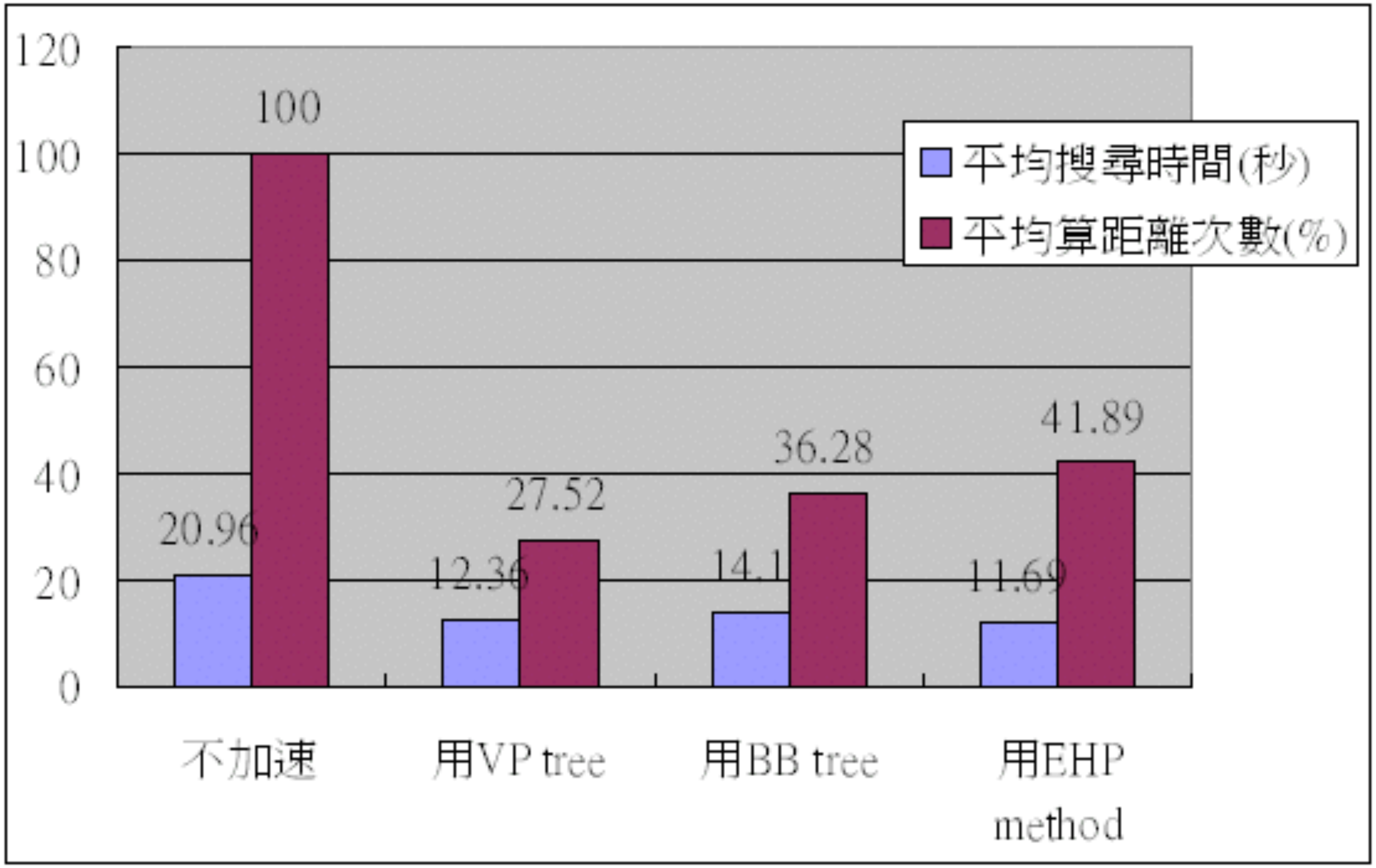


實驗結果—全曲比對

- 用兩階段式比對
- 前三名39.63%



實驗結果—全曲比對比較圖



錯誤分析

- 資料庫錯誤
  - 有前奏
  - 同一時間不只一個音符
  - 主旋律錯誤
  - 相同歌曲但不同歌名
- 測試歌聲不良
  - 有雜音
  - 氣不足、音不準
  - 唱錯
  - 拍子不準

結論

- 從頭比對
  - 只用dtw → linear scaling + 改良式dtw
    - 大約快 34 倍
    - 前三名比例下降不到 4%
- 全曲比對
  - 只用改良式dtw → linear scaling + 改良式dtw
    - 大約快 23 倍
  - 第一階段不用加速方法 → 第一階段用equal-average hyperplane partitioning method
    - 大約快 1.8 倍
    - 平均一首歌約多佔 2.7 K byte 的記憶體

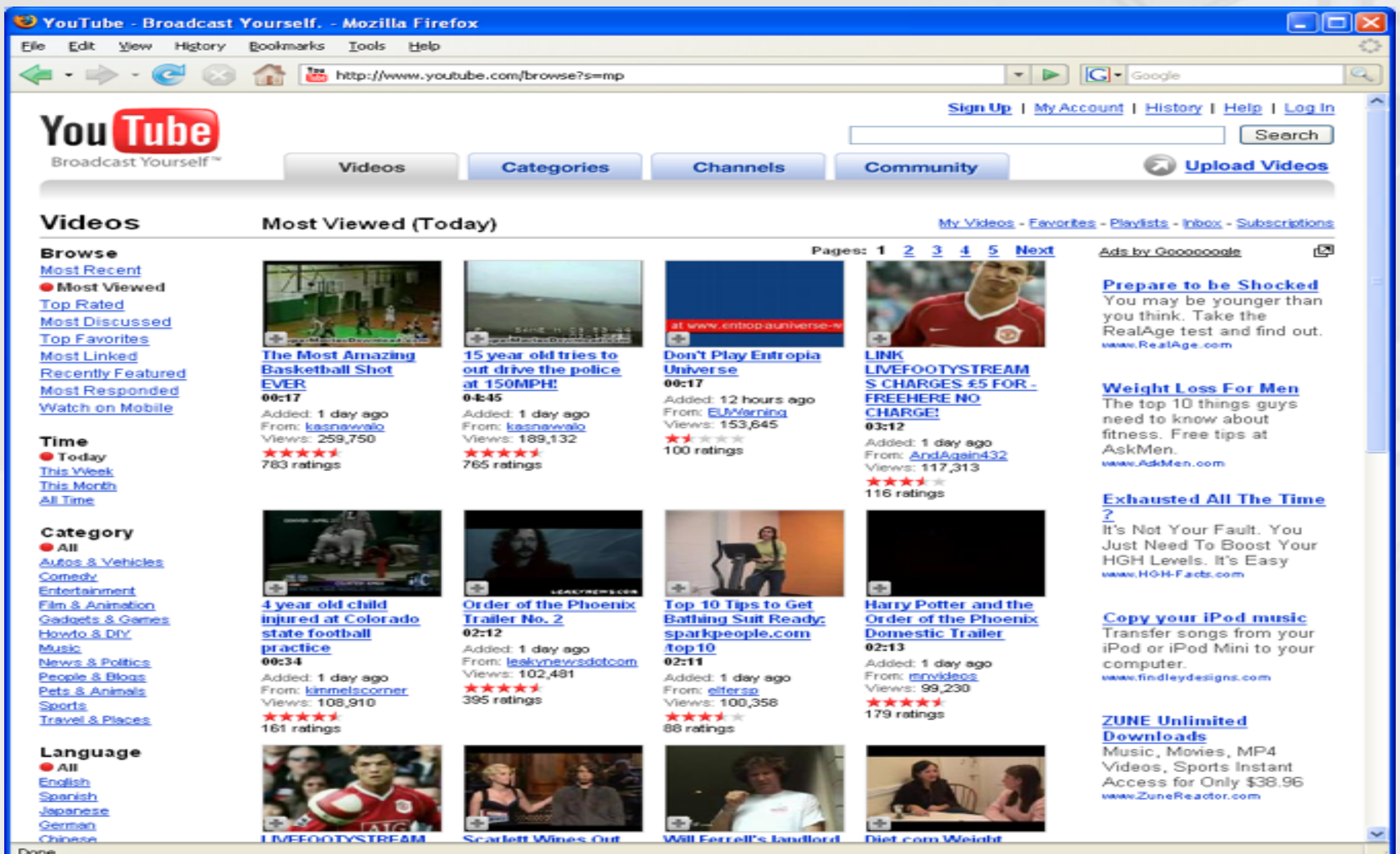
3. Video retrieval techniques

Differences and relations between image and video

- Images are **static**, but video are **dynamic**.
- Video stream can be viewed as sequence of image frames.

CBVR

- Sample YouTube Video page:



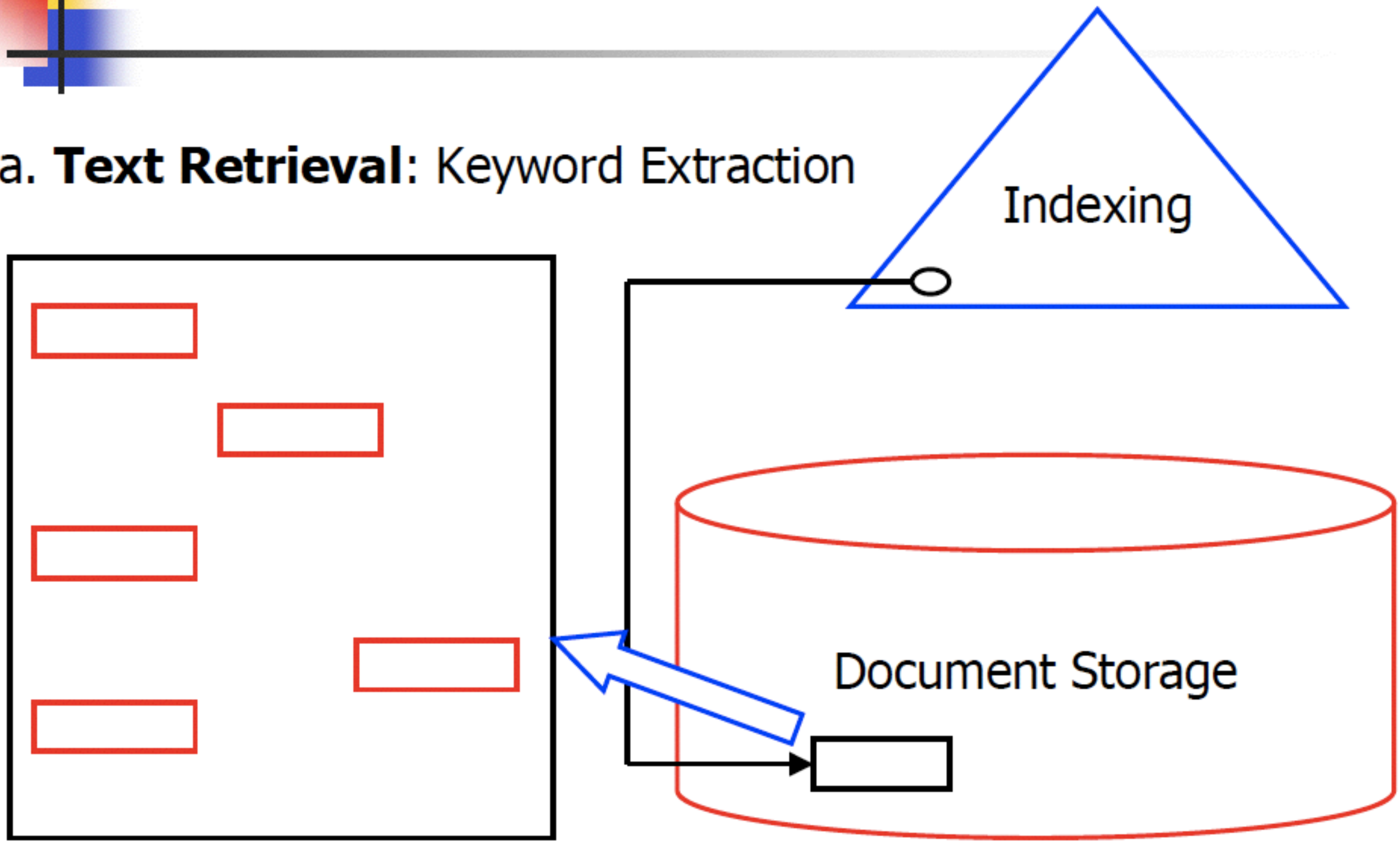
Main methods of digital media retrieval

- **Text-based** digital media retrieval

- **Content-based** digital media retrieval

Why we need video shots?

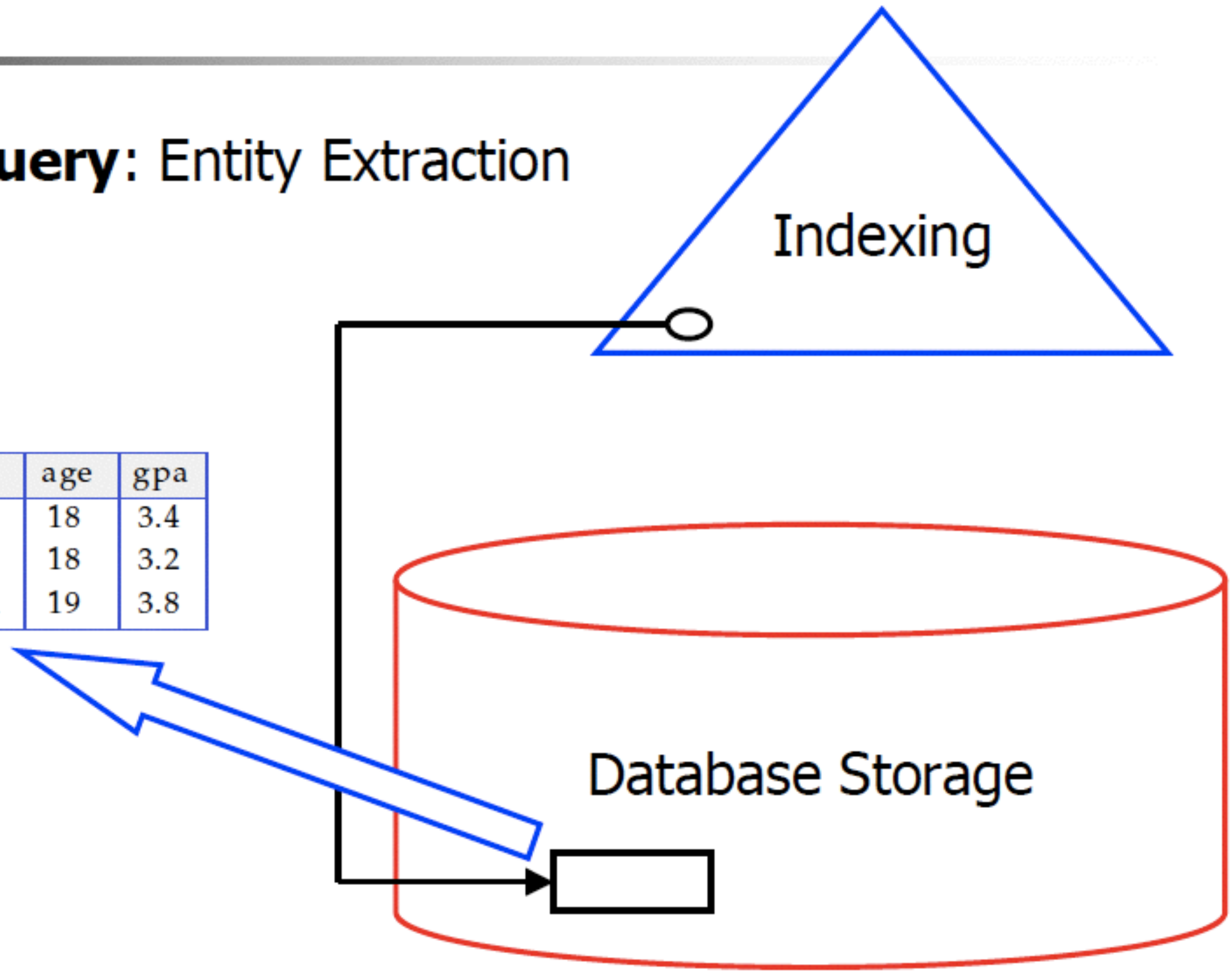
a. **Text Retrieval**: Keyword Extraction



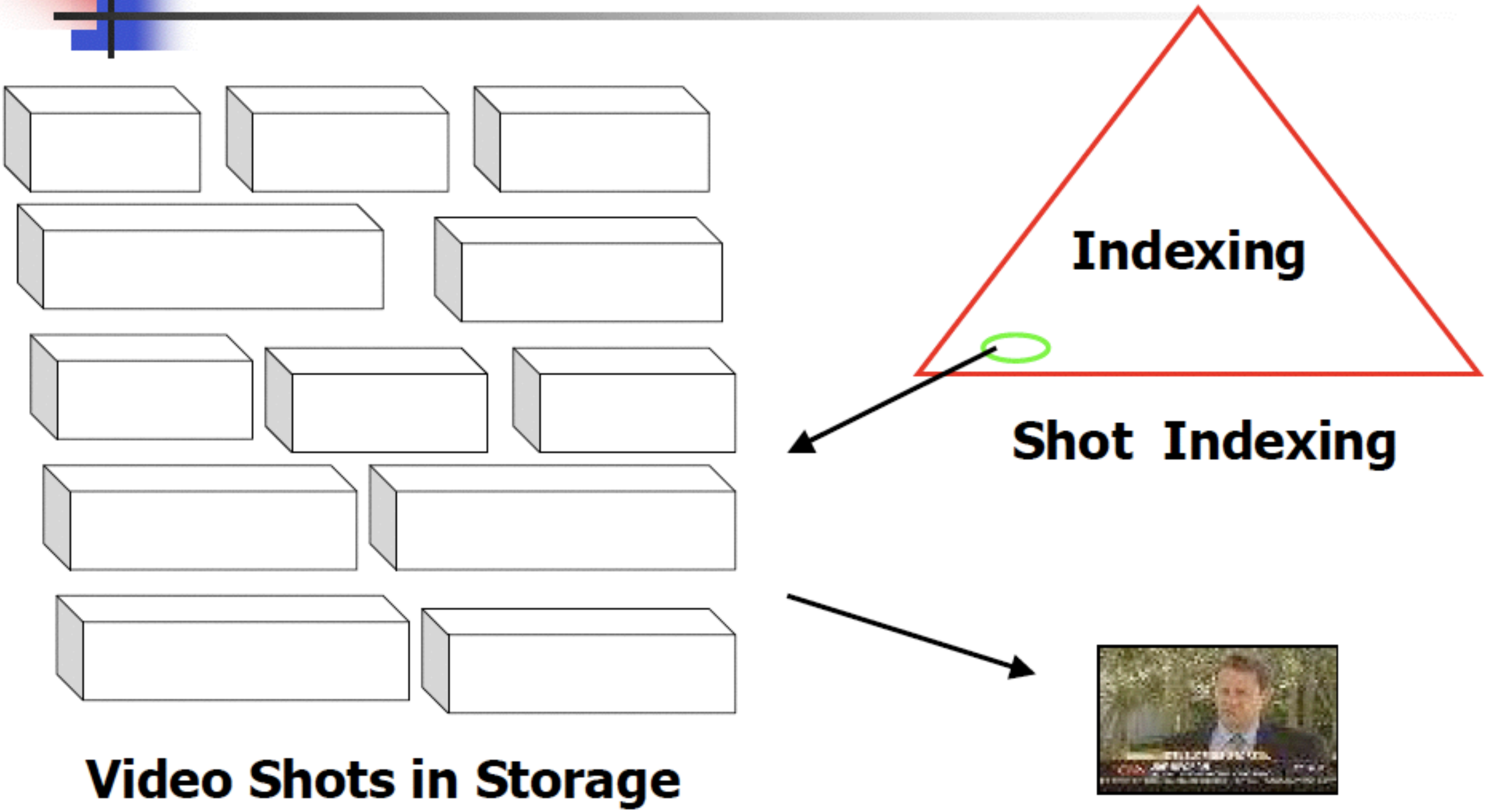
Why we need video shots?

b. **Database Query**: Entity Extraction

sid	name	login	age	gpa
53666	Jones	jones@cs	18	3.4
53688	Smith	smith@eecs	18	3.2
53650	Smith	smith@math	19	3.8

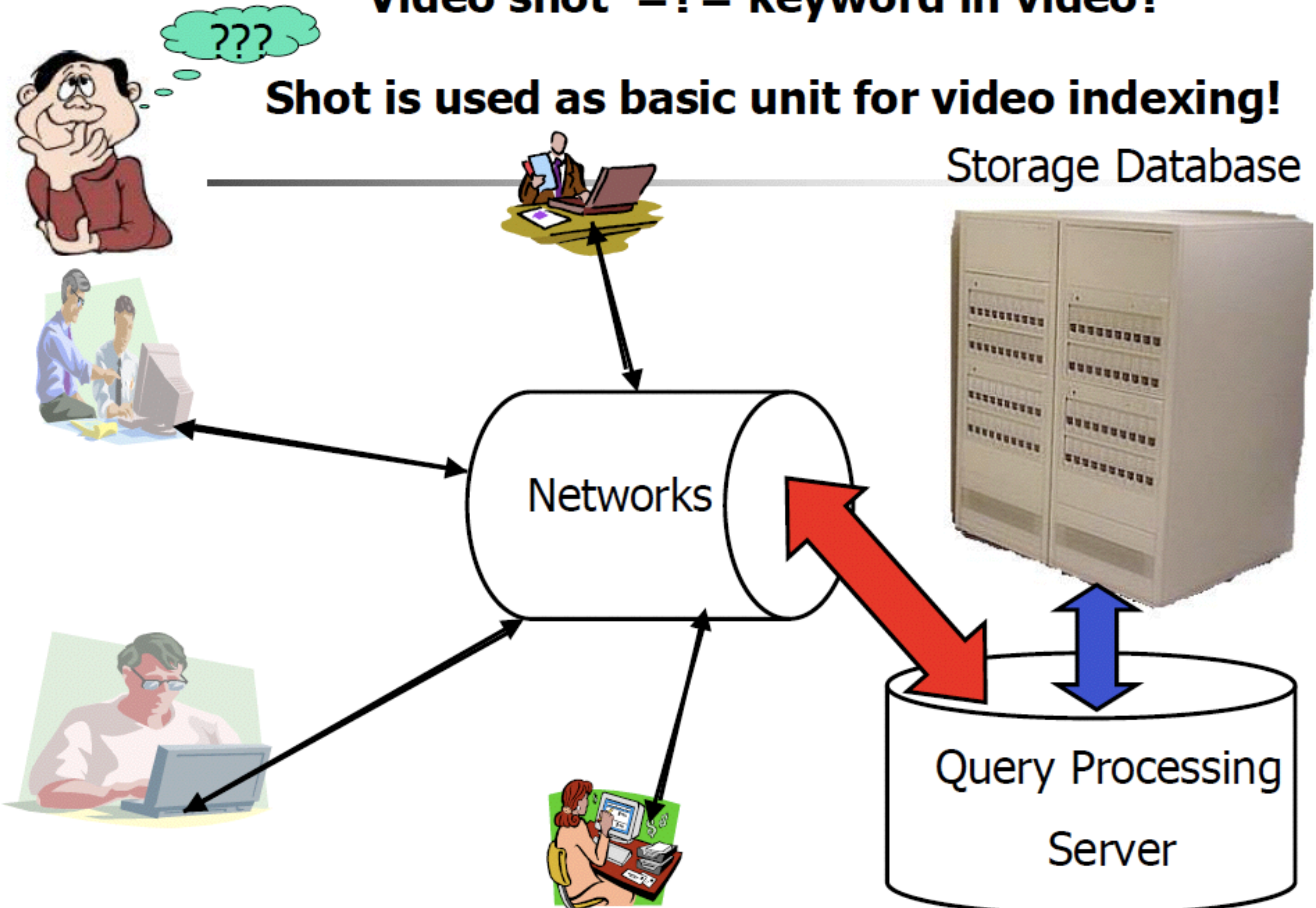


Why we need video shots?



Video shot =?= keyword in video?

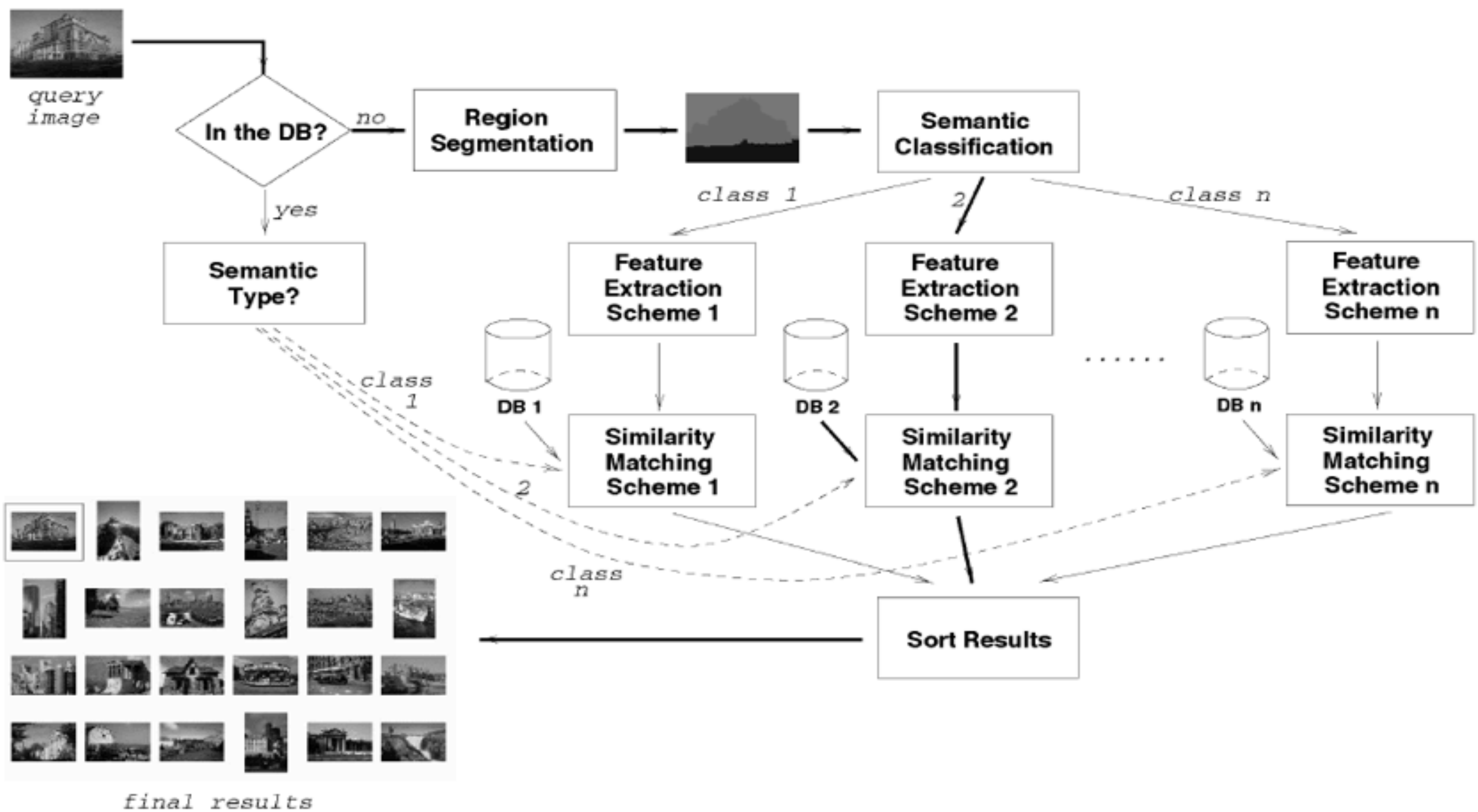
Shot is used as basic unit for video indexing!



CBVR Overview

- 2 phases:
  - –Database Population phase
    - Video shot boundary detection
    - Key Frames selection
    - Feature extraction
  - –Video Retrieval phase
    - Similarity measure

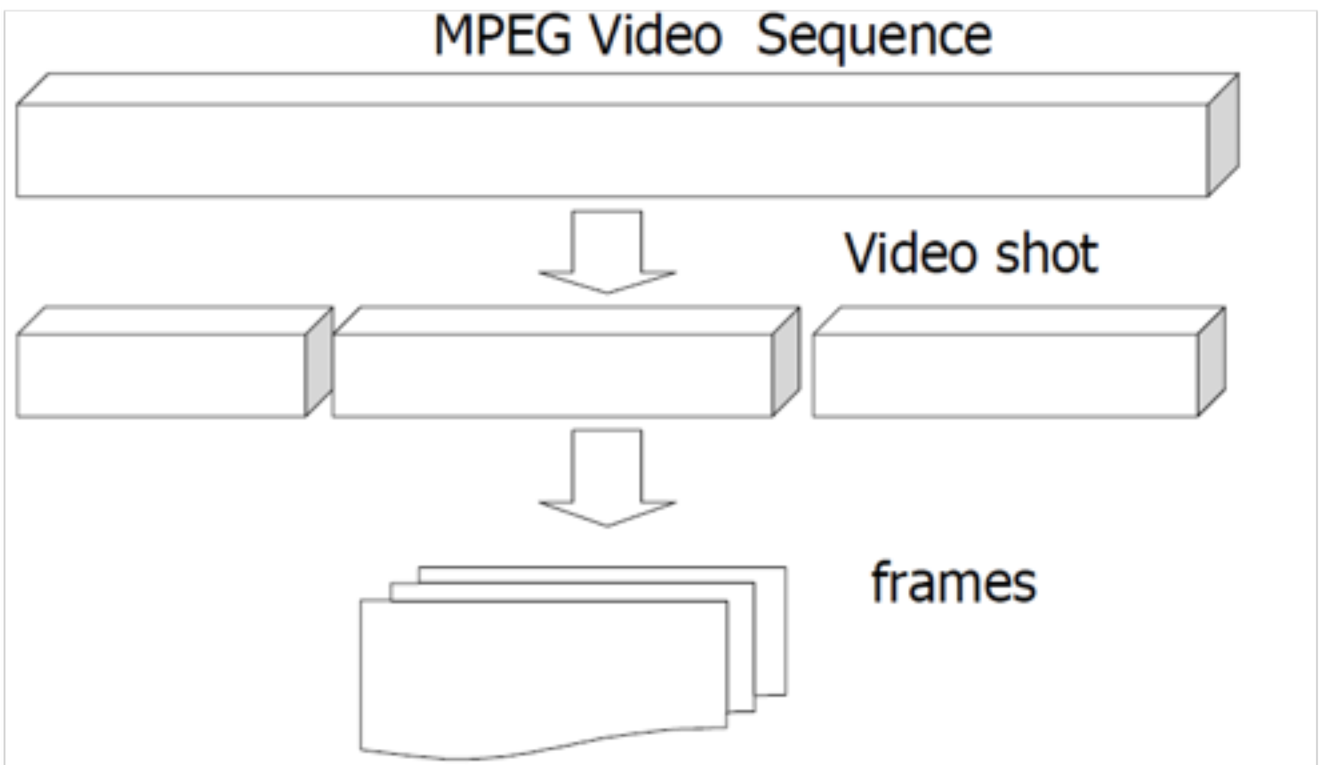
Overview (cont.)



[Wang, Li, Wiederhold, 2001]

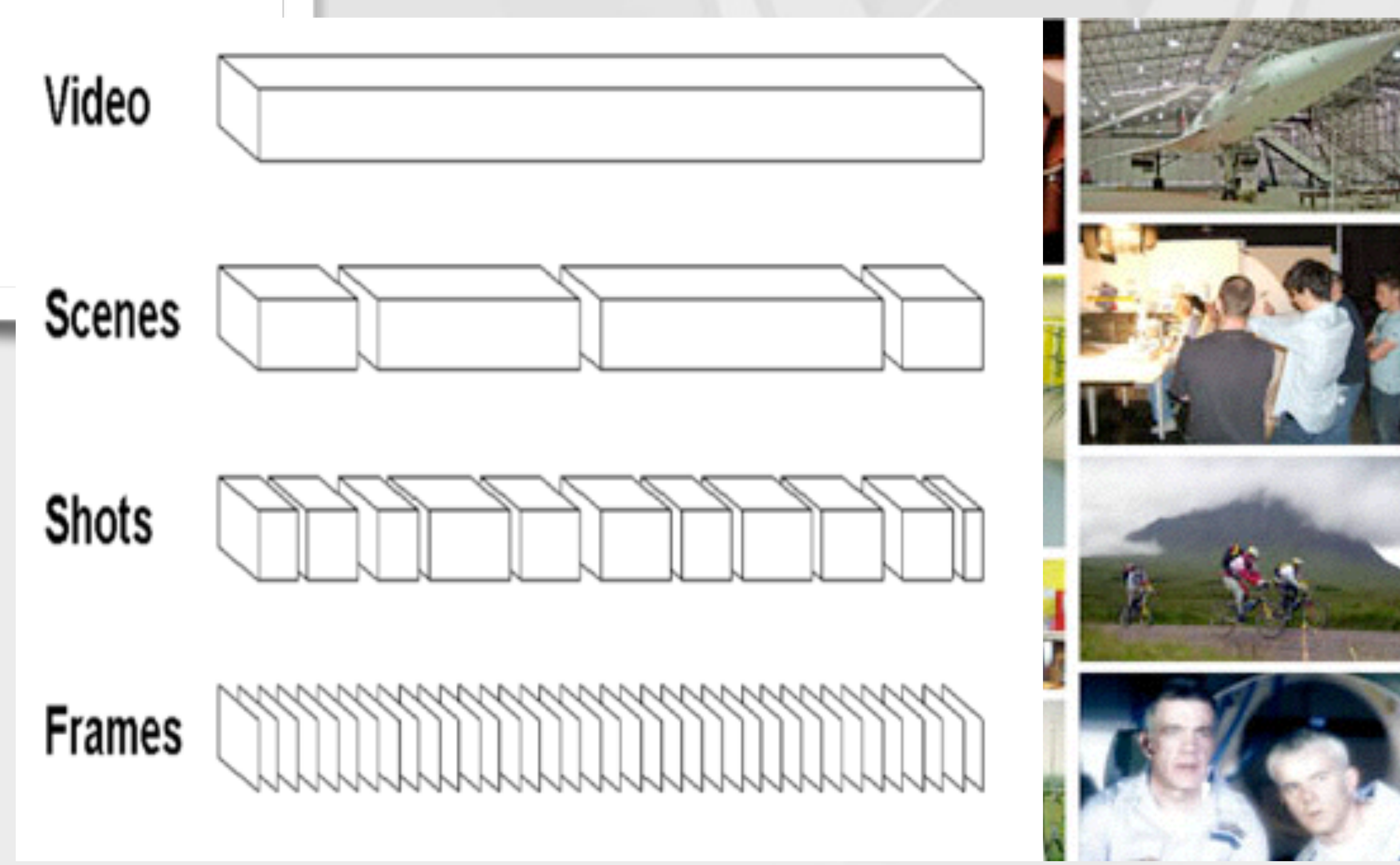
Structuralizing video data

- **semantic content layers**, e.g., scenes and shots in a video program.
  - These layers are erased when they are displayed for audience, which weakens the ability for user dealing with raw video data.



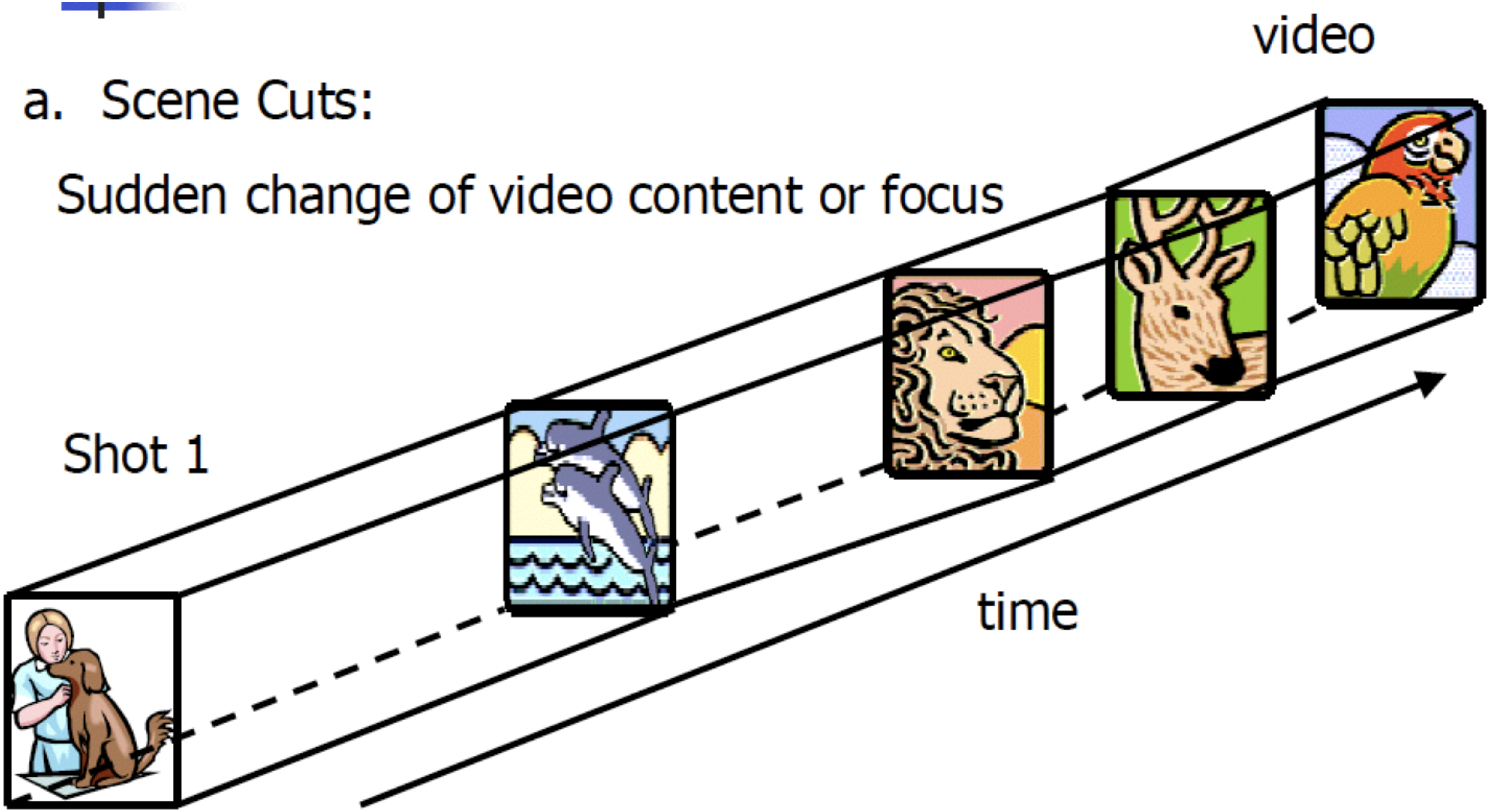
Fundamental definitions in video structurization

- Frame (帧)
- Shot (镜头)
- Key frame (关键帧)
- Scene (场景)
- Group (组)



a. Scene Cuts:

Sudden change of video content or focus



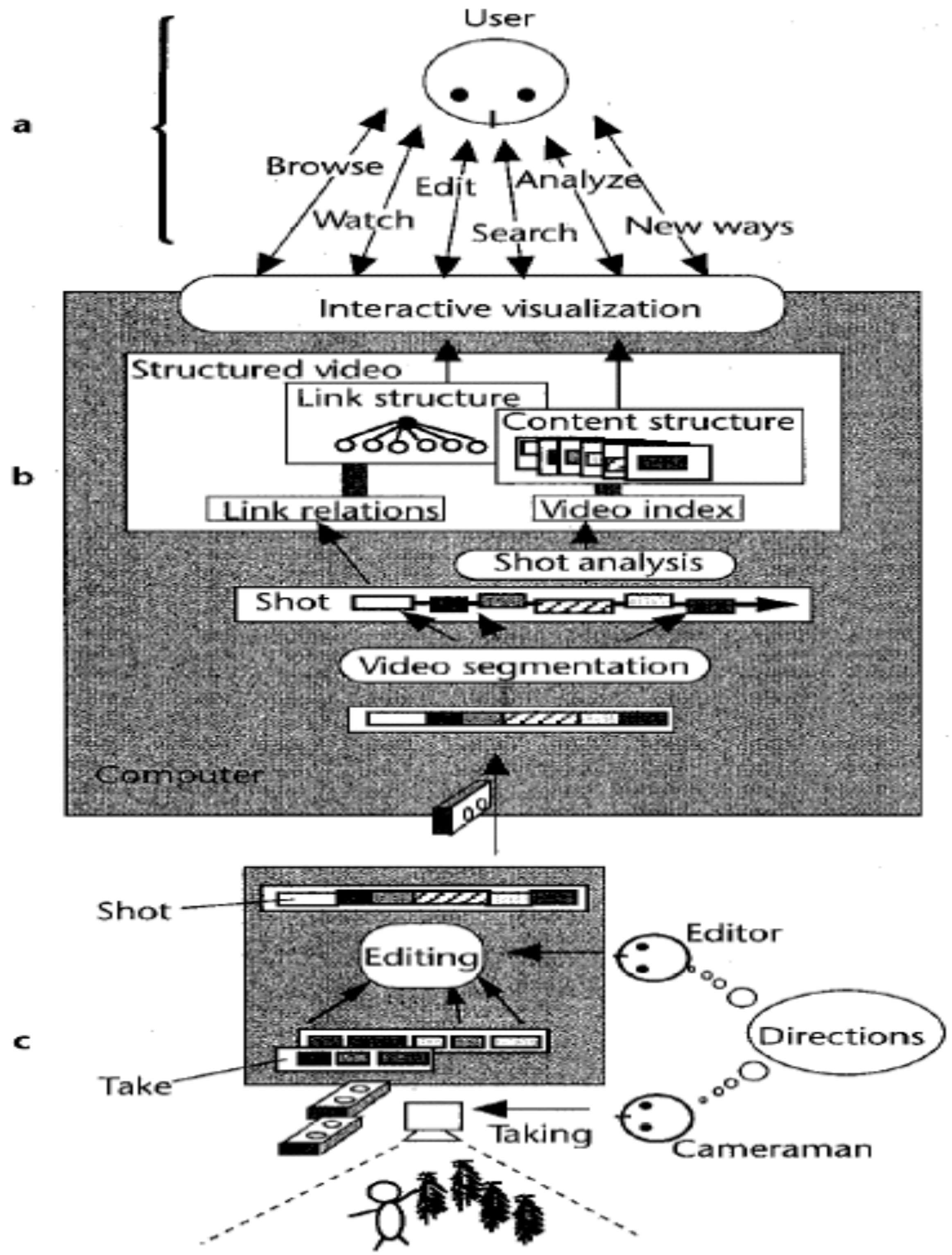
Proposal

- Analyze a video stream
- Segment the stream into shots
- Index shots using extracted features
  - Camera work characteristics (Long, Middle, Short ...)
  - Color representations
- Browsing methods and user interfaces

Desired Video Interaction

- Focus on fast visual browsing.
- Ability to grasp idea of lengthy video in short time.
- Not simply fast forward.
- Challenge: find and manage essential visual cues, then present them visually in an effective way

Viewer-Video Interaction: Conceptual Model

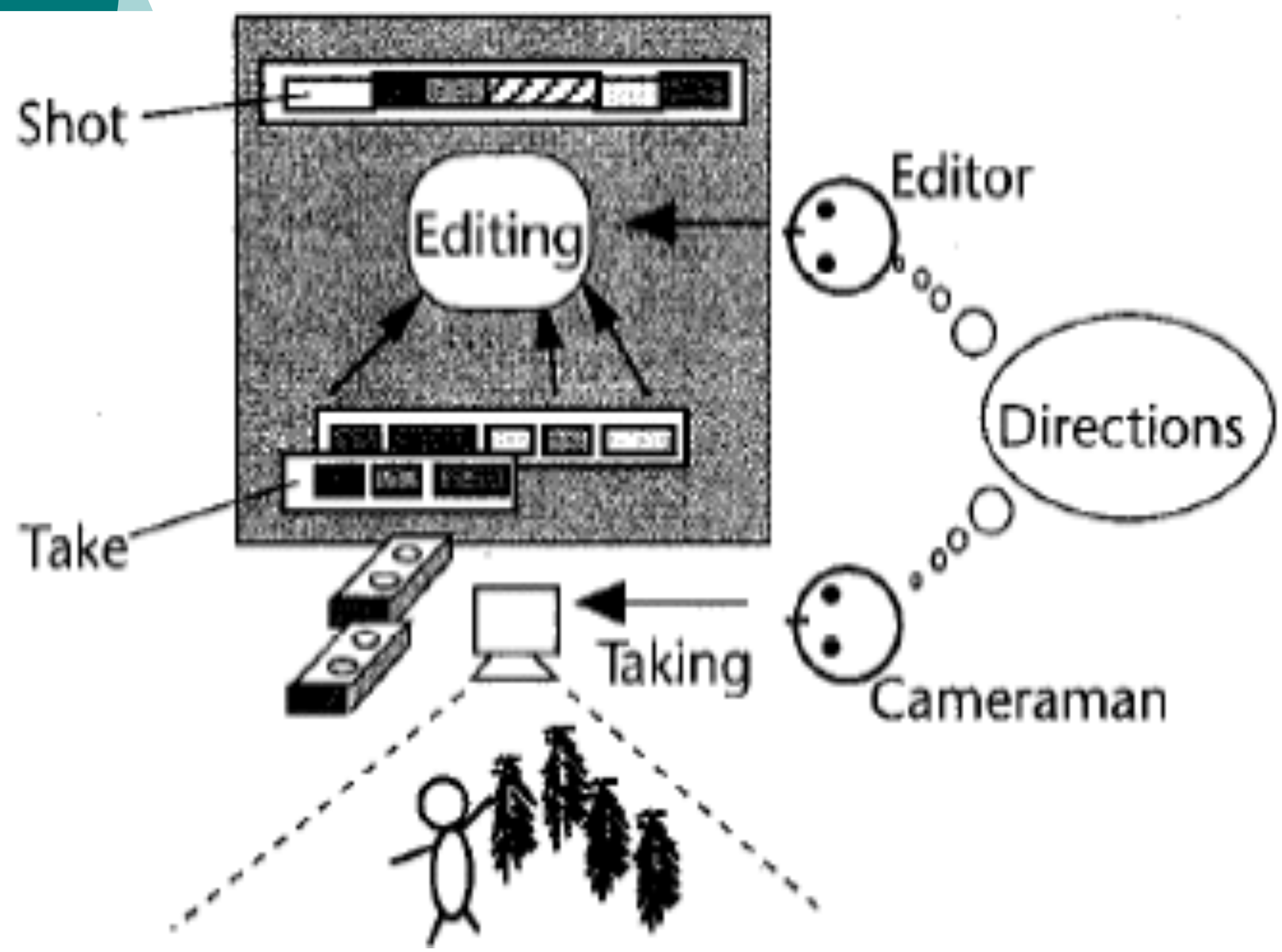


a) Viewer Interaction

b) Video Computing

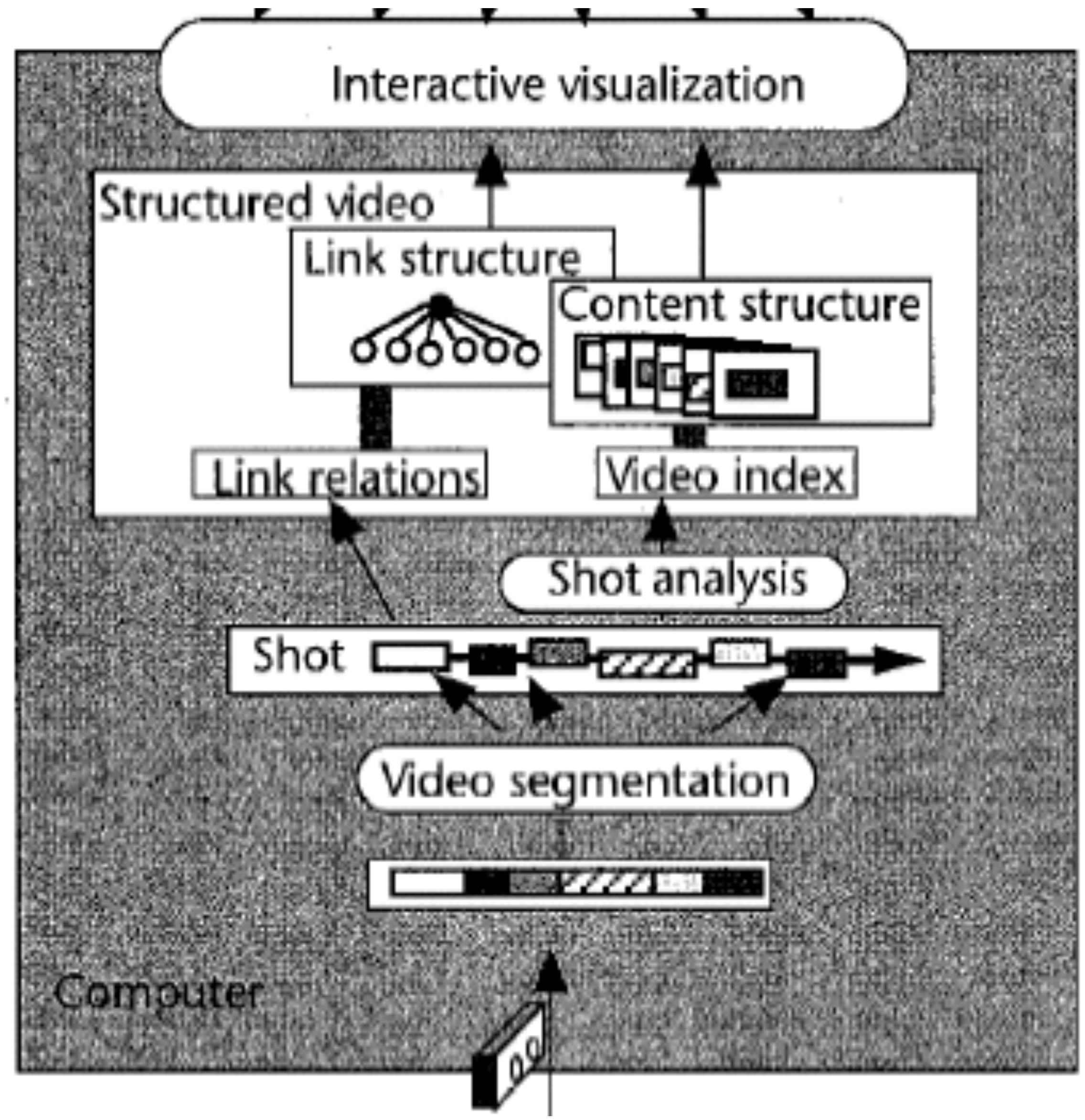
c) Video Production & Editing

Video Production



- Key Concepts:
  - Take: continuous video
  - Cut: separates takes
  - Camera characteristics
    - Pan, tilt, zoom, etc.
  - Shot: edited takes
- Resulting video contains embedded info: cut points, camera characteristics

Video Computing



- Main Function: Make the implied video structure explicit.

Video Segmentation: Problems

- Traditional Cut Detection – detect differences between frames using inter-frame comparisons (intensity, RGB, motion vectors).
- Mis-detection due to rapid object motion, slow motion, animation, strobes, fading, wiping, dissolving, etc.
- Result: Low successful detection rate.

Basic video segmentation metrics

- Pair-wise comparison
  - Pixel-level
    - Sensitive to camera movement and motion
  - Block-level (Likelihood ratio)
    - Can tolerate small motion

$$DP_i(k,l) = \begin{cases} 1 & \text{if } |P_i(k,l) - P_{i+1}(k,l)| > t \\ 0 & \text{otherwise} \end{cases}$$

$$\frac{\sum_{k,l=1}^{M,N} DP_i(k,l)}{M * N} * 100 > T$$

$$\frac{\left[\frac{S_i + S_{i+1}}{2} + \left(\frac{m_i - m_{i+1}}{2}\right)^2\right]^2}{S_i * S_{i+1}} > t$$

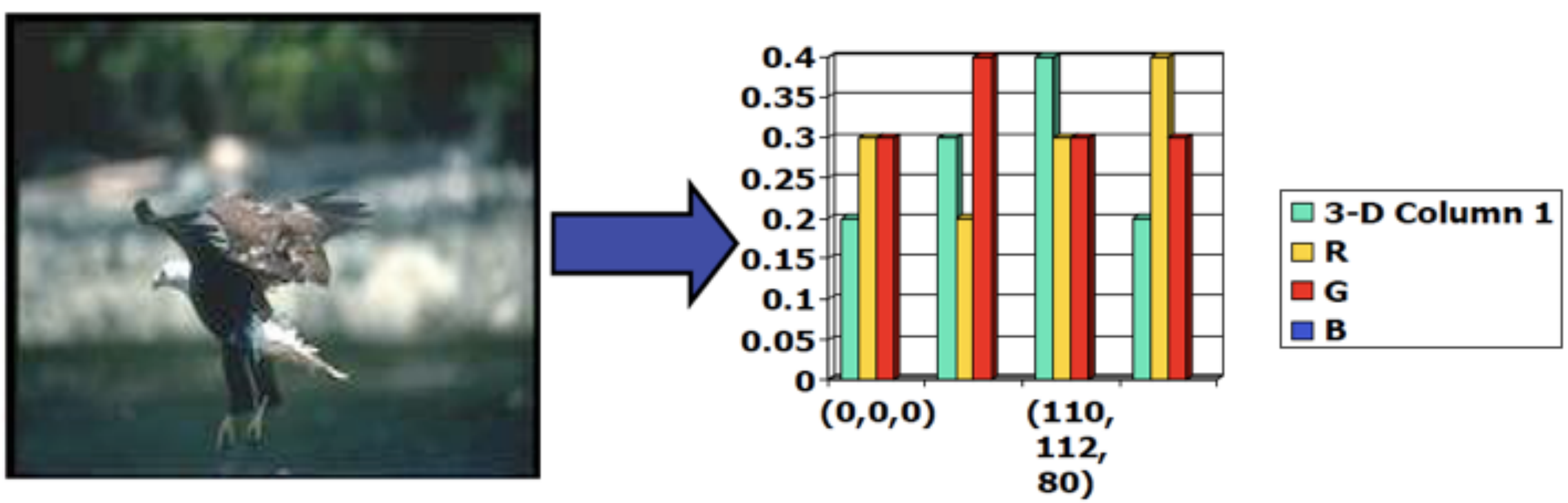
mi: mean intensity

Si: corresponding variance

Basic video segmentation metrics

How to measure statistical property of video frames?

Color Histogram



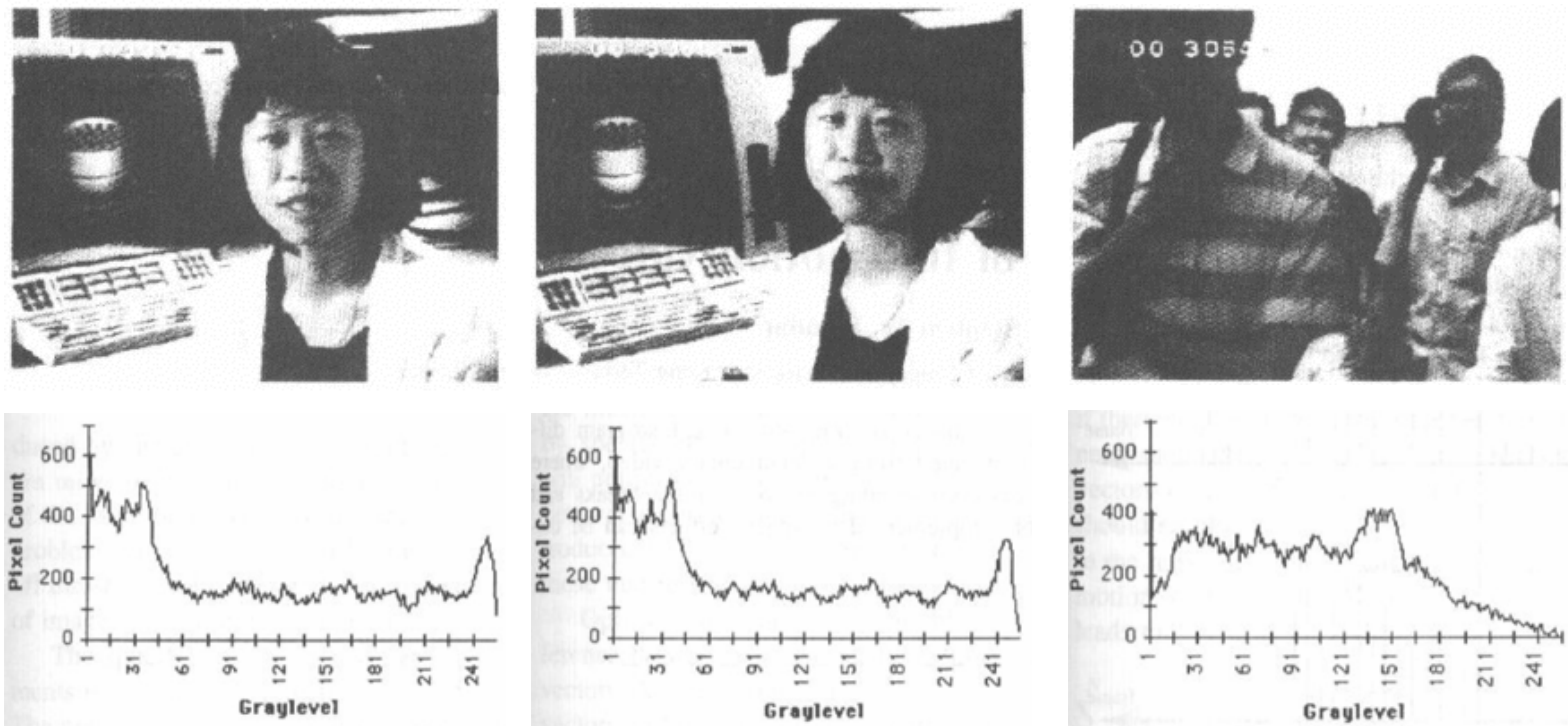
Basic video segmentation metrics

- Histogram comparison
  - Basic
    - Tolerate motion better
  - χ2-test
  - Color level can also be used but only the MSB to save the number of bins

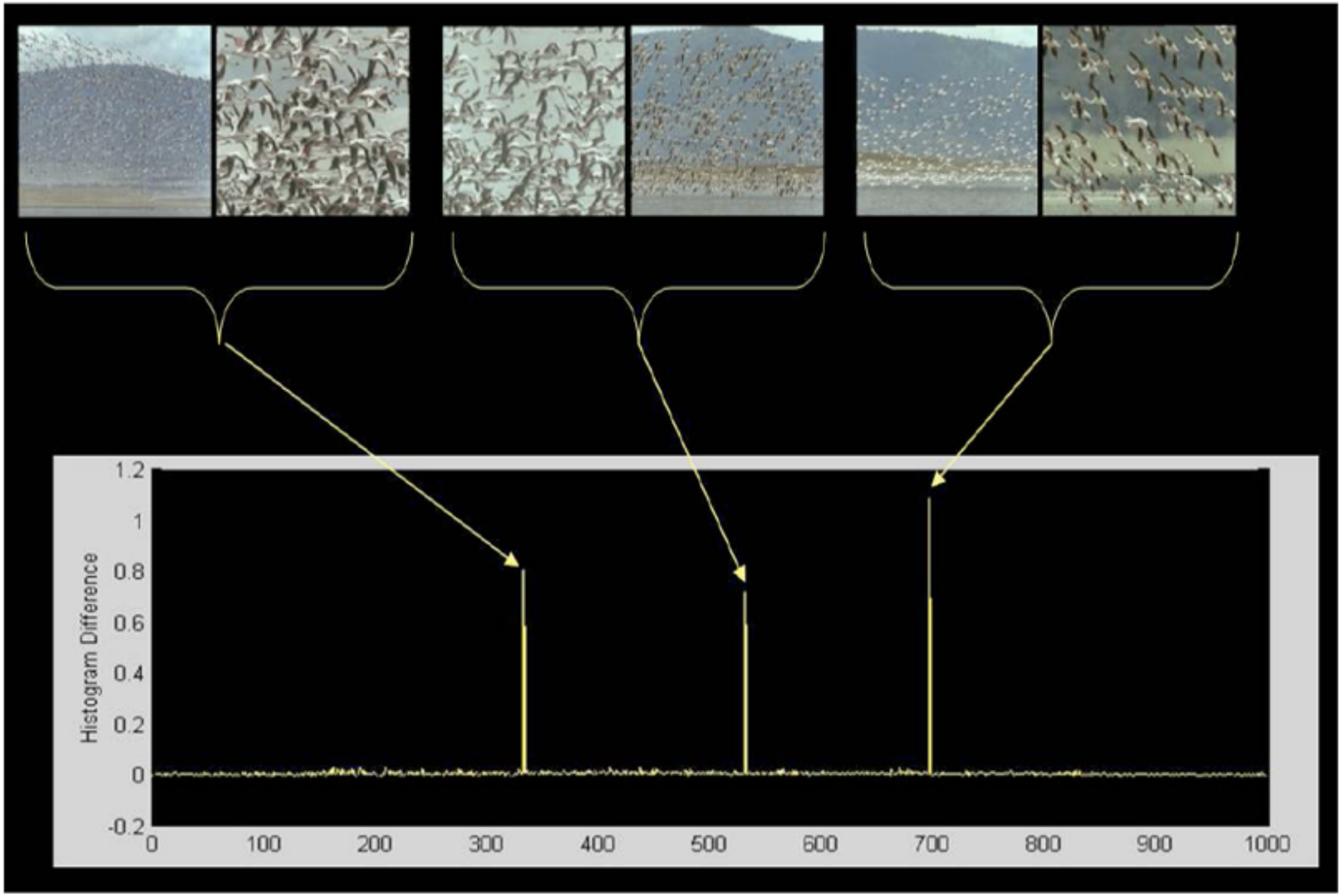
$$SD_i = \sum_{j=1}^{G} |H_i(j) - H_{i+1}(j)|$$

$$SD_i = \sum_{j=1}^{G} \frac{|H_i(j) - H_{i+1}(j)|^2}{H_{i+1}(j)}$$

Sample of using histogram

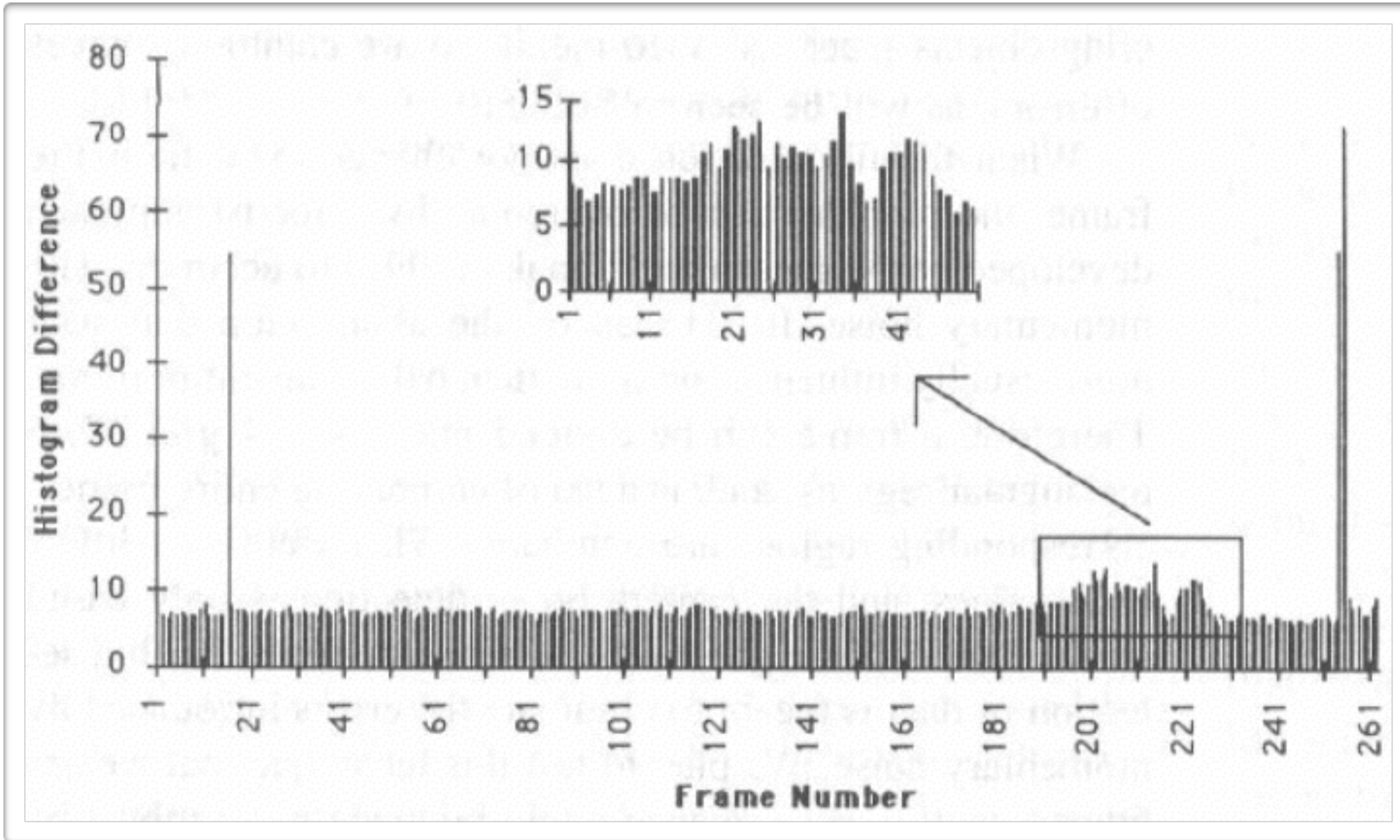


Scene Cut



Histogram Difference
1.2
1
0.8
0.6
0.4
0.2
0
-0.2
0
100
200
300
400
500
600
700
800
900
1000

Gradual transition detection



Gradual transition detection

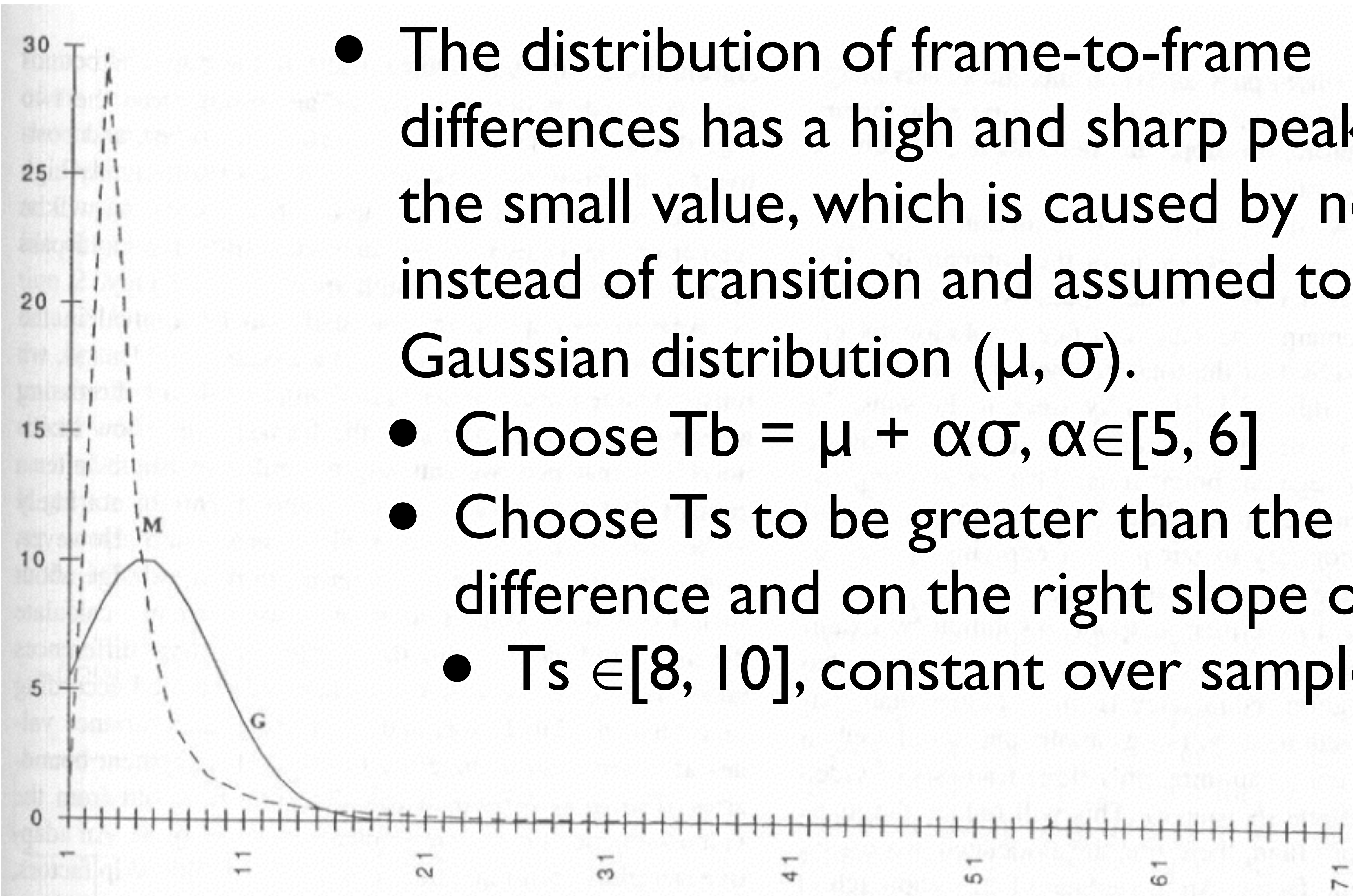
- Twin-comparison
  - Use two thresholds Tb and Ts to accommodate both **short-term** and **long-term** transitions
  - Differences of (F1, F2), (F2, F3), (F3, F4) are small, but difference of (F1, F4) is still big

1 2 3 4

- Twin-comparison
  - F_s— the potential beginning frame of the transition
  - F_e— the ending frame of the transition

```
scan frame
if (Diff(F_i) ≥ T_b)
    detect as camera break
else if (T_b > Diff(F_i) ≥ T_s)
    F_s ← F_i
    i ← i + 1
    while (Diff(F_i) ≥ T_s)
        i ← i + 1
    if (Diff(F_s, F_i) ≥ T_b)
        F_e ← F_i
```

Threshold selection (Tb, Ts)



- The distribution of frame-to-frame differences has a high and sharp peak near the small value, which is caused by noise instead of transition and assumed to follow Gaussian distribution (μ, σ).
  - Choose $Tb = \mu + \alpha\sigma$, $\alpha \in [5, 6]$
  - Choose Ts to be greater than the mean difference and on the right slope of M
    - $Ts \in [8, 10]$, constant over samples

Multi-pass approach

- Scanning all frames could be computationally hard
- Temporal skipping is more useful
  - e.g. one out of every 10 frames
  - Better for detecting gradual transition
  - May miss camera break
  - May get false detection (distance increased)
- Multi-pass approach
  - First pass, use either pair-wise or histogram with large skip factor and smaller Tb to collect the potential regions.
  - Second pass, two methods may be applied together (hybrid) to search the candidate regions while increasing the confidence.

Distinguish camera movement

- To distinguish gradual transitions from changes made by camera movements
- Basic approach— observing optical flow via motion vectors

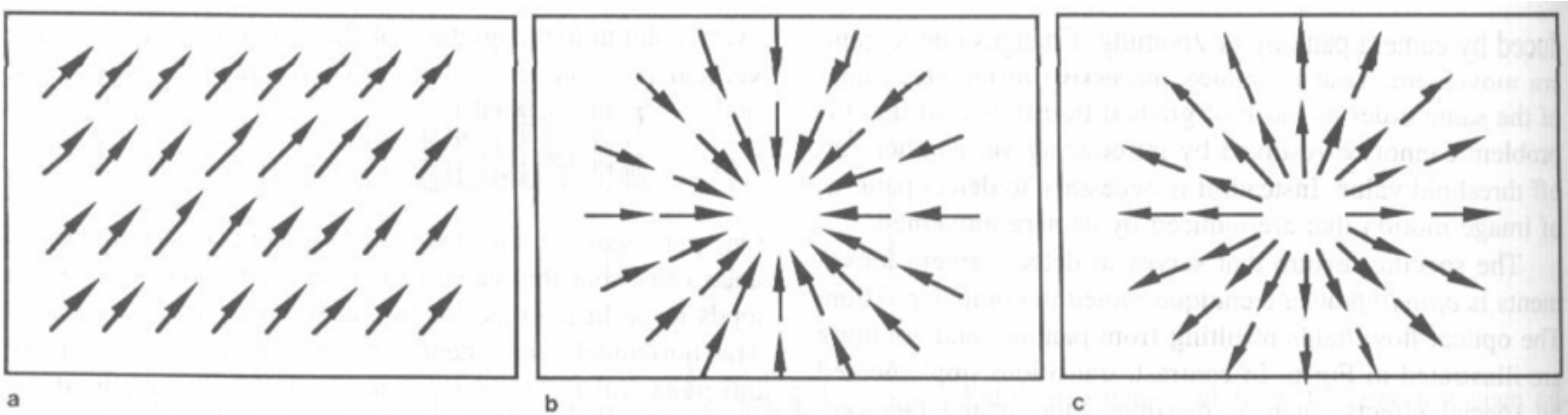


Fig. 6a–c. Motion vector patterns resulting from camera panning and zooming. a ⁄ Camera panning direction. b Camera zoom-out. c Camera zoom-in

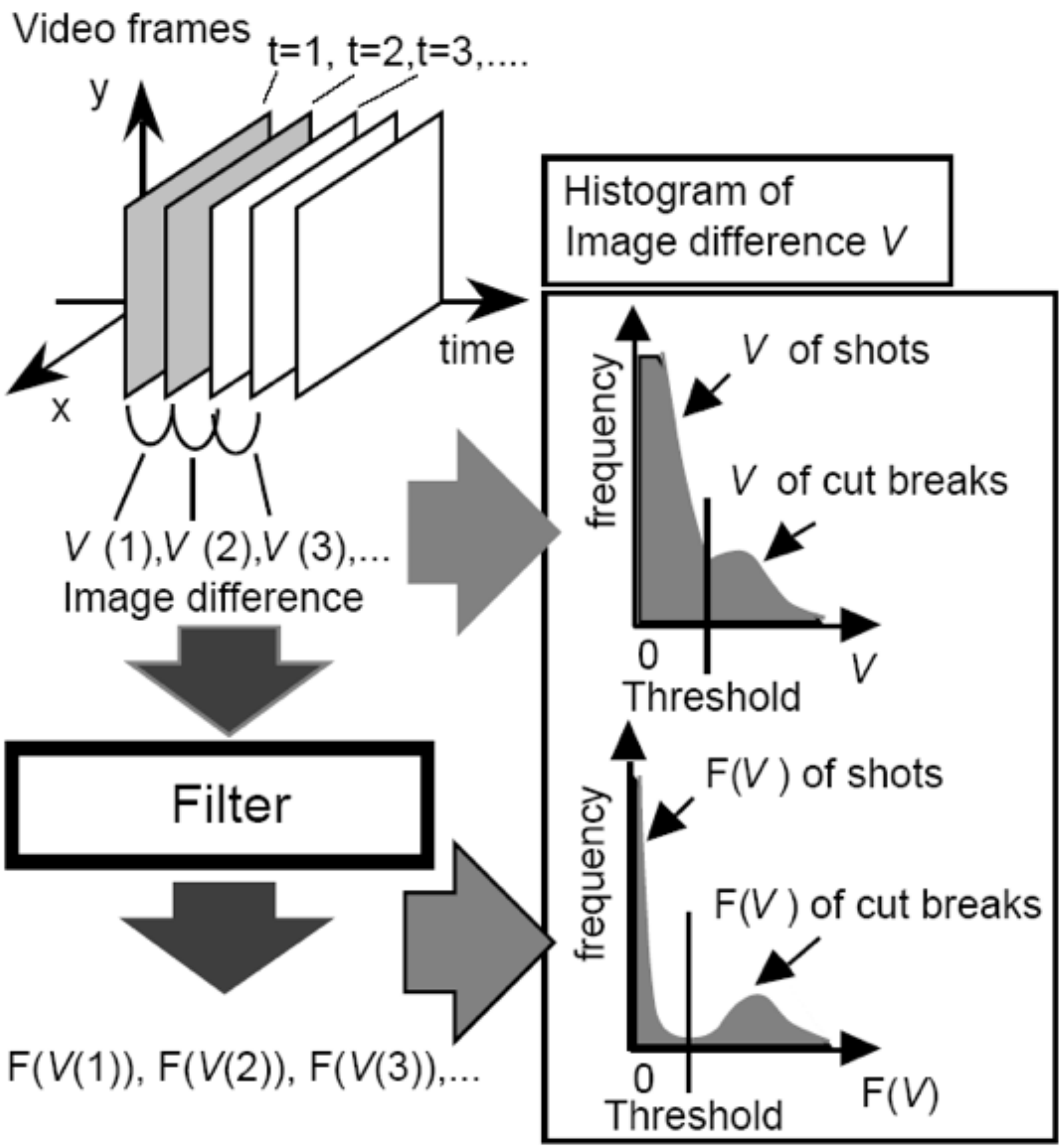
Distinguish camera movement

- Panning
  - Distribution of motion vectors has a single modal value (θm) that corresponds to the panning direction.

$$\sum_{k}^{N} |\theta_k - \theta_m| \leq \Theta_p$$

- Zooming
  - The vertical components of top and bottom motion vectors have different signs.
  - Similarly for horizontal components of left and right motion vectors.

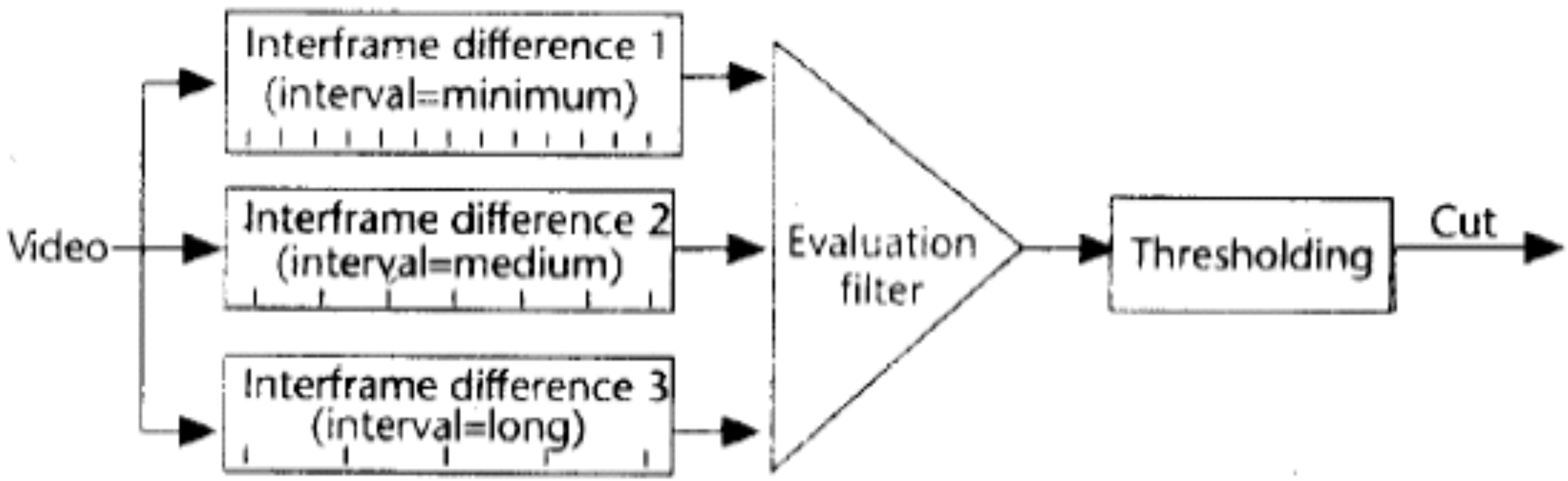
Yet Another Video Segmentation



V = image difference

Video Segmentation: Solution

- 92-98% success rate over 4.5 hours of video (news, movies, documentaries)
- 90% success when 1/3 of all cuts were via special affects

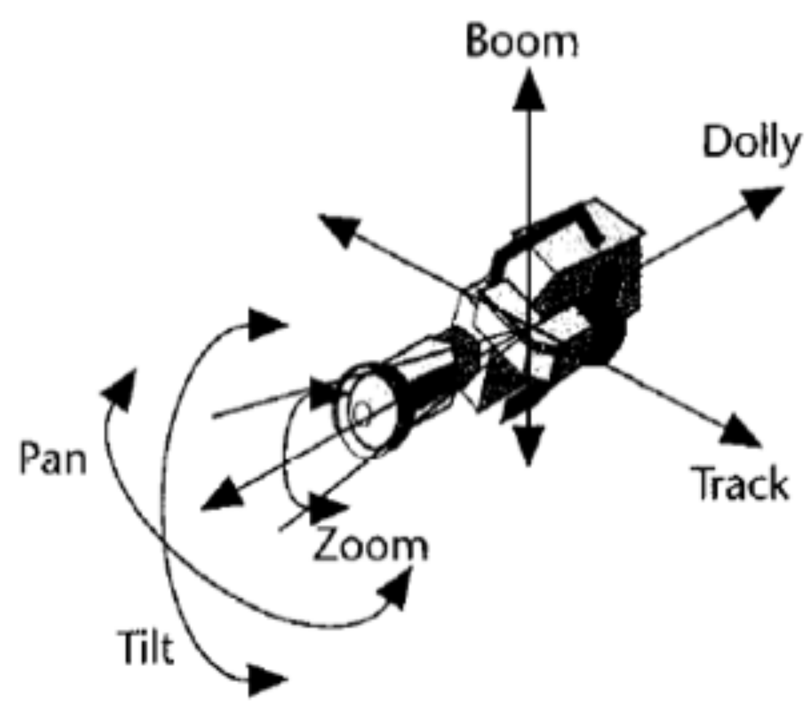


Shot Analysis

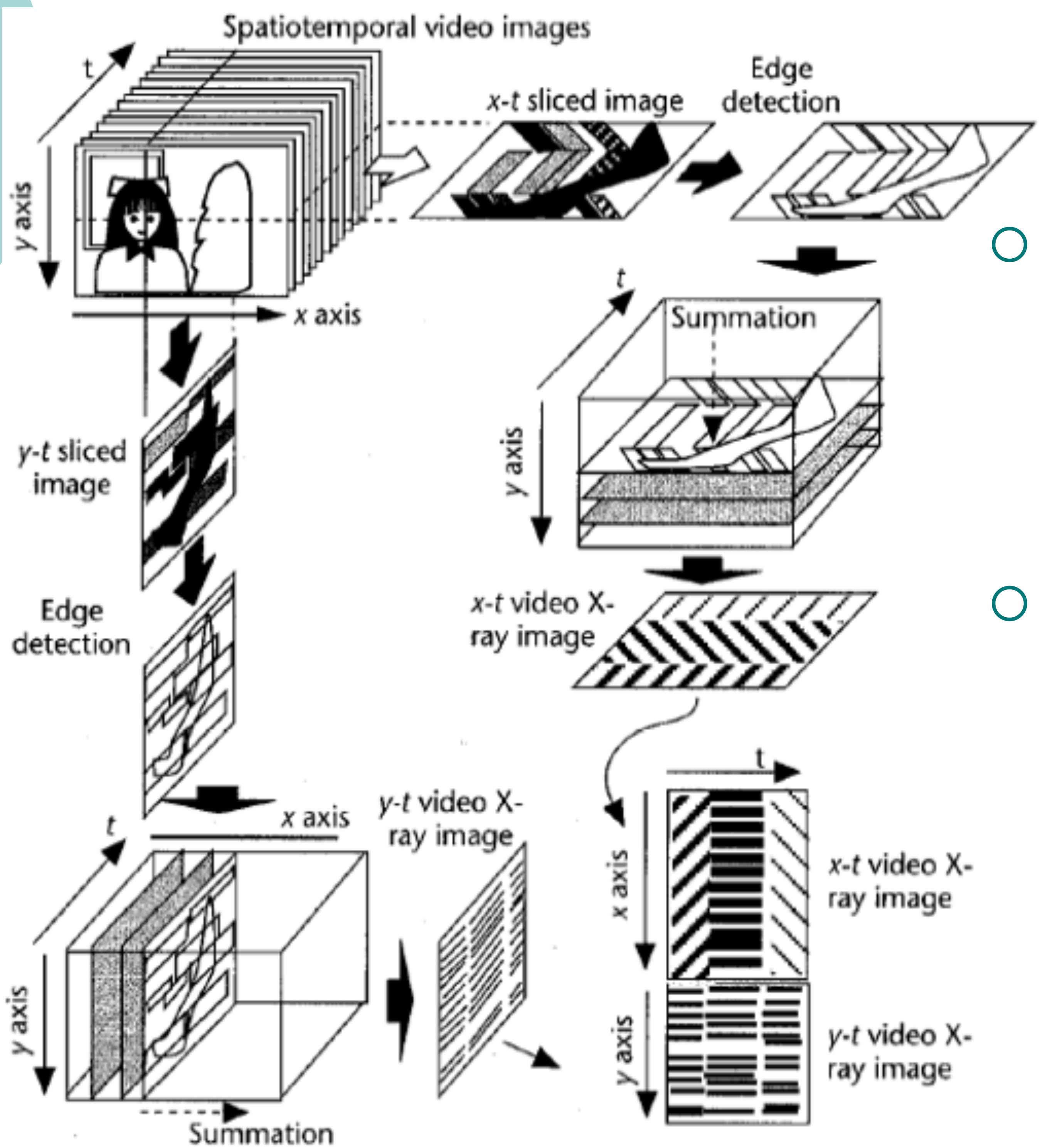
- Shot is simply sequence of frames capturing a scene’s spatial and temporal context.
- Extract this information:
  - Camera work yields spatial situation
  - Color info yields object information

Camera Work Information Extraction

- Camera movement causes global change in objects.
- Resulting point traces = motion vectors
- Motion vectors yield camera work parameters
- Computationally complex, not robust



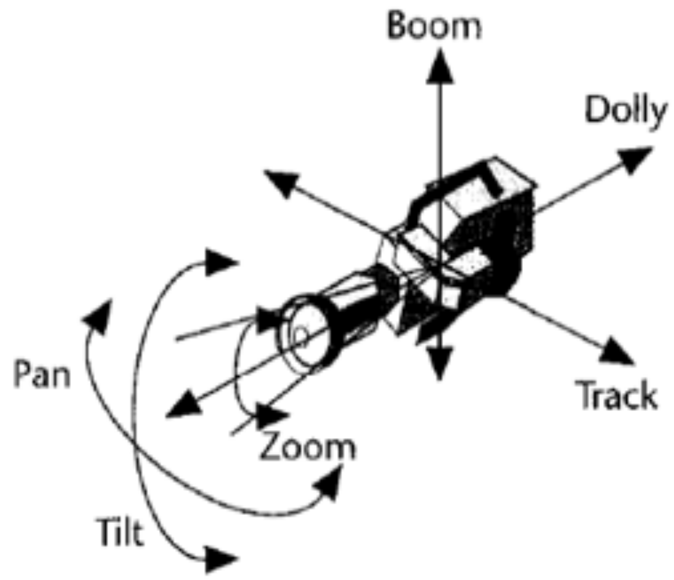
Camera Work Information Extraction

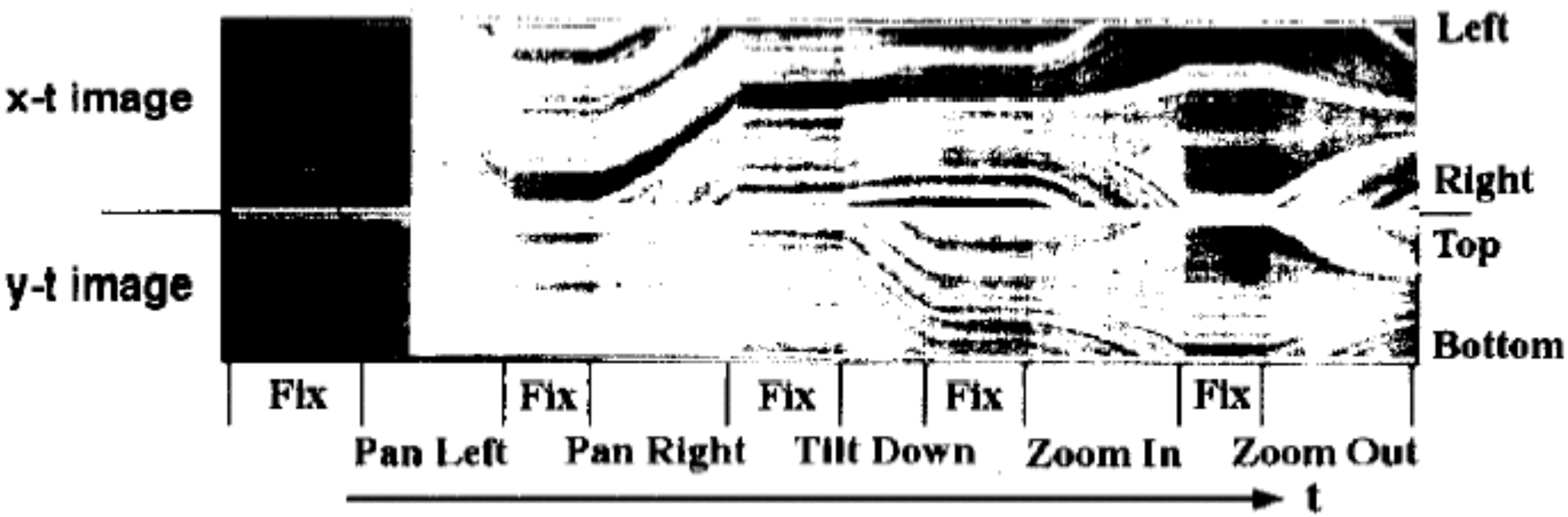


- Proposal based on video x-ray imaging.
- Easy calculations, robust

Camera Work Information Extraction

- Parallel to time = fixed camera
- Slant = camera pan
- Degree of slant = speed of pan
- Line spread = zoom
- No information present for track and dolly

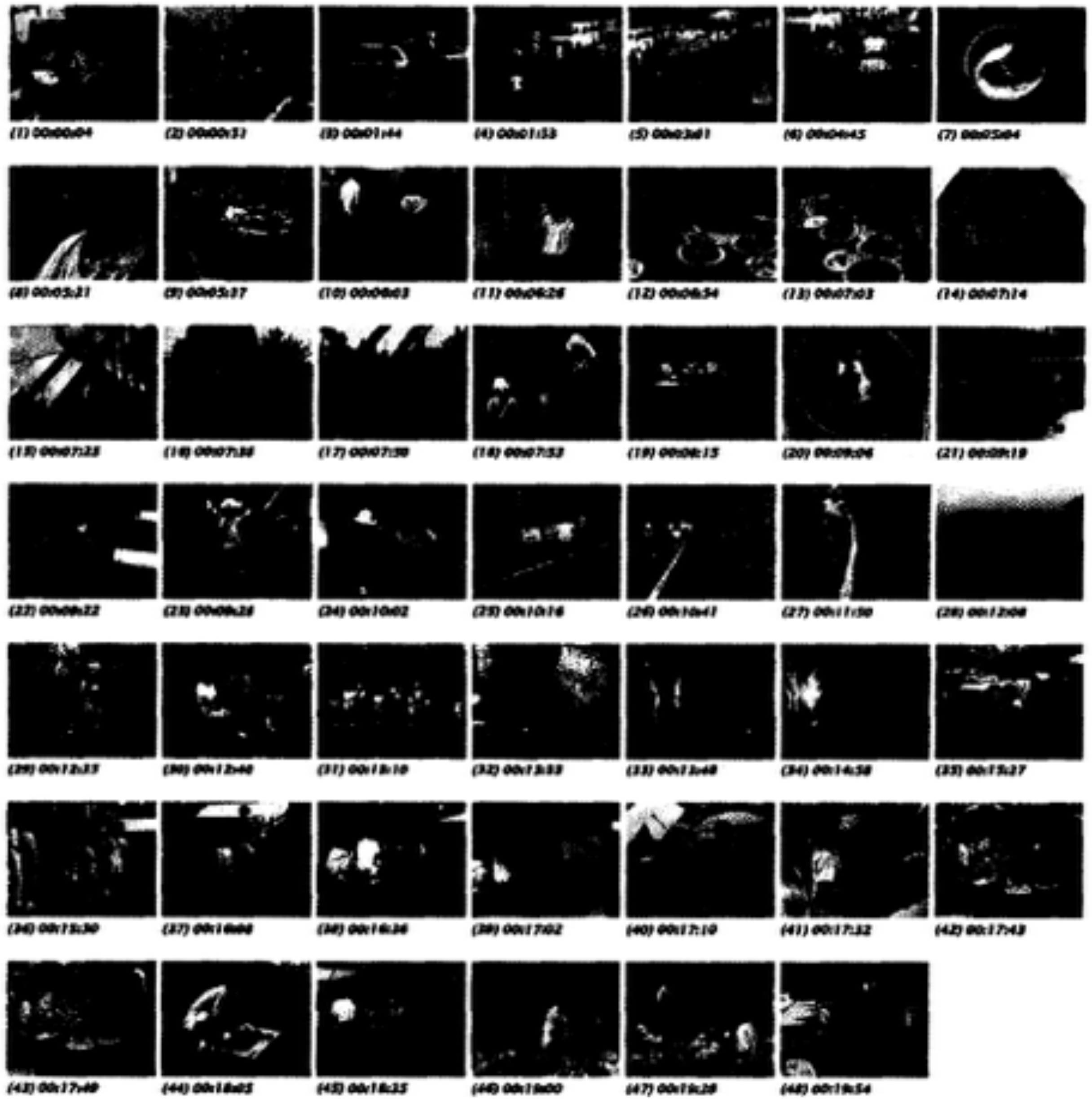




New Video Interfaces

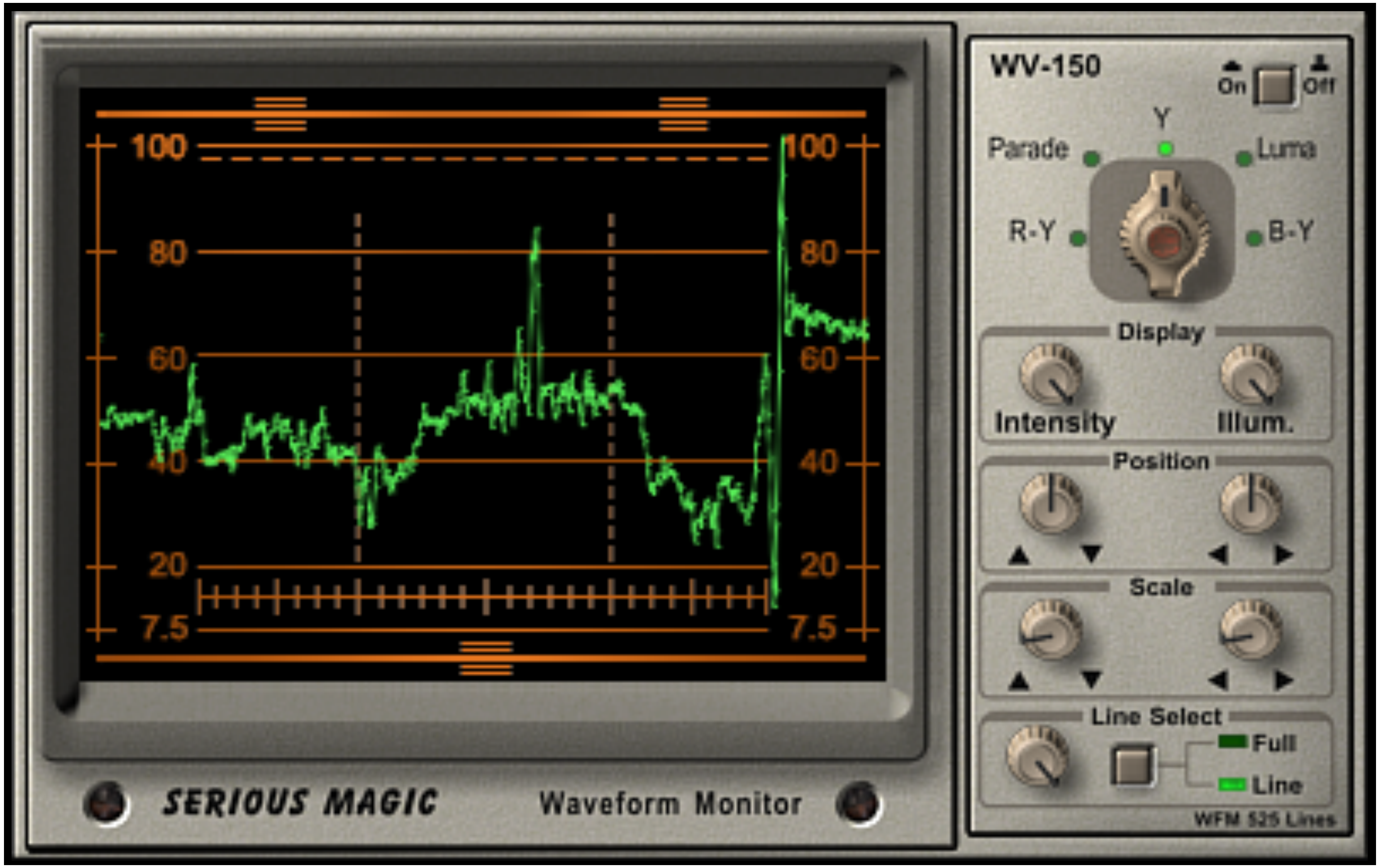
- VideoScope
- VideoSpaceIcon
- ViewSpaceMonitor
- PaperVideo

PaperVideo



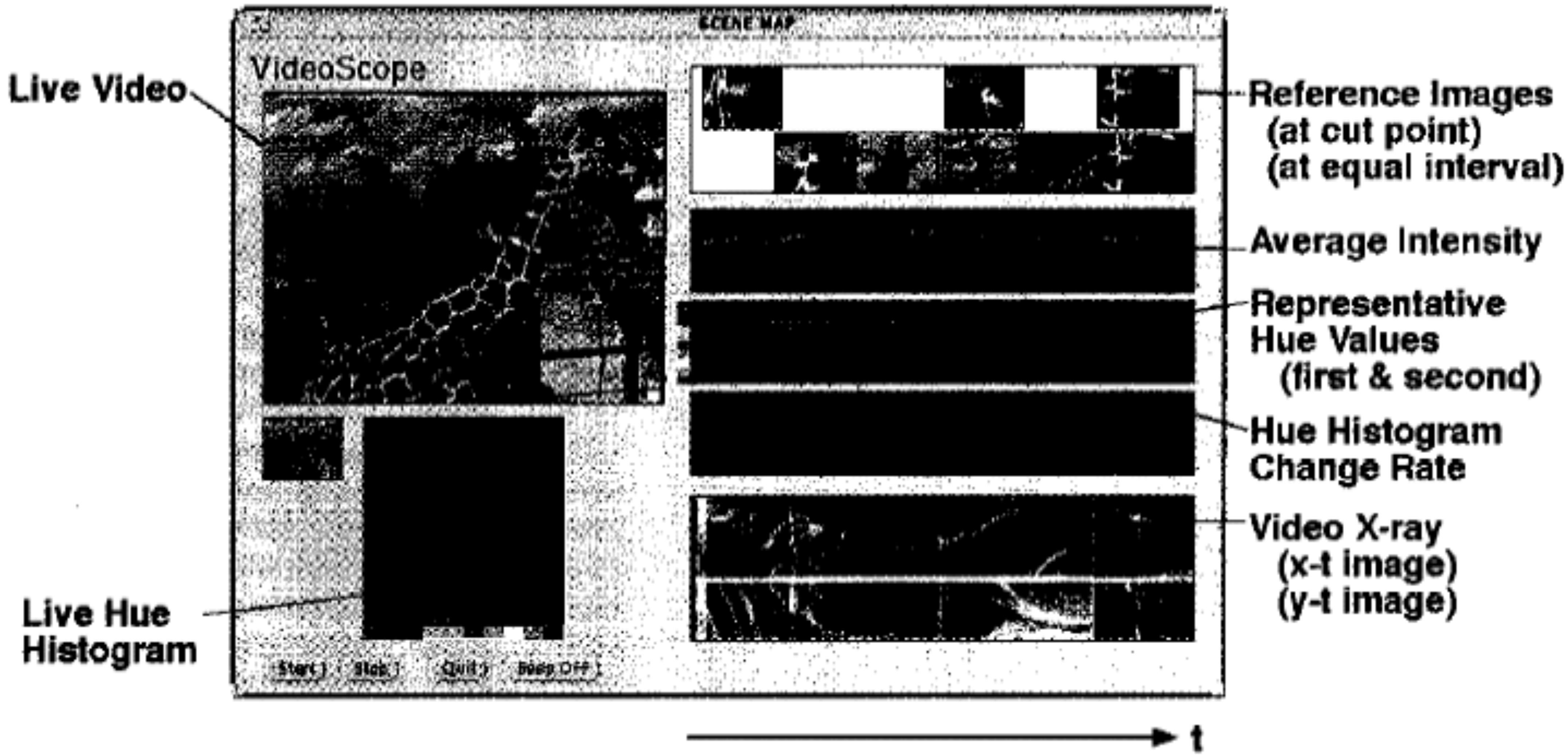
- Photo albums and video indexing.
- Shows potential simplicity of structured video apps.

VideoScope



100
80
60
40
20
7.5
WV-150
On
Off
Y
Parade
Luma
R-Y
B-Y
Display
Intensity
Illum.
Position
Scale
Line Select
Full
Line
WFM 525 Lines
SERIOUS MAGIC
Waveform Monitor

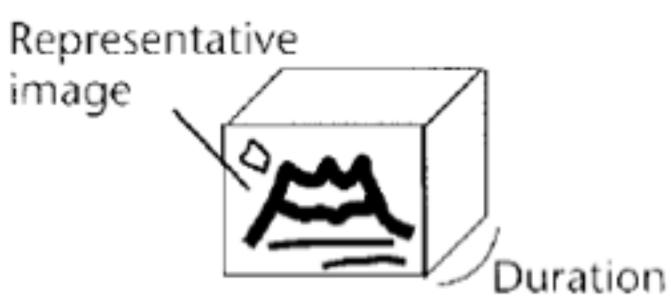
VideoScope

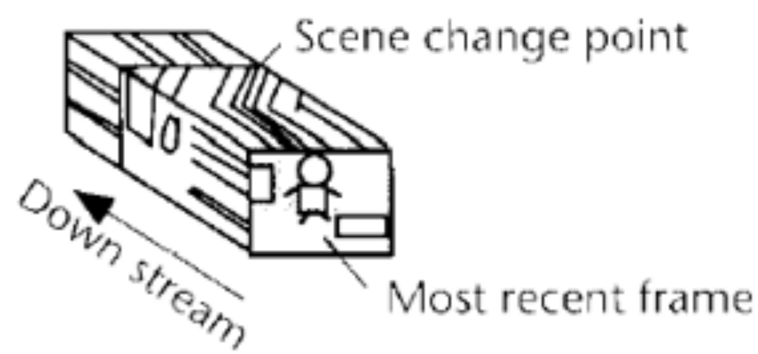


- Possible use as video engineering tool.
- Shows potential complexity of structured video apps.

Related Work

- Importance of visual interface
  - Must activate user's visual sense
  - Must stimulate user's ability to manipulate video





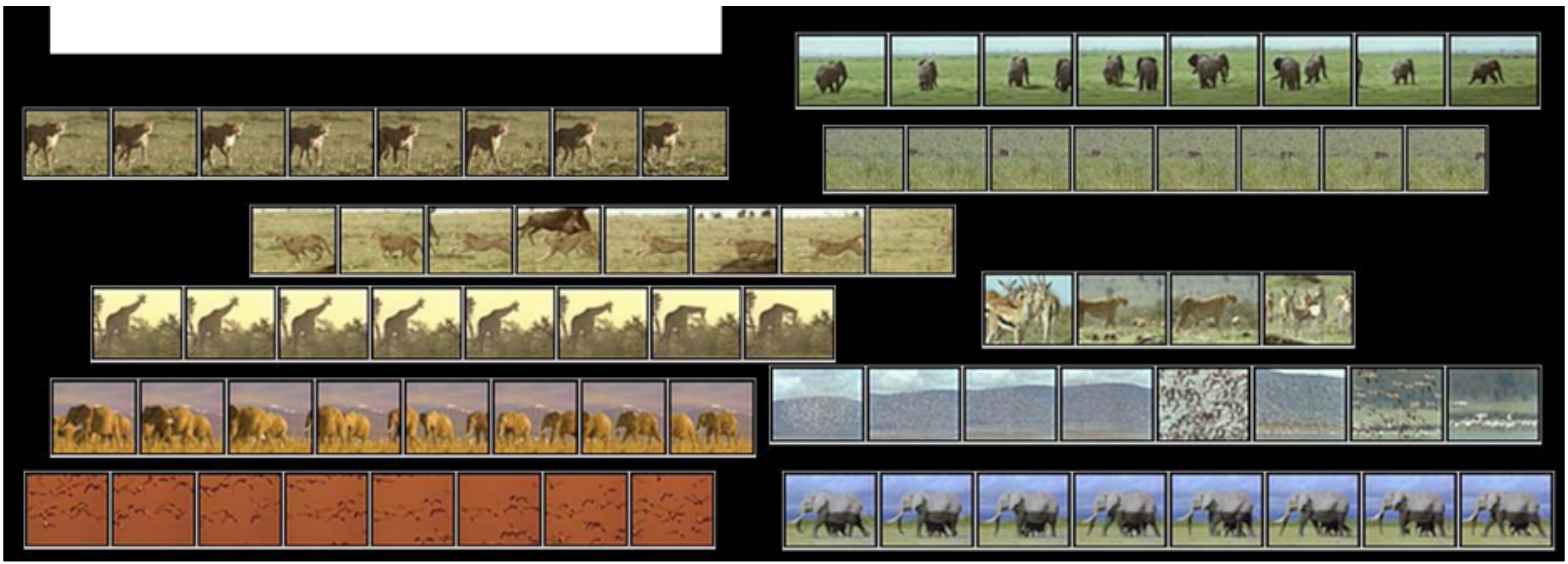
- What can be done in video production stage?

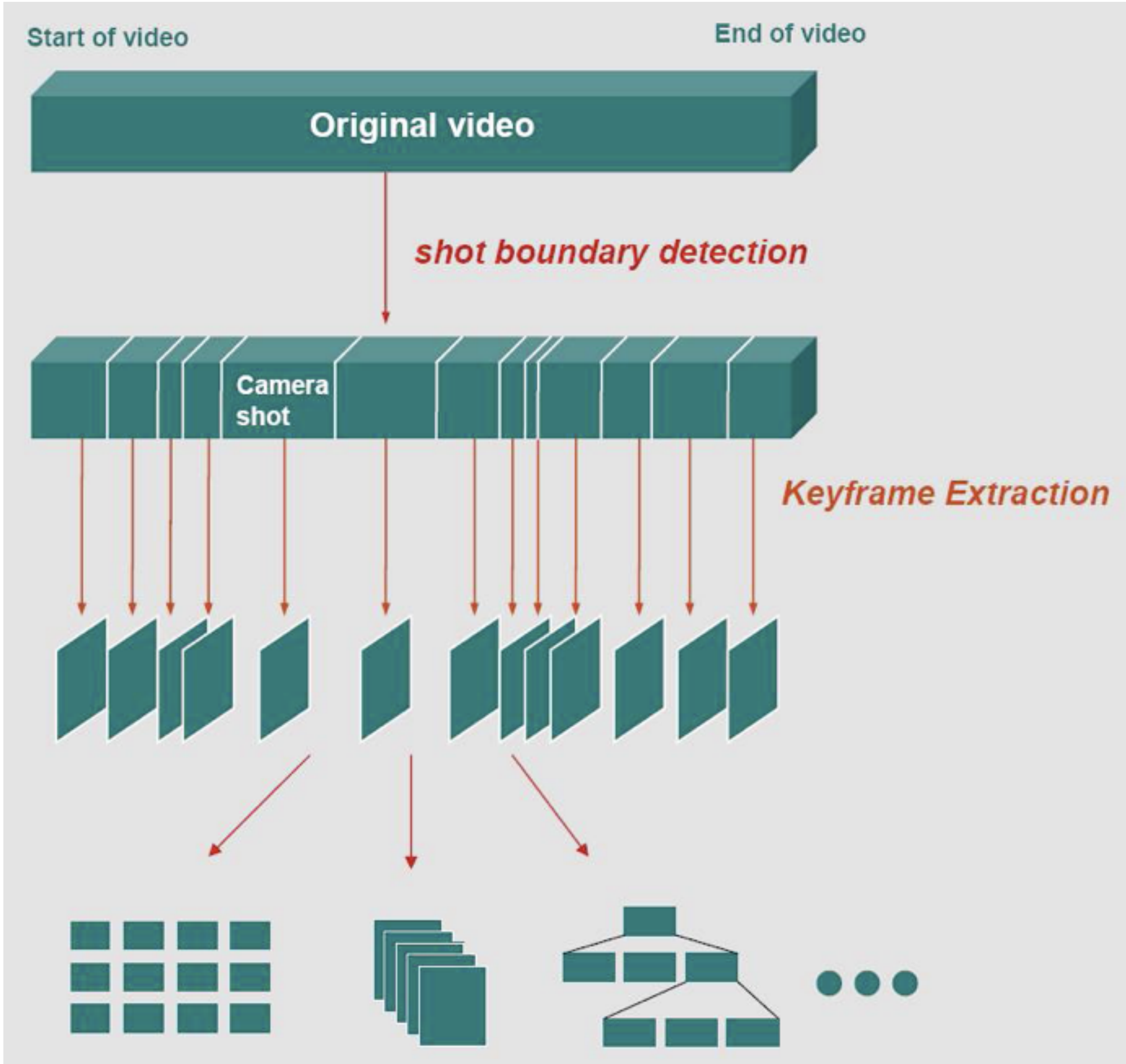
Notable Reference

Cut Detection

- K. Otsuji, Y. Tonomura, “Projection Detecting Filter for Video Cut Detection,” *Proc. ACM Multimedia 93*, ACM Press, New York, 1993.

Keyframe extraction

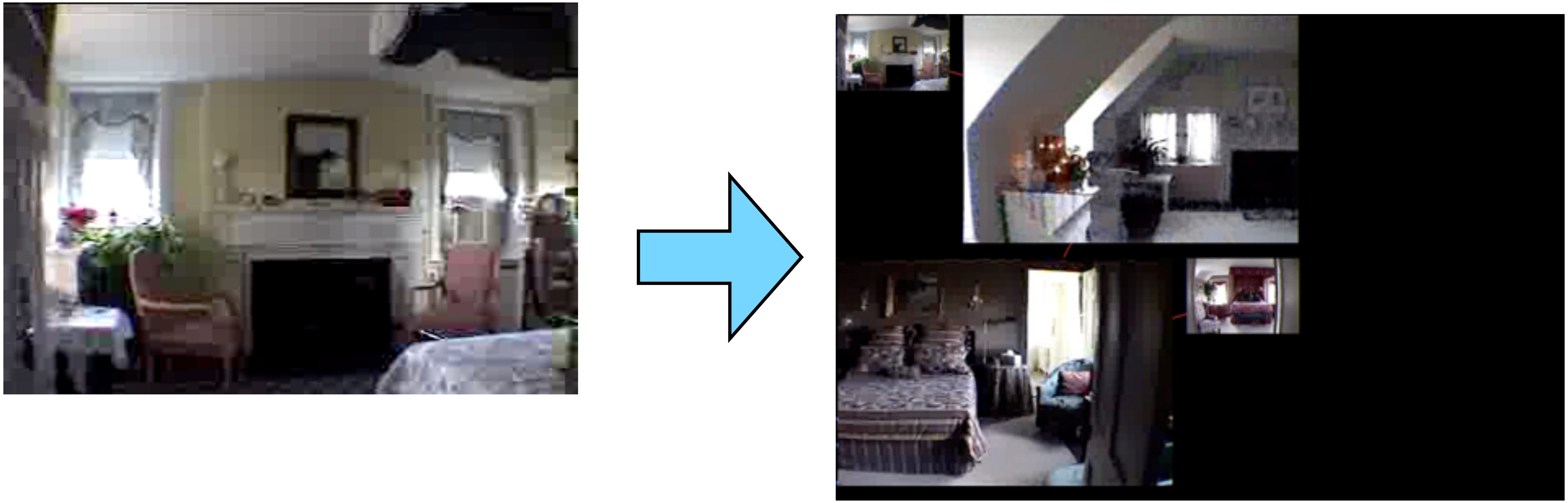



Start of video
End of video
Original video
shot boundary detection
Camera
shot
Keyframe Extraction

Reference

- Key Frame Extraction

http://www.cs.ust.hk/~rossiter/mm_projects/video_key_frame/key_frame_index.html



关键帧提取技术

- 镜头边界法
  - 选取镜头中的首帧和末帧
- 颜色特征法
  - 首帧为关键帧，其后比较与前面关键帧的颜色差异
- 运动分析法
  - 分析相机的运动
- 聚类分析法

聚类分析法

- 设一个镜头 $S = \{f_1, f_2, ..., f_m\}$
  - 找关键帧 $[F_1, F_2, ..., F_n]$
  - 定义帧间距离 $d(f_i, f_j)$

Step 0. 设定阈值，选定初始n个关键帧位置

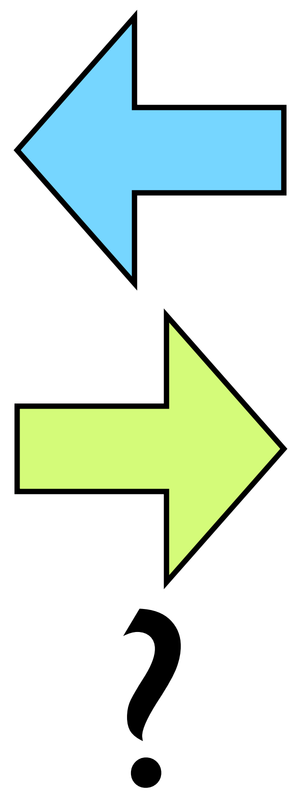
Step 1. 按照到关键帧的最小距离重新划分

Step 2. 指定每一聚类的中心帧为新的关键帧。

如果与上次划分区别不大则停止，否则重复 Step 1和Step 2.

Brain storm

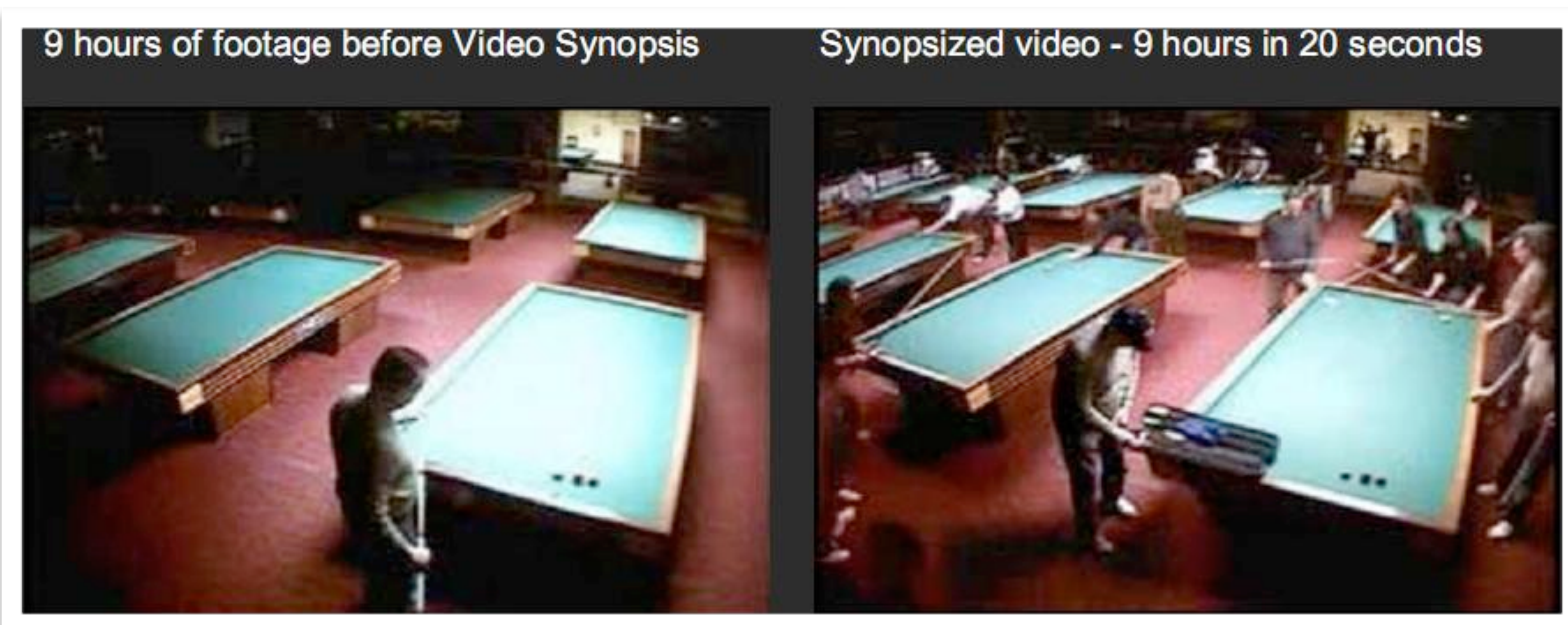

专辑:强殖装甲
视频:26 时长:10:10:51 播放:42,021
专辑:强殖装甲
视频:26 时长:10:00:23 播放:2,400
专辑:强殖装甲
视频:27 时长:10:11:26 播放:1,982
更多相关专辑>>
强殖装甲 1:31:28
强殖装甲强殖装甲强殖装甲
强殖装甲
malinkof 3个月前
播放: 17,361 | 评论: 21 | 收藏: 21
2条相似结果


?

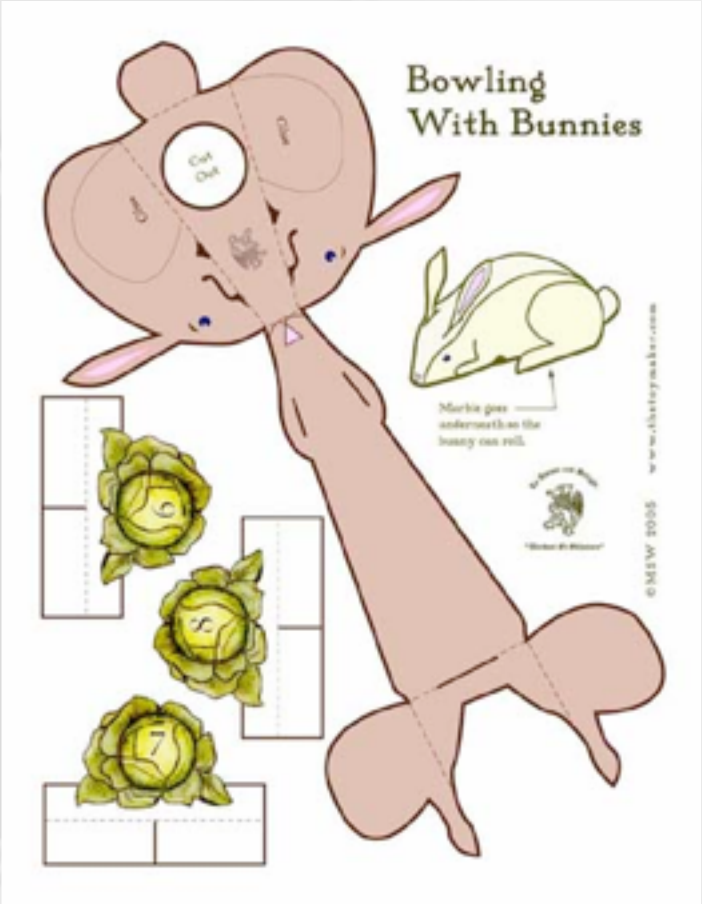

女的……!?
▼突然襲來的新的殖裝者……!其真正身份是……!?
■強殖裝甲凱普——4月號待續

BriefCam

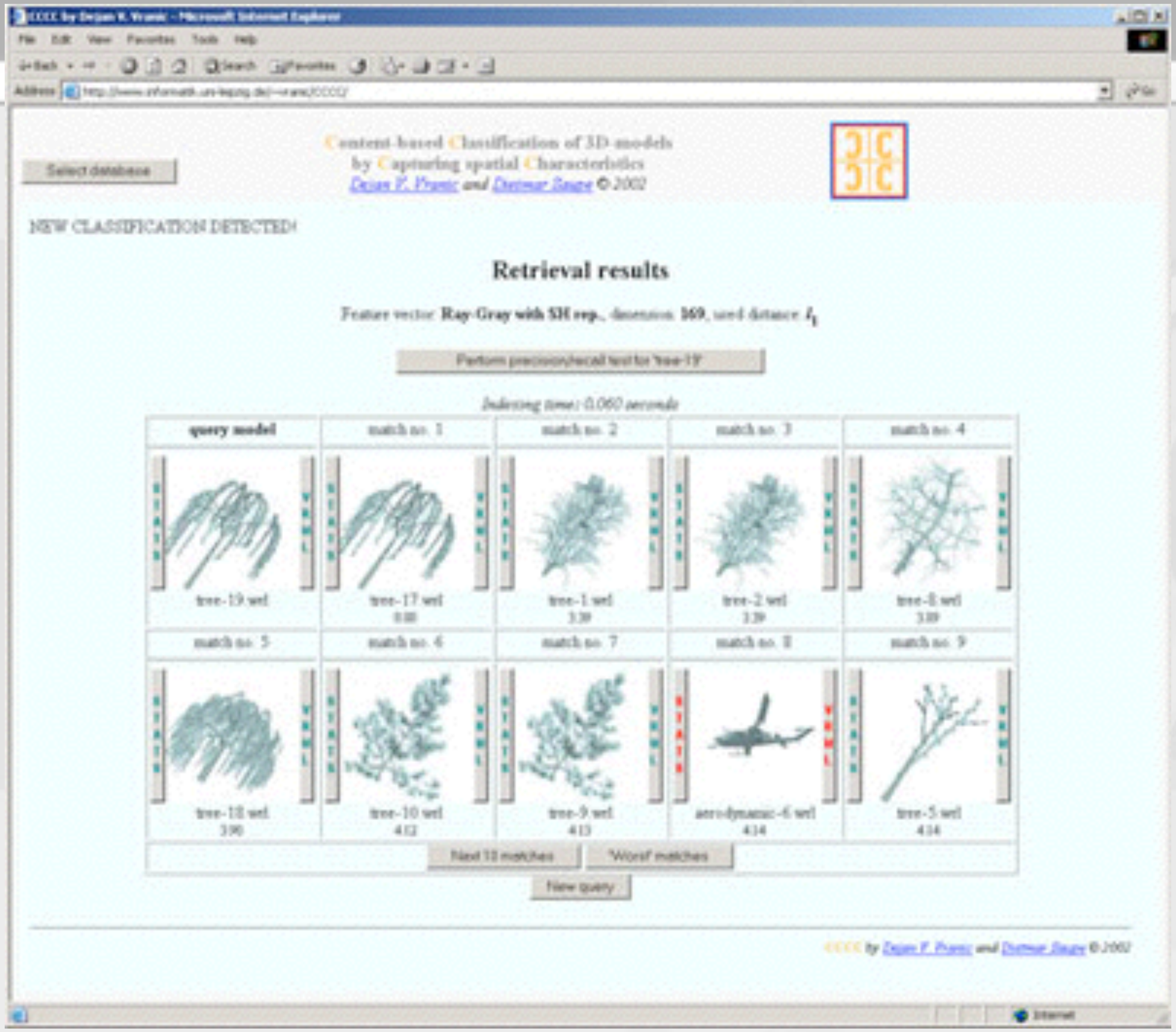
- Making a long videoshort: Dynamic video synopsis
  - http://www.vision.huji.ac.il/video-synopsis/



4. Graphics retrieval techniques

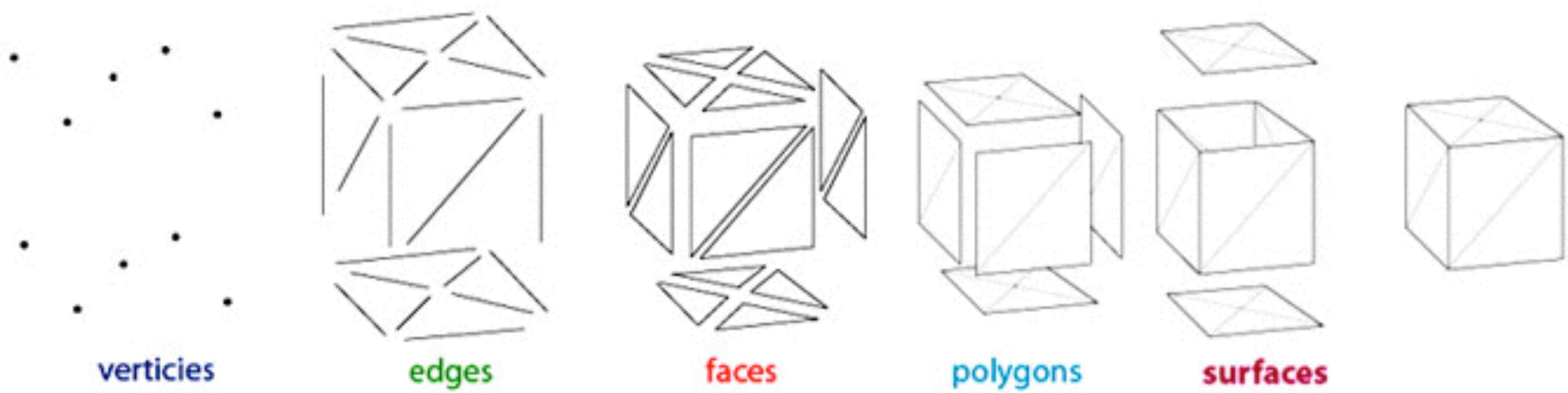


3D Model Similarity Search

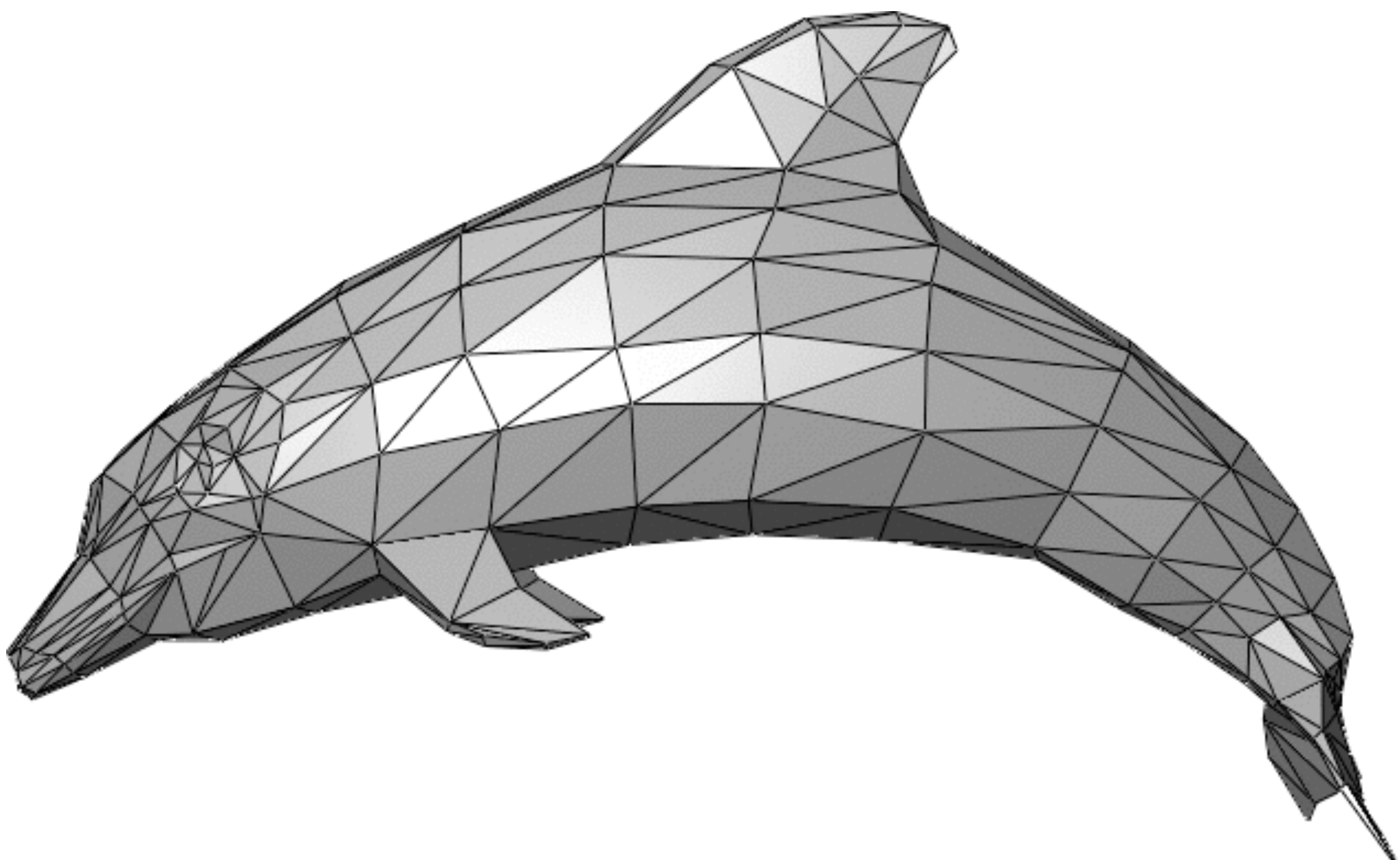


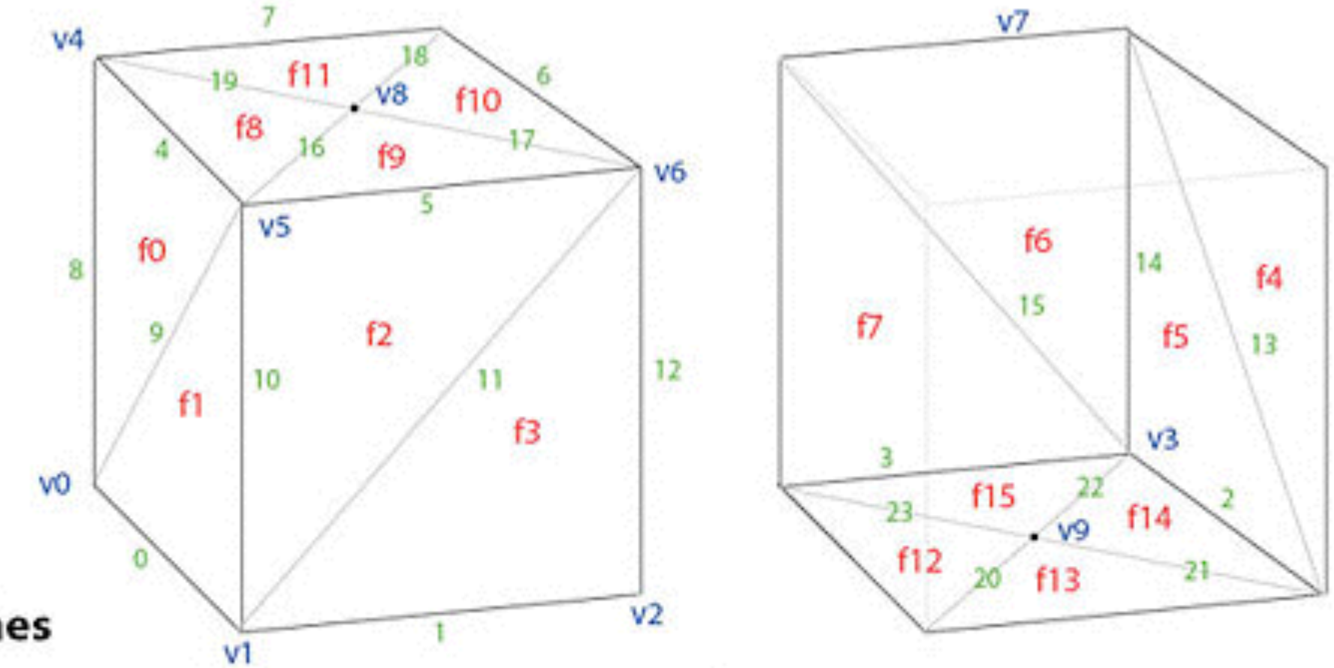
http://infovis.uni-konstanz.de/research/projects/SimSearch3D/

Elements of polygonal mesh modeling



Triangle mesh





Winged-Edge Meshes

Face List

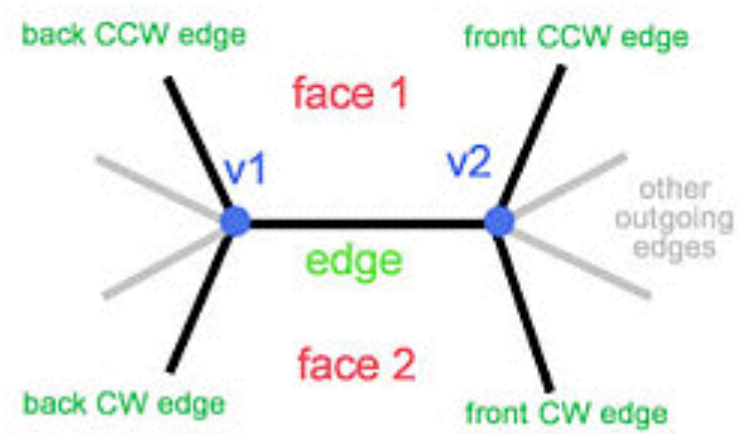
f0	4 8 9
f1	0 10 9
f2	5 10 11
f3	1 12 11
f4	6 12 13
f5	2 14 13
f6	7 14 15
f7	3 8 15
f8	4 16 19
f9	5 17 16
f10	6 18 17
f11	7 19 18
f12	0 23 20
f13	1 20 21
f14	2 21 22
f15	3 22 23

Edge List

e0	v0 v1	f1 f12	9 23 10 20
e1	v1 v2	f3 f13	11 20 12 21
e2	v2 v3	f5 f14	13 21 14 22
e3	v3 v0	f7 f15	15 22 8 23
e4	v4 v5	f0 f8	19 8 16 9
e5	v5 v6	f2 f9	16 10 17 11
e6	v6 v7	f4 f10	17 12 18 13
e7	v7 v4	f6 f11	18 14 19 15
e8	v0 v4	f7 f0	3 9 7 4
e9	v0 v5	f0 f1	8 0 4 10
e10	v1 v5	f1 f2	0 11 9 5
e11	v1 v6	f2 f3	10 1 5 12
e12	v2 v6	f3 f4	1 13 11 6
e13	v2 v7	f4 f5	12 2 6 14
e14	v3 v7	f5 f6	2 15 13 7
e15	v3 v4	f6 f7	14 3 7 15
e16	v5 v8	f8 f9	4 5 19 17
e17	v6 v8	f9 f10	5 6 16 18
e18	v7 v8	f10 f11	6 7 17 19
e19	v4 v8	f11 f8	7 4 18 16
e20	v1 v9	f12 f13	0 1 23 21
e21	v2 v9	f13 f14	1 2 20 22
e22	v3 v9	f14 f15	2 3 21 23
e23	v0 v9	f15 f12	3 0 22 20

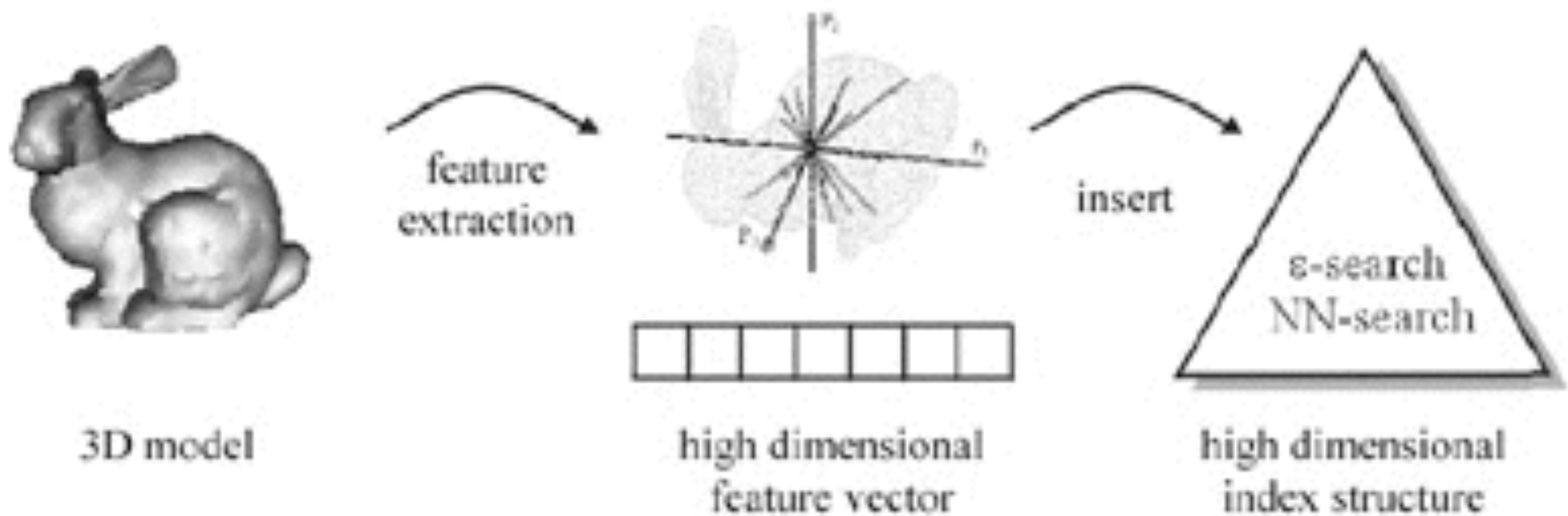
Vertex List

v0	0,0,0	8 9 0 23 3
v1	1,0,0	10 11 1 20 0
v2	1,1,0	12 13 2 21 1
v3	0,1,0	14 15 3 22 2
v4	0,0,1	8 15 7 19 4
v5	1,0,1	10 9 4 16 5
v6	1,1,1	12 11 5 17 6
v7	0,1,1	14 13 6 18 7
v8	.5,.5,0	16 17 18 19
v9	.5,.5,1	20 21 22 23



Winged Edge Structure

Main idea

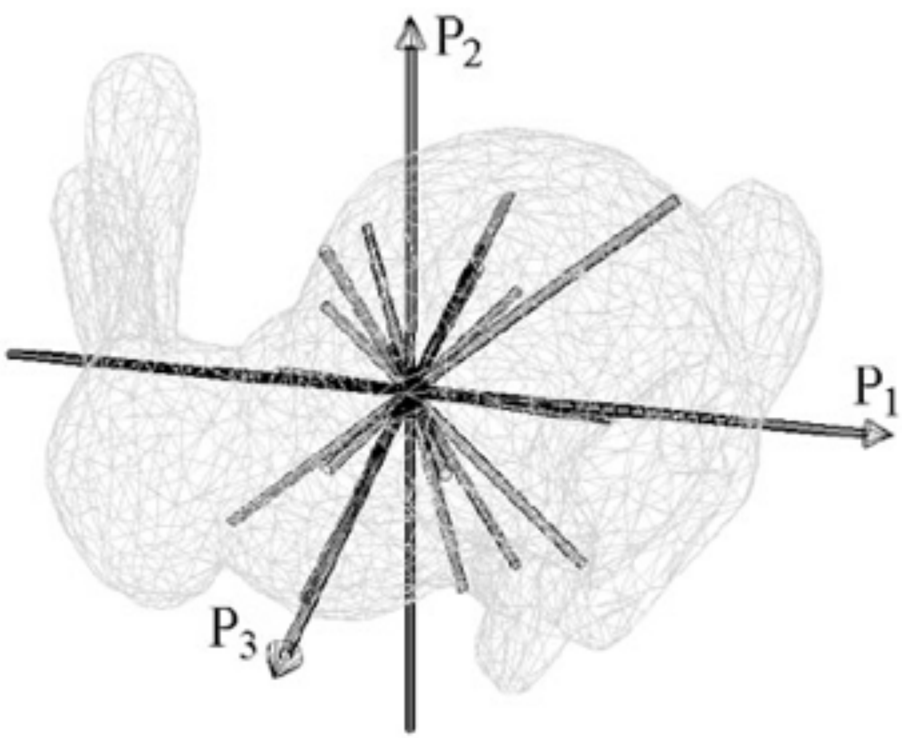

feature
extraction
insert
ε-search
NN-search
3D model
high dimensional
feature vector
high dimensional
index structure

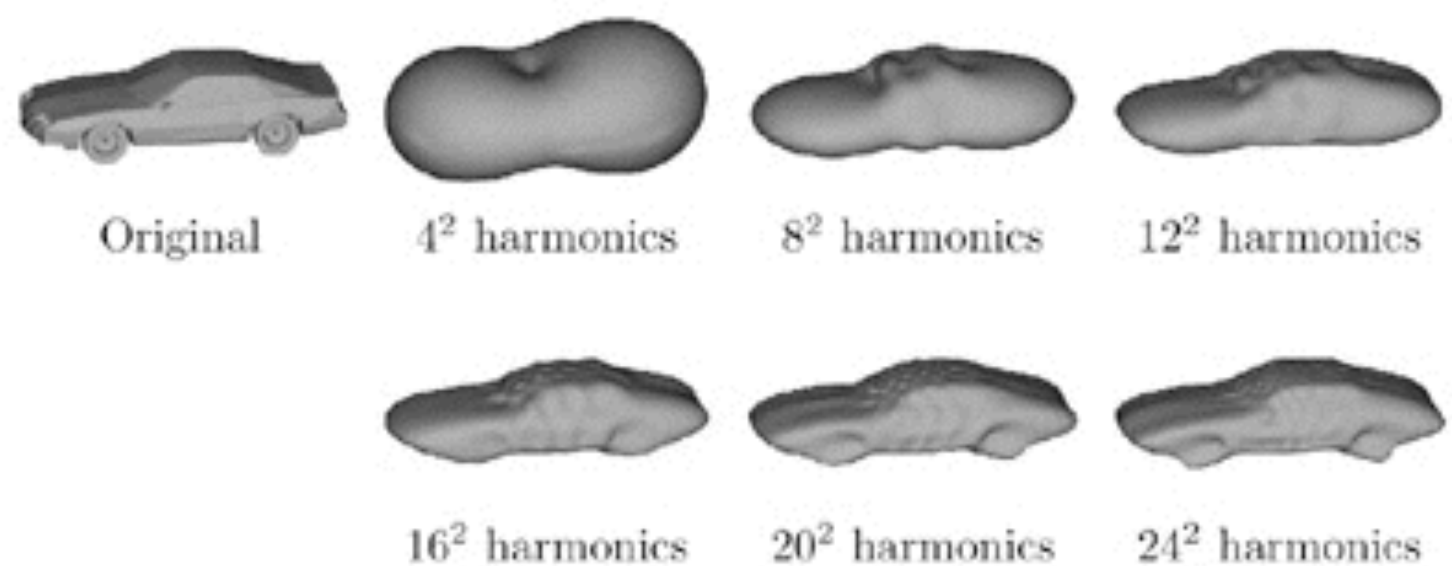
Feature vectors

- geometry based
- image based

Feature vectors

- Geometry based





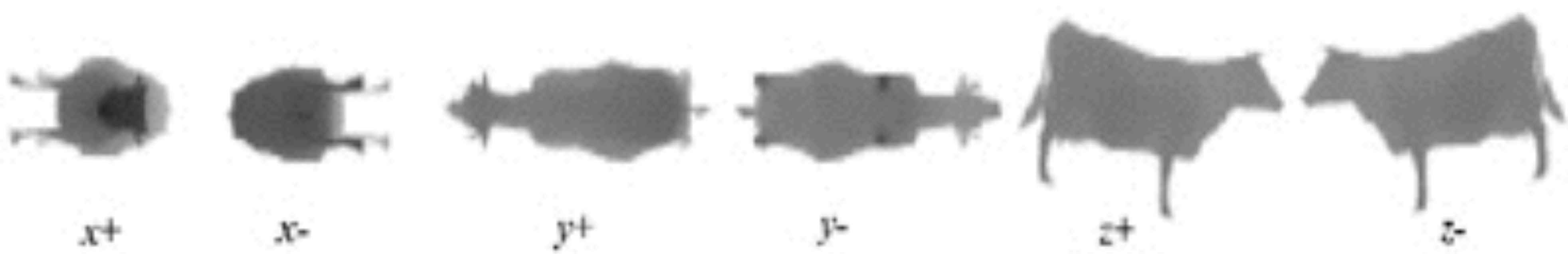
Ray-based scanning after principal axes transformation

Multi-resolution spherical harmonics representation

Feature vectors

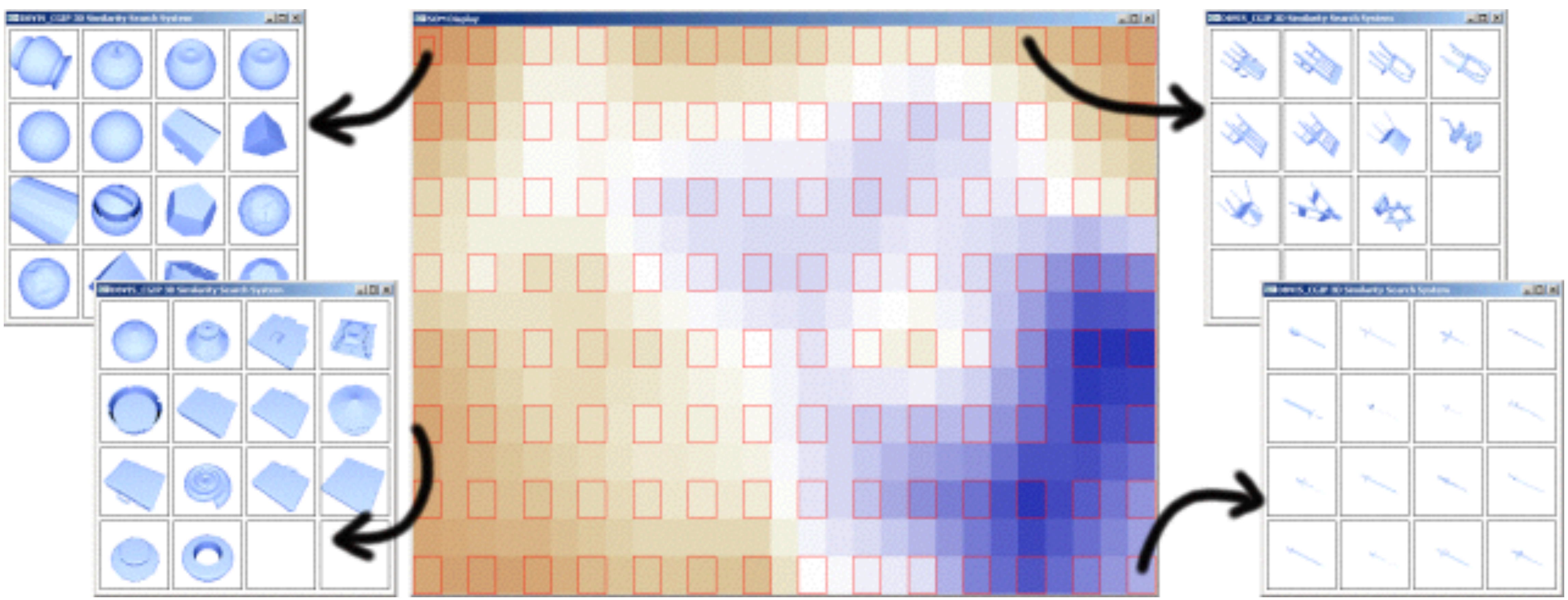
- Image based

Flat 2D silhouettes with Fourier coefficients



Depth buffer maps from 6 directions

What's good?



Self-organizing map of a 3D database

未来 ?!?

数字媒体资源管理
Digital Asset Management

Two applications

- Game design and film production
  - course note #16
- Digital library
  - course note #17

Possible DAM future

- Challenges on:
  - Techniques
  - Business
  - Culture

Possible DAM future

- Techniques
  - cloud computing / mobile computing
  - (super) large scale storage
  - new games or new UI (brain interface?)
  - new applications, new standards and new protocols

Possible DAM future

- New business models
  - What will be the next giant after MicroSoft, Google, Apple, Facebook, Oracle, Intel and IBM?
  - Information always has large value

Possible DAM future

- Culture and philosophy
  - to explore new dimensions of our world
  - immortal information (forever?)
  - complicated or simple live

This is NOT the dam end

The best way to predict future is to create it.