



*Open up the box of a computer, and you won't find any numbers in there. You'll find electromagnetic fields. Just as if you open up a person's brain case, you won't find symbols; you'll find neurons. You can use those things, either neurons or electromagnetic fields, to represent any patterns you like. A computer could care less whether those patterns denote words, numbers, or pictures. Sure, in one sense, there are bits inside a computer, but what's important is not that they can do fast arithmetic but that they can manipulate symbols. That's how humans can think, and that's the basic hypothesis I operate from.*

*- Herbert Simon, OMNI Magazine (June 1994)*

“omni”: of all things



## Digital Modeling

- Digital computers perform operations on represented data.
- All data is represented by numbers.
  - What kinds of data (or information) are there?
- Mixed success:
  - A wide range of things can be modeled.
  - Some things are very difficult to model.

kinds of data that need to modeled: numbers, letters, colors, pictures, sounds, videos, etc.

more to model: chemical reactions, disease spread, airflows, population growth/decline, etc etc etc . . .



# Binary

- Binary is a *base-2* numbering system.
- A *bit* is a “binary digit”:
  - 0 (or “off”)
  - 1 (or “on”)
- Binary is just as powerful as decimal -- no more or less.

Decimal	Binary
0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111



## Decimal Numbers

Decimal numbers are base-10 (using digits 0-9)

$$\begin{array}{c} 123 \\ \swarrow \quad \downarrow \quad \searrow \\ 1*10^2 + 2*10^1 + 3*10^0 \\ \downarrow \quad \downarrow \quad \downarrow \\ 1*100 + 2*10 + 3*1 \\ \downarrow \quad \downarrow \quad \downarrow \\ 100 + 20 + 3 \end{array}$$



## Binary Numbers

Binary numbers are base-2 (using digits 0 & 1)

$$\begin{array}{ccccc} & & 110_2 & & \\ & \swarrow & \downarrow & \searrow & \\ 1*2^2 & + & 1*2^1 & + & 0*2^0 \\ \downarrow & & \downarrow & & \downarrow \\ 1*4 & + & 1*2 & + & 0*1 \\ \downarrow & & \downarrow & & \downarrow \\ 4 & + & 2 & + & 0 = 6_{10} \end{array}$$

© Keith Vander Linden, 2005  
Jeremy D. Frens, 2008



## Digitizing Numbers

- Numbers are represented in memory using a binary encoding scheme.
  - Storing positive numbers is pretty obvious.
  - What about negative numbers?
  - What about “decimals”?
  - What about really really really big numbers?
- That’s why there are standard “encoding schemes”.

It’s all binary underneath!

*2’s compliment* for integers

*Floating point* for real numbers



## Digitizing Characters

- Each character is assigned an integer value.
  - Programs keep track of which memory locations store character data.
  - Programs display the right glyph on the screen.
- Two common schemes:
  - ASCII
  - Unicode

It's all binary underneath!



# ASCII

- **American Standard Code for Information Interchange.**
- **Uses 7 bit integers**
  - $2^7 = 128$  different characters
- **Extended ASCII uses 8 bit integers**
  - $2^8 = 256$  characters
- **ASCII is the most common code currently used.**

Character	ASCII Code
A	100 0001
B	100 0010
C	100 0011
D	100 0100
...	...
a	110 0001
b	110 0010
c	110 0011
d	110 0100
...	...
0	011 0000
1	011 0001
2	011 0010
3	011 0011
...	...
<space>	010 0000
.	010 1110
...	...

© Keith Vander Linden, 2005  
Jeremy D. Frens, 2008

Note the “numerals” have their own ASCII code which means that “1” is different from 1. Confusing!





# Unicode

- Uses 8 - 32 bit integers
  - over a million characters defined.
- Unicode supports a number of different character types.
  - Cyrillic
  - Ancient Coptic
  - All charts

3400		CJK Unified Ideographs Extension A																34DF	
		3400	3401	3402	3403	3404	3405	3406	3407	3408	3409	340A	340B	340C	340D	340E	340F		
0	北	𠄎	𠄏	𠄐	𠄑	𠄒	𠄓	𠄔	𠄕	𠄖	𠄗	𠄘	𠄙	𠄚	𠄛	𠄜	𠄝	𠄞	𠄟
1	𠄠	𠄡	𠄢	𠄣	𠄤	𠄥	𠄦	𠄧	𠄨	𠄩	𠄪	𠄫	𠄬	𠄭	𠄮	𠄯	𠄰	𠄱	𠄲
2	𠄳	𠄴	𠄵	𠄶	𠄷	𠄸	𠄹	𠄺	𠄻	𠄼	𠄽	𠄾	𠄿	𠅀	𠅁	𠅂	𠅃	𠅄	𠅅
3	𠅆	𠅇	𠅈	𠅉	𠅊	𠅋	𠅌	𠅍	𠅎	𠅏	𠅐	𠅑	𠅒	𠅓	𠅔	𠅕	𠅖	𠅗	𠅘
4	𠅙	𠅚	𠅛	𠅜	𠅝	𠅞	𠅟	𠅠	𠅡	𠅢	𠅣	𠅤	𠅥	𠅦	𠅧	𠅨	𠅩	𠅪	𠅫
5	𠅬	𠅭	𠅮	𠅯	𠅰	𠅱	𠅲	𠅳	𠅴	𠅵	𠅶	𠅷	𠅸	𠅹	𠅺	𠅻	𠅼	𠅽	𠅾
6	𠅿	𠆀	𠆁	𠆂	𠆃	𠆄	𠆅	𠆆	𠆇	𠆈	𠆉	𠆊	𠆋	𠆌	𠆍	𠆎	𠆏	𠆐	𠆑
7	𠆒	𠆓	𠆔	𠆕	𠆖	𠆗	𠆘	𠆙	𠆚	𠆛	𠆜	𠆝	𠆞	𠆟	𠆠	𠆡	𠆢	𠆣	𠆤
8	𠆥	𠆦	𠆧	𠆨	𠆩	𠆪	𠆫	𠆬	𠆭	𠆮	𠆯	𠆰	𠆱	𠆲	𠆳	𠆴	𠆵	𠆶	𠆷
9	𠆸	𠆹	𠆺	𠆻	𠆼	𠆽	𠆾	𠆿	𠇀	𠇁	𠇂	𠇃	𠇄	𠇅	𠇆	𠇇	𠇈	𠇉	𠇊
A	𠇋	𠇌	𠇍	𠇎	𠇏	𠇐	𠇑	𠇒	𠇓	𠇔	𠇕	𠇖	𠇗	𠇘	𠇙	𠇚	𠇛	𠇜	𠇝
B	𠇞	𠇟	𠇠	𠇡	𠇢	𠇣	𠇤	𠇥	𠇦	𠇧	𠇨	𠇩	𠇪	𠇫	𠇬	𠇭	𠇮	𠇯	𠇰
C	𠇱	𠇲	𠇳	𠇴	𠇵	𠇶	𠇷	𠇸	𠇹	𠇺	𠇻	𠇼	𠇽	𠇾	𠇿	𠈀	𠈁	𠈂	𠈃
D	𠈄	𠈅	𠈆	𠈇	𠈈	𠈉	𠈊	𠈋	𠈌	𠈍	𠈎	𠈏	𠈐	𠈑	𠈒	𠈓	𠈔	𠈕	𠈖
E	𠈗	𠈘	𠈙	𠈚	𠈛	𠈜	𠈝	𠈞	𠈟	𠈠	𠈡	𠈢	𠈣	𠈤	𠈥	𠈦	𠈧	𠈨	𠈩
F	𠈪	𠈫	𠈬	𠈭	𠈮	𠈯	𠈰	𠈱	𠈲	𠈳	𠈴	𠈵	𠈶	𠈷	𠈸	𠈹	𠈺	𠈻	𠈼

The Unicode Standard 5.0 Copyright © 1991-2009 Unicode, Inc. All rights reserved. 601

© Keith Vander Linden, 2005  
Jeremy D. Frens, 2008

This is just a small part of the full unicode support for chinese characters.

Unicode is becoming more and more common.

[unicode.org/charts](http://unicode.org/charts)

<http://www.unicode.org/charts/PDF/U0400.pdf>

<http://www.unicode.org/charts/PDF/U2C80.pdf>



## Social justice and computing

- The *accessibility* of computers and readable character sets can be seen as an issue of social justice.

Related to the digital divide material included in the computer anatomy lectures.

Given the English-centric nature of the web, one might more accurately call it the **Western-wide web**.

Digital divide – the WWW is hard to access in:

- the developing world
- the non-western world
- underprivileged social classes
- the disabled community

What could we do to help bridge this divide?

- Unicode
- internationalized domain name resolution
- better translation tools
- better international/disabled design and testing



## Digitizing Big Data

- Using numbers and characters, we can digitize and model bigger things:
  - Documents
  - Accountant's ledger
  - City maps
  - Human behaviors
  - Calendars
  - Images, audio, video
- Programs make sense of the data.



## Digitizing Multimedia Data

- Multimedia data is usually
  - **HUGE**
  - And highly patterned
- *Compress* the data by taking advantage of the patterns to take up less space.
  - Lossless compression doesn't lose any information.
  - Lossy compression loses some information for better compression.



## Digitizing Images

- An image is an array of pixels.
- Each pixel has:
  - intensity values for Red, Green & Blue
  - an optional *alpha* value for transparency
- Common image file formats include:
  - PNG: Lossless, often seen on the web
  - GIF: Lossless, 256 colors max, but patented
  - JPEG: Lossy, compression 3:1 to 60:1
  - TIFF: Lossless
  - BMP: Uncompressed Windows format

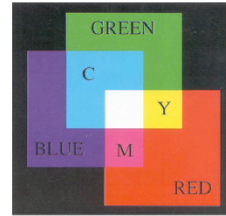


image from Harry Plantinga

© Keith Vander Linden, 2005  
Jeremy D. Frens, 2008

Check out this article on digitally edited photos:

Can Photos Be Trusted? - Popular Science

<http://www.popsci.com/popsci/technology/generaltechnology/d6002684e4646010vgnvcm1000004eecbccdrd.html>



## Use of Image Formats

- PNG: web, logos and text
- GIF: web, logos and text, animation
- JPEG: photos
- TIFF: imaging software
- BMP: never

Check out this article on digitally edited photos:

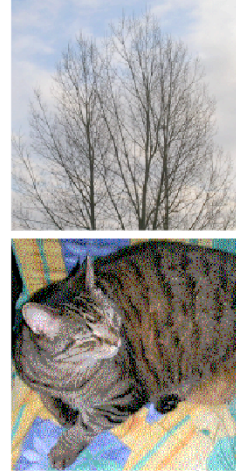
Can Photos Be Trusted? - Popular Science

<http://www.popsci.com/popsci/technology/generaltechnology/d6002684e4646010vgnvcm1000004eecbccdrerd.html>



## Steganography

- Cryptography encrypts messages using encryption keys.
- Steganography *hides* messages in other digital media.



© Keith-VanderLinden, 2009  
Jeremy D. Frens, 2008

The bottom picture was hidden in the last 2 bits of the pixel codes of the top picture and recovered. People looking at the original would never have known that the cat was there, except, perhaps, by noticing that the relatively large size of the image file is not congruent with the relatively poor resolution of image.

<http://en.wikipedia.org/wiki/Steganography>

<http://www.calvin.edu/~lave/s-tools/>

This works for GIF but not for JPG (because of the way JPG codes the colors for compression).

<http://www.stegoarchive.com/>



## Digitizing Audio

- Sound can also be digitized.
- Common sound file formats:
  - mp3** – open, patented, no DRM, older/less effective
  - wma** – Windows Media Audio, patented/proprietary, DRM
  - AAC** – Apple’s iTunes, patented, proprietary, DRM
  - RealAudio** – patented, proprietary, DRM
  - Ogg Vorbis** – unpatented, open, no DRM

DRM – digital rights management (see wikipedia)

Lossy vs. lossless

Fights over “standards”

DRM is an ever-more-important issue





## Digitizing Video

- Common movie file formats:
  - mpeg** – Open (but patented) standard
  - avi** – Windows Media Player
  - DV** – As used in digital camcorders
  - divx** – very high compression ratios



## How does a computer know what it is looking at?

- Windows tells what kind of thing is being modeled by looking at a file's suffix (or *extension*) :
  - .txt: text file (in ASCII or Unicode)
  - .jpg, .png, .bmp: image
  - .xls, .xlsx: Excel spreadsheet
  - .zip: a compressed folder of files/folders.
  - .doc, .docx, .rtf: Word documents
- Linux stores file type in the file itself.

If you change the suffix of a file name, Windows thinks it is a different kind of file and will try to open it with a different program.



## The Difficulty of Modeling

- Not everything can be easily modeled.
  
- “I praise you because I am fearfully and wonderfully made.” - Psalm 139:14

The weather (just too much stuff) (although this is getting much better every year)

The human genome (just too much stuff we don't understand)

Human intelligence (AI – to the sussman anomaly example here) - Easy things are hard, hard things are easy. E.g., Being human is harder than it looks. “One year in AI is enough to make one believe in God” – Alan Perlis.