05-CharacterEncoding

Introduzione

- Problema della codifica dei caratteri nel contesto della globalizzazione di Internet.
- Standard: ASCII, ISO/IEC 10646, UNICODE, UCS, UTF.
- Content encoding.

Curiosità

- ASCII: 8 bit ma solo 128 caratteri definiti (7 bit + 1 bit di parità).
- Codici di ritorno a capo: LF (Line Feed, `
`) e CR (Carriage Return, `
`).
- Codici di cancellazione: BS (Backspace, `&#08;`) e DEL (Delete, `&#127;`).
- DEL: unico codice di controllo non raggruppato (0-31).
- Errori di visualizzazione: lettere accentate scorrette a causa di codifiche errate.
- Codice FFFE proibito in UCS-2: Zero-Width No-Break Space (ZWNBSP).

Digitalizzazione di Dati Non-Numerici

- Digitalizzazione: associazione di un numero a un dato per identificarlo.
- Approccio "Divide et Impera": digitalizzazione di componenti atomici (caratteri nel testo, pixel in immagini).
- Testo: digitalizzazione tramite giustapposizione dei valori numerici associati ai singoli caratteri.

Set di Caratteri

- Problema: rappresentare correttamente gli alfabeti di migliaia di lingue.
- Necessità di un criterio non ambiguo per associare blocchi di bit a caratteri.

Rappresentazione Binaria del Testo

- Identificare gli elementi fondanti (caratteri).
- Identificare lo spazio di rappresentazione.
- Creare un mapping standardizzato.

Caratteri

- Entità atomica di un testo scritto.
- Variazioni tra alfabeti:
  - Maiuscole/minuscole (alfabeti di derivazione greca).
  - Segni diacritici (alfabeti di derivazione latina).
  - Modificatori grafici per vocali (ebraico).
  - Cambiamento di forma in base alla vicinanza (arabo).
  - Composizione di caratteri (cinese).
- Tre aspetti del carattere:
  - Natura (difficile da attribuire, es. A e À in italiano vs. A e Å in danese).
  - Forma (glifo, ambiguità tra alfabeti e font diversi).
  - Codice numerico (variabile in base alla tabella).

Spazio di Rappresentazione

- Associazione di valori numerici ai caratteri (arbitraria o secondo regole).

- Regole importanti:
  - Ordine: valori numerici seguono l'ordine alfabetico.
  - Contiguità: ogni valore tra il minimo e il massimo è associato a un carattere.
  - Raggruppamento: appartenenza a gruppi logici riconoscibile numericamente (es. in ASCII).

Altri Termini

- **Shift**: codice riservato che cambia la mappa dei caratteri.
- **Codici liberi**: codici non associati a nessun carattere (indicano errori).
- **Codici di controllo**: codici associati alla trasmissione, non al messaggio.

Notazioni Binarie ed Esadecimali

- Binario (0b): base 2 (0, 1).
- Esadecimale (0x): base 16 (0-9, A-F).
- Connessione diretta tra binario ed esadecimale: 4 bit = 1 cifra esadecimale.
- Esempio: 0b11001110 = 0xCE = 206.

Breve Ricapitolo di Matematica Binaria

- Numero di combinazioni possibili in base al numero di bit: 2^n (dove n è il numero di bit).

Baudot

- Inventato nel 1870 da Emile Baudot.
- Codice a 5 bit (32 codici possibili).
- Uso di shift per lettere e numeri (totale 64 codici).
- Codifica non contigua né ordinata.

ASCII (American Standard Code for Information Interchange)

- Standard ANSI (X3.4 - 1968).
- 128 caratteri (7 bit).
- 33 caratteri di controllo (inclusi BS, DEL, CR, LF).
- 95 caratteri dell'alfabeto latino, numeri e punteggiatura.
- Codifica contigua e ordinata.
- Nessun codice libero.

EBCDIC (Extended Binary Characters for Digital Interchange Code)

- Codifica proprietaria IBM (1965) a 8 bit.
- Usata nei mainframe IBM.
- 56 codici di controllo, molte locazioni vuote.
- Lettere dell'alfabeto NON contigue.

ISO 646-1991

- Codifica ISO per caratteri nazionali europei in contesto ASCII.
- International Reference Version (IRV) identica ad ASCII.
- Versioni nazionali con 12 codici liberi per caratteri specifici.
- Caratteri sacrificati: # $ @ \ ¬ ` { | } ~

Code Page di ASCII

- Estensioni di ASCII per usare i rimanenti 128 caratteri (128-255).
- IBM e Microsoft: code page multiple e indipendenti.
- Centinaia di code page esistenti.
- Necessità di fonti esterne per identificare la code page usata.
- Caratteri 0-127 sempre ASCII.
- Esempi:
    - CP 737 (greco).
    - KOI7 e KOI8 (cirillico).
    - Codepage per arabo (720, 864, 1256).
    - Codepage per ebraico (862).

Codifiche CJK (Cinese, Giapponese, Koreano)

- Condivisione di molti caratteri.
- Codifiche a 16 bit o più larghe (Unicode: 70.000 caratteri CJK).
- Supporto per caratteri Han e script fonetici specifici.
- Esempi: Big5, EUC-JP, GB18030, EUC-KR.

Alfabeti delle CJK

- Han-Kanji (cinese tradizionale e semplificato, giapponese tradizionale, coreano tradizionale).
- Pinyin (traslitterazione fonetica con alfabeto latino).
- Bopomofo (traslitterazione per cinese, usata a Taiwan).
- Hiragana (alfabeto fonetico giapponese).
- Katakana (alfabeto fonetico giapponese per parole straniere).
- Hangul (scrittura moderna coreana, sillabico).

ISO 8859/1 (ISO Latin 1)

- Estensione standard di ASCII (unica standard).
- Include caratteri di alfabeti europei (accenti, ecc.).
- Usato automaticamente da HTTP e alcuni sistemi operativi.
- Retrocompatibile con ASCII (solo caratteri >127 estesi).

Esigenza di uno Standard Internazionale

- Molte codifiche a 8 e 16 bit per alfabeti diversi.
- Interpretazioni diverse per lo stesso codice numerico.
- Necessità di meccanismi esterni per specificare la codifica.
- Problema dei flussi misti (necessità di shift).

Unicode e ISO/IEC 10646

- Standard unico sviluppato da due commissioni: ISO/IEC 10646 (dal 1989) e Unicode (dal 1991).
- Convergenza delle due commissioni.
- Versione 15.0 di Unicode e ISO/IEC 10646:2022: stessi codici per gli stessi caratteri.
- Codifiche a lunghezza fissa (UCS-2, UCS-4) e variabile (UTF-8, UTF-16, UTF-32).
- Scomparsa delle differenze tra UTF-32 e UCS-4 (2020).
- Versione 15.1 (Settembre 2023): 149.813 caratteri, 161 script, ~140.000 per scopi privati (PUA).
- Categorie di caratteri: script moderni, script antichi, segni speciali.
- Discussioni su emoji e toni della pelle.

Star Trek, Lord of the Ring e Unicode

- Alfabeti inventati (Klingon, Elfico) con grammatiche e vocabolari complessi.
- Proposta di inserimento dell'alfabeto Klingon (1997), rifiutata (2001).
- Creazione della Private Use Area (PUA).
- ConScript Unicode Registry: catalogo non ufficiale di assegnazioni PUA.

Principi di Unicode

1. **Repertorio universale**: tutti i caratteri di tutti gli alfabeti.
2. **Efficienza**: minimo uso di memoria, massima velocità di parsing.
3. **Caratteri, non glifi**: i font sono esclusi.
4. **Semantica**: significato preciso per ogni carattere.
5. **Testo semplice**: solo caratteri di testo semplice (no bold, ecc.).
6. **Ordine logico**: ordine alfabetico naturale.
7. **Unificazione**: caratteri comuni unificati in un singolo codice.
8. **Composizione dinamica**: caratteri composti da frammenti indipendenti.
9. **Stabilità**: codici immutabili una volta assegnati.
10. **Convertibilità**: facile conversione tra Unicode e altre codifiche.

UCS-2 e UCS-4

- UCS-2: schema a due byte (estensione di ISO Latin 1).
- UCS-4: schema a 31 bit in 4 byte (estensione di UCS-2).
  - Divisione in gruppi, piani, righe, celle.
- Piano 0 (BMP - Basic Multilingual Plane): equivalente a UCS-2 (alfabeti moderni).
- Altri piani: alfabeti antichi (SMP), caratteri ideografici CJK (SIP), caratteri cinesi antichi (TIP), caratteri tag (SSP, in disuso), Private Use Areas (piani 15 e 16).

Da UCS a UTF

- UTF (Unicode Transformation Format): sistema a lunghezza variabile.
- Accesso a tutti i caratteri di UCS in modo efficiente.
- Utilizzo in base all'alfabeto e al documento.

UTF-8

- Accesso a tutti i caratteri di UCS-4.
- Da 1 a 4 byte per carattere.
  - 1 byte: codici ASCII (0-127, primo bit 0).
  - 2 byte: alfabeti non ideografici.
  - 3 byte: alfabeti ideografici.
  - 4 byte: piani alti.
- Regole di codifica basate sui primi bit del byte:
  - 0xxxxxxx: carattere ASCII.
  - 110yyyyx: carattere non ideografico.
  - 1110zzzz: carattere ideografico.
  - 11110uuu: carattere non BMP ma già noto (coppie di surrogati in UTF-16).
- Byte di continuazione: 10xxxxxx.
- Lunghezza del carattere indicata dal numero di 1 nel primo byte.

Little-Endian, Big-Endian

- Differenza nell'ordine dei byte in processori diversi.
  - Big-endian: byte più significativo prima (Motorola, IBM, RISC).
  - Little-endian: byte meno significativo prima (Intel, DEC, CISC).
- Problemi nell'interpretazione di flussi di byte.

Byte Order Mark (BOM)

- Codice FFFE (Unicode) come segnalatore di ordinamento.
- FEFF: Zero-Width No-Break Space (ZWNBSP).
- FFFE: carattere proibito in Unicode.
- Uso di ZWNBSP all'inizio di flussi UTF-16 e UCS-2.
  - FEFF: sistema sorgente big-endian.
  - FFFE: sistema sorgente little-endian (riconversione).
- BOM anche in flussi UTF-8 per trasparenza.

Differenze tra UTF-8 e ISO Latin-1

- Identici per caratteri ASCII.
- Differenze per caratteri con decorazioni (accenti, ecc.).
  - UTF-8: 2 byte.
  - ISO Latin 1: 1 byte.
- Errore comune: interpretazione errata di UTF-8 come ISO Latin-1 ("Ã©" invece di "é").

Conclusioni

- Set di caratteri:
  - Lunghezza fissa: 7, 8 bit (ASCII, EBCDIC, ISO Latin 1).
  - Lunghezza fissa: 16, 31 bit (UCS-2, UCS-4).
  - Lunghezza variabile: 1-4 * 8 bit (UTF-8, UTF-16).