Chapter 1

Introduction

The Unicode Standard is the universal character encoding scheme for written characters and text. It defines a consistent way of encoding multilingual text that enables the exchange of text data internationally and creates the foundation for global software. As the default encoding of HTML and XML, the Unicode Standard provides a sound underpinning for the World Wide Web and new methods of business in a networked world. Required in new Internet protocols and implemented in all modern operating systems and computer languages such as Java, Unicode is the basis of software that must function all around the world.

With Unicode, the information technology industry gains data stability instead of proliferating character sets; greater global interoperability and data interchange; and simplified software and reduced development costs.

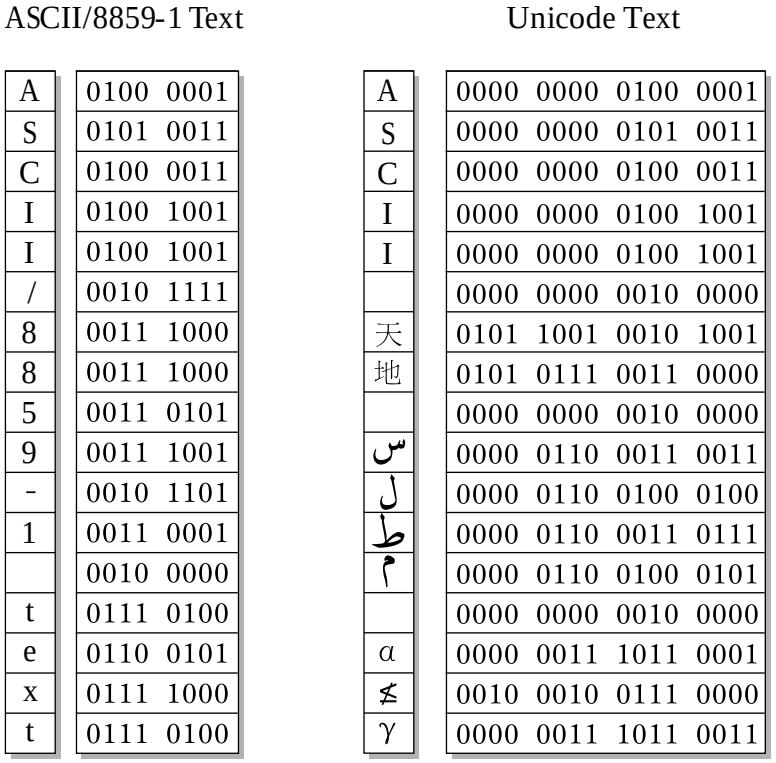
While modeled on the ASCII character set, the Unicode Standard goes far beyond ASCII's limited ability to encode only the upper- and lowercase letters A through Z. It provides the capacity to encode all characters used for the written languages of the world—more than 1 million characters can be encoded. No escape sequence or control code is required to specify any character in any language. The Unicode character encoding treats alphabetic characters, ideographic characters, and symbols equivalently, which means they can be used in any mixture and with equal facility (see *Figure 1-1*).

The Unicode Standard specifies a numeric value and a name for each of its characters. In this respect, it is similar to other character encoding standards from ASCII onward. In addition to character codes and names, other information is crucial to ensure legible text: a character's case, directionality, and alphabetic properties must be well defined. The Unicode Standard defines this and other semantic information, and includes application data such as case mapping tables and mappings to the repertoires of international, national, and industry character sets. The Unicode Consortium provides this additional information to ensure consistency in the implementation and interchange of Unicode data.

Unicode provides for two encoding forms: a default 16-bit form and a byte-oriented form called UTF-8 that has been designed for ease of use with existing ASCII-based systems. *The Unicode Standard, Version 3.0*, is code-for-code identical with International Standard ISO/IEC 10646. Any implementation that is conformant to Unicode is therefore conformant to ISO/IEC 10646.

Using a 16-bit encoding means that code values are available for more than 65,000 characters. While this number is sufficient for coding the characters used in the major languages of the world, the Unicode Standard and ISO/IEC 10646 provide the UTF-16 extension mechanism (called *surrogates* in the Unicode Standard), which allows for the encoding of as many as 1 million additional characters without any use of escape codes. This capacity is sufficient for all known character encoding requirements, including full coverage of all historic scripts of the world.

Figure 1-1. Wide ASCII



1.1 Coverage

The Unicode Standard, Version 3.0, contains 49,194 characters from the world's scripts. These characters are more than sufficient not only for modern communication, but also for the classical forms of many languages. Scripts include the European alphabetic scripts, Middle Eastern right-to-left scripts, and scripts of Asia. The unified Han subset contains 27,484 ideographic characters defined by national and industry standards of China, Japan, Korea, Taiwan, Vietnam, and Singapore. In addition, the Unicode Standard includes punctuation marks, mathematical symbols, technical symbols, geometric shapes, and dingbats.

Many new scripts and characters have been added in Version 3.0, including Ethiopic, Canadian Aboriginal Syllabics, Cherokee, Sinhala, Syriac, Myanmar, Khmer, Mongolian, Braille, and additional ideographs. Overall character allocation and code ranges are detailed in *Chapter 2, General Structure*.

Note, however, that the Unicode Standard does not encode idiosyncratic, personal, novel, rarely exchanged, or private-use characters, nor does it encode logos or graphics. Graphologies unrelated to text, such as dance notations, are likewise outside the scope of the Unicode Standard. Font variants are explicitly not encoded. The Unicode Standard reserves 6,400 code values in the basic 16-bit encoding for the *Private Use Area*, which may be used to assign codes to characters not included in the repertoire of the Unicode Standard.

There are 7,827 unused code values, which will permit future expansion in the basic encoding space. Provision has also been made for another 917,476 code points through the use of surrogate pairs. Surrogate pairs make another 131,068 private-use code points available should 6,400 prove insufficient for particular applications.

Standards Coverage

The Unicode Standard is a superset of all characters in widespread use today. It contains the characters from major international and national standards as well as prominent industry character sets. For example, Unicode incorporates the ISO/IEC 6937 and ISO/IEC 8859 families of standards, the SGML standard ISO/IEC 8879, and bibliographic standards such as ISO 5426. Important national standards contained within Unicode include ANSI Z39.64, KS C 5601, JIS X 0209, JIS X 0212, GB 2312, and CNS 11643. Industry code pages and character sets from Adobe, Apple, Fujitsu, Hewlett-Packard, IBM, Lotus, Microsoft, NEC, and Xerox are fully represented as well.

For a complete list of ISO and national standards used as sources, see *References*.

New Characters

The Unicode Standard continues to respond to new and changing industry demands by encoding important new characters. As an example, when the need to support the euro sign arose, *The Unicode Standard, Version 2.1*, with euro support was issued to ensure a conforming version.

As the universal character encoding scheme, the Unicode Standard must also respond to scholarly needs. To preserve world cultural heritage, important archaic scripts are encoded as proposals are developed.

For instructions on how to submit proposals for new characters to the Unicode Consortium, see *Appendix B, Submitting New Characters.*

1.2 Design Basis

The primary goal of the development effort for the Unicode Standard was to remedy two serious problems common to most multilingual computer programs. The first problem was the overloading of the font mechanism when encoding characters. Fonts have often been indiscriminately mapped to the same set of bytes. For example, the bytes 0x00 to 0xFF are often used for both characters and dingbats. The second major problem was the use of multiple, inconsistent character codes because of conflicting national and industry character standards. In Western European software environments, for example, one often finds confusion between the Windows Latin 1 code page 1252 and ISO/IEC 8859-1. In software for East Asian ideographs, the same set of bytes used for ASCII may also be used as the second byte of a double-byte character. In these situations, software must be able to distinguish between ASCII and double-byte characters.

The ASCII 7-bit code space and its 8-bit extensions, although used in most computing systems, are limited to 128 and 256 code positions, respectively. These 7- and 8-bit code spaces are inefficient and completely inadequate in the global computing environment.

When the Unicode project began in 1988, the groups most affected by the lack of a consistent international character standard included publishers of scientific and mathematical software, newspaper and book publishers, bibliographic information services, and academic researchers. More recently, the computer industry has adopted an increasingly global outlook, building international software that can be easily adapted to meet the needs of particular locations and cultures. The explosive growth of the Internet has merely added to the demand for a character set standard that can be used all over the world.

The designers of the Unicode Standard envisioned a uniform method of character identification that would be more efficient and flexible than previous encoding systems. The new system would satisfy the needs of technical and multilingual computing and would encode

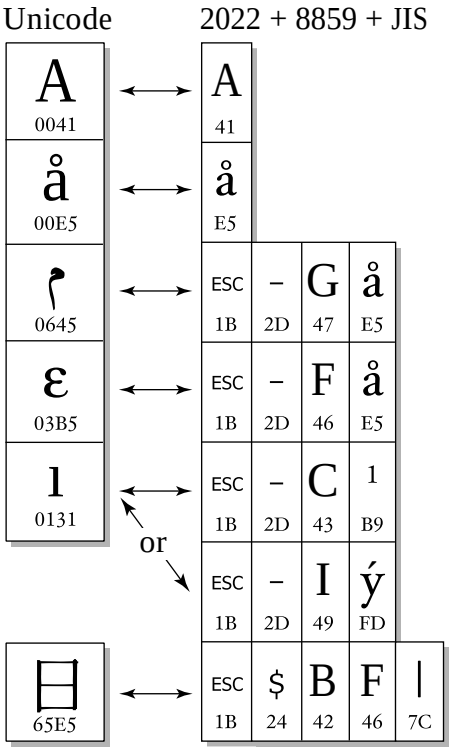
a broad range of characters for professional-quality typesetting and desktop publishing worldwide.

The Unicode Standard was designed to be:

- *Universal.* The repertoire must be large enough to encompass all characters that are likely to be used in general text interchange, including those in major international, national, and industry character sets.
- *Efficient.* Plain text is simple to parse: software does not have to maintain state or look for special escape sequences, and character synchronization from any point in a character stream is quick and unambiguous.
- *Uniform.* A fixed character code allows for efficient sorting, searching, display, and editing of text.
- *Unambiguous.* Any given 16-bit value always represents the same character.

Figure 1-2 demonstrates some of these features, contrasting the Unicode encoding with mixtures of single-byte character sets with escape sequences to shift the meanings of bytes.

Figure 1-2. Universal, Efficient, and Unambiguous



1.3 Text Handling

Computer text handling involves processing and encoding. When a word processor user types in text via a keyboard, the computer's system software receives a message that the user pressed a key combination for "T", which it encodes as U+0054. The word processor stores

the number in memory and also passes it on to the display software responsible for putting the character on the screen. This display software, which may be a windows manager or part of the word processor itself, then uses the number as an index to find an image of a "T", which it draws on the monitor screen. The process continues as the user types in more characters.

The Unicode Standard directly addresses only the encoding and semantics of text and not any other actions performed on the text. In the preceding scenario, the word processor might check the typist's input after it has been encoded to look for misspelled words, and then highlight any errors it finds. Alternatively, the word processor might insert line breaks when it counts a certain number of characters entered since the last line break. An important principle of the Unicode Standard is that the standard does not specify how to carry out these processes as long as the character encoding and decoding is performed properly and the character semantics are maintained.

Interpreting Characters

The difference between identifying a code value and rendering it on screen or paper is crucial to understanding the Unicode Standard's role in text processing. The character identified by a Unicode code value is an abstract entity, such as "LATIN CAPITAL LETTER A" or "BENGALI DIGIT 5". The mark made on screen or paper, called a glyph, is a visual representation of the character.

The Unicode Standard does not define glyph images. That is, the standard defines how characters are interpreted, not how glyphs are rendered. Ultimately, the software or hardware rendering engine of a computer is responsible for the appearance of the characters on the screen. The Unicode Standard does not specify the size, shape, or orientation of on-screen characters.

Text Elements

The successful encoding, processing, and interpretation of text requires appropriate definition of useful elements of text and the basic rules for interpreting text. The definition of text elements often changes depending on the process handling the text. For example, when searching for a particular word or character written with the Latin script, one often wishes to ignore differences of case. However, correct spelling within a document requires case sensitivity.

The Unicode Standard does not define what is and is not a text element in different processes; instead, it defines elements of encoding, called code values. A code value, commonly called a character, is fundamental and useful for computer text processing. For the most part, code values correspond to the most commonly used text elements.

1.4 The Unicode Standard and ISO/IEC 10646

The Unicode Standard is fully compatible with the international standard ISO/IEC 10646-1:2000, *Information Technology—Universal Multiple-Octet Coded Character Set (UCS)—Part 1: Architecture and Basic Multilingual Plane*, which is also known as the Universal Character Set (UCS). During 1991, the Unicode Consortium and the International Organization for Standardization (ISO) recognized that a single, universal character code was highly desirable. A formal convergence of the two standards was negotiated, and their repertoires were merged into a single character encoding in January 1992. Since then, close cooperation and formal liaison between the committees have ensured that all additions to

either standard are coordinated and kept synchronized, so that the two standards maintain exactly the same character repertoire and encoding.

Version 3.0 of the Unicode Standard is code-for-code identical to ISO/IEC 10646-1:2000. This code-for-code identity holds true for all encoded characters in the two standards, including the East Asian (Han) ideographic characters. ISO/IEC 10646 provides character names and code values; the Unicode Standard provides the same names and code values plus important implementation algorithms, properties, and other useful semantic information.

For details about the Unicode Standard and ISO/IEC 10646, see *Appendix C, Relationship to ISO/IEC 10646*, and *Appendix D, Changes from Unicode Version 2.0*.

1.5 The Unicode Consortium

The Unicode Consortium was incorporated in January 1991, under the name Unicode, Inc., to promote the Unicode Standard as an international encoding system for information interchange, to aid in its implementation, and to maintain quality control over future revisions.

To further these goals, the Unicode Consortium cooperates with the International Organization for Standardization (ISO/IEC/JTC1). It holds a Class C liaison membership with ISO/IEC JTC1/SC2; it participates in the work of both JTC1/SC2/WG2 (the technical working group for the subcommittee within JTC1 responsible for character set encoding) and the Ideographic Rapporteur Group (IRG) of WG2. The Consortium is a member company of the National Committee for Information Technology Standards, Technical Committee L2 (NCITS/L2), an accredited U.S. standards organization. In addition, Full Member companies of the Unicode Consortium have representatives in many countries who also work with other national standards bodies.

A number of organizations are Liaison Members of the Unicode Consortium: the Center for Computer & Information Development (CCID, China), the Internet Engineering Task Force (IETF), the Kongju National Library (Chung-nam, Korea), the Technical Committee on Information Technology (TCVN/TC1, Viet Nam), and the World Wide Web Consortium (W3C) I18N Working Group.

Membership in the Unicode Consortium is open to organizations and individuals anywhere in the world who support the Unicode Standard and who would like to assist in its extension and widespread implementation. Full and Associate Members represent a broad spectrum of corporations and organizations in the computer and information processing industry. The Consortium is supported financially solely through membership dues.

The Unicode Technical Committee

The Unicode Technical Committee (UTC) is the working group within the Consortium responsible for the creation, maintenance, and quality of the Unicode Standard. The UTC controls all technical input to the standard and makes associated content decisions. Full Members of the Consortium vote on UTC decisions. Associate and Specialist Members and Officers of the Unicode Consortium are nonvoting UTC participants. Other attendees may participate in UTC discussions at the discretion of the Chair, as the intent of the UTC is to act as an open forum for the free exchange of technical ideas.

This PDF file is an excerpt from *The Unicode Standard, Version 3.0*, issued by the Unicode Consortium and published by Addison-Wesley. The material has been modified slightly for this online edition, however the PDF files have not been modified to reflect the corrections found on the Updates and Errata page (see http://www.unicode.org/unicode/uni2errata/UnicodeErrata.html). More recent versions of the Unicode standard exist (see http://www.unicode.org/unicode/standard/versions/).

Dai Kan-Wa Jiten used as the source of reference Kanji codes was written by Tetsuji Morohashi and published by Taishukan Shoten.

ISBN 0-201-61633-5

This book is set in Minion, designed by Rob Slimbach at Adobe Systems, Inc. It was typeset using FrameMaker 5.5 running under Windows NT. ASMUS, Inc. created custom software for chart layout. The Han radical-stroke index was typeset by Apple Computer, Inc. The following companies and organizations supplied fonts:

Apple Computer, Inc.
Atelier Fluxus Virus
Beijing Zhong Yi (Zheng Code) Electronics Company
DecoType, Inc.
IBM Corporation
Monotype Typography, Inc.
Microsoft Corporation
Peking University Founder Group Corporation
Production First Software

Additional fonts were supplied by individuals as listed in the *Acknowledgments*.

The publisher offers discounts on this book when ordered in quantity for special sales. For more information please contact:

Corporate, Government, and Special Sales
Addison Wesley Longman, Inc.
One Jacob Way
Reading, Massachusetts 01867

Visit A-W on the Web: http://www.awl.com/cseng/

First printing, January 2000.