Design Considerations for Multilingual Web Sites

Joan Starr

The most powerful marketing, service, and information-distribution tool a library has today is its Web site, but providing Web content in many languages is complex. Before allocating scarce technical and financial resources, it is valuable to learn about writing systems, types of writing, how computers render and represent writing systems, and to study potential problem areas and their possible solutions. The accepted Web standard for presenting languages is Unicode and a full understanding of its history and the coding tools it provides is essential to making appropriate decisions for specific multilingual and internationalization projects. Actual coding examples, as well as a sampling of existing multilingual library services, also serve to illuminate the path of implementation.

The most powerful marketing, customer service, and information-distribution tool a library has today is its Web site. Providing dynamic, cost-effective service delivery, along with twenty-four-hour availability, Web sites are fast surpassing any other form of widespread information delivery and exchange, especially as Internet access approaches ubiquity. Unlike other types of mass media, Web sites are instantly and automatically available in every country, and Web designers can, with some basic knowledge of Hypertext Markup Language (HTML), present sites to virtually any Internet user in any language.

This ease of distribution belies somewhat the real challenges inherent in planning for an intercultural, multilingual reach. Recent research efforts show that the United States now represents only 27 percent of the world's Internet users and that English is the first language of only 36 percent of the world's users.[1] Moreover, many academic and public libraries serve an increasingly diverse population within the U.S. For example, the Queens Borough (New York) Library reported in 2000 that "almost half the [local] residents speak a language other than English at home."[2] This knowledge constitutes a mandate not only to internationalize one's Web site to extend its reach to non-English-speaking people, but also to do so with care, in order to ensure true communication and excellent service for all customers, regardless of country of origin.

Jakob Nielsen, a leading Web-usability expert, has proposed three principal areas for investigation by designers converting a preexisting local site to one with an international scope: navigation and layout, language handling, and regional content variation.[3] While navigation and content are critical components, the focus here will be on written language, its intrinsic characteristics, and their implications for Web site design.

Before allocating scarce technical and financial resources, it is important to gain a fuller sense of the topic of Web site internationalization. For effective translation of Web site content into other languages, it is valuable to learn about different writing systems, types of writing, how computers render and represent writing systems, and to study potential problem areas and their possible solutions in Web site internationalization. Scholars in fields as varied as marketing, anthropology, and psychology have produced a wealth of information about writing and a summary of their observations contributes a variety of insights that are useful to Web designers. The accepted standard for presenting languages on a Web site is Unicode, and a full understanding of Unicode history and the coding tools it provides to developers is essential to making appropriate decisions for specific multilingual and internationalization projects. Actual coding examples—specifically, the codes for character set, language, and direction of the text—as well as a sampling of existing multilingual library services also serve to illuminate the path of implementation.

Writing Systems Defined

In the *Blackwell Encyclopedia of Writing Systems*, Coulmas defined a writing system as:

> A set of visible or tactile signs used to represent units of language in a systematic way, with the purpose of recording messages which can be cited by everyone who knows the language in question and the rules by virtue of which its units are encoded in the writing system.[4]

That is, signs and symbols are the writing system itself, with the language's rules being applied (and supplied) by the reader. Note that this description is inclusive of nongraphic systems such as Braille. Writing twenty years earlier, Pulgram was considerably more restrictive, but similarly sign-oriented, reducing writing to "visible marks that evoke or recall linguistic performance."[5]

By contrast, the Unicode Consortium has defined a writing system as "a set of rules for using one or more

Joan Starr (joanstarr@earthlink.net) is a Senior Project Consultant in the Information and Technology Services Department at the Federal Reserve Bank of San Francisco. The views herein expressed are those of the author and not necessarily those of the Federal Reserve Bank of San Francisco or any federal entity.

scripts to write a particular language."[6] Writing, then, is an abstraction of logic, with the visible symbols swapped in and out of use according to precise rules. This clearly lays the groundwork for the treatment of writing as a framework of programmable codes and algorithms. While attractive from a practical perspective, a purely rule-based approach makes less sense when applied to writing systems that utilize meaningful symbols rather than meaningless phonetic marks. Meaningful symbols are characteristic of some of the world's major writing systems and the challenges they present to automation efforts merit special attention.

A complete definition of a writing system, then, requires "a set of characters and the basic rules for their use in creating a visual depiction of language."[7] Only with both perspectives is it possible to ensure the semantic and phonetic integrity of messages the system encodes. As Coulmas noted, recording and retrieving messages are the fundamental purposes of writing.[8]

Major Writing System Types

Scholars and students of the world's written languages have proposed a great variety of categorizations for writing systems. Pulgram described seven types: pictorial, logographic, syllabic, alphabetic, phonemic, phonetic, and spectrographic.[9] Ager used five categories: alphabetic, syllabic, logographic, undeciphered, and alternative.[10] Both Pulgram's and Ager's typologies made allowances for writing systems composed for pedagogical or theoretical purposes, such as the Pitman Initial Teaching Alphabet and the International Phonemic Alphabet. Coulmas restricted his evaluation to spoken (or formerly spoken) tongues, and, as such, delineated only two high-level groupings, cenemic (sense-discriminative), and pleremic (sense-determinative).[11]

The Coulmas division is clearest, though more familiar terminology—alphabetic and ideographic—would be helpful. The key distinction between these two groupings is their respective approaches to the relationships between sense, shape, and sound. Alphabetic systems use meaningless shapes, called graphemes or, less formally, letters, to correspond to the sounds of speech (phonemes) that in turn combine into meaningful words. Some alphabetic systems include only consonants, such as Hebrew and Arabic, while others provide representations for vowels and consonants, like Cyrillic, Greek, and the numerous Roman-based systems (including English). Still other alphabets use single symbols to correspond with whole syllables. Examples of syllabic alphabets include Amharic, the national language of Ethiopia, as well as Limbu, a Tibeto-Burman script.

Alphabets vary greatly in the closeness of the letter-to-sound correspondence, with some writing systems providing an almost one-to-one relationship; examples include Spanish, Finnish, and Serbo-Croatian. [12] More often, the systems retain some degree of etymological spelling, as with English. Another distinction among alphabets relates to their derivation. "Roman alphabets" are what linguists call the systems that have descended from an adaptation of Latin. These include the various European languages, such as English, Spanish, French, and German. The Roman alphabet contains twenty-six letters and, along with various diacritics, serves as the basis for more scripts than any other alphabetic set.[13] Diacritics are accents and other marks added to characters to modify their pronunciation or significance. New writing systems imposed upon previously unwritten languages, such as Cherokee, are typically Roman alphabet-based. Computer industry trade journals sometimes use the uncapitalized term "roman alphabet" to represent all alphabets, although this is not strictly accurate. As noted above, non-Roman alphabets include such major groupings as Arabic and Cyrillic writing.

Three-fourths of the world's languages employ alphabetic systems.[14] These systems are notably efficient in terms of the number of elements required to represent the language's vocabulary. Alphabets typically consist of between two- and five-dozen elements and are, therefore, considered to have small character sets. Ideographic systems, on the other hand, are inherently open-ended and quite large, although there are undoubtedly some limits to the human capacity for memorization.[15] Chinese, for example, contains an estimated 1.6 million ideograms, if the count includes all ethnic language minorities.[16]

Ideographic systems use meaningful shapes, called logograms or ideograms, to correspond to meaningful words or phrases. Thus, ideographic systems alone do not provide any pronunciation clues to the reader. Some ideographic systems operate alongside separate phonetic-based syllabic systems. An example of this is Japanese kanji, coupled with hiragana and katakana.[17] Certainly the most common ideographic systems are Chinese hanzi, Japansese kanji, and Korean hanja. Many writers refer to these three as a group, using the acronym CJK. For a reader accustomed to an alphabetic system, it may be useful to point out the logographic elements embedded in Roman numeric scripts. For example, when a teacher places two numbers, one over the other, with a line between them, this conveys mathematical information to the student. Likewise, when the mathematician shifts a number upward, shrinking it, to represent squared or cubed numbers, this imparts a meaningful concept rather than a sound.[18]

Research in the fields of marketing and psychology has indicated that the cognitive differences between users of alphabetic and ideographic writing systems are quite profound. Scientists have compared Chinese and English speakers in connection with their affective response to visual and auditory cues and have found ample evidence to suggest that speakers of ideographic languages are most strongly influenced by visual cues in information, whereas speakers of alphabetic languages respond most to audible cues.[19] This has significant implications for multilingual information architecture, inasmuch as designers often establish site identity and navigational elements with sensory cues of one type or another.

Two additional characteristics of writing systems have implications for the information architecture of multilingual sites: directionality and contextuality. Directionality refers to the normal orientation of the script. In strict terms, all contemporary writing systems are internally unidirectional, that is, they are presented and read in one direction only. However, in some cases, the numeric writing system associated with a language follows a different directional convention, resulting in a bidirectional total system. The most common examples of this type of bidirectionality (bidi, for short), are Hebrew and Arabic, which use a right-to-left pattern for text and a left-to-right pattern for numbers.[20]

For text display, the three most common directions are right to left, as in Arabic and Hebrew; left to right, as in all Roman alphabets; and top to bottom combined with right to left, as in CJK.[21] Linguists find incidences of internal textual bidi, that is, a script that alternates lines of right to left with those of left to right, only in antiquity; for example, in the ancient Greek boustrophedon writing.[22] In the multilingual environment of the Internet, bidi refers additionally to sections of text that combine writing systems with conflicting directions.[23] In other words, designers must treat pages that present Hebrew text with English terms inserted as bidi.

Contextuality refers to an aspect of character stability. A written system is contextual when it contains characters that change shape depending upon the context of neighboring characters. For example, in Arabic, different display forms exist for the standalone, initial, medial, and final placements of specific characters.[24] The alternative presentation forms are contextual variants.[25] Examples of contextuality in English include the differences found in cursive lettering from initial to medial placement, as well as in ligatures, which occur when the script combines two or more letters into a unit. Ligatures are relatively uncommon in contemporary English writing, but they remain a stylistic choice.[26]

Finally, in approaching the internationalization of a Web site, it is important to respect that, for good or ill, people have linked both alphabetic and ideographic writing systems to their ethnic, racial, or national identity. In the early twentieth century, for example, Western scholars routinely portrayed non-Roman writing systems as culturally inferior and considered the alphabet to be evidence of "man's emergence from his primitive barbarism."[27] During this same period, Mustafa Kamal, known now as Atatürk, introduced a language-reform campaign in Turkey. He demanded and achieved the adoption of a romanized script both as more effective than Arabic in representing the abundant vowel sounds of spoken Turkish, and also, as a mechanism for modernization.[28] In recent years, critics have accused Japanese kanji purists of language nationalism. The Japanese are, in fact, engaged in a debate as to whether a romanized script—romaji—would permit greater participation in global information exchanges, and, if so, whether or not this is desirable.[29] Site designers should explore these issues when planning for the selection and display of language options to avoid unintended political or cultural overtones.[30]

Writing Systems as Computerized Data

The discussion of how computers handle different writing systems begins at the most basic level. Fundamentally, computers manipulate and store text (and all other data) as binary number codes governed by algorithms, or processing rules. A computer operating system must support an array of such algorithms and codes. The sum of these algorithms and codes is a computer's script management system. Early systems were constrained by the high cost of computer memory, especially prior to 1960; developed in the U.S., they handled only Roman alphabets, primarily English.[31] These early systems utilized only six binary pairs, or bits, allowing for sixty-four distinct values. To the contemporary eye, the resulting displays would appear to be very limited even for English, because they omitted lowercase letters and many punctuation marks.

As the price of memory declined and the demand for better text support rose, software vendors extended the character-code size to eight bits, providing for a possible 128 different values. The computer industry refers to this bit-group size as a byte.[32] The character-set systems developed to encode characters using one byte are known as single byte (or 8-bit) systems. The American Standard Code for Information Interchange (ASCII) is a single-byte character set and it accommodates 127 characters and marks using all but one of the possible combinations.[33] Software firms have developed other, more complex single-byte character sets to support bidi and contextual alphabets by using a second 8-bit "extension."[34]

In order to support large character sets such as CJK, vendors developed competing double-byte (or 16-bit) encoding standards. Zhang and Zeng noted a number of commercially available packages, such as GB, HanZi, Chinese Character Code for Information Exchange, and East Asian Character Code.[35] A double-byte standard has approximately sixty-five thousand unique coding combinations available. Zhang and Zeng further found that multiple character-set standards have severely hampered CJK information exchange, as any given character may have multiple encodings.

In an attempt to preserve compatibility with ASCII's single-byte character set, while providing the extended capacity of 16 bits, developers have also created a number of large character sets using a variable byte size. These so-called multibyte character sets have one- and two-byte characters, and they are both the predecessors to and principle competitors of Unicode.[36] As with the double-byte standards, resources encoded using competing multibyte character sets are not able to exchange information without costly modifications.

In all of the aforementioned character sets, there has been no mention of the specific presentation of text on the screen or page. This is because character-set encodings are applicable to characters, not glyphs. As Aliprand explained, in this context, the term character means the letter or character in its elemental form, such as lower case "a." The term glyph specifies the appearance of the character.[37] For lower case "a," this refers to design choices such as whether it has one or two storeys, that is, whether or not the letter has a curved line over the lower loop or bowl.

Collections of glyphs used to depict character data are called fonts.[38] Fonts are software that is stored on a computer's hard drive either as graphic elements or as mathematical sets of points that are then converted into graphic elements for display. Fonts are associated with specific character sets and have properties such as size, orientation, weight, and whether or not the letters have serifs (added decorative strokes). Web designers can specify a font for page display, but if the site visitor's computer does not have that font, the visitor's browser software will swap in a default font set; this can result in considerable alteration of the appearance and readability of the site's content. Thus, it is significant that the presence or absence of the preferred font is essentially out of the designer's control. Fortunately, as Unicode 4.0 achieves acceptance, support for multilingual fonts is increasing, with free or low-cost font sets becoming available for many languages, so it is now practical to provide referring links for users of writing systems with significant online representation.[39]

Another challenge for multilingual Web page design and layout is arranging lists in a sortable order. This is due to the widely varied collation patterns of character sets. Within the Roman alphabets, for example, to achieve alphabetical order, one must resolve issues regarding the ordering of mixed-case text, letters with diacritical marks, ligatures, and so on.[40] In addition, the newly romanized systems, such as Turkish, may use only a portion of the Roman character set, which raises the question of how to order the omitted elements when a page includes non-Turkish Roman text.[41]

The East Asian writing systems also present sorting complexities. Although the CJK character sets contain a common core of Chinese characters, Japanese and Korean sets include additional language-specific characters. Chinese readers expect sorting by the number of strokes in a character, Japanese users expect sorting by the pronunciation of the Japanese hiragana characters, and Korean users expect the Korean characters to sort in Korean hangul alphabet order.[42] To provide accurate information ordering, then, encoding systems must establish a sorting key for each character set and include algorithms that calculate the proper character weights for that set. Unicode has such a feature.[43]

Two final page-layout issues remain: justification and line breaking. Justification is a variety of text alignment in which lines of text are forced to fill an exact line width. In Roman text, scripts can achieve this in one of two ways, by compressing or expanding the space between words (tracking) or by compressing or expanding the space between letters within words (kerning). Ideographic writing systems typically do not have a concept of space between words, so scripts achieve justification strictly by changing the intercharacter spacing. The Arabic writing system presents a special situation because its letters may be extended to justify a line.[44] Therefore, the Arabic character sets must include an extending-bar character, called a kashida. Interestingly, type stylists consider the kashida element a reflection of the calligraphic heritage of Arabic texts.[45]

As with justification, line breaks relate to the boundaries between words, phrases, and sentences, and different character sets handle these differently. Japanese and Chinese writing systems allow line breaks between any characters, but certain Japanese characters (kinsoku) may not begin or end a line.[46] Roman systems allow for word hyphenation, but the rules governing hyphenation vary by language. In a World Wide Web Consortium (W3C) technical note, Dürst proposed that designers handle line breaks as well as letter- and word-spacing issues by using eXtensible Markup Language (XML) syntax, linking to external style files.[47]

The Unicode Approach

By the early 1990s, a compelling argument existed for a single, all-encompassing character-encoding scheme. Information-exchange requirements, software-marketing forces, and the dawn of the Internet combined to drive together a group of industry representatives, government officials, and education and library science partners in pursuit of this goal. Two standards bodies undertook the effort: the International Organization for Standardization (ISO) with its 10646 project and the Unicode Consortium, sponsored, in part, by U.S. software companies.[48] To achieve a unified standard, the organizations agreed to synchronize their respective character-encoding standards, and the result is commonly referred to as Unicode.

Simply stated, Unicode is the "universal character-encoding system, a scheme that assigns standard numbers—called code points—to characters and symbols across all languages."[49] In addition to the global standard-character codes, Unicode provides algorithms to handle all alphabetic writing systems and large subsets of the major ideographic systems, along with their specific requirements for bidi, sort ordering, and contextuality. The W3C has unequivocally endorsed Unicode "as the base of the architecture of the Web."[50] The Unicode Consortium has expanded the original double-byte scheme to four bytes, or 32 bits, allowing for more than 1 million possible code points, organized in so-called "planes." Some planes are presently unused, and some are deliberately reserved for private or local use.[51]

Unicode is critically important for a number of stakeholders. First, it has already empowered a worldwide market in software manufacture, which is, of course, why vendors like Microsoft, Apple, Hewlett-Packard, IBM, and Oracle are members of the consortium.[52] Of greatest interest to the information profession and the international Internet-user base, however, is Unicode's implicit promise of easy information exchange and real, multinational, multilingual resource sharing. Unicode provides solutions to the problems of conflicting national and proprietary encoding schemes and addresses the major gaps in support for uncommon writing systems. In the W3C's words, "Unicode is the hub for conversion between different character encodings, making sure data can be handled in a uniform way and displayed, searched, sorted, and manipulated without fear of data corruption."[53]

Of course, Unicode is not perfect. As noted above, support for ideographic languages is incomplete. In addition, for efficiency's sake, the consortium made the decision to treat the common characters in CJK systems as a single set of encodings.[54] The problem is that the individual ideograms, while appearing identical, have different meanings in the different languages. Japanese critic Sakamura Ken has been vocal in his accusations that "the code was developed by people from non-kanji countries . . . without any real understanding of the compatibility needs of the separate [kanji] countries."[55] Therefore, any Web page using the CJK character set must clearly establish the language context.

Implications for Coding

Recall that "The major thing the [Unicode] standard does not define is the actual rendering of the character image."[56] This puts the responsibility for text presentation squarely upon the page designer. Addressing the challenges of coding and designing for Web pages that may appear in multiple languages, Nielsen offered this advice, "remember . . . not to overdesign to the extent that the page will not work if some words are pushed around or if some table cells become a little wider."[57] An alternative approach is to design style sheets for specific-language character sets and then, based on the language, programmatically alternate the markup instructions.

The most common strategies for storing and manipulating language information include making use of <meta> tags in HTML document headers and utilizing the lang attribute, available in virtually all HTML elements.

Coding Meta Tags

Meta tags are placed within the <head> tags at the beginning of HTML documents in order to establish information about that document.[58] The head section can include any number of meta elements inside. Within a <meta> element, the name and http-equiv attributes identify the type of metadata being stored in that element, including content type, language, and direction. The difference between the two attributes is that, in addition to the identification function, the http-equiv attribute also serves to pass information to http-response headers.[59] The examples here use the http-equiv attribute.

By default, HTML documents have a content-type with the value of text/html. To specify a character set, the developer simply extends the content-type value with a character set statement, as follows:

```
<meta http-equiv="content-type" content="text/html;
charset=UTF-8">
```

This instructs the browser to decode content as 8-bit Unicode. Texin and Savourel have suggested that designers place the charset encoding as early in the document as possible, because until the browser parses this statement, "the document's encoding is unknown."[60]

To specify a language, the value of the http-equiv or name attribute is "content-language," and the value of the content attribute is a language code. In a 1999 tutorial, Yergeau and Dürst explained that W3C's Request for Comments (RFC) 1766 specification on language tags allows the ISO-639 two-letter language codes, optionally appended by an ISO-3166 country code, as well as codes from the Internet Assigned Numbers Authority, and experimental codes prefixed by the letter x.[61] A sample follows:

```
<meta http-equiv="content-language" content="fr-ca">
```

This demonstrates how to establish the page language as the dialect of French spoken in Canada.

Directionality is an attribute of many elements in the HTML Specification, so the designer has a number of options for coding bidi pages. The designer can indicate directionality for an entire document by the selection of character set and by using the dir attribute of the meta element. The allowable values are rtl for right to left, and ltr for left to right. A sample follows:

```
<meta dir="rtl">
```

The default value of this attribute is ltr, so it is most important to utilize if the desired result is a right-to-left display. Developers should take note that the W3C has warned against mixing approaches to the handling of directionality:

> . . . conflicts can arise if the dir attribute is used on inline elements . . . concurrently with the corresponding UNICODE formatting characters . . . If both methods are used, great care should be exercised to insure proper nesting of markup and directional embedding or override, otherwise, rendering results are undefined.[62]

Coding the Lang Attribute

As with directionality, the HTML Specification allows for language designation at every level, using the lang attribute. The meta-tag technique described above sets a language attribute for the entire document. If the designers wish to assign a different language value for a particular textual segment, they can set the lang attribute for that segment.[63] For example, this code enables the page to display a quotation using Greek characters, in the context of another base language:

```
<q lang="el"> . . . place text of Greek quotation here
. . . </q>
```

In all such uses, the W3C's RFC 1766 governs allowable language codes.

When implementing this technique using style sheets, developers should employ classes, as in this Cascading Style Sheets (CSS) example that supports a multilingual page:

```
body {font-family: "Times New Roman", serif;}
.ar  {font-family: "Traditional Arabic", serif; font-size:
1.2em;}
.zht {font-family: PMingLiU,MingLiU, serif;}
.zhs {font-family: SimSum-18030;SimHei, serif;}
.din {font-family: "Doulos SIL", serif;}
```

In the future, it is likely that browsers will also support a pseudo-class language selector technique for CSS.[64]

Unicode in Practice

The decision to embark upon a multilingual and internationalization project for library services is typically contingent upon one or more of the following conditions:

1. The library has a multilingual collection.
2. The library serves a community (geographic, academic, or commercial) with readers of languages other than English.
3. The library determines a need to exchange information or build cooperative services with libraries of the other two types.

The projects themselves range from multilingual Web directories or finding aids and multilingual OPACs to the creation of new multilingual content.

Multilingual Finding Aids

Multilingual Web directories constitute one of the earliest efforts by librarians and others to provide information access in multiple languages. Librarians created the first of these in languages that use Roman-based scripts, principally Spanish, with library Web site designers simply using the special characters in HTML to accommodate non-English characters—for example, using <&ccedil;> to represent the letter "c" with a cedilla (ç). Public and academic libraries now face an increasing demand for Web directories in non-Roman scripts, as geographic and educational communities absorb populations from Asia, Eastern Europe, the Middle East, and Africa. Two examples of existing multilingual directories are the Queens Borough (New York) Public Library's (QBPL) WorldLinq (www.worldlinq.org) and the Danish State and University Library's FINFO (www.finfo.dk/wwwfinfo/html/default.html).

WorldLinq offers a librarian-selected directory to global Web resources for the areas from which local residents emigrated.[65] QBPL's technical staff engaged an outside vendor to develop the site, with adoption of 32-bit Unicode as a system requirement. This ensured support of the initial language offerings and enabled WorldLinq's scope to grow and adapt over time.[66] The number of languages supported has indeed increased since inception and now includes Arabic, Chinese, Croatian, French, Korean, Romanian, Russian, Spanish, and Ukrainian.

The Danish Central Library for Immigrant Literature, a branch of the Danish State and University Library, provides FINFO. Regional and municipal libraries contribute maintenance assistance. The site includes information about all aspects of Danish life in Albanian, Arabic, Croatian, Danish, English, French, Kurdish (Kurmanji), Persian, Tamil, Turkish, Urdu, and Vietnamese.[67] When the library originally launched the site in 1999, there was no standard character set that could accommodate the many languages FINFO covered. Users had to typically apply plug-ins and download fonts. When widespread support for Unicode became available, the library staff converted FINFO to Unicode encoding and "focused on developing input methods for the languages in question, based on the Unicode standard."[68]

Multilingual Catalogs

In addition to directories and finding aids, many librarians find that multilingual access to the OPAC is compelling, as they "collect materials in many different languages and want to be able to display the native scripts in their Web catalogs as well as allow users to search by typing in the native scripts."[69] Responding to this need in 1998, the MARC Advisory Committee approved Unicode as an interchange format for MARC21 records.[70] The MARC Standards Office followed with a set of specifications to govern the exchange, declaring that library systems are in a period of transition to the Unicode character set. For this reason, the specification restricted the use of Unicode to a subset "made up of the [Unicode] characters that correspond to the over 16,000 characters defined in the separate MARC8 character sets for MARC21."[71]

During this transition, librarians should look for library catalog systems that support "record exchange in both MARC21 [*sic*, MARC8] character sets and Unicode until Unicode becomes the dominant character set used by most systems."[72] Current users of the high-end library systems already have this feature as a fully integrated capability. Other libraries should conduct an evaluation of their catalog systems to determine the status of Unicode compliance.

Multilingual Content

Unicode's international character set greatly enables multilingual digital-library collection building. This is especially important for languages and cultures that have been traditionally underrepresented in archives and libraries worldwide. Cunningham of the Vicnet State Library of Victoria, Australia, asserted, "The new paradigm of electronic multicultural library services should be based on the principles of universal access and linguistic rights—that everyone should be able to see their language and culture reflected in the resources available on the Internet."[73] Accordingly, Vicnet staff members determined to place a high priority on what they termed "new and emerging" linguistic communities. They conducted an extensive content-and-system development effort, having found that, for some languages, digital resources did not yet exist. An example of their work is the "Naath online" site (http://home.vicnet.net.au/~naath/) presented in the southern Sudanese language of Nuer. Working closely with the language community, Vicnet built the site using Unicode, rather than non-standard legacy coding, for the reasons that Unicode support is already widespread, and that new fonts and tools are becoming increasingly available.[74]

Cunningham acknowledged that most public libraries would not have the resources to undertake the system-development effort, calling for "a content infrastructure, including Web development toolkits."[75] Computer-science researchers at New Zealand's University of Waikato drew the same conclusion and developed the Greenstone digital-library software.[76] Greenstone is an open-source, freely licensed system that provides "a collection-building 'wizard' that allows nonprogramming users to create and organize new digital-library collections from source documents."[77] The Greenstone software uses Unicode, so "users are able to translate the interface into local languages without having to delve into the detailed operation of the software"[78] As an open-source system, Greenstone benefits from the efforts of its users, so that many interfaces are now available for download, including Arabic, Armenian, Catalan, Chinese, Croatian, Czech, Dutch, English, Farsi, Finnish, French, Galician, Georgian, German, Greek, Hebrew, Hindi, Indonesian, Italian, Japanese, Kannada, Kazakh, Latvian, Maori, Portuguese (Brazil), Portuguese (Portugal), Russian, Serbian, Spanish, Thai, Turkish, Ukrainian, Vietnamese.[79]

The International Children's Digital Library (ICDL) is a five-year project of the University of Maryland's Human-Computer Interaction Lab. Its first goal is "to create a collection of more than 10,000 books in at least 100 languages that is freely available to children, teachers, librarians, parents, and scholars throughout the world via the Internet."[80] ICDL is, in part, a recognition that resource limitations constrain collection activities at free libraries worldwide, and the project aims to counterbalance the inequity by offering "the largest bookmobile in history [for] children around the world."[81] The site currently provides direct access to 573 books in twenty-one languages (Arabic, Croatian, English, Filipino, Finnish, French, German, Hebrew, Italian, Japanese, Macedonian, Maori, Persian/Farsi, Portuguese, Russian, Serbian, Shona, Spanish, Swedish, Vietnamese, and Yiddish), with multilingual search and display supported by the Unicode character set. ICDL provides a very complete Help page for its users with original material as well as a link to Alan Woods' Unicode resources page.[82]

Conclusions

To review, then, there are substantial differences among written languages, and these differences pose a significant challenge to the project of internationalizing an existing local Web site. The two major writing system types, alphabetic and ideographic, are opposites in their approach to the relationships between characters, meaning, and pronunciation. Within the two types, there are vast differences in the areas of directionality, sorting, word discrimination, and more. There are cognitive distinctions between users of alphabetic and ideographic writing systems that are likely to affect their response to site navigation and identity elements. Designers should consider the potential implications for branding approaches aimed at target language groups.

To address the challenges of multilingual Web site service provision, library managers without qualified internal resources may choose to outsource the development effort, either by obtaining a Unicode-compliant software package or by engaging a technical consultant (paid or volunteer). In monitoring the progress of an internal project or evaluating the alternatives offered by external service providers, it may be useful to refer to the following summary of the principal considerations for site-design teams.

1. The choice of which languages to support and how to display their selection may have political and cultural overtones. The team should explore this possibility and validate its design with user testing if at all possible.
2. It is now practical to provide direct links to font-download facilities for many languages, and project teams should investigate the best way to present these referrals.
3. Many text display characteristics are language-specific, so the team should employ variable style sheets to provide the appropriate markup. Markup elements that will be key include dir, lang, and charset.
4. Pages that combine writing systems with opposing directionality are bidirectional, as are pages provided in writing systems with internal bidi. The designers should use either inline markup *or* Unicode- formatting specifications to handle this variation.
5. By using Unicode, the team can ensure that the character-encoding system uses a language-appropriate sorting key for each character set, allowing for accurate information collation. Unicode also provides correct encoding for contextual variants in writing systems where characters change shape based on context.

Providing Web site content in many languages is a complex undertaking, even without considering the substantial challenges of translation. There are clear and practical methodologies available for handling the complexity, and, as the 2004 Library of Congress study on expanding the MARC21 character set noted, "the adoption of Unicode . . . bring[s] the library world more into line with mainstream computer developments."[83] Fortunately, it is likely that the tools and techniques supporting multilingual presentations will continue to improve, inasmuch as the Web's governing body, the W3C, is firmly committed to a Web that is "usable worldwide in all languages and in all writing systems."[84]

References and Notes

1. Central Intelligence Agency, "Rank Order—Internet Users," *World Fact Book* (2003). Accessed Feb. 20, 2004, www.odci.gov/cia/publications/factbook/rankorder/2153rank.html; Global Reach, "Global Internet Statistics (By Language)," (2003). Accessed Feb. 20, 2004, www.glreach.com/globstats/index.php3.

2. G. Strong, "LinQing the World to Queens—and Queens to the World," *American Libraries* 31, no. 9 (2000): 44–46.

3. J. Nielsen, *Designing Web Usability: The Practice of Simplicity* (Indianapolis, Ind.: New Riders, 2000).

4. F. Coulmas, *The Blackwell Encyclopedia of Writing Systems* (Oxford, England: Blackwell, 1996), 560.

5. E. Pulgram, "The Typologies of Writing Systems," in W. Haas (ed.), *Writing Without Letters* (Manchester, England: Manchester Univ. Press, 1976), 1–28.

6. Unicode, "Glossary of Unicode Terms" (2003). Accessed Feb. 16, 2004, www.unicode.org/glossary.

7. Apple Computer, "Features of the World's Writing Systems," in Inside Macintosh: Text (1996). Accessed Feb. 21, 2004, http://developer.apple.com/documentation/mac/Text/Text-25.html.

8. Coulmas, *Blackwell Encyclopedia of Writing Systems.*

9. Pulgram, "The Typologies of Writing Systems."

10. S. Ager, "Omniglot: A Guide to Writing Systems" (n.d.). Accessed Feb.15, 2004, www.omniglot.com.

11. Coulmas, *Blackwell Encyclopedia of Writing Systems,* 521.

12. Ibid., 11.

13. Ibid., 438.

14. N. T. Tavassoli and J. K. Han, "Auditory and Visual Brand Identifiers in Chinese and English," *Journal of International Marketing* 10, no. 2 (2002): 13–28.

15. Pulgram, "The Typologies of Writing Systems."

16. J. Felici, "Unicode: The Quiet Revolution," *The Seybold Report: Analyzing Publishing Technologies* 2, no. 10 (2002): 11–15.

17. N. Gottlieb, *Word-Processing Technology in Japan* (Richmond, Va.: Curzon, 2000).

18. W. C. Brice, "The Principles of Non-Phonetic Writing," in W. Haas (ed.), *Writing Without Letters* (Manchester, England: Manchester Univ. Press, 1976), 29–44.

19. Y. Pan and B. Schmitt, "Language and Brand Attitudes: Impact of Script and Sound Matching in Chinese and English," *Journal of Consumer Psychology* 5, no. 3 (1996): 263–77; Tavassoli and Han, "Auditory and Visual Brand Identifiers in Chinese and English."

20. Apple Computer, "Features of the World's Writing Systems."

21. Coulmas, *Blackwell Encyclopedia of Writing Systems*.

22. Ibid., 49.

23. R. Ishida, “What You Need to Know about the Bidi Algorithm and Inline Markup” (2003). Accessed Feb. 15, 2004, www.w3.org/international/articles/inline-bidi-markup.

24. Apple Computer, “Features of the World’s Writing Systems.”

25. Unicode 2003, “Glossary of Unicode Terms.”

26. Coulmas, *Blackwell Encyclopedia of Writing Systems*, 295.

27. W. A. Mason, *A History of the Art of Writing* (New York: Macmillan, 1920), 17.

28. S. Atkin and A. Irmakkesen, “Computing in Turkish,” *MultiLingual Computing & Technology* 13, no. 7 (n.d.). Accessed Feb. 11, 2004, www.multilingual.com/FMPro?-db=archives&-format=ourpublication%2ffeaturedarticlesdetail.htm&-lay=cgi&-sortfield=magazine%20number&-sortorder=descend&-op=eq&Ad%20Type=reprint&-recid=33417&-find.

29. Gottlieb, *Word-Processing Technology in Japan*.

30. Nielsen, *Designing Web Usability*, 327.

31. A. Dumestre, “Digging Deeper—Text and Computers,” *Orange Bytes* 1, Article 394 (1999). Accessed Feb. 11, 2004, www.noccc.org/bytes/articles/v01/394.html.

32. Unicode 2003, “Glossary of Unicode Terms,” Byte.

33. Dumestre, “Digging Deeper.”

34. Apple Computer, “Features of the World’s Writing Systems.”

35. F. Zhang and M. L. Zeng, “Multiscript Information Processing on Crosswords: Demands for Shifting from Diverse Character Code Sets to the Unicode™ Standard in Library Applications,” *IFLA Journal* 25, no. 3 (1999): 162–67.

36. Microsoft Developers Network, “Support for Multibyte Character Sets (MBCS)” (2004). Accessed Feb. 22, 2004, http://msdn.microsoft.com/library/default.asp?url=/library/en-us/vccore/html/_core_support_for_multibyte_character_sets_.28.mbcs.29.asp.

37. J. M. Aliprand, “The Unicode Standard: Its Scope, Design Principles, and Prospects from International Cataloging,” *Library Resources & Technical Services* 44, no. 3 (2000): 160–67.

38. Unicode 2003, “Glossary of Unicode Terms,” Font.

39. A. Woods, “Unicode Fonts for Windows Computers,” in *Alan Wood’s Unicode Resources* (2004). Accessed Feb. 22, 2004, www.alanwood.net/unicode/fonts.html.

40. Apple Computer, “Features of the World’s Writing Systems.”

41. Atkin and Irmakkesen, “Computing in Turkish.”

42. L. Tull, “Library Systems and Unicode: A Review of the Current State of Development,” *Information Technology and Libraries* 21, no. 4 (2002): 181–85.

43. Unicode, “Unicode Collation Algorithm,” in *Unicode Technical Standard* #10 (n.d.). Accessed Feb. 23, 2004, www.unicode.org/reports/tr10/.

44. Apple Computer, “Features of the World’s Writing Systems.”

45. H. S. AbiFares, “Arabic Type: A Challenge for the Second Millennium” (1998). Accessed Feb. 23, 2004, www.sakkal.com/articles/Arabic_Type_Article/Arabic_Type1.html.

46. Antenna House, “Solutions for Multilingual Literature by XSL Formatter” (n.d.). Accessed Feb. 23, 2004, www.antennahouse.com/DMCHTML/sample/html/PowerPoint2.htm.

47. M. Dürst, “A Notation of Character Collections for the WWW,” World Wide Web Consortium Note (Nov. 5, 1999). Accessed Feb. 23, 2004, www.w3.org/TR/1999/NOTE-charcol-19991105.

48. M. Kuhn, “UTF-8 and Unicode FAQ for Unix/Linux” (2004). Accessed Feb. 12, 2004, www.cl.cam.ac.uk/~mgk/unicode.html.

49. Felici, “Unicode: The Quiet Revolution,” 11.

50. World Wide Web Consortium, “Internationalization Activity Statement” (2003). Accessed Feb. 27, 2004, www.w3.org/International/Activity.html.

51. Felici, “Unicode: The Quiet Revolution,” 12.

52. Ibid.

53. World Wide Web Consortium, “Internationalization Activity Statement.”

54. M. Needleman, “The Unicode Standard,” *Serials Review* 26, no. 2 (2000): 51–55. Accessed Feb. 9, 2004. A

55. Gottlieb, *Word-Processing Technology in Japan*.

56. Needleman, “The Unicode Standard.”

57. Nielsen, *Designing Web Usability*, 318.

58. J. Niederst, *Web Design in a Nutshell* (Sebastopol, Calif.: O’Reilly, 1999), 94.

59. World Wide Web Consortium, “Internationalization Activity Statement.”

60. T. Texin and Y. Savourel, “Web Internationalization, Standards and Practice” (2003), 24. Tutorial presented at the 24th Internationalization and Unicode Conference. Accessed Feb. 15, 2004, www.xencraft.com/resources/web18ntutorial.pdf.

61. M. Yergeau and M. Dürst, “Weaving the Multilingual Web” (1999). Tutorial presented at the 15th Internationalization and Unicode Conference. Accessed Feb. 28, 2004, www.w3.org/Talks/1999/0830-tutorial-unicode-mjd.

62. World Wide Web Consortium, “HTML 4.01 Specification” (1999). Accessed Feb. 28, 2004, www.w3.org/TR/REChtml40/cover.html#minitoc.

63. Niederst, *Web Design in a Nutshell*, 460.

64. World Wide Web Consortium, “W3C Web Internationalization FAQ” (2004). Accessed Feb. 28, 2004, www.w3.org/International/questions.html.

65. Strong, “LinQing the World to Queens.”

66. Ibid.

67. Danish Central Library for Immigrant Literature, “About FINFO” (n.d.). Accessed Dec. 28, 2004, www.finfo.dk/wwwfinfo/HTML/engelsk/Finfo_Danmark/OmFinfo.html.

68. P. Jessen, “Re: Inquiry about FINFO Site,” personal e-mail, Jan. 3, 2005.

69. Tull, “Library Systems and Unicode,” 181.

70. Ibid.

71. Library of Congress Network Development and MARC Standards Office, “MARC21 Specifications for Record Structure, Character Sets, and Exchange Media” (2003). Accessed Dec. 28, 2004, www.loc.gov/marc/specifications/speccharucs.html.

72. Tull, “Library Systems and Unicode,” 181.

73. A. Cunningham, “Global and Local Support Dimensions for Emerging Community Languages,” *Australia’s Public Library Information Service* 17, no. 3 (2004): 113–24. Accessed Dec. 18, 2004. Available from WilsonWeb database.

74. Ibid.

75. Ibid., 124.

76. I. H. Witten et al., “The Promise of Digital Libraries in Developing Countries,” *The Electronic Library* 20, no. 1 (2002): 7–13. Accessed Dec. 18, 2004.

77. Ibid.

78. Ibid.

79. Greenstone Digital Library Software, "About Greenstone." Accessed Dec. 31, 2004, www.greenstone.org/cgi-bin/library.

80. International Children's Digital Library, "Overview." Accessed Dec. 18, 2004, www.icdlbooks.org/project/overview.shtml.

81. International Children's Digital Library, "Executive Summary." Accessed Dec. 30, 2004, www.icdlbooks.org/project/summary.shtml.

82. International Children's Digital Library, "Foreign Character Help." Accessed Dec. 30, 2004, www.icdlbooks.org/library/basic/helpfont.html; Woods, "Unicode Fonts for Windows Computers."

83. J. Cain, "Assessment of Options for Handling Full Unicode Character Encodings in MARC21: A Study for the Library of Congress" (2004). Accessed Dec. 30, 2004, www.loc.gov/marc/marbi/2004/2004-report01.pdf.

84. World Wide Web Consortium, "W3C Web Internationalization FAQ."