Character Sets

Vermont Technical College
Peter C. Chapin

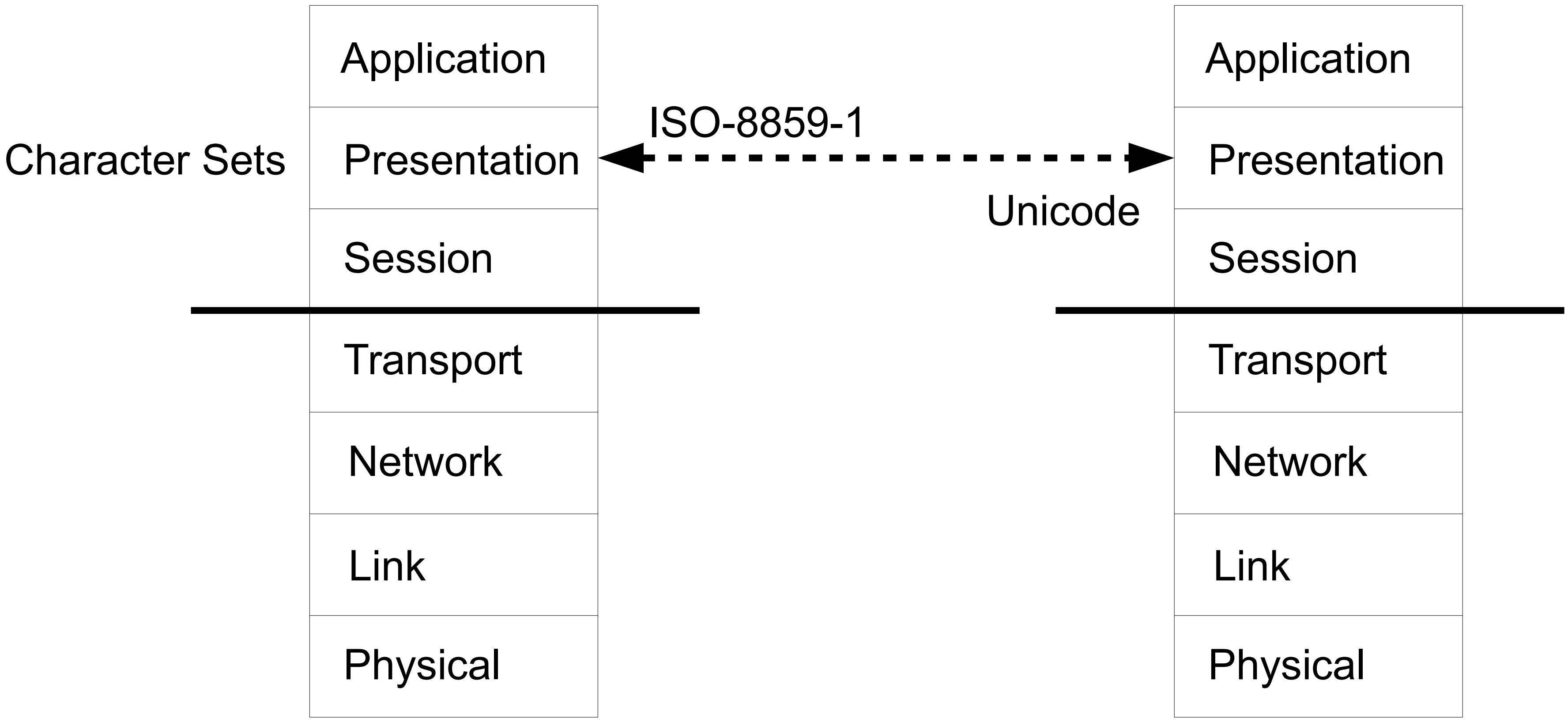
Problem?

- Sharing text requires common character sets.
  - Computers deal with numbers.
  - Humans deal with characters.
  - Must define bidirectional mapping from numbers to characters.
    - 0x41 <=> 'A'
- Many different writing systems...
  - Prompts many different character sets.
  - Conversion, Translation, Interpretation complicated.

Network Problem?

- Why does this matter to networking?
  - Low level protocols don't care
    - Data just a bag of bytes.
    - TCP/IP transports the bytes raw.
    - No semantic interpretation.
  - High level (application) protocols do care
    - Interact with humans, and humans care about characters
  - Any other text sharing application
    - Shared files
    - Databases
    - etc...

Presentation Layer



Character set translation is a network protocol issue

Some Terminology

- *Code Point*
  - The numeric value corresponding to a character
- *Encoding*
  - The way in which code points are mapped to octets
- *Glyph*
  - The image of a character
- *Font*
  - A collection of glyphs for every code point

Encoding?

- The way in which code points are represented
  - For simple character sets, there is no encoding (or maybe it is just the “nop” encoding)
    - The octets used are the same as the code points (ASCII is like this)
  - For complex character sets, there can be multiple encodings
    - Sender/receiver must agree on both the character set and the encoding used (Unicode is like this)
  - There is often confusion/ambiguity about this issue.
    - The terms “encoding” and “character set” are sometimes used interchangeably

ASCII

- “**A**merican **S**tandard **C**ode for **I**nformation **I**nterchange.”
  - Very old character set used for “plain text.”
  - 7 bits
    - Uses least significant bits of an 8 bit byte.
    - Most significant bit reserved for parity (error detection). Usually zero.
  - Only 128 characters.
    - Letter of the Latin alphabet (upper and lower case)
    - Digits
    - Various punctuation symbols
    - Control characters.

ASCII Trivia

- Various interesting characteristics of ASCII...
  - Codes assigned to letters are contiguous and in alphabetical order: 'A' => 0x41, 'B' => 0x42, etc.
    - BUT... All upper case letters come before any lower case letters.
      - So a simple comparison puts 'Z' < 'a'
      - Sometimes informally called “ASCII order.”
  - Digits are contiguous
    - BUT... codes assigned to digits are not the digit's numeric value: '0' => 0x30, '1' => 0x31, etc.
  - Control characters are 0x7F and 0x00 .. 0x1F.
    - Subtract 0x40 from an upper case letter: ^A => 0x01

More ASCII Trivia

- Control characters...
  - Most control characters are officially used to control data flow over certain (old) communications systems.
    - Such as RS-232 serial ports, etc.
  - Some control characters are used for formatting:
    - Backspace (0x08)
    - Horizontal tab (0x09)
    - Carriage return (0x0D)
    - Line feed (0x0A)
    - Form feed (0x0C)
  - Otherwise no formatting control.

ASCII Usage

- ASCII is very widely used.
  - “Plain text” usually means ASCII to most people.
  - Good for program source code.
    - Many (older) programming languages assume source will be in a subset of ASCII. Modern languages tend to use Unicode.
  - Good for configuration files.
- Very generic.
  - “Everybody” can read it.
  - Good for documents to be widely shared.
  - Good for archival documents (RFCs use ASCII).

ASCII Limitations

- There are many limitations to ASCII
  - No significant formatting control.
  - Limited character set.
    - Good for English but not much else.
    - Limited collection of special symbols.
      - Mathematical symbols.
      - Special punctuation symbols
    - NOTE: The lack of special symbols is a problem even for English speakers.
  - Lacks the usual typographical characters used in publishing (like curly double quotation marks)

Extended ASCII

- Use the 8th bit to double the number of characters.
    - Not many applications need the parity bit *and* are limited to only 8 bit transmission units.
        - In most cases it is possible to send 8 data bits *and* parity if necessary.
- What to do with the extra characters?
    - Many variations exist.
        - Creates compatibility problems.

Additional References

- ISO-646
  - http://en.wikipedia.org/wiki/Iso-646
    - International 7 bit character set family.
    - An old character set family. Not commonly used today.
- ISO-8859
  - http://en.wikipedia.org/wiki/ISO-8859
    - International 8 bit character set family.
    - Common usage today (especially ISO-8859-1).