ERICSSON

JPEG2000

The next generation still image compression standard

JPEG2000
The next generation still image compression standard

Charilaos Christopoulos
Media Lab
Ericsson Research
Ericsson Radio Systems AB
Sweden

charilaos.christopoulos@era.ericsson.se

Athanassios Skodras
University of Patras
Electronics Laboratory
Greece

skodras@cti.gr

Why another still image compression standard?

In order to address areas that the current standards fail to produce the best quality or performance, as for example:

- **Low bit-rate compression**: Current standards, such as IS10918-1 (JPEG), offer excellent rate-distortion performance in the mid and high bit-rates. However, at low bit-rates (e.g., below 0.25 bpp for highly detailed gray-level images) the subjective distortion becomes unacceptable.
- **Lossless and lossy compression**: There is currently no standard that can provide superior lossless compression and lossy compression in a single codestream.
- **Large images:** The JPEG image compression algorithm does not allow for images greater then 64K by 64K without tiling.

Why another still image compression standard? (cont'd)

- **Single decompression architecture:** The current JPEG standard has 44 modes, many of which are application specific and not used by the majority of the JPEG decoders.
- **Transmission in noisy environments**: The current JPEG standard has provision for restart intervals, but image quality suffers dramatically when bit errors are encountered.
- **Computer generated imagery:** The current standard was optimized for natural imagery and does not perform well on computer generated imagery.
- **Compound documents:** Currently, JPEG is seldom used in the compression of compound documents because of its poor performance when applied to bi-level (text) imagery.

JPEG2000 Targets

Thus, the aim of the JPEG2000 is to develop a new still image coding standard for different types of still images (bi-level, gray-level, color, multicomponent, hypercomponent), with different characteristics (natural, scientific, medical, remote sensing, text, rendered graphics, compound, etc.), allowing different imaging models (client/server, real-time transmission, image library archival, limited buffer and bandwidth resources, etc.) preferably within a unified and integrated system.

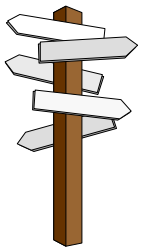
This coding system is intended for low bit-rate applications, exhibiting rate-distortion and subjective image quality performance superior to existing standards.

JPEG2000

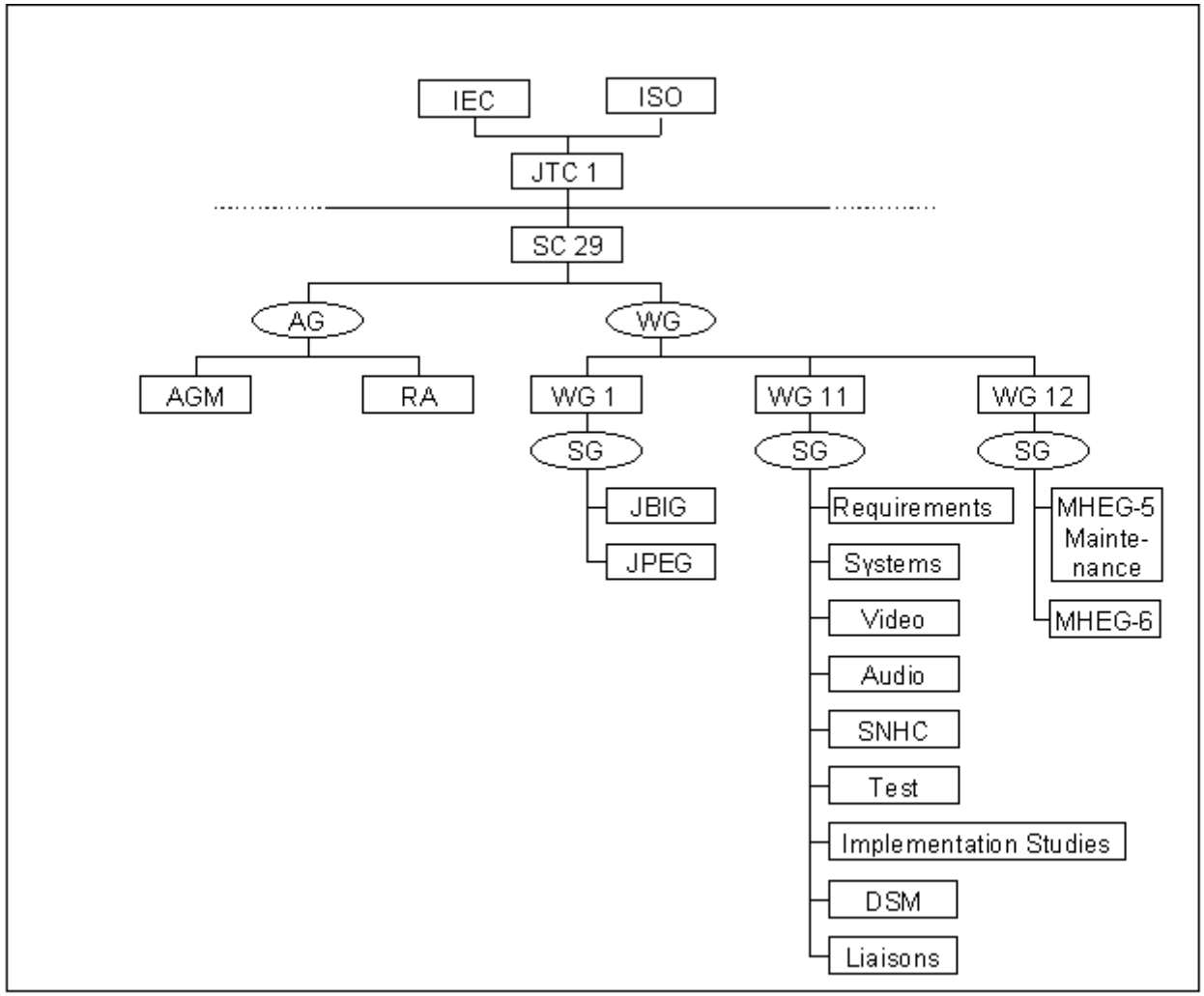
- How is the standard being developed ?
- Who are the contributors ?
- What is the schedule ?

ISO / IEC Terminology and ...

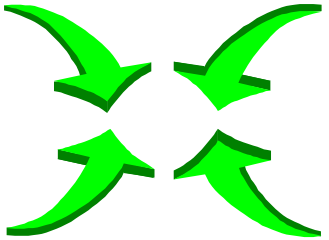
- ISO: International Standardization Organization
- IEC: International Electrotechnical Committee
- ISO/IEC JTC1: Joint Technical Committee
- SC29: Information Technologies
  - WG1: still images, JPEG and JBIG
    - Joint Photographic Experts Group and Joint Bi-level Image Group
  - WG11: video, MPEG
    - Motion Picture Experts Group
  - WG12: multimedia, MHEG
    - Multimedia Hypermedia Experts Group



... Structure


IEC
ISO
JTC 1
SC 29
AG
WG
AGM
RA
WG 1
WG 11
WG 12
SG
SG
SG
JBIG
JPEG
Requirements
Systems
Video
Audio
SNHC
Test
Implementation Studies
DSM
Liaisons
MHEG-5 Mainte-nance
MHEG-6

JPEG2000 contributors



- 15 countries / 80-100 meeting attendees
  - EUROPE
    - **Ericsson, Nokia, Philips, Canon, Motorola, Alcatel, EPFL, NTNU, Technical University of Denmark, ...**
  - USA/Canada
    - **Kodak, HP, Rockwell, Motorola, TI, Ricoh, Sharp, Adobe, University of Maryland, UBC, RPI**
  - ASIA
    - **Samsung, Panasonic, Sony, OKI, Mitsubishi, CISRA...**
- 3-4 meetings per year

JPEG2000 Development

- Timeline
  - Feb 96 (Geneva) started with original proposal
  - Nov 96 (Palo Alto) test method agreed
  - Mar 97 (Dijon) call for proposals
  - Jul 97 (Sapporo) requirements analysis started
  - Nov 97 (Sydney) algorithm competition & selection
  - VM 1 (Mar 98), VM 2 (Aug 98), split to VM 3A and 3B Nov 98. Converged to VM4 and WD in Mar 99
- Current status: VM 5.2, WD 2.0

JPEG2000 work plan

- Part I: A set of tools covering a good proportion of application requirements (20-80 rules)
- Other parts will contain different application tools in the form of profiles
- Schedule for part I:

Elevation to CD:	12/99
Elevation to FCD:	07/00
Elevation to FDIS:	11/00
Elevation to IS:	03/01

JPEG2000 Objectives

- Advanced standardized image coding system to serve applications into next millennium
- Provide features vital for high-end and emerging imaging applications
- Address areas where current standards fail to produce the best performance
- Provide capabilities to markets that currently do not use compression

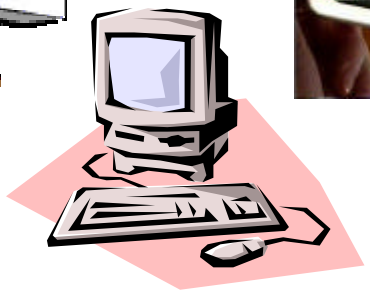
JPEG2000: Objectives in detail

- Superior low bit-rate performance
- Continuous-tone and bi-level compression
- Lossless and lossy compression
- Progressive transmission by pixel accuracy and resolution
- Fixed-rate, fixed-size, limited workspace memory
- Random codestream access and processing
- Robustness to bit-errors
- Open architecture
- Sequential build-up capability (real time coding)
- Backward compatibility with JPEG
- Content-based description
- Protective image security
- Side channel spatial information (transparency)

JPEG2000
Markets and Applications

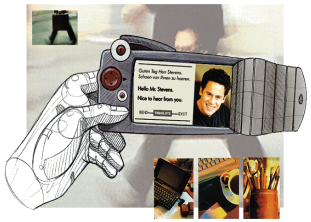
- Internet
- Mobile
- Printing
- Scanning
- Digital Photography
- Remote Sensing
- Facsimile
- Medical
- Digital Libraries
- E-Commerce



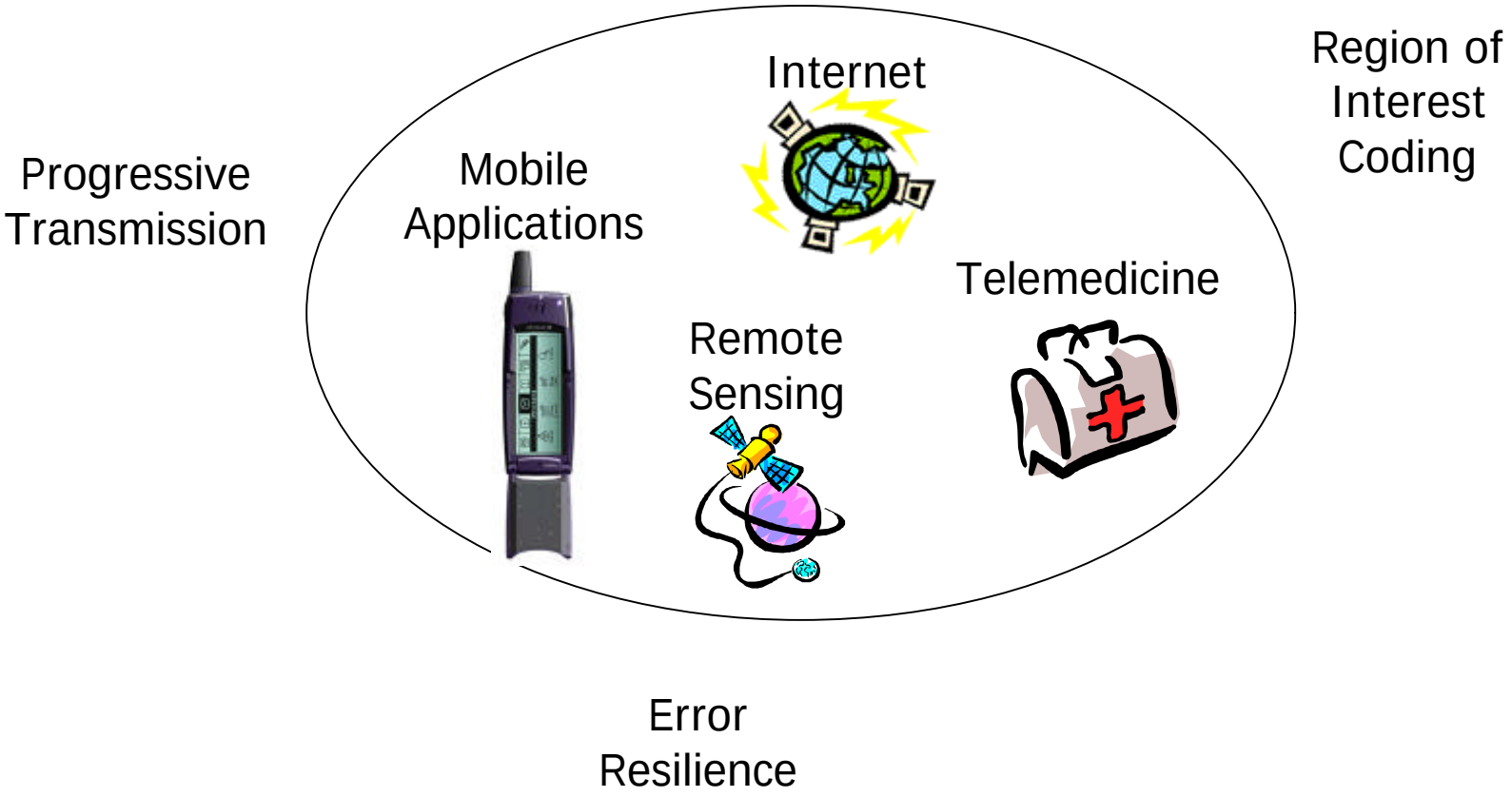





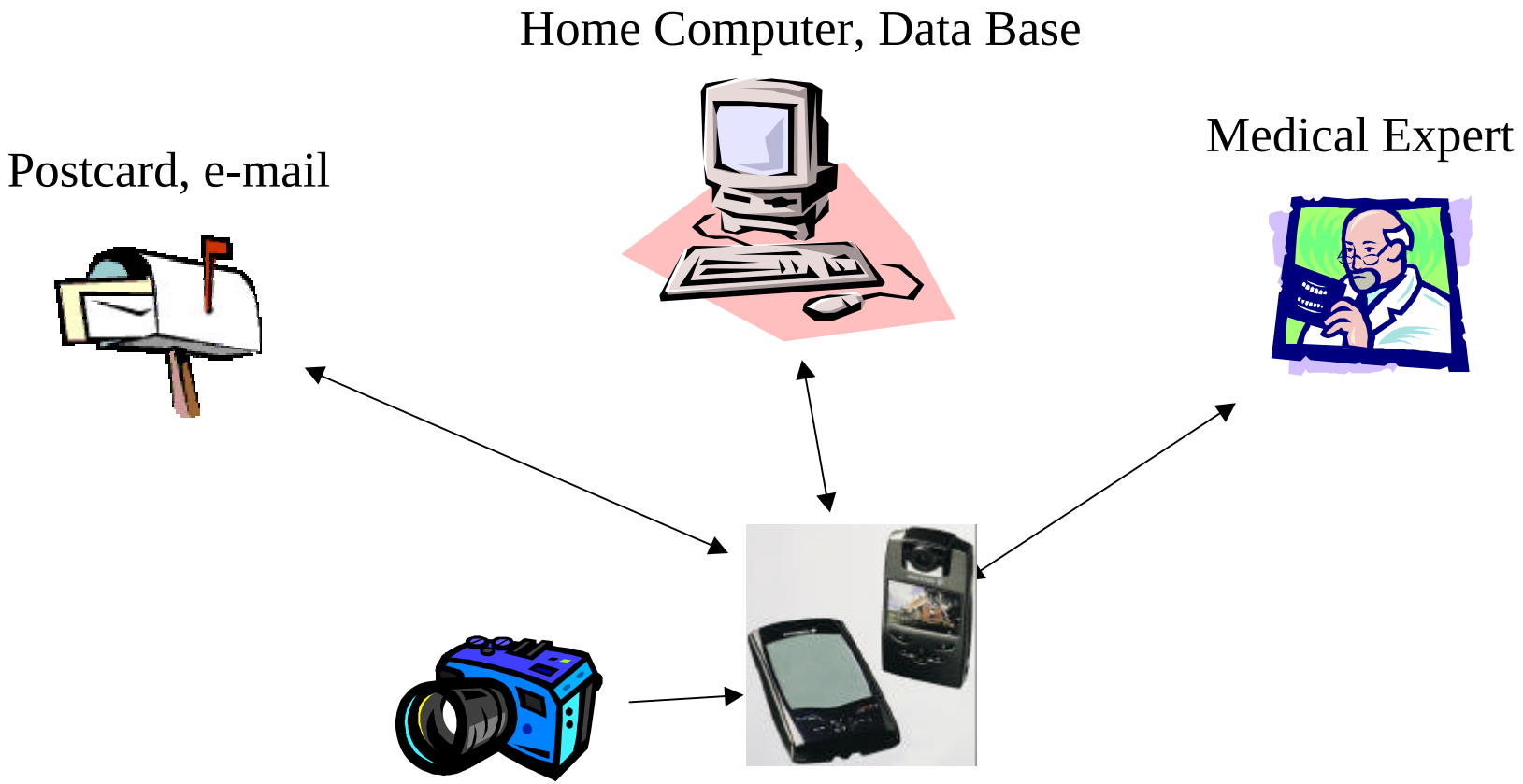


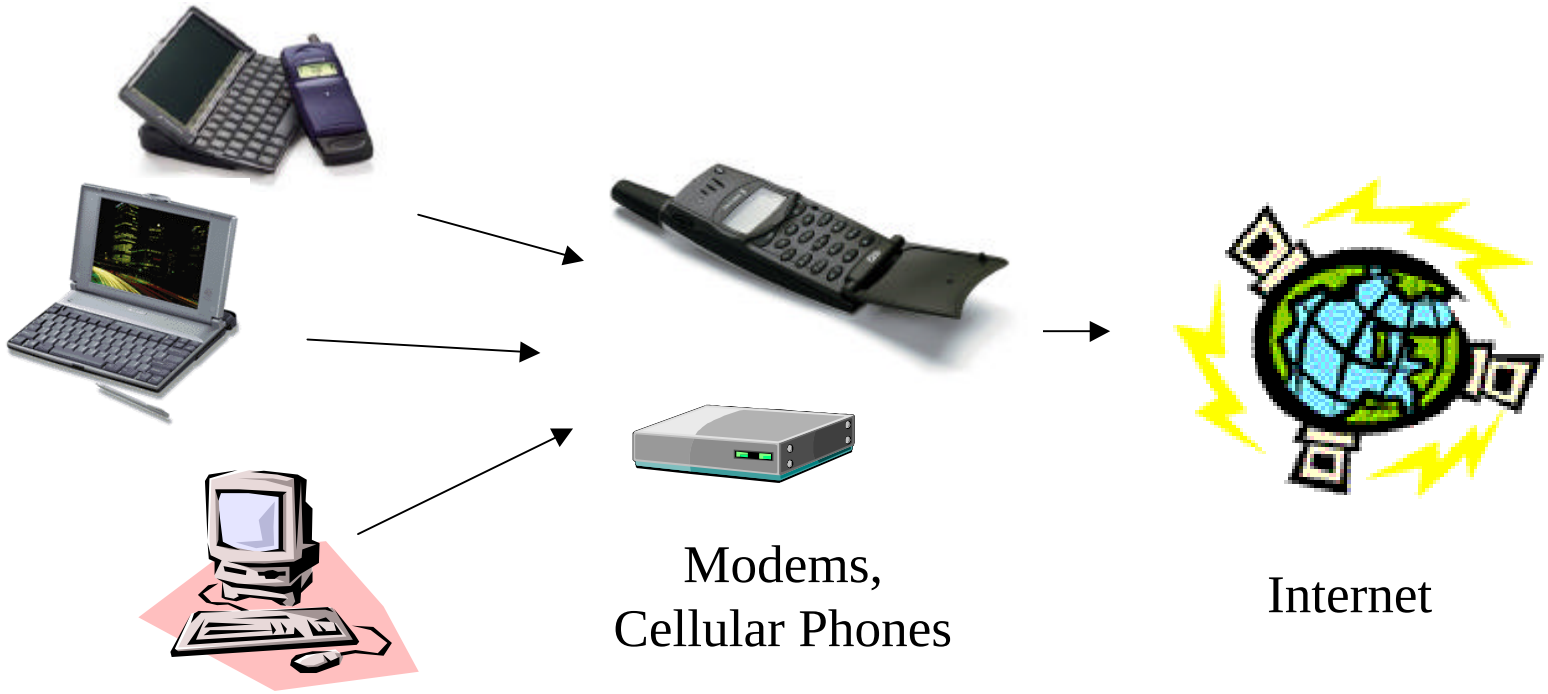
JPEG2000 applications



Mobile Applications



Internet applications



Lap Tops, PDAs,
Computers

Modems,
Cellular Phones

Internet

JPEG2000
Application Requirements

- Image type
  - Image width and height: 1 to $(2^{32} - 1)$
  - Component depth: 1 to 32 bits
  - Number of components: 1 to 255 (or more)
  - Dissimilar component depths (each component can be a different depth)
  - Dissimilar component spans (each component can have a different coverage)

Application profiles:

- **Internet**: Image sizes from 32 x 32 up to at least 4K x 4K pixels with 1, 3 (Y, RGB, YUV,...) or 4 components including alpha channel and from 1 to 8 bits/component
- **Printing**: Compound images, with typical sizes of 4800 by 6600 pixels (600ppi, 8in by 11in image) with 1, 3, and 4 components and 8 bits/component
- **Scanning**: Compound images, with typical sizes of 10K x 10K up to at least 20K x 20K pixels with 1, 3 and 4 components and up to 16 bits/component

JPEG2000
Application Requirements (cont'd)

- Application profiles (cont'd):
  - **Digital Photography**: Natural images, with sizes of at least up to 4K x 4K pixels with 1, 3 components (with spatially correlated components), with a minimum of 8 bits/component and a maximum of 16 bits/component
  - **Remote Sensing**: Infra-red, electro-optical, multi-spectral, hyper-spectral and SAR images, with virtually unlimited vertical definition and fixed horizontal definition depending on the line scan sensor upto 24000 pixels with 1 up to 500 components, and 8 up to 20 bits/component precision
  - **Mobile**: Compound images, with sizes from 32 x 32 up to at least 4K x 4K pixels with 1 or 3 components (Luminance, RGB, ...) and 1 to 8 bits/component
  - **Medical**: Natural images, with sizes from 32 x 32 to at least 10K x 10K pixels with 1 and 3 (Luminance, RGB, ...) or 4 components (plus alpha) and up to 16 bits/component
  - **Digital libraries**: Same as the Internet
  - **E-commerce**: Same as the Internet

JPEG2000
Application Requirements (cont'd)

- **Uncompressed**: The image is stored in the bitstream without compression

- **Lossless Compression**: The reconstructed image is identical, bit for bit, to the original image. Provide performance at least as good as JPEG LS. Performance includes speed, complexity, memory requirements, etc

- **Visually Lossless Compression**: The reconstructed image may differ numerically from the original image, but any such differences are not perceptible under normal viewing conditions

- **Visually Lossy Compression**: The reconstructed image contains perceptible differences from the original image under normal viewing conditions

- **Progressive Spatial Resolution**: Ability to extract lower resolution images from a codestream without redundant decoding

JPEG2000
Application Requirements (cont'd)

- **Progressive Quality Resolution:** Ability to extract lower bit-rate images from a codestream without redundant decoding or sacrifice of image quality (at that bit-rate)

- **Security:** Three purposes: 1) protect access to the image, 2) identify the image, source or owner in a secure way that cannot be removed or modified by unauthorized parties, 3) indication of the integrity

- **Error resilience:** To be "robust" (allow complete or acceptable partial decoding) in the presence of errors in the codestream such as random errors, burst errors, and packet or byte loss or insertion errors

- **Complexity Scalability**: should be scalable in complexity, so that depending on the applications, different levels of complexity can be implemented

JPEG2000
Application Requirements (cont'd)

- **Strip Processing:** The ability to compress and decompress images with a single sequential pass
- **Information embedding:** Efficient embedding of non-image information such as text, voice annotation, web links, and other types of meta-data information into compressed images
- **Repetitive Encoding/Decoding:** The ability to decode and re-encode iteratively without adding distortion (Idempotency)
- **ROI Encoding/Decoding**
- **Fast/Random Data Access**

JPEG2000 Application Profiles

		Overall System															Decoder Specific							
		Image Type	Uncompressed	Lossless Compression	Visually Lossless Compression	Visually Lossy Compression	Progressive Spatia	Progressive Quality	Security	Error Resilience	Complexity Scaleability	Strip Processing	Sensor Specific Compression	Information Embedding	Repetitive Encoding/Decoding	Object-Based Functionality	MPEG4 VTC Compatability	Backward Compatability Client Side Ease of Transcodin	Dynamic ROI	Fast/Random Data Access	Implementation Complexity	Geometric Manipulation	Static ROI	ITU-T compatibility	
	Internet	M(1,3) O(4+)			M	M	M	M	O	O				M(rec. obj)		M	O	M(Baseline) O(non-baseline)	O	M			O		
	Facsimile	M		O	O		O	O		O		M			M			O		O	M	M			
	Printing	M			M	O			O			M						O		O	M	O			
	Scanning (Consumer, pre-press)	M	O	M	M				O											O					
	Digital Photography	M		O	M	O	O	O	O	O				M(rec.obj.)	O	O	O	O			O	O			
	Remote Sensing	M(1,3) O(4+)	O	M	M	M	M	O	O	O	O	M	M	M(LUTs)		O		O	M	M		O	M		
	Mobile	M(1,3) O(4+)		O	M	M	O	M	O	M	O					O	O	O	O	M	M		O		
	Medical	M(1,3) O(4+)		M	M	M	M	M	M	O		M	M	M(rec.obj)	O	O		M(Baseline) O(non-baseline)	O	M			M		
	Digital Library	M(1,3) O(4+)		O	M	M	M	M	O	O				M(rec.obj)		M		M(Baseline) O(non-baseline)	O	M			O		
	E-Commerce	M(1,3) O(4+)		O	M	M	M	M	M	O				M(rec.obj)		M		M(Baseline) O(non-baseline)	O	M			O		

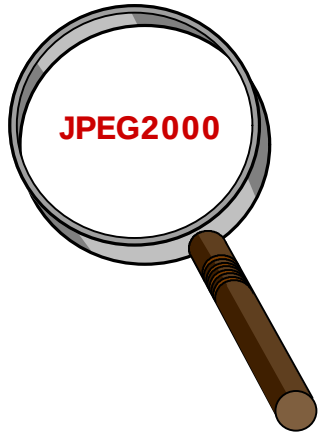
JPEG2000 Features

- High compression efficiency
- Lossless colour transformations
- Lossy and lossless coding in one algorithm
- Embedded lossy to lossless coding
- Progressive by resolution and quality
- Static and dynamic Region-of-Interest
- Error resilience
- Visual (fixed and progressive) coding
- Multiple component images
- Block and line based transforms
- Compressed image manipulation methods

JPEG2000 Requirements

- Higher compression efficiency than current JPEG
- Backward compatibility with current JPEG
- Progressive coding (by accuracy and by resolution)
- ROI coding (static and dynamic)
- Error resilience capabilities
- Object oriented functionalities (coding, information embedding, ...)
- ...

First steps of algorithm development



- November 1997 (Sydney)
  - about 100 participants
  - 24 candidate algorithms
  - All of them intensively tested
    - objective tests (quality metrics) ran on 22 test images at lossless and 6 different lossy bit rates (2, 1, 0.5, 0.25, 0.125, 0.0625 bpp)
    - subjective tests by 40 evaluators at the 3 lowest bit rates
  - selection WTCQ
  - VM established in March 98

Current status of VM 5.2

- **Wavelet based coding**
  - more advanced than DCT-based with many functionalities
- **Software status**
  - C implementation (SAIC / Univ. of Arizona / HP)
  - JavaTM implementation (EPFL, Canon, Ericsson)
  - Core experiments carried out in C up to now

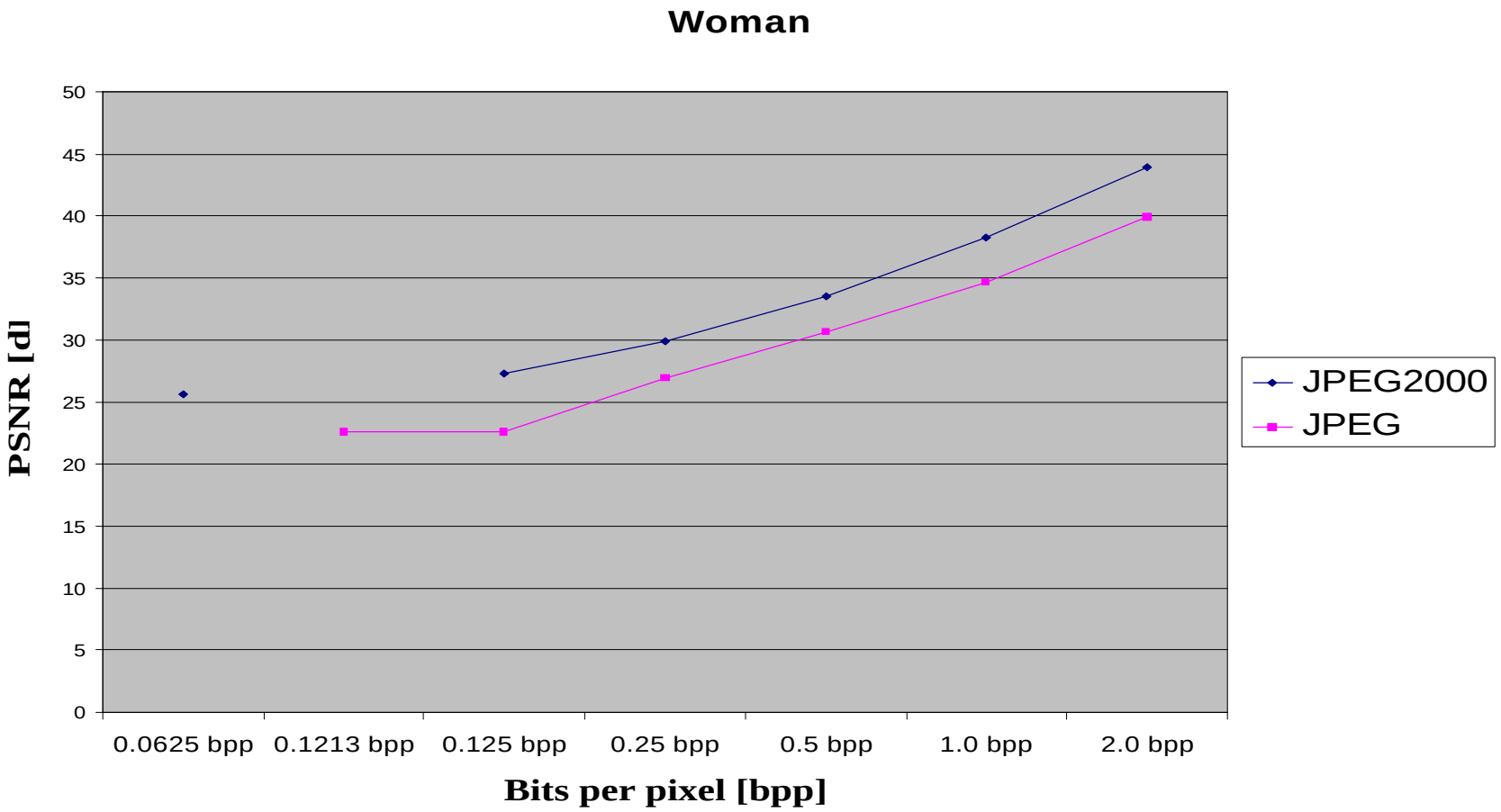
Some examples

JPEG2000

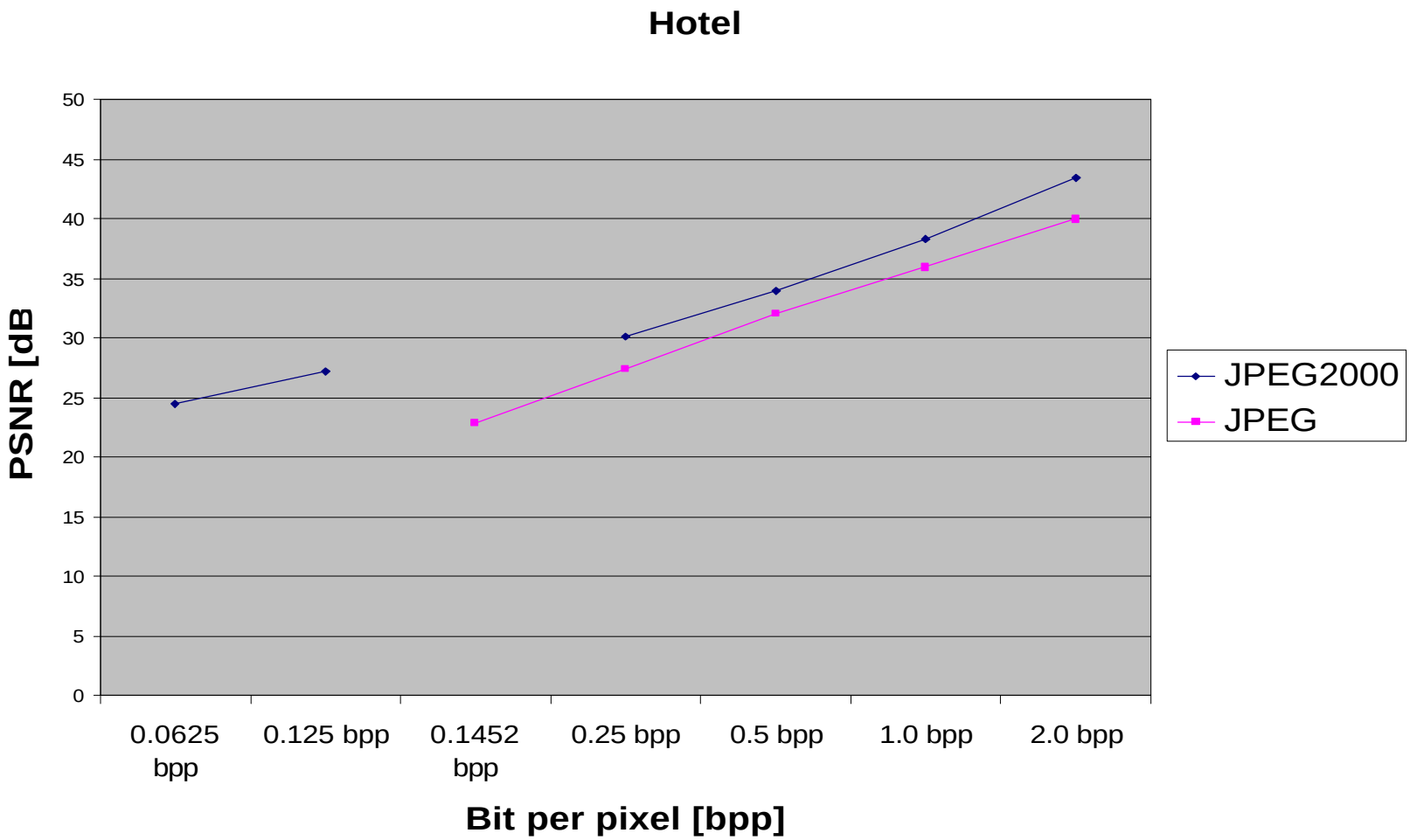
versus

JPEG baseline

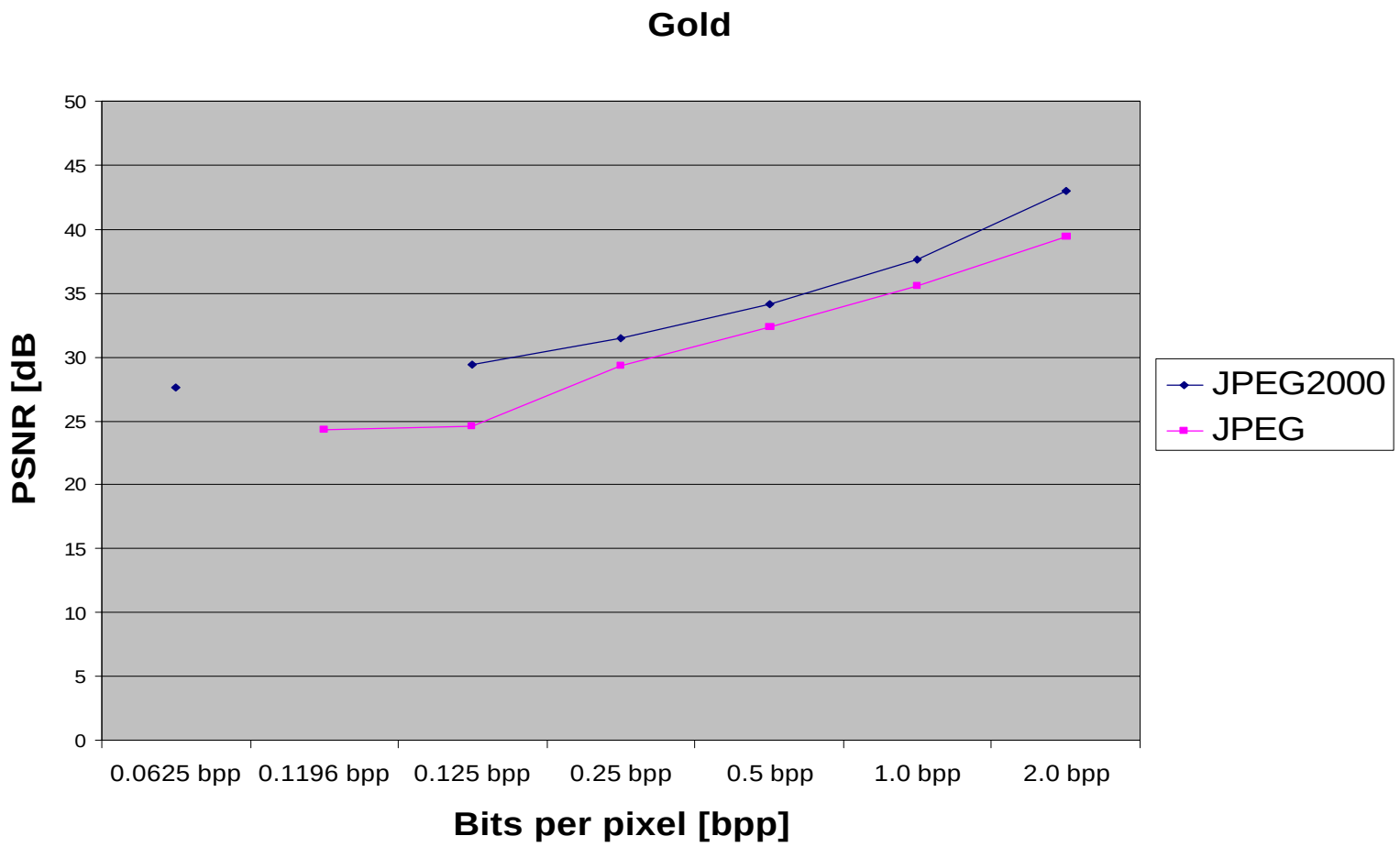
JPEG2000 vs JPEG baseline



JPEG2000 vs JPEG baseline (cont'd)



JPEG2000 vs JPEG baseline (cont'd)



JPEG at 0.125 bpp

JPEG2000 at 0.125 bpp

JPEG at 0.25 bpp

JPEG2000 at 0.25 bpp

JPEG at 0.125 bpp

JPEG2000 at 0.125 bpp

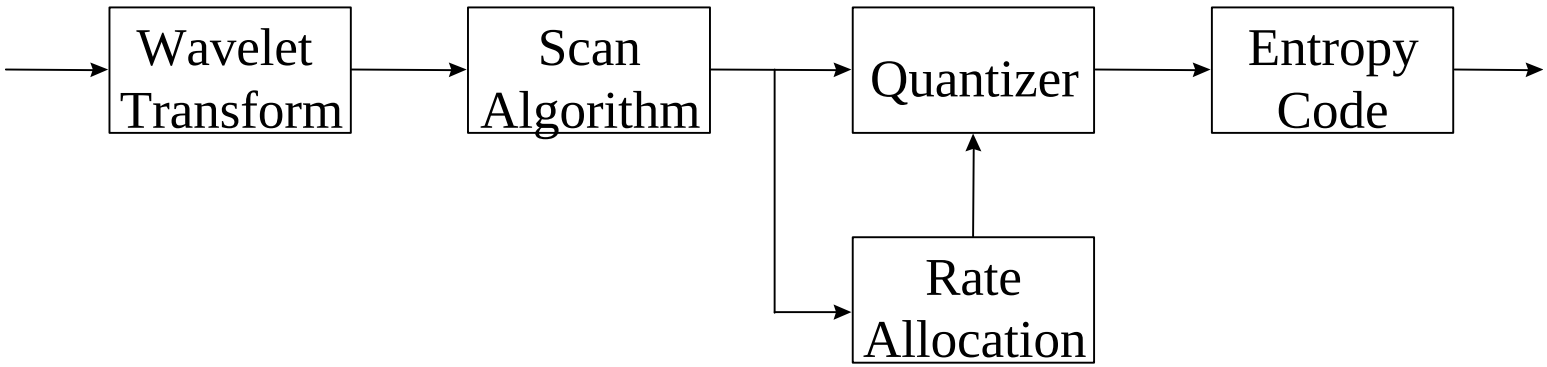
JPEG at 0.25 bpp

JPEG2000 at 0.25 bpp

JPEG2000

Current technical status

JPEG2000: Basic encoding scheme



Embedded Block Coding with Optimized Truncation (EBCOT)

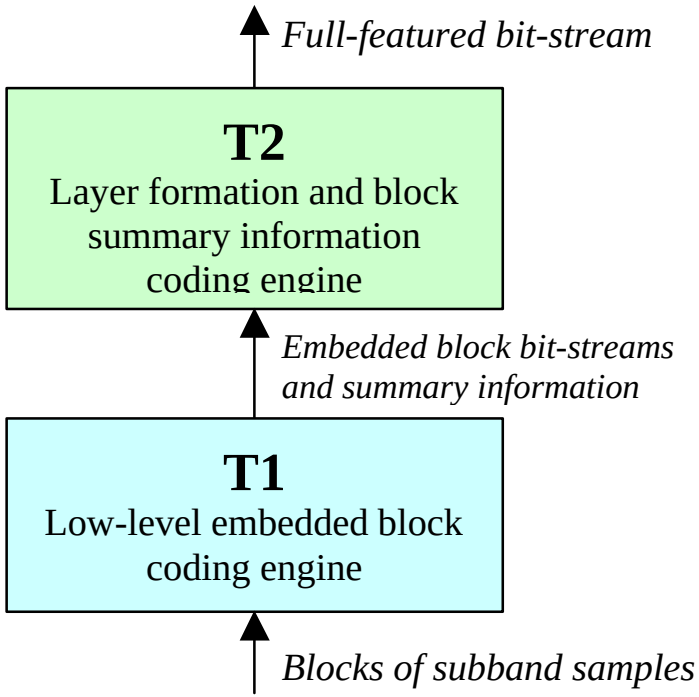
- Each subband is partitioned into a set of blocks
- All blocks within a subband have the same size (possible exception for the blocks at the image boundaries)
- Blocks are encoded independently
- Post-processing operation determines the extent to which each block's bitstream should be truncated
- Final bitstream is composed of a collection of "layers"

Why block coding?

- exploit local variations in the statistics of the image from block to block
- provide support for applications requiring random access to the image
- reduce memory consumption in hardware implementations of the compression or decompression engine
- Allow for parallel implementation

EBCOT coding operations

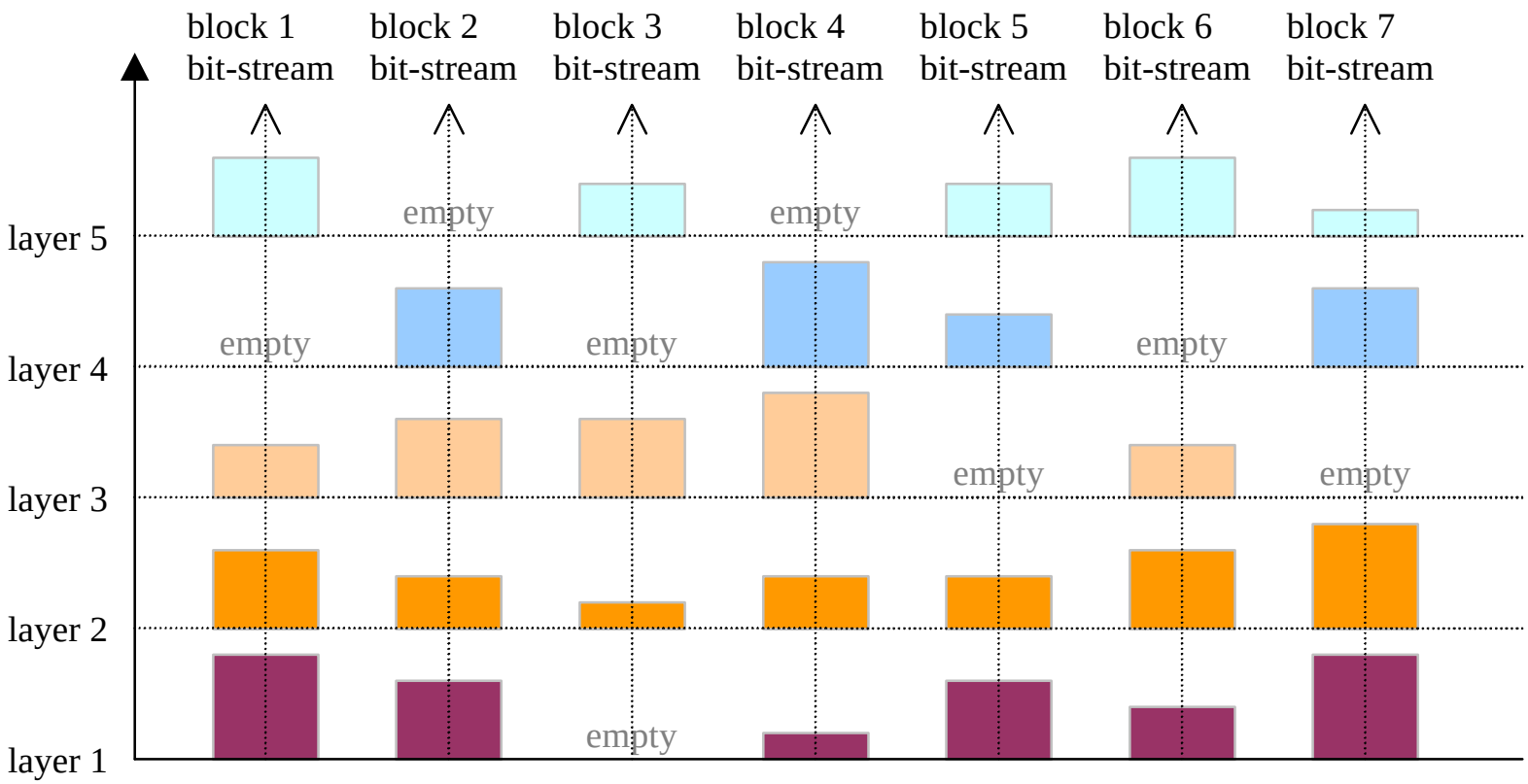
- **T2**: layered bitstream formation
- **T1**: generation of embedded block bit-streams



EBCOT: layered bitstream formation

- Each bitstream is organized as a succession of layers
- Each layer contains additional contributions from each block (some contributions might be empty)
- Block truncation points associated with each layer are optimal in the rate distortion sense
- rate distortion optimization is performed but it does not need to be standardized

EBCOT layered formation



Types of Coding Operations

- Zero coding (ZC)
- Run-Length coding (RLC)
- Sign coding (SC)
- Magnitude refinement (MR)

  – Arithmetic coding is used
  – Reduced complexity in “lazy coding mode”

Type of coding operations (cont'd)

Run-Length Coding (RLC)

used in conjunction with the ZC primitive, in order to reduce the average number of binary symbols which must be encoded using the arithmetic coding engine

Sign coding (SC)

used at most once for each sample in the block immediately a previously insignificant symbol is found to be significant during a Zero Coding or Run-Length Coding operation

Magnitude Refinement (MR)

used to encode an already significant sample

Types of Coding Operations (cont'd)

- If the sample is non yet significant, a combination of the "Zero Coding" (ZC) and "Run-Length Coding" (RLC) primitives is used to encode whether or not the symbol is significant in the current bit-plane

- If so, the "Sign Coding" (SC) primitive must also be invoked to send the sign

- If the sample is already significant, the "Magnitude Refinement" primitive is used to encode the new bit-position

Zero Coding (ZC)

- Use of 1 of 9 different context states to code the value of the symbol, depending upon the significance state variables of:

- Immediate horizontal neighbors (h)
- Immediate vertical neighbors (v)
- Immediate diagonal neighbors (d)
- Non-immediate neighbors (f)