UNICODE™ 5.0
Dr. Bruce K. Haddon

Character Encodings

- Morse Code
- Baudot Code
- Hollerith
- ASCII
- EBCDIC
- etc.
ASCII (ANSI-X3.4)

- ANSI-X3.4-1986 (R1997); ISO-14962-1997
- 7-bit code
- Purpose: information interchange
- Popular choice for programming languages (e.g., C/C++, Pascal, Ada, Java/C#, etc.)
- Became the *de facto* code set and encoding for (too?) many applications

ASCII—The Code

<p>| | | | | | | | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
</tr>
<tr>
<td>0</td>
<td>NUL</td>
<td>SOH</td>
<td>STX</td>
<td>ETX</td>
<td>EOT</td>
<td>ENQ</td>
<td>ACK</td>
<td>BEL</td>
<td>BS</td>
<td>HT</td>
<td>LF</td>
<td>VT</td>
<td>FF</td>
<td>CR</td>
<td>SO</td>
</tr>
<tr>
<td>1</td>
<td>DLE</td>
<td>DC1</td>
<td>DC2</td>
<td>DC3</td>
<td>DC4</td>
<td>NAK</td>
<td>SYN</td>
<td>ETB</td>
<td>CAN</td>
<td>EM</td>
<td>SUB</td>
<td>ESC</td>
<td>FS</td>
<td>GS</td>
<td>RS</td>
</tr>
<tr>
<td>2</td>
<td>SP</td>
<td>&quot;</td>
<td>$</td>
<td>%</td>
<td>&amp;</td>
<td>'</td>
<td>(</td>
<td>)</td>
<td>*</td>
<td>+</td>
<td>,</td>
<td>-</td>
<td>.</td>
<td>/</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>:</td>
<td>;</td>
<td>&lt;</td>
<td>=</td>
<td>&gt;</td>
</tr>
<tr>
<td>4</td>
<td>@</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
<td>G</td>
<td>H</td>
<td>I</td>
<td>J</td>
<td>K</td>
<td>L</td>
<td>M</td>
<td>N</td>
</tr>
<tr>
<td>5</td>
<td>P</td>
<td>Q</td>
<td>R</td>
<td>S</td>
<td>T</td>
<td>U</td>
<td>V</td>
<td>W</td>
<td>X</td>
<td>Y</td>
<td>Z</td>
<td>[</td>
<td>\</td>
<td>]</td>
<td>^</td>
</tr>
<tr>
<td>6</td>
<td>`</td>
<td>a</td>
<td>b</td>
<td>c</td>
<td>d</td>
<td>e</td>
<td>f</td>
<td>g</td>
<td>h</td>
<td>i</td>
<td>j</td>
<td>k</td>
<td>l</td>
<td>m</td>
<td>n</td>
</tr>
<tr>
<td>7</td>
<td>p</td>
<td>q</td>
<td>r</td>
<td>s</td>
<td>t</td>
<td>u</td>
<td>v</td>
<td>w</td>
<td>x</td>
<td>y</td>
<td>z</td>
<td>{</td>
<td></td>
<td></td>
<td>~</td>
</tr>
</tbody>
</table>
ISO 646

First version of “ASCII” by the International Standards Organization (with “National” variants)

7-bit codes Currently 25 National variants

(changes certain characters, e.g., 5B₁₆ “[” in ASCII is “Æ” in 646-DK)

Largely obsolete

ISO 646—Basis Code

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>NUL</td>
<td>SOH</td>
<td>STX</td>
<td>ETX</td>
<td>EOT</td>
<td>ENQ</td>
<td>ACK</td>
<td>BEL</td>
<td>BS</td>
<td>HT</td>
<td>LF</td>
<td>VT</td>
<td>FF</td>
<td>CR</td>
<td>SO</td>
</tr>
<tr>
<td>1</td>
<td>DLE</td>
<td>DC1</td>
<td>DC2</td>
<td>DC3</td>
<td>DC4</td>
<td>NAK</td>
<td>SYN</td>
<td>ETB</td>
<td>CAN</td>
<td>EM</td>
<td>SUB</td>
<td>ESC</td>
<td>FS</td>
<td>GS</td>
<td>RS</td>
</tr>
<tr>
<td>2</td>
<td>SP</td>
<td>!</td>
<td>&quot;</td>
<td>#</td>
<td>$</td>
<td>%</td>
<td>&amp;</td>
<td>‘</td>
<td>(</td>
<td>)</td>
<td>*</td>
<td>+</td>
<td>,</td>
<td>-</td>
<td>.</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>:</td>
<td>;</td>
<td>&lt;</td>
<td>=</td>
<td>&gt;</td>
</tr>
<tr>
<td>4</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
<td>G</td>
<td>H</td>
<td>I</td>
<td>J</td>
<td>K</td>
<td>L</td>
<td>M</td>
<td>N</td>
<td>O</td>
</tr>
<tr>
<td>5</td>
<td>P</td>
<td>Q</td>
<td>R</td>
<td>S</td>
<td>T</td>
<td>U</td>
<td>V</td>
<td>W</td>
<td>X</td>
<td>Y</td>
<td>Z</td>
<td>[</td>
<td>\</td>
<td>]</td>
<td>^</td>
</tr>
<tr>
<td>6</td>
<td>a</td>
<td>b</td>
<td>c</td>
<td>d</td>
<td>e</td>
<td>f</td>
<td>g</td>
<td>h</td>
<td>i</td>
<td>j</td>
<td>k</td>
<td>l</td>
<td>m</td>
<td>n</td>
<td>o</td>
</tr>
<tr>
<td>7</td>
<td>p</td>
<td>q</td>
<td>r</td>
<td>s</td>
<td>t</td>
<td>u</td>
<td>v</td>
<td>w</td>
<td>x</td>
<td>y</td>
<td>z</td>
<td>{</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
ISO 646—UK variant

ISO 646—Swedish/Finnish Variant
**C Programming in ISO 646**

æa=xÆlÆø'Ø02';â

äa=xÄlÄö'Ö02';â

??<a=x??(1??) ??!'??/02';??>

{a=x[1] | '\02';}

- Danish
- Swedish / Finnish
- C Standard trigraphs
- What was really meant

---

**ISO/IEC 8859**

- 8 bit codes
- Currently, 16 variants (called “Parts”)
- 7-bit subset of each ≡ ASCII (exactly)
- Each 8859 variant (Part) redefines the code points from 80₁₆-FF₁₆
- e.g., ISO/IEC 8859-1 is “Latin-1”, ISO/IEC 8859-5 is “Latin/Cyrillic”
  ISO/IEC 8859-9 is “Latin-5”
  ISO/IEC 8859-11 is “Latin/Thai5”
  ISO/IEC 8859-15 is “Latin-9” (or “Latin-0”)

---

Copyright © 2000 - 2007 Paladin Software Incorporated, Inc.
### ISO/IEC 8859-1—The “Latin-1” Code

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
</tr>
<tr>
<td>8</td>
<td>PAD</td>
<td>HOP</td>
<td>BPH</td>
<td>NBH</td>
<td>IND</td>
<td>NEL</td>
<td>SSA</td>
<td>ESA</td>
<td>HTS</td>
<td>HTJ</td>
<td>VTS</td>
<td>PUD</td>
<td>PIU</td>
<td>RI</td>
</tr>
<tr>
<td>9</td>
<td>DCS</td>
<td>PU1</td>
<td>PU2</td>
<td>STS</td>
<td>COH</td>
<td>MW</td>
<td>SPA</td>
<td>EPA</td>
<td>SOS</td>
<td>SCI</td>
<td>SCI</td>
<td>ST</td>
<td>DSC</td>
<td>PM</td>
</tr>
<tr>
<td>A</td>
<td>NSGP</td>
<td>i</td>
<td>t</td>
<td>l</td>
<td>e</td>
<td>y</td>
<td>!</td>
<td>£</td>
<td>$</td>
<td>©</td>
<td>a</td>
<td>≈</td>
<td>¬</td>
<td>$</td>
</tr>
<tr>
<td>B</td>
<td>o</td>
<td>±</td>
<td>³</td>
<td>²</td>
<td>µ</td>
<td>t</td>
<td>³</td>
<td>°</td>
<td>»</td>
<td>½</td>
<td>¼</td>
<td>¾</td>
<td>¿</td>
<td>Š</td>
</tr>
<tr>
<td>C</td>
<td>á</td>
<td>á</td>
<td>á</td>
<td>å</td>
<td>å</td>
<td>å</td>
<td>Å</td>
<td>Æ</td>
<td>Ç</td>
<td>È</td>
<td>É</td>
<td>Ê</td>
<td>Ë</td>
<td>Ì</td>
</tr>
<tr>
<td>D</td>
<td>Ð</td>
<td>Ñ</td>
<td>Ò</td>
<td>Ó</td>
<td>Ô</td>
<td>Õ</td>
<td>Ö</td>
<td>×</td>
<td>Ø</td>
<td>Ù</td>
<td>Ú</td>
<td>Û</td>
<td>Ü</td>
<td>Ý</td>
</tr>
<tr>
<td>E</td>
<td>à</td>
<td>à</td>
<td>à</td>
<td>ã</td>
<td>ã</td>
<td>ã</td>
<td>Ä</td>
<td>Å</td>
<td>Æ</td>
<td>Ñ</td>
<td>Ñ</td>
<td>Ñ</td>
<td>Ñ</td>
<td>Ñ</td>
</tr>
<tr>
<td>F</td>
<td>ð</td>
<td>ñ</td>
<td>ò</td>
<td>ó</td>
<td>ô</td>
<td>õ</td>
<td>ö</td>
<td>÷</td>
<td>ø</td>
<td>ù</td>
<td>ú</td>
<td>û</td>
<td>ü</td>
<td>ý</td>
</tr>
</tbody>
</table>

### ISO/IEC 8859-15—The “Latin-9” Code

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
</tr>
<tr>
<td>8</td>
<td>XXX</td>
<td>XXX</td>
<td>BHP</td>
<td>NBH</td>
<td>IND</td>
<td>NEL</td>
<td>SSA</td>
<td>ESA</td>
<td>HTS</td>
<td>HTJ</td>
<td>VTS</td>
<td>PLO</td>
<td>PIU</td>
<td>RI</td>
<td>SS2</td>
</tr>
<tr>
<td>9</td>
<td>DCS</td>
<td>PU1</td>
<td>PU2</td>
<td>STS</td>
<td>COH</td>
<td>MW</td>
<td>SPA</td>
<td>EPA</td>
<td>SOS</td>
<td>XXX</td>
<td>SCI</td>
<td>SCI</td>
<td>ST</td>
<td>DSC</td>
<td>PM</td>
</tr>
<tr>
<td>A</td>
<td>NSGP</td>
<td>i</td>
<td>t</td>
<td>l</td>
<td>x</td>
<td>y</td>
<td>¥</td>
<td>$</td>
<td>©</td>
<td>a</td>
<td>≈</td>
<td>¬</td>
<td>$</td>
<td>¬</td>
<td>¬</td>
</tr>
<tr>
<td>B</td>
<td>o</td>
<td>±</td>
<td>³</td>
<td>²</td>
<td>µ</td>
<td>t</td>
<td>³</td>
<td>°</td>
<td>»</td>
<td>½</td>
<td>¼</td>
<td>¾</td>
<td>¿</td>
<td>Š</td>
<td>Š</td>
</tr>
<tr>
<td>C</td>
<td>á</td>
<td>á</td>
<td>á</td>
<td>å</td>
<td>å</td>
<td>å</td>
<td>Å</td>
<td>Æ</td>
<td>Ç</td>
<td>È</td>
<td>É</td>
<td>Ê</td>
<td>Ë</td>
<td>Ì</td>
<td>Í</td>
</tr>
<tr>
<td>D</td>
<td>Ð</td>
<td>Ñ</td>
<td>Ò</td>
<td>Ó</td>
<td>Ô</td>
<td>Õ</td>
<td>Ö</td>
<td>×</td>
<td>Ø</td>
<td>Ù</td>
<td>Ú</td>
<td>Û</td>
<td>Ü</td>
<td>Ý</td>
<td>Þ</td>
</tr>
<tr>
<td>E</td>
<td>à</td>
<td>à</td>
<td>à</td>
<td>ã</td>
<td>ã</td>
<td>ã</td>
<td>Ä</td>
<td>Å</td>
<td>Æ</td>
<td>Ñ</td>
<td>Ñ</td>
<td>Ñ</td>
<td>Ñ</td>
<td>Ñ</td>
<td>Ñ</td>
</tr>
<tr>
<td>F</td>
<td>ð</td>
<td>ñ</td>
<td>ò</td>
<td>ó</td>
<td>ô</td>
<td>õ</td>
<td>ö</td>
<td>÷</td>
<td>ø</td>
<td>ù</td>
<td>ú</td>
<td>û</td>
<td>ü</td>
<td>ý</td>
<td>þ</td>
</tr>
</tbody>
</table>
Other “National” and ISO Standards

- Japanese Industrial Standards
  - Series of encodings (>15), all including ASCII and “wide character” ASCII
- Big 5, GB, GB/T, CNS (Chinese)
- KS (Korean)
- TCVN (Vietnamese)
- UNIX Extended Code (UEC)
  - Escaping convention to allow intermixing of ASCII and any of the above (Open Consortium, OSF, UI, USLP: 1991)

Shift-JIS

- ASCII
  - 21-7E16
- half-width katakana
  - A1-DF16
- JIS X 0208:1977
  - 1st byte 81-9F16, E0-EF16
  - 2nd byte 40-7E16, 80-FC16

EUC-JP

- ASCII or JIS-Roman
  - 21-7E16
- half-width katakana
  - 8E16 followed by A1-DF16
- JIS X 0208:1977
  - 1st byte 81-9F16, E0-EF16
  - 2nd byte 40-7E16, 80-FC16
- JIS X 0212:1990
  - 8F16 followed by:
    - 2nd byte A1-FE16
    - 3rd byte A1-FE16
Terminology (1)

- “Character Set”
- “Glyph”
- “(Natural) Encoding”
  - “Code page/set”
- “Code point”
- “Transcoding”
- “Transformation”

Terminology (2)

- “Single byte, simple”
- “Double byte (simple)”
  - “Multi-byte (simple)”
- “Single byte, complex”
- “Bi-Directional” (‘bi-di’)
- “Universal”
ISO/IEC-10646

- “Universal” character set—each code point is 32 bits, “0” + 31 bits (UCS-4)
- Initial approach, use “planes,” each containing defined national subsets
- 15 bits for “plane” number, 16 bits define character encoding within plane

The Unicode Standard

- Consortium of, now, 13 “full” members, 3 “institutional” members, 2 “supporting” member, 32 “associate” members, and a long list of individual and liaison members
- Interoperability with ISO 8859-1 Latin-1 (including ASCII)
- Encompassing all scripts in use—now, all scripts ever used (or shall be used!)
The Unicode Consortium®

Full Members
- Adobe Systems, Inc.
- Apple Computer, Inc.
- DENIC eG
- Google
- Hewlett-Packard Company
- IBM Corporation
- Justsystem Corporation
- Microsoft Corporation
- Oracle Corporation
- SAP AG
- Sun Microsystems, Inc.
- Sybase, Inc.
- Yahoo

Institutional Members
- Government of India
- Government of Pakistan
- University of California at Berkeley

Supporting Members
- Basis Technology Corporation
- Monotype Imaging

Associate Members
- AOL Online
- Beijing Founder Electronics
- Beijing Zhong Yi Electronics
- La Bibliothèque universitaire des langues et civilisations
- Booz, Allen & Hamilton
- The Church of Jesus Christ of Latter-day Saints
- Columbia University
- DecoType, Inc.
- DigiCert SSL Certificate Authority
- EdgeNet, Inc.
- EmuraSoft, Inc.
- Evertype
- Ex Libris
- Fidelity National Information Services, Inc.
- Innovative Interfaces, Inc.
- The Library Corporation
- Linotype GmbH
- NCR Corporation
- Nokia
- OCLC, Inc.
- The perl Foundation
- SAS Institute, Inc.
- SIL International
- SIRSI Corporation
- Sony Ericsson
- Sybase, Inc.
- United Bible Societies
- Utilika Foundation
- Verisign, Inc.
- Vernacular Information Society Project
- VTLS, Inc.

Plus, Individual and Liaison Members

Copyright © 2000 - 2007 Paladin Software Incorporated, Inc.
History of Unicode Standard

- Unicode 5.0 (November, 2006)
- Unicode 4.1.0 (March, 2005)
- Unicode 4.0.1 (March, 2004)
- Unicode 4.0 (March, 2003)
- Unicode 3.2.0 (March, 2002)
- Unicode 3.1.1 (August, 2001)
- Unicode 3.1.0 (March, 2001)
- Unicode 3.0.1 (August, 2000)
- Unicode 3.0 (September, 1999)
- Unicode 2.0 (July, 1996)
- Unicode 1.0 (October, 1991)

The Unicode Standards.
- Version 5.0, 2006
- Version 2.0, 1996.

Addison-Wesley Developers Press, Reading, MA.

Unicode Design Principles

- Universality
  - A single, universal, repertoire for all human (and some non-human) writing—see next slide
- Efficiency
  - Easy to parse and process
  - A compact representation that fits into an average of no more than sixteen bits.
- Characters, not glyphs
  - Encode each abstract character once
- Semantics
  - Well-defined character semantics
- Plain text
  - Characters represent plain text
- Logical Order
  - Storage default is logical order, not printed order
- Unification
  - Han, and other, unification, e.g., CJKV conceptually same ideograms unified
- Dynamic Composition
  - Accented forms may be composed
- Stability
  - Characters once assigned cannot be reassigned
- Convertibility
  - Round trip preservation, hence many a’s, alpha, aleph, etc.
  - Compatibility with “wide” characters, Arabic contextual forms, ligatures, etc.
Natural Encoding: UTF-32
(subset of UCS-4: since Version 2)

0 – 10FFFF\textsubscript{16}

Unicode Character Set

- Arabic
- Armenian
- Balinese
- Basic Latin
- Bengali
- Buginese
- Buhid
- Cherokee
- Coptic
- Cuneiform
- Cypriot
- Cyrillic
- Deseret
- Devanagari
- Ethiopic
- Etruscan
- IPA Numbers (Decimal, Counting Rods, Cuneiform)
- General Diacritics
- General Punctuation
- General Symbols
- Mathematical Symbols
- Technical Symbols
- Tone Symbols
- Dingbats
- Arrows, Blocks, Box Drawing Forms, and Geometric Shapes
- Miscellaneous Symbols
- Presentation Forms
- Braille Patterns
- Musical Symbols (Western, Byzantine, & Ancient Greek)
Efficient Encoding: UTF-16

"Surrogates" D800–DFFF 2048 values

p-1: 6 bits

1,048,576 code points

Count of Unicode Characters
The Groups — BMP (approximate)

Interoperable Encoding: UTF-8
Unicode Transformation Formats

Summary

- **UTF-32**
  - Is a subset of UCS-4, i.e., 0-10FFFF16
  - Is the natural representation of Unicode in 32-bit units

- **UTF-16**
  - Transforms UTF-32 into a stream of 16-bit units
  - Is the standard representation of Unicode in 16-bit units (i.e., with surrogates)

- **UTF-8**
  - Transforms UCS-4 (hence UTF-32) into a stream of 8-bit units
  - Interoperates with ASCII

Compatibility—“round tripping”

- ASCII is in twice (21-7E16, FF01-FF5E16)
- 29 sets of decimal digits, 0-9
- 18 space characters (not counting tabs, etc.)
- 18 hyphen or dash characters
- Composed and decomposed characters
Han Unification

<table>
<thead>
<tr>
<th>Unicode</th>
<th>China</th>
<th>Taiwan</th>
<th>Japan</th>
<th>Korea</th>
</tr>
</thead>
<tbody>
<tr>
<td>U+4E00</td>
<td>一</td>
<td>一</td>
<td>一</td>
<td>一</td>
</tr>
<tr>
<td>U+4E0E</td>
<td>与</td>
<td>与</td>
<td>与</td>
<td>与</td>
</tr>
<tr>
<td>U+5224</td>
<td>判</td>
<td>判</td>
<td>判</td>
<td>判</td>
</tr>
<tr>
<td>U+5668</td>
<td>器</td>
<td>器</td>
<td>器</td>
<td>器</td>
</tr>
<tr>
<td>U+5B57</td>
<td>字</td>
<td>字</td>
<td>字</td>
<td>字</td>
</tr>
<tr>
<td>U+6D77</td>
<td>海</td>
<td>海</td>
<td>海</td>
<td>海</td>
</tr>
<tr>
<td>U+9038</td>
<td>逸</td>
<td>逸</td>
<td>逸</td>
<td>逸</td>
</tr>
<tr>
<td>U+9AA8</td>
<td>骨</td>
<td>骨</td>
<td>骨</td>
<td>骨</td>
</tr>
</tbody>
</table>

Unicode Characteristics

- Character name
- General Category
- Canonical Combining Classes
- Bi-directional Category
- Character Decomposition Mapping
- Decimal digit value
- Digit value
- Numeric value
- Mirrored
- Unicode 1.0 Name
- 10646 comment field
- Upper case Mapping
- Lower case Mapping
- Title case Mapping

UnicodeData.txt
Example (1) of UnicodeData.txt

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>Category</th>
<th>Widths</th>
<th>Unicode Versions</th>
</tr>
</thead>
<tbody>
<tr>
<td>0C66</td>
<td>TELUGU DIGIT ZERO</td>
<td>Nd</td>
<td>0, 0, 0</td>
<td>N</td>
</tr>
<tr>
<td>0C67</td>
<td>TELUGU DIGIT ONE</td>
<td>Nd</td>
<td>1, 1, 1</td>
<td>N</td>
</tr>
<tr>
<td>0C68</td>
<td>TELUGU DIGIT TWO</td>
<td>Nd</td>
<td>2, 2, 2</td>
<td>N</td>
</tr>
<tr>
<td>0C69</td>
<td>TELUGU DIGIT THREE</td>
<td>Nd</td>
<td>3, 3, 3</td>
<td>N</td>
</tr>
<tr>
<td>0C6A</td>
<td>TELUGU DIGIT FOUR</td>
<td>Nd</td>
<td>4, 4, 4</td>
<td>N</td>
</tr>
<tr>
<td>0C6B</td>
<td>TELUGU DIGIT FIVE</td>
<td>Nd</td>
<td>5, 5, 5</td>
<td>N</td>
</tr>
<tr>
<td>0C6C</td>
<td>TELUGU DIGIT SIX</td>
<td>Nd</td>
<td>6, 6, 6</td>
<td>N</td>
</tr>
<tr>
<td>0C6D</td>
<td>TELUGU DIGIT SEVEN</td>
<td>Nd</td>
<td>7, 7, 7</td>
<td>N</td>
</tr>
<tr>
<td>0C6E</td>
<td>TELUGU DIGIT EIGHT</td>
<td>Nd</td>
<td>8, 8, 8</td>
<td>N</td>
</tr>
<tr>
<td>0C6F</td>
<td>TELUGU DIGIT NINE</td>
<td>Nd</td>
<td>9, 9, 9</td>
<td>N</td>
</tr>
</tbody>
</table>

Example (2) of UnicodeData.txt

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>Category</th>
<th>Widths</th>
<th>Unicode Versions</th>
</tr>
</thead>
<tbody>
<tr>
<td>0024</td>
<td>DOLLAR SIGN</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>00A2</td>
<td>CENT SIGN</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>00A3</td>
<td>POUND SIGN</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>00A4</td>
<td>CURRENCY SIGN</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>00A5</td>
<td>YEN SIGN</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>20A0</td>
<td>EURO-CURRENCY</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>201A</td>
<td>COLON</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>20A2</td>
<td>CRUZEIRO SIGN</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>20A3</td>
<td>FRENCH FRANC</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>20A4</td>
<td>LIRA SIGN</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>20A5</td>
<td>LI LL SIGN</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>20A6</td>
<td>NAIRA SIGN</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>20A7</td>
<td>PESETA SIGN</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>20AC</td>
<td>EURO SI SIGN</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>20AD</td>
<td>KI PROUSSA</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>20AE</td>
<td>TUGRII SIGN</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>20AF</td>
<td>DRACHMA SI SIGN</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>20B0</td>
<td>GERMAN PENNY</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>20B1</td>
<td>PESO SI SIGN</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>20B2</td>
<td>ALI ALI</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>FEE9</td>
<td>SMALL DOLLAR</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>FF04</td>
<td>FULLWI DTH DOLLAR</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>FFE0</td>
<td>FULLWI DTH CENT</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>FFE1</td>
<td>FULLWI DTH POUND</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
<tr>
<td>FFF5</td>
<td>FULLWI DTH YEN</td>
<td>Sc</td>
<td>0, 0</td>
<td>N</td>
</tr>
</tbody>
</table>
Composed and Decomposed Characters

- Composed:
  
  e.g., å è î ô ü (â ≡ U+00E5)

- Decomposed:
  
  e.g., a° (≡ U+0061 U+030A)

- Multiple accents:
  
  e.g., â (≡ U+00E5 U+0334)
  or U+0061 U+030A U+0334)

Example (3) of UnicodeData.txt

00E4;LATIN SMALL LETTER A WITH DIAERESIS;Ll;0;L;0061
0308;;;;;N;LATIN SMALL LETTER A
DIAERESIS;;00C4;;00C4

00E5;LATIN SMALL LETTER A WITH RING
ABOVE;Ll;0;L;0061 030A;;;;;N;LATIN SMALL LETTER A
RING;;00C5;;00C5

00E6;LATIN SMALL LETTER AE;Ll;0;L;;;;;N;LATIN SMALL
LETTER A E;ash *;00C6;;00C6

00E7;LATIN SMALL LETTER C WITH CEDILLA;Ll;0;L;0063
0327;;;;;N;LATIN SMALL LETTER C CEDILLA;;00C7;;00C7

00E8;LATIN SMALL LETTER E WITH GRAVE;Ll;0;L;0065
0300;;;;;N;LATIN SMALL LETTER E GRAVE;;00C8;;00C8
Unicode Publications

- **US**: Unicode Standard
  - The standard, *i.e.*, the book, with the character set, code points, and conformance requirements

- **UAX**: Unicode Standard Annexes
  - Subsections of the standard, included in the Standard, containing explanatory details.

- **UTS**: Unicode Technical Standard (electronic only)
  - Associated standards, such as compression, collation, XML usage, *etc.*

- **UTR**: Unicode Technical Report (electronic only)
  - Other informative material, *e.g.*, the encoding model, property model, mathematical support, security, *etc.*

Comparison and Normalization (UAX #15)

What does it mean to ask:

"When are two (Unicode) strings **equal?**"

- The bytes are identical
- The characters are the same ignoring differences in the ways accents are combined
- The characters are the same ignoring compatibility differences
- What if there are multiple accents on a character?
- Is “decomposed characters” the best way to represent accented characters?

YES! BUT INADEQUATE
Normalization (UAX #15)

**NFD  Canonical Decomposition**

All accented characters are “canonically decomposed,” as well as certain Han decompositions. Accents are placed in “canonical order.”

**NFC  Canonical Composition**

Canonical decomposition is followed by recomposition, i.e., for a given list of composed characters, base characters followed by an accent are recomposed.

**NFKD  Compatibility Decomposition**

Characters are canonically decomposed, and concurrently, all compatibility characters are replaced by their “non-compatible” forms.

**NFKC  Compatibility Composition**

Characters are compatibility decomposed, and then canonically recomposed.

Examples: Combining / Compatibility Forms

- Singletons never combine
- Combining accents come after the character
- Compatible forms arise for a variety of reasons

<table>
<thead>
<tr>
<th>Combining sequence</th>
<th>Ç</th>
<th>←</th>
<th>Ç</th>
<th>Ọ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hangul</td>
<td>ṅ</td>
<td>←</td>
<td>ṛ</td>
<td>ḥ</td>
</tr>
<tr>
<td>Singleton</td>
<td>Ω</td>
<td>←</td>
<td>Ω</td>
<td>dz</td>
</tr>
</tbody>
</table>

Font variants

Breaking differences

Cursive forms

Circled

Width, size, rotated

Superscripts/subscripts

Squared characters

Fractions

Others
Example: Composition / Compatibility

In the first example, decomposition separates the accent, but composition puts it back together again.

In the second, the ångstrom character and the ohm character are replaced by the letter a and the letter omega.

Example: Canonical Composition

<table>
<thead>
<tr>
<th>Source</th>
<th>NFD</th>
<th>NFC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Š</td>
<td>0073 0323</td>
<td>0307 1E69</td>
</tr>
<tr>
<td>Ń</td>
<td>0071 0307</td>
<td>0323 0307</td>
</tr>
<tr>
<td>ķ</td>
<td>0071 0323</td>
<td>0307 0307</td>
</tr>
</tbody>
</table>
Example: the (Normalization) works!

<table>
<thead>
<tr>
<th>Source</th>
<th>NFD</th>
<th>NFC</th>
<th>NFKD</th>
<th>NFKC</th>
</tr>
</thead>
<tbody>
<tr>
<td>fi</td>
<td>fi</td>
<td>fi</td>
<td>f i</td>
<td>f i</td>
</tr>
<tr>
<td>FB01</td>
<td>FB01</td>
<td>FB01</td>
<td>0066</td>
<td>0066</td>
</tr>
<tr>
<td>2⁵</td>
<td>2⁵</td>
<td>2⁵</td>
<td>2⁵</td>
<td>2⁵</td>
</tr>
<tr>
<td>0032 2075</td>
<td>0032 2075</td>
<td>0032 2075</td>
<td>0032 2075</td>
<td>0032 2075</td>
</tr>
<tr>
<td>ʃ̃</td>
<td>ʃ̃</td>
<td>ʃ̃</td>
<td>ʃ̃</td>
<td>ʃ̃</td>
</tr>
<tr>
<td>1E9B 0323</td>
<td>017F 0323 0307</td>
<td>1E9B 0323</td>
<td>0073 0323 0307</td>
<td>1E9B</td>
</tr>
</tbody>
</table>

Programming Language Identifiers (UAX #31)
Unicode Standard Identifier Requirements

R4 Normalized Identifiers
To meet this requirement, an implementation shall specify the Normalization Form and shall provide a precise list of any characters that are excluded from normalization.

R5 Case-Insensitive Identifiers
To meet this requirement, an implementation shall specify either simple or full case folding, and adhere to the Unicode specification for that folding.

Mathematical Characters (UTR #25)

\[ H = \int d\tau (\varepsilon E^2 + \mu H^2) \]

- Unicode does have all these characters
- If a compatibility normalization were applied:

\[ h = \int d\tau (\varepsilon E^2 + \mu H^2) \]
Mathematical Alphabets

<table>
<thead>
<tr>
<th>Math Style</th>
<th>Characters from Basic Set</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>plain (upright, serifed)</td>
<td>Latin, Greek and digits</td>
<td>BMP</td>
</tr>
<tr>
<td>bold</td>
<td>Latin, Greek and digits</td>
<td>Plane 1</td>
</tr>
<tr>
<td>italic</td>
<td>Latin and Greek</td>
<td>Plane 1*</td>
</tr>
<tr>
<td>bold italic</td>
<td>Latin and Greek</td>
<td>Plane 1</td>
</tr>
<tr>
<td>script (calligraphic)</td>
<td>Latin</td>
<td>Plane 1*</td>
</tr>
<tr>
<td>bold script (calligraphic)</td>
<td>Latin</td>
<td>Plane 1</td>
</tr>
<tr>
<td>Fraktur</td>
<td>Latin</td>
<td>Plane 1*</td>
</tr>
<tr>
<td>bold Fraktur</td>
<td>Latin</td>
<td>Plane 1</td>
</tr>
<tr>
<td>double-struck</td>
<td>Latin and digits</td>
<td>Plane 1*</td>
</tr>
<tr>
<td>sans-serif</td>
<td>Latin and digits</td>
<td>Plane 1</td>
</tr>
<tr>
<td>sans-serif bold</td>
<td>Latin, Greek and digits</td>
<td>Plane 1</td>
</tr>
<tr>
<td>sans-serif italic</td>
<td>Latin</td>
<td>Plane 1</td>
</tr>
<tr>
<td>sans-serif bold italic</td>
<td>Latin and Greek</td>
<td>Plane 1</td>
</tr>
<tr>
<td>monospace</td>
<td>Latin and digits</td>
<td>Plane 1</td>
</tr>
</tbody>
</table>

Mathematical italics

- italic a
- italic v (pointed)
- italic v (rounded)
- script X
- plain Y

Some careful distinctions need to be made
Bidirectionality (UAX #9)

- Mid-Eastern texts inherently bi-directional
- The Unicode standard (Unicode Standard Annex #9) specifies an embedding algorithm
  - Direction characteristics (strong, weak, neutral)
  - Directionality overrides
  - Language overrides
- The order of characters in a file follows the “natural" order (no directionality).

Sorting (UTS #10 Collation)

- Sorting by code value does not do the job!
- Unicode specifies five “levels" of collation, applied to NFKD normalization
  - *i.e.*, base, accent, case, punctuation, identity
- Orderings
  - dictionary, language specific
  - telephone directory
  - radical and stroke order, or phonetic, for the Han characters
  - *etc.*
- There are other considerations (UTS#10 is 62 pages long!)
Regular Expressions (UTS #18)

- Ranges specified by:
  - hex codes,
  - Type (digit, letter, separator, etc.)
  - Language block (Latin, Greek, Thai, etc.)
  - Function (SOL, EOL, white space, etc.)

- Other features
  - Level (see Sorting/Collation)
  - Normalized/Un-normalized

Byte Order Marker

- U+FEFF \texttt{ZERO WIDTH NO-BREAK SPACE}
- U+FFE \texttt{not a character code}

Bytes at the beginning of a file:

- U+FFFE \texttt{UTF-16 high byte first}
- U+FE \texttt{UTF-16 low byte first}
- U+BB \texttt{UTF-32 high byte first}
- U+BF \texttt{UTF-32 low byte first}
New Scripts in Version 4.0

- BMP, Plane 0
  - Limbu
  - Tai Le
- Plane 1
  - Shavian
  - Linear B
  - Ugaritic Cuneiform
  - Cypriot syllabary
  - Osmanya
- High Voltage Sign (26A116)
- Rejected for 4.0
- Klingon
- Total 1226 new

4.0 Statistics

- Graphic 96,245
- Format 137
- Control 65
- Private Use 137,468
- Noncharacter 66

Shavian Script

And sit down sensibly on the ground on her left. This dirty dog (cowering Sphinx) is a real Christian. He mobs the temple, he does [at each accusation he gives the neck of Sphinx's tunic a twist]; he goes smashing things mad drunk, he does; he steals the gold vessels, he does; he assaults the priestesses, he does — yah! [He flings Sphinx into the middle of the group of prisoners], You're the sort that makes duty a pleasure, you are. Sphinx (gaping) That's it: strangle me. Kick me. Beat me. Revile me. Our Lord was beaten and reviled. That's my way to heaven. Every martyr goes to heaven, no matter what he's done. That is so, isn't it, brother?

Centurion. Well, if you're going to heaven, I don't want to go there. I wouldn't be seen with you.

Lentulus. Hah! Good! [indicating the kneeling Flora]. Is this one of the turn-the-other-check gentlemen, Centurion? Centurion. Yes, sir. Lucky for you too, sir, if you want to take any liberties with him.
New Scripts in Version 4.1

- BMP, Plane 0
  - New Tai Lue
  - Buginese
  - Glagolitic
  - Coptic
  - Tifinagh
  - Syriac
  - New Tai Lue
  - Plane 1
    - Old Persian
    - Kharoshthi
  - Additions to Arabic, ancient
    Greek, Ethiopic, and Hebrew
- Recommended SPACE 00A016
- Total 1273 new

4.1 Statistics

<table>
<thead>
<tr>
<th>Category</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graphic</td>
<td>97,517</td>
</tr>
<tr>
<td>Format</td>
<td>138</td>
</tr>
<tr>
<td>Control</td>
<td>65</td>
</tr>
<tr>
<td>Private Use</td>
<td>137,468</td>
</tr>
<tr>
<td>Noncharacter</td>
<td>66</td>
</tr>
</tbody>
</table>

New Scripts in Version 5.0

- BMP, Plane 0
  - N’ko
  - Balinese
  - Phags-Pa
  - Plane 1
    - Cuneiform
    - Counting Rods
    - Phoenician
  - Small additions to Latin,
    Greek, Cyrillic, Hebrew,
    Devanagari, Kannada
  - Some symbols
  - Total 1369 new

5.0 Statistics

<table>
<thead>
<tr>
<th>Category</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graphic</td>
<td>98,884</td>
</tr>
<tr>
<td>Format</td>
<td>140</td>
</tr>
<tr>
<td>Control</td>
<td>65</td>
</tr>
<tr>
<td>Private Use</td>
<td>137,468</td>
</tr>
<tr>
<td>Noncharacter</td>
<td>66</td>
</tr>
</tbody>
</table>
Accepted Proposals for New Scripts

Scripts for the Basic Multilingual Plane (BMP)
- Draughts/checkers, mahjong, and dominos symbols
- Avestan (and Pahlavi)
- Batak
- Methei/Manipuri

Scripts for Plane 1
- Basic Egyptian Hieroglyphics
- Brahmi
- Manichaean
- Tengwar (but not Klingon?)

Proposals for New Scripts

Scripts for the Basic Multilingual Plane (BMP)
- Chakma
- Javanese
- Mandarin
- Newari
- Old Hungarian
- Pahaw h Hmong
- Samaritan
- Siddham
- Sorang Sompeng
- Varang Kshiti
- Viet Thai

Scripts for Plane 1
- Ahom
- Balti
- Bassa
- Blissymbols
- Cirth
- Hittite (Anatolian)
- Hieroglyphs/Luvian
- Indus Valley Script
- Kaiti
- Khambti
- Linear A
- Meroitic
- Naxi Geba
- Old Permic
- Palmyrene
- Pollard
- Rongo Rongo
- South Arabian
- Sogimbo

"Help me!" he said. "The baby is sick! Can you come here?"
Unicode Applications

- HTML
- XML
- Windows NT/2000/2003/XP, CE, 95/98/Me, Vista
- Mac 9.2, X
- IBM AIX
- Java & C#
- C/C++ (wchar_t)
- JavaScript
- Browsers (Netscape 4+, IE 5+)
- VB
- Ingres 2.6+
- IBM DB2
- Solaris 8, 9, 10 (UTF-8)
- Perl 5.6 (UTF-8), 5.8
- Oracle 8+ (UTF-8)
- TCL 8.1 (UTF-8)
- Mac 9.0 (UTF-8)
- many others

Unicode in Java Source Code

- Basically, all alpha and all numeric characters from any language may be used in identifiers, plus "$" and "_
- The notation "\uXXXX" may be used anywhere to represent a 16-bit Unicode character
- Identifiers are NOT normalized
- However, the Java source file must be an (7-bit) ASCII file
- String denotations, i.e., "..." are Strings
Unicode in Java programming

- The `char` primitive data type represents a UTF-16 value (may be half a surrogate).
- `CharBuffer`, `String`, `StringBuffer`, `StringBuilder` are classes encapsulating arrays of `char`.
  - All implement the `CharSequence` interface (i.e., UTF-16 representation, including surrogates).
- Code points (i.e., UTF-32) are represented by the 32-bit `int` primitive data type.
  - Methods `CodePointAt()`, and similar, convert elements of `Strings` to (arrays of) code points, and v.v.

Unicode classifications in Java

- The class `Character` encapsulates `char`s, and provides access to Unicode characteristics.
- `Character.Subset` and `Character.UnicodeBlock` describe features of `char`s (e.g., `ARABIC`, `CURRENCY_SYMBOLS`, `BASIC_LATIN`, `CJK_UNIFIED IDEOGRAPHIC`, etc.).
- Similarly, methods `isDigit()`, `isLetter()`, `isHighSurrogate()`, `isJavaIdentifierPart()`, etc.
Java input/output and encodings

- Classes derived from abstract Reader and Writer perform transcodings from (and to, respectfully) other character encodings to (and from) (arrays of) char's
- E.g.,

  ```java
  ... isr = new InputStreamReader(filename, "SJIS");
  ...
  osw = new OutputStreamWriter(filename, "UTF8");
  ```

Summary

- Unicode is more than just another character set, or encoding
- Unicode is “multi-byte, complex”
- Calls into question many of the basic assumptions we make about Western languages
- Is gaining much deeper acceptance and understanding in text applications (but still not fully understood)
Research Areas

- Sort Specification Languages
- Sort implementation techniques
- "Large" font management
- Converting to a 21 bit world
- Normalization libraries (IBM)
- Han refinement
- Archeological research
- Extension of Unicode to further scripts

Further information

- Unicode
  - http://www.unicode.org/
- HTML, XML, the Web
  - http://www.w3.org/TR/unicode-xml/
    Unicode in XML and other Markup Languages (Unicode Technical Report #20 W3C Note 13 June 2003)
  - http://www.w3.org/TR/charmod/
    Character Model for the World Wide Web 1.0 (W3C Working Draft 22 August 2003)
- History
  - http://www.loc.gov/marc/specifications/speccharucs.html
- Support