Advanced Lecture on Internet Applications

4. Text based Communication:

Character Code and Internationalization

Masataka Ohta

mohta@necom830.hpcl.titech.ac.jp

ftp://ftp.hpcl.titech.ac.jp/appli4e.ppt

Reference Material

- 太田昌孝、「いま日本語が危ない」、光芒社、ISBN4－89542－146－5、平成9年

What is “Character”

- unit to represent language by graphic symbol
  - phonetic character
  - ideogram

What is Script（用字系）?

- system to represent language by characters
- never confuse language and script
  - 「でぃすいずあぺん」 is English by Kana script
  - “Koreha pen desu.” is Japanese by Roman (Latin) script

Scripts to Represent Japanese

- kana (hiragana, katakana, manyogana)
- mixed kanji kana
- romaji (Hepburn, Monbusho, etc.)
  - “masataka” in French should be “massataka”
- and phonetic representations in various local script systems such as Hangul

Digital and Analog

- digital ignores small differences
  - can remove noise
- language (incl. spoken one) is digital
  - voice and song are analog
- character is digital
  - can represent very subtle feelings with 17 characters
  - calligraphy is analog
- to what extent, small differences should be ignored? (how many bits should be used?)

Character Code

- an encoding rule for strings using characters of a character set
  - not merely assign code (number) to characters
    - the rule may be very complicated
- the number of characters of a character set matters
  - if large, many bits are necessary
  - if small, many characters can’t be represented
    - small differences between similar characters can’t be represented

Byte

- originally mean # of bits to represent a character
  - 1 byte is not always 8 bits
- 4 bit byte can represent 16 characters
  - enough for numbers and “,.+-$ “
- 6 bit byte can represent 64 characters
  - enough for capital Latin letters, numbers and symbols
  - used on 36bit/word computers
- ASCII use 7 (not 8) bit byte

Multi Byte Character Code

- self-contradictory concept
- simple if 1 byte = 1 character
- as the number of bits of a byte is fixed for a long time
  - becomes practically impossible to extend byte
- to represent a character by multiple bytes
  - represent a character by sequence of bytes
    - multi byte character
  - switch character sets by special bytes (control “character”)

ASCII (American Standard Code for Information Interchange)

- US standard 7 bit byte character code
  - – 95 (incl. space) graphic character set
    - capital and small letters, digits, symbols
  - – 33 control characters
  - – US local version of ISO 646
- enough to represent English
- simple character set in various ways
  - – easy to computerise

ISO 646

- have same structure as ASCII
- among 95 (graphic) characters
  - 83 characters are internationally common
  - 12 characters can be different country by country
- japanese version of JIS X 0201 (JIS C 6220) Latin have two characters different from ASCII
  - “＼” to “￥”, “～” to “￣”

Simplicity of ASCII (or Latin Script for English) (1)

- small number of characters
- horizontal only
- single (left to right) directional only
- ligature (variation of character shape by previous/next characters) is not necessary
- commonly shared recognition for character identifications and character shapes

Simplicity of ASCII (or Latin Script for English) (2)

- correspondence between small/capital characters is clear and regular
- no characters with diacritical marks
  - such as “ä”
- character width can be constant
- widely spread and usable everywhere

Small Number of Characters of ASCII

- can represent all the characters by a single byte
- multi byte characters or character set switching not necessary

Latin Script of English needs Horizontal Writing only

- proper writing of Kanji is vertical
- Mongolian script is vertical, too

Latin Script of English is Unidirectional (left to right)

- horizontally written Kanji script is written from right to left
  - actually is vertical writing with 1 character/line
  - left to right horizontal writing introduced in Meiji era
- Arabic script is written right to left
  - numbers and Latin characters (quoted English etc.) are written left to right
    - directionality changes may be nested

Ligature is not necessary for Latin Script of English

- ligature
  - – variation of character shape by previous/next characters
  - – “i” of “fi” and “ffi” may be combined with “f”
    - may not be combined and not available with ASCII
- shape of Arabic and Devanagari (Indian) characters affected by previous/next characters
  - – natural with hand writing
  - – printing type not adopted

Commonly Shared Recognition on Latin Characters in English

- originally with Latin characters
  - “u” and “v” are same character (BVLGARI)
  - “W” is “UU” (double “U”)
- they are separate characters in modern Latin script for English
  - “a” and “ɑ” are same character

Regular Correspondence between Small/Capital Characters

- some Latin characters may have irregular case correspondences
  - capital form of “ÿ” may be “Y”, “Ÿ” or “IJ”
  - “y” “ÿ” “ij”
- no such irregularity in modern Latin script of English

No characters with Diacritical Marks in ASCII

- diacritical marks are introduced
  - to represent intermediate pronounciation
    - “Å” is “A” pronounced with flavor of “o”
    - “ö” is “o” pronounced with flavor of “e”
      - “e” became “..”

Latin Characters may have Fixed Spacing

- # of bytes of a string is proportional to display width
- “character display” was popular
  - to display 80*25 characters with 2kB RAM
    - and character generator ROM of, say, 7*9*128 dots
- “line printer” was popular
  - to print a line of 132 characters at once
    - as a drum with 132 columns rotates once

ASCII is Widely Spread and may be used as Default

- not necessary to specify character set

Wrap Up

- character enables graphical representation of language
- character code is a rule to translate strings to byte sequences
- character code is restricted by the number of bits of a byte
- ASCII is “simple” character code in various ways