Data Storage

JMU Computer Science Content Teaching Academy 2014

Florian Buchholz
buchhofp@jmu.edu

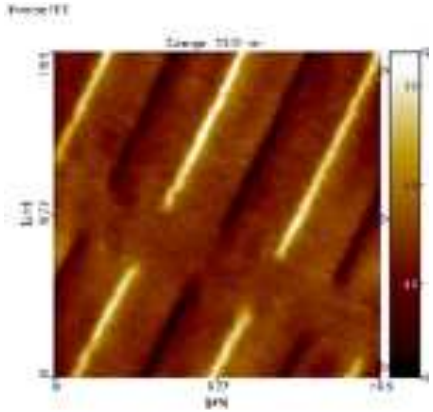
Abstraction layers to interpret data and information

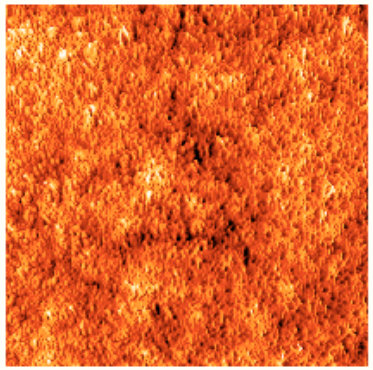
- Physical layer
  - Data is physically stored
- Device BIOS
  - Chunks of data can be read/written
- Partitions
- File System
  - Naming capabilities
  - Metadata
- Application layer(s)
  - Data becomes information
  - Several layers possible

How data is stored on physical media

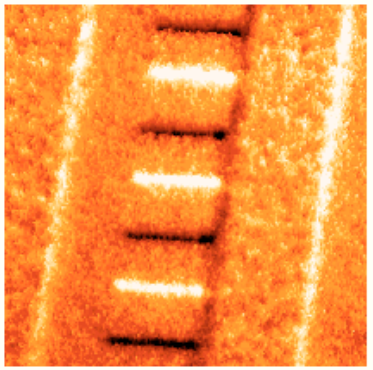
- Link between analog and digital world
- Encode bits as physical manifestations
  - Magnetic (hard drives)
  - Optical (CD, DVD)
- Bits read and written by head off disk surface
  - Magnetic head
  - Laser (optical)
- Surface imperfections, magnetic interference, dirt, etc. can cause errors
  - Limited error correction possible

Microscope images of disk surfaces

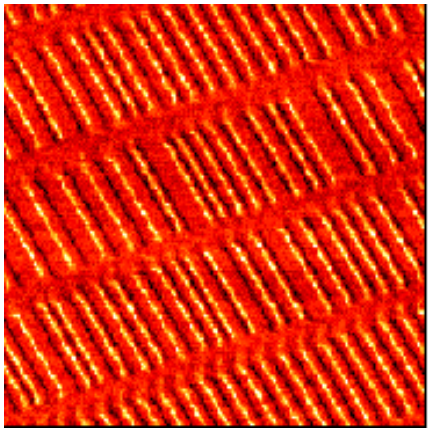


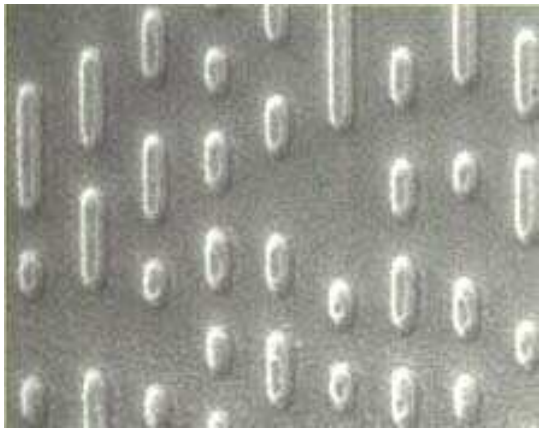


Surface Texture

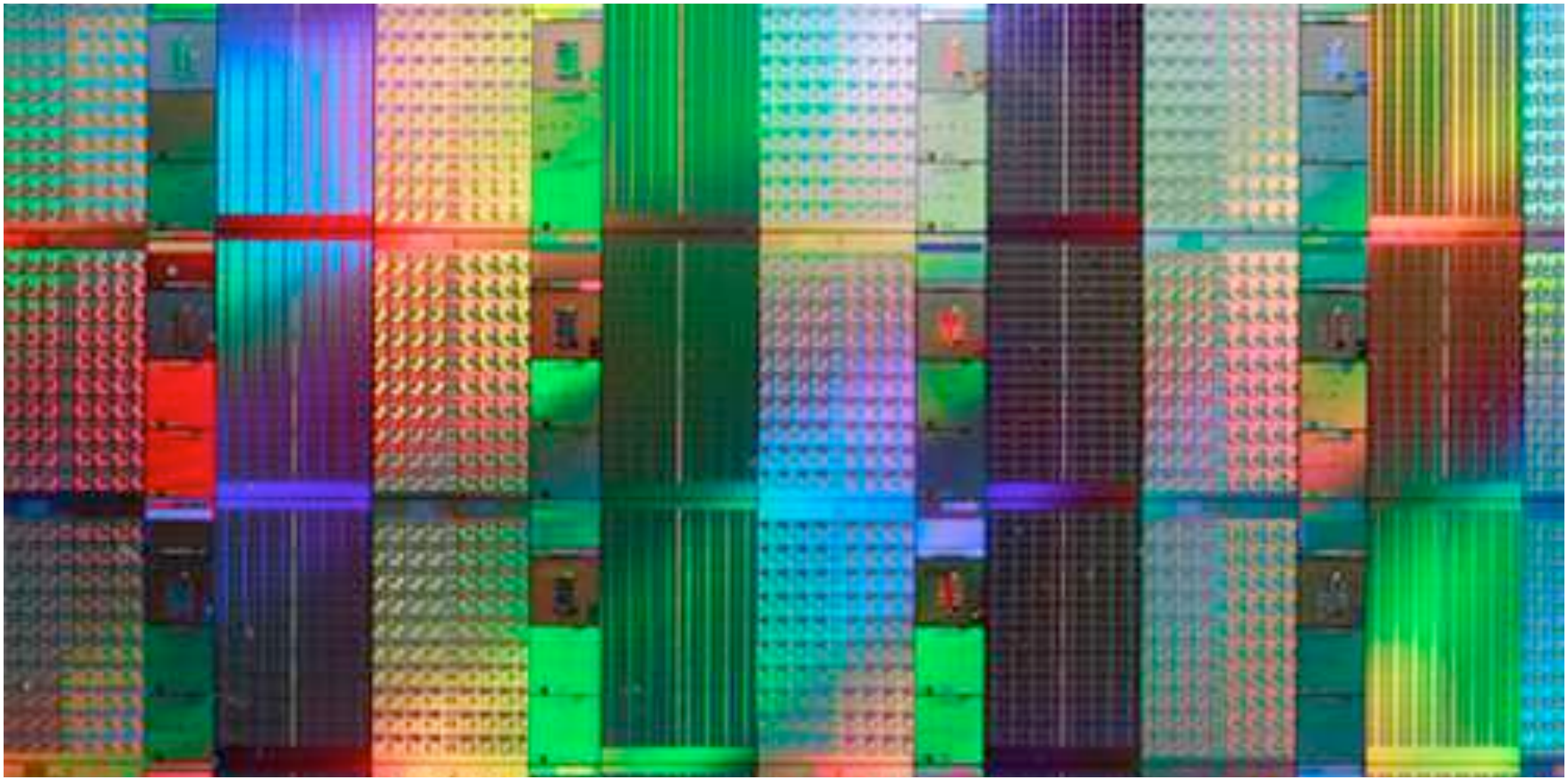


Magnetic Force Image

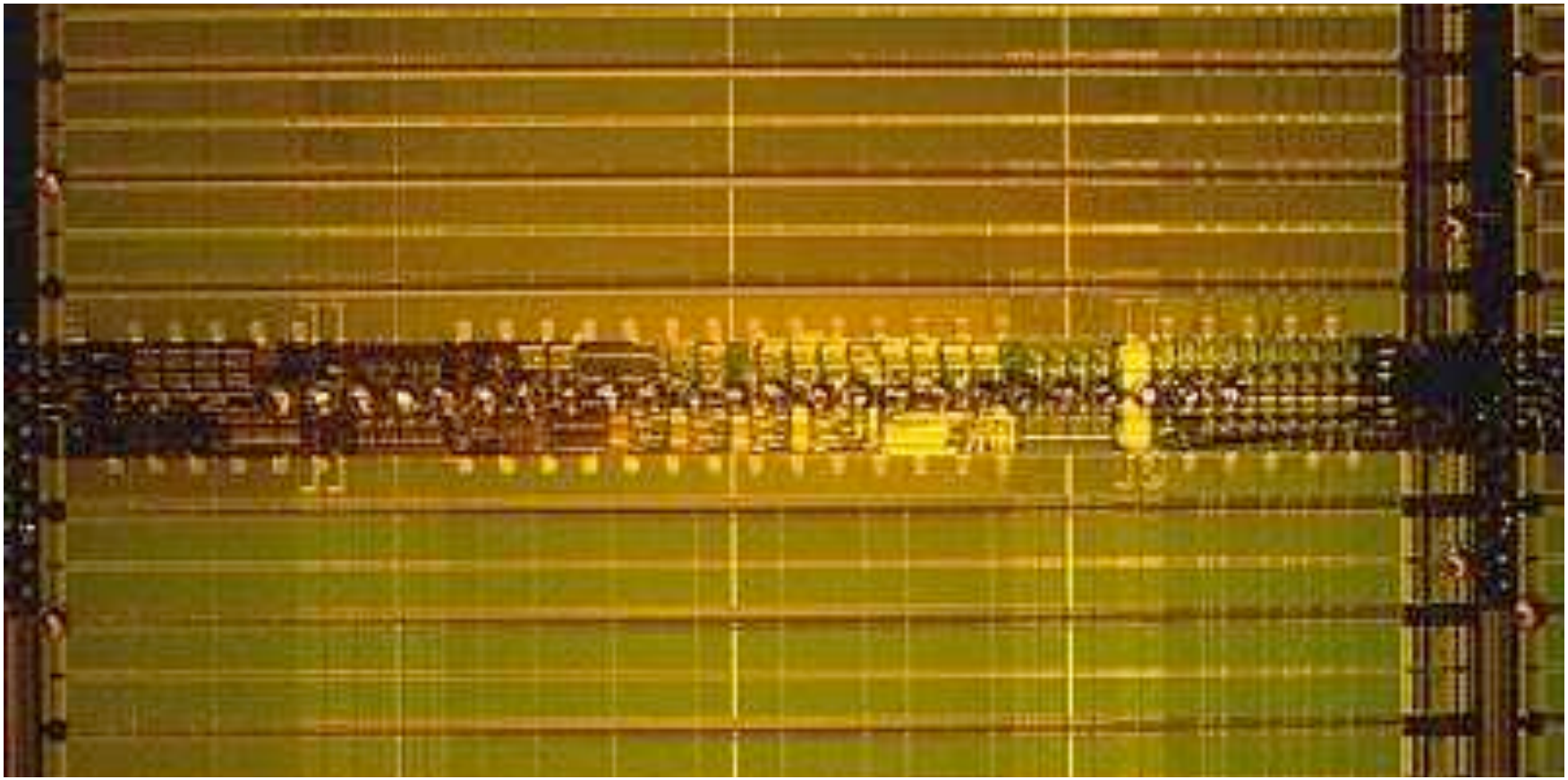




Flash chip



Memory chip



Physical data organization

- Platter/Cylinder/Layer/Head
  - Denotes the layer on which data resides if more than one is present
- Track
  - Concentric circle track on a layer
- Sector
  - Fragment of a track
  - Result of “intersection of lines” from the center of the disk to outer edge
- Cluster
  - Several successive sectors on the same track

Organization of one disk layer

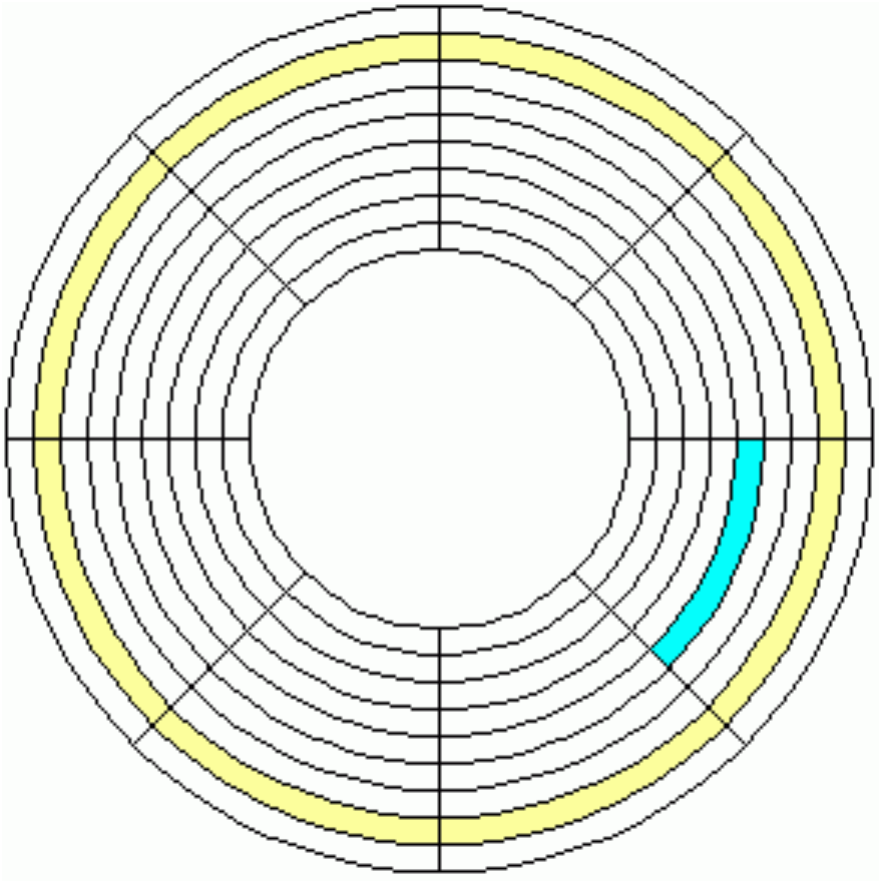
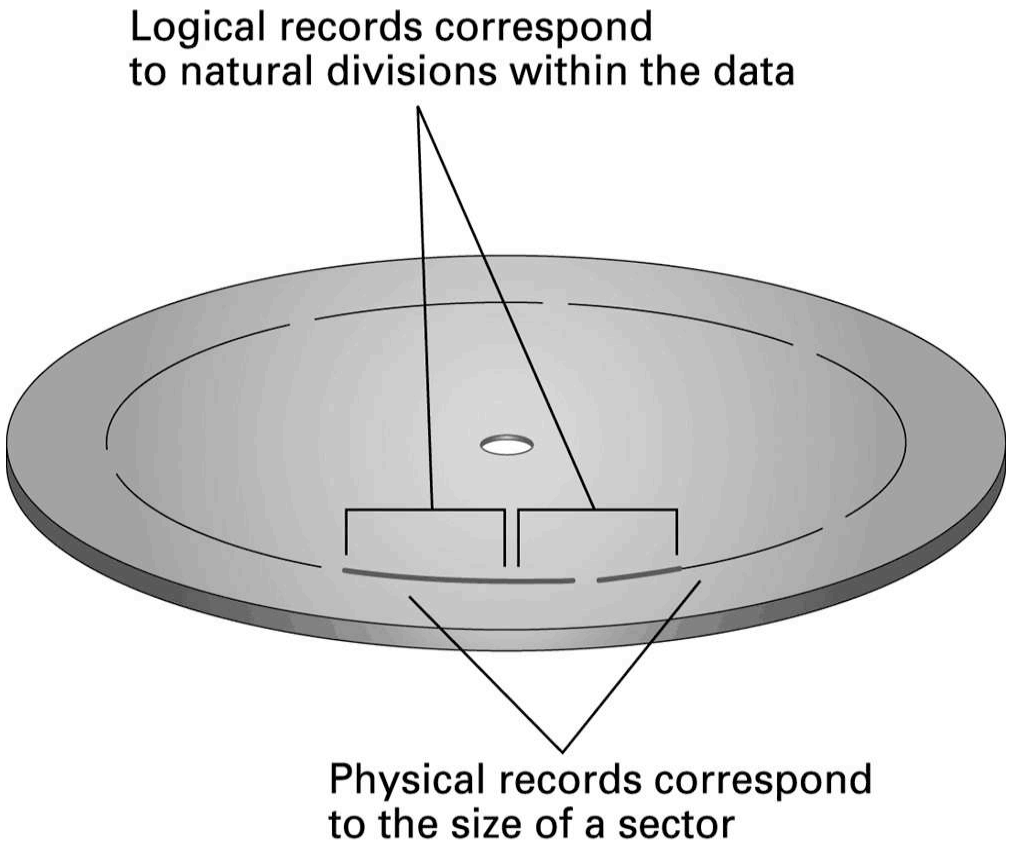


Figure 1.12 **Logical records versus physical records on a disk**



Device BIOS

- Abstraction layer to the next level
- Translates physical addressing scheme to logical
  - Cyl/Track/Sec to a logical sector/cluster number
  - Disk is seen as contiguous space
- Device can be queried and modified
  - “Give me the data from Sector x”
  - “Write this data to Sector y”
  - Sectors are typically 512 bytes
- BIOS also re-maps bad sectors
- Host Protected Area
  - Keeps statistics and mappings

Partition layer

- Divides disk space up into logical units
- Master boot record
- Primary partitions
- Logical partition
- Separation of information
- Different file systems can be used on the same disk

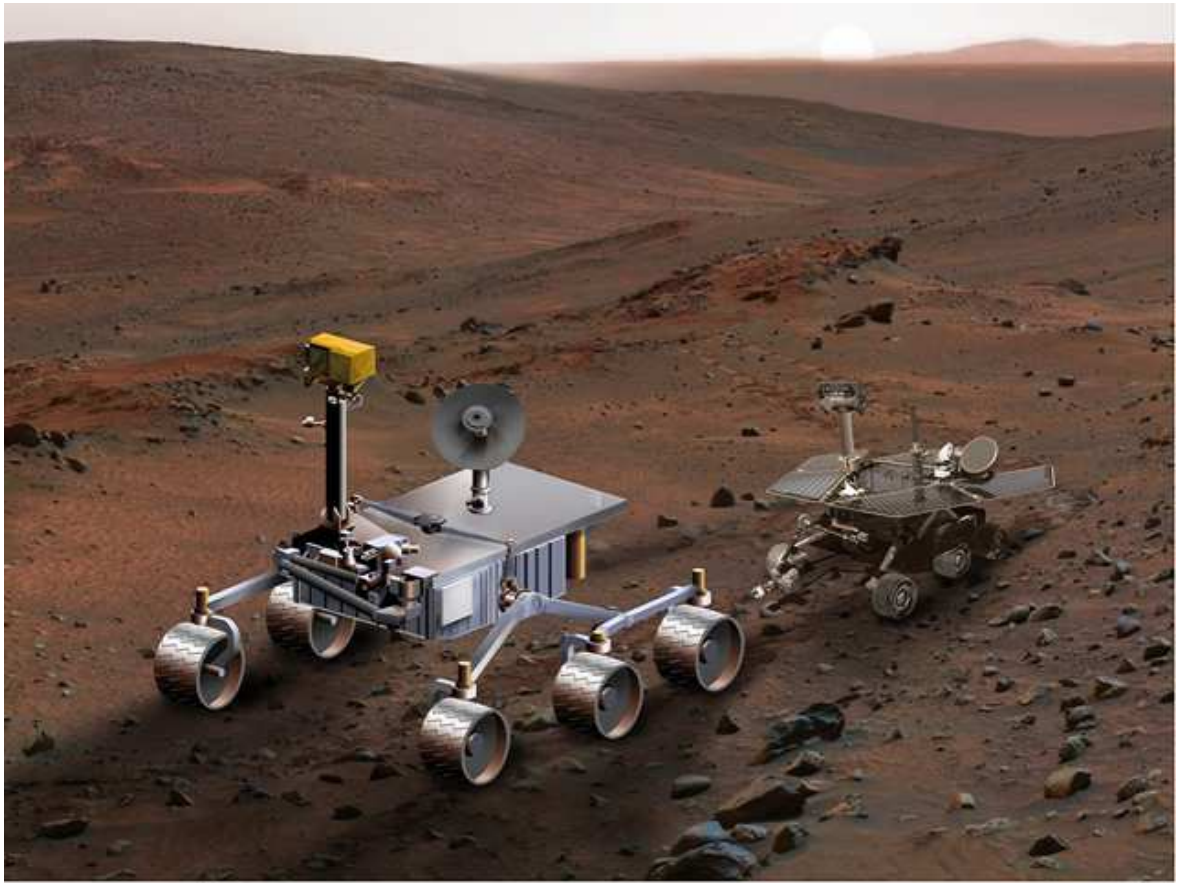
File system layer

- View partition as contiguous block of space.
  - Smallest unit that can be addressed is a sector/block/cluster
- Allows naming of files
  - Data is associated with a name, where data is stored within the partition is irrelevant.
  - Maps space “belonging” to file to sectors/blocks
- Directory hierarchy
  - Organize files
  - Unique paths: files of same name can exist
- Metadata
  - Timestamps
  - User information
  - Permissions

Application layer

- To this point all abstraction layers merely provide a “blob of bits”
  - Raw data has no meaning
- Applications interpret the data as information
- Extract data necessary for application
  - Image data, text, sound, etc.
- Metadata
  - Comments, timestamps, user information, etc.

Different views of a file



Different views of a file

Format: JPEG (Joint Photographic Experts Group JFIF format)
Geometry: 640x481
Class: DirectClass
Type: TrueColor
Endianess: Undefined
Colorspace: RGB
Depth: 8 bits
Channel depth:
Red: 8-bits
Green: 8-bits
Blue: 8-bits
Channel statistics:
Red:
Min: 0 (0)
Max: 255 (1)
Mean: 101.175 (0.396763)
Standard deviation: 48.6853 (0.190923)
Green:
Min: 0 (0)
Max: 255 (1)
Mean: 81.2703 (0.318707)
Standard deviation: 49.2262 (0.193044)
Blue:
Min: 0 (0)
Max: 255 (1)
Mean: 70.9726 (0.278324)
Standard deviation: 51.1614 (0.200633)
Colors: 32917

Rendering-intent: Undefined
Resolution: 72x72
Units: PixelsPerInch
Filesize: 280kb
Interlace: None
Background Color: white
Border Color: #DFDFDF
Matte Color: grey74
Dispose: Undefined
Iterations: 0
Compression: JPEG
Quality: 99
Orientation: Undefined
JPEG-Colorspace: 2
JPEG-Sampling-factors: 1x1,1x1,1x1
Signature:
0f439ec4bd57f9311a709ea5feed539865563aee181fa9495ba5188b0b197fa1
Profile-xmp: 6461 bytes
Profile-exif: 5859 bytes
Orientation: 1
X Resolution: 72/1
Y Resolution: 72/1
Resolution Unit: 2
Software: Adobe Photoshop CS Macintosh.
Date Time: 2005:12:01 17:40:18.
Exif Offset: 164
Color Space: 65535

Different views of a file

```
00000000  ff d8 ff e0 00 10 4a 46  49 46 00 01 02 01 00 48  |......JFIF.....H|
00000010  00 48 00 00 ff e1 16 e5  45 78 69 66 00 00 4d 4d  |.H......Exif..MM|
00000020  00 2a 00 00 00 08 00 07  01 12 00 03 00 00 00 01  |.*..............|
00000030  00 01 00 00 01 1a 00 05  00 00 00 01 00 00 00 62  |...............b|
00000040  01 1b 00 05 00 00 00 01  00 00 00 6a 01 28 00 03  |...........j.(..|
00000050  00 00 00 01 00 02 00 00  01 31 00 02 00 00 00 1d  |.........1......|
00000060  00 00 00 72 01 32 00 02  00 00 00 14 00 00 00 8f  |...r.2..........|
00000070  87 69 00 04 00 00 00 01  00 00 00 a4 00 00 00 d0  |.i..............|
00000080  00 00 00 48 00 00 00 01  00 00 00 48 00 00 00 01  |...H.......H....|
00000090  41 64 6f 62 65 20 50 68  6f 74 6f 73 68 6f 70 20  |Adobe Photoshop |
000000a0  43 53 20 4d 61 63 69 6e  74 6f 73 68 00 32 30 30  |CS Macintosh.200|
000000b0  35 3a 31 32 3a 30 31 20  31 37 3a 34 30 3a 31 38  |5:12:01 17:40:18|
…
```

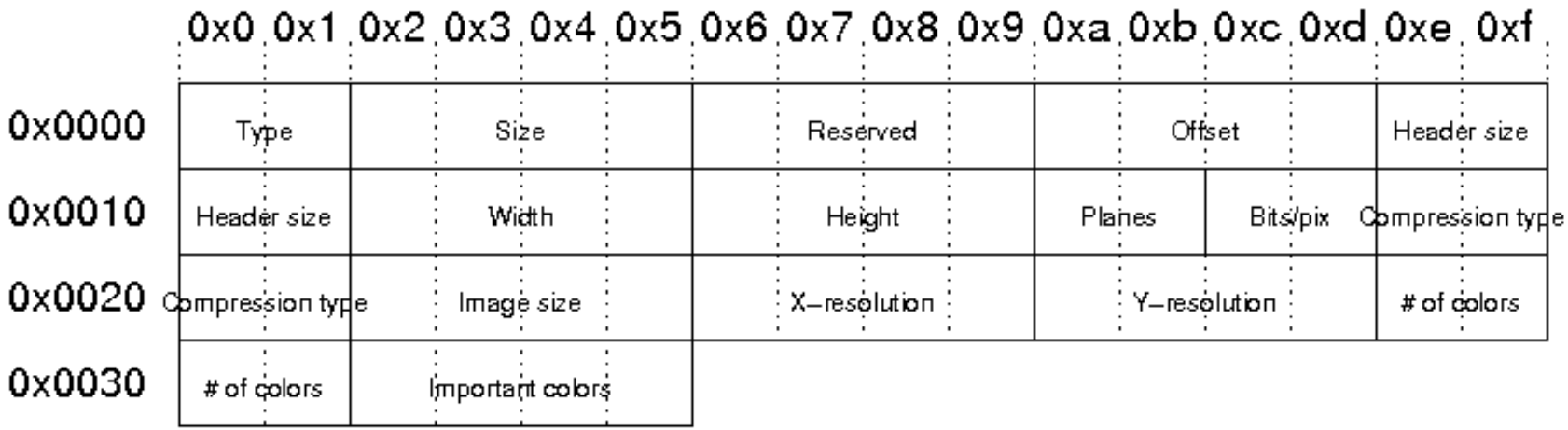
Representing Images

- Bit map techniques
  - Pixel: short for “picture element”
  - RGB
  - Luminance and chrominance
- Vector techniques
  - Scalable
  - TrueType and PostScript

Bitmap files

- Header information at start of the file
  - Contains information about the image
- The remainder of the file is a list of pixel data
  - Three color values for each pixel
    - Red, green, blue
    - Generally 1 byte for each color (24-bit bitmap)
  - First pixel in the file is generally in lower right corner of the image
    - Can be in upper right corner if “height” value is negative
  - Information from the header is used to determine when a new line starts

Bitmap header



Byte order (Endianness)

- When more than one byte is needed to represent an integer, it is important to know how the computer architecture stores the bytes
- Little-endian
  - The least significant byte is stored first
    - If we want to store 0x01020304, then the on-disk (or in-memory) bytes look like this:
      04 03 02 01
- Big-endian
  - The most significant byte is stored first
    - If we want to store 0x01020304, then the on-disk (or in-memory) bytes look like this:
      01 02 03 04