

# Hur ser det ut idag?

- **ISO 8859-1 (ISO Latin 1)**
- **EBCDIC 278 (EBCDIC Finland/Sverige motsvarande Latin 1).**
- **Problem att lagra och hantera texter på språk som inte omfattas av dessa.**

# Vad är UNICODE?

- **Provides a unique number for every character**
  - **No matter what the platform**
  - **No matter what the program**
  - **No matter what the language**
- **Unicode standarden är en universell teckenkodnings standard som används för representation av text för datorprocesser.**
- **Unicode standarden är nära kopplad till den internationella ISO/IEC 10646 standarden, även kallad UCS (Universal Character Set).**
- **Majoriteten av alla språks vanligaste tecken ryms inom "Basic Multilingual Plane" (BMP) dvs. inom 2 bytes.**

# UNICODE varianter

- **UNICODE varianter med förkortningen UTF står för Unicode (alt. UCS) Transformation Format.**
  - UTF-8 är ett variabelt format baserat på en byte.
  - UTF-16 är ett variabelt format baserat på två bytes.  
UCS-2 som Java använder (V1.4), skiljer sig från UTF-16 genom att man inte hanterar "supplementary characters". Båda har dock samma 16-bitars teckenkodning. Beror på Unicode 2.0 som expanderade code point range från 64k till 1,1 M.
  - UTF32 är ett fast format baserat på 4 bytes.
  - Exempel på UTF-8:  
Åsa Åström blir X" C385 73 61 C385 73 74 72 C3B6 6D"

<i>Bits</i>	<i>Standard</i>	<i>Binary</i>	<i>Hex</i>	<i>Dec</i>	<i>Char</i>
7	ASCII	1000001	41	65	A
8	8859-1	01000001	41	65	A
16	UTF-16, UCS-2	00000000 01000001	41	65	A
32	UTF-32, UCS-4	00000000 00000000 00000000 01000001	41	65	A

# UNICODE i DB2, Java och COBOL

- **Vilka format använder DB2?**
  - SBCS CCSID 367 ASCII subset
  - Mixed dvs. varchar/char fält CCSID 1208 UTF-8
  - DBCS Graphic fält CCSID 1200 UTF-16
  - Internt och i Catalogen används UTF-8.
- **Java använder i V1.4 UCS-2**
- **Cobol Pic n (National) använder UTF-16.**
- **Cobol Pic x kan hantera UTF-8 vid ENCODING=UNICODE. Tillagt efter presentationstillfället.**

# Var börjar man?

- **Viktigt att sätta sig in i hela sammanhanget.**
- **Försök hitta lösning som står i proportion till behovet.**
- **Vad har verksamheten för strategi (allt är inte bara teknik).**
- **Unicode krävs inte i alla sammanhang. Grundregeln är dock att undvika destruktiv konvertering.**

## Några angreppssätt

- Börja med UNICODE i "tunna" tjänster där konvertering hanteras av DB2.
- Det kan vara svårt att ändra i befintliga komplicerade tjänster ("lager på lager") som körs i ramverksstrukturer.
- Viktigt att sätta sig in i hur "Legacy" systemens beteende kan påverkas.
- Språkneutral stavning som fungerar för dagens ASCII/EBCDIC kan lösa en del baksidor.