The Open Road:  
language technology developments  
and public library services

Andrew Cunningham  
Multilingual Technical Project Officer,  
Accessibility and Evaluation Unit,  
VICNET, State Library of Victoria  
andrewc@vicnet.net.au

Larry Stillman,  
Accessibility and Evaluation Unit  
VICNET, State Library of Victoria  
&  
Senior Research Fellow  
Centre for Community Networking Research  
Monash University  
larrys@vicnet.net.au

Abstract:  
Language technology and access to the internet are crucial components in the development of flexible and responsive library services to culturally and linguistically diverse populations. Minority and emerging languages need public access. Internet access is one way of breaking the linguistic barriers that have been erected around minority and emerging languages. This paper reviews the Open Road multilingual website (www.openroad.vic.gov.au) from a technical perspective, outlining current strengths and limitations of major operating systems and browsers in the display of different character sets. Some solutions for entry and display of languages are offered for public access systems in public library environments.
The Open Road: language technology developments and public library services

Introduction

The last few years have seen major changes in the linguistic makeup of the internet, with increased diversity. It is no longer an English, particularly American English space. The increased linguistic diversity includes not only languages with large populations, but also smaller minority languages and even endangered languages.

Materials in many languages are available as HTML documents and PDF files. Increasingly material is available as digitised radio and television programs via the internet. Major multilingual radio broadcasters are digitising their programs. The list of multilingual broadcasters includes the BBC World Service, Deutsche Welle, Voice of America, Radio Free Asia, Radio Free Europe/Radio Liberty, Radio Australia and Radio Japan, among others. Programs are available in a large range of languages, including languages that are difficult for libraries to locate through traditional resources.

These changes have provided public libraries with the opportunities to expand their services to their culturally and linguistically diverse communities. Public libraries, traditionally, have only been able to develop and sustain collections in a limited number of languages. Public internet access allows public libraries a way to provide access to resources in a much wider range of languages.

The Open Road\(^1\) at its core is an ideal: the provision of multilingual public internet access in Victorian public libraries. Any member of the public should be able to go to their local library to use the internet and access internet resources in their preferred language, assuming that the language is available somewhere on the internet.

This is the first goal of the Open Road project. The second goal is to assist public libraries to extend multicultural library services through the use of technology. This goes beyond the provision of multilingual public internet access, and includes the internationalization of library management systems and bibliographic records, communication between multicultural library service specialists around the world, acquisitions, provision of reference services to multicultural library services and many other facets of library services. MCL-net\(^2\), a companion web site to the Open Road web directory, is a starting point in addressing this second goal.

The Open Road project evolved and developed through collaboration and the exchange of information, knowledge and skills. The Port Phillip Library Service, located in bayside Melbourne, and the Maribyrnong Library Service, located in the Melbourne’s industrial west, introduced multilingual public internet workstations (MPAWs) a number of years ago. Port Phillip developed a small web directory, indexing resources in a range of languages to supplement existing language collections. Maribyrnong developed a multilingual web site, and started to explore and develop Unicode web pages as an interface to resources in minority and emerging languages in Maribyrnong’s community. However, anecdotal evidence suggests that the initial uptake of multilingual internet access by Victorian public libraries was artificially limited to large and identifiable language groups in libraries’ communities.
The Open Road project is an attempt to make MPAWs universally accessible in Victorian public libraries. Accessibility in this sense refers to the language barriers we build into our electronic information resources.3

A growing amount of material on the internet is available in languages other than English. It is possible for libraries to extend the services they provide to their linguistically diverse communities through the provision of access to these resources.

There are several strategies the Open Road project has used to assist public libraries in realising the potential of a multilingual internet and providing multilingual internet access for the public.

Access has been fostered by:

- a series of training programs for Victorian public library staff covering the administration and configuration of computers for multilingual use and assistance to the public to facilitate public use of the internet in other languages; and

- a small multilingual web directory, which provides a starting point for exploring the internet in selected languages4. The design and implementation of version 2 of the directory (the current version) commenced in late 1999.

This paper will explore some of the language technology issues relevant to the current implementation of the Open Road’s public directory, and possible future developments. A basic knowledge of Unicode and current W3C HTML, XML and CSS recommendations is assumed.

**Open Road Mark 1, 1999**

The first ‘build’ of Open Road was officially launched in May, 1999. It was a collaborative effort, drawing on the knowledge gained by Port Phillip and Maribyrnong Libraries and VICNET.

The site was static, and links were collaboratively collected. It was not updated, but served to demonstrate that a multilingual site could be built for a small amount of money and in a short amount of time (around 6 months). This site, now archived, can be viewed at http://www.openroad.vic.gov.au/old/. Links were made available in Arabic, Chinese (Traditional script), Greek, Russian, Turkish, and Vietnamese, with a few extra links in Albanian, Amharic, and Vietnamese. These six languages were chosen since they represented major community languages in Victoria with significant demand for access.

A core decision in the development of both the early version of Open Resources did not permit the creation of original webpages. In addition, it would not be possible to provide full interactivity in different scripts for email to the Open Road team, submission of forms etc. Instead, it provided read-only access to annotated links around key categories (youth and children, women’s issues, arts, culture, education and media), though some languages could build extra categories if there were particular special needs.
**Mark II: the Current State of Play**

The current version of the Open Road web directory (Stage II) was designed in late 1999. The language technology incorporated into the design of the site reflects the software and language technology options available at the time the specifications for the directory were written.

The Open Road directory has utilised open source solutions. It runs on a Linux server, running Apache web software and the PostgreSQL RDMS. CGI scripts generate the web pages. The scripting language used is Perl. Unicode data is stored in the database in UTF-8 encoding.

PostgreSQL must be compiled with Unicode support. If PostgreSQL is compiled with Unicode support, the result is that the default character set is Unicode. PostgreSQL, like many Unix and Linux applications and servers, uses UTF-8 for the character encoding.

Since Perl 5.005_50 it has been possible to use UTF-8 encoding strings in Perl, although it is preferable to use Perl 5.6 with appropriate patches. UTF-8 support in Perl, PostgreSQL and current web browsers avoided the necessity of transcoding data at different stages in the input or output of data. There are limitations to current implementations of Unicode in Perl, and these centre around collation issues. More information on the current state of Unicode support in Perl is available in the *Perl, Unicode and i18N FAQ*.

**Limitations**

The Open Road project has developed and been shaped by:

- the possibilities and limitations of Unicode implementations in applications and operating systems; as well as
- the choice of operating systems and web browsers public libraries use for their public internet workstations.

At the present time, Victorian public libraries use Windows 95, Windows 98, and Windows NT4, with Windows 95/98 operating systems being more common. There is little to no use of Windows 2000 or the recently released Windows XP. With respect to browsers, libraries use Netscape 4.x or Internet Explorer 5.x, with little use of Opera, Netscape 6.x, or any of the other spin-offs of the Mozilla source code.

Each browser has its strengths and weaknesses. Some browsers are more suited to particular languages than others. Likewise, each operating system has a differing range of language support. Each operating system Microsoft releases has greater language support than the previous operating systems. When it released the Windows NT4 operating system, Microsoft committed itself to a greater support for Unicode within its operating systems and applications.

Microsoft’s definition of operating system support for a language includes support for an IME, locale specific functions such as time, date, number and currency formats, sorting/collation, an appropriate font, and support for screen rendering of the script.
Windows 2000 indicated a new trend within Microsoft. By this stage Microsoft had added support for most national and international character set standards. All additional languages that are not supported by existing international standards were only supported through Unicode.

Internationally, the preferred operating system for multilingual use was Windows 2000, in which Microsoft, cobbled together limited Unicode support for the Windows 95/98/ME operating systems. This added support allows users to display Unicode web pages in Internet Explorer and Netscape. Unfortunately there are major limitations to this support. Unicode input is limited to languages supported by existing Windows code pages. This support is not always provided by the major browsers.

When Stage II of the Open Road web site was being planned, a list of target languages was identified by Open Road’s steering committee. These languages were Arabic, Chinese (Simplified and Traditional scripts), Greek, Indonesian, Italian, Polish, Russian, Sinhalese (Sinhala) and Turkish. The second list of languages to be implemented at a later date included Amharic, Hindi, Oromo, Persian (Farsi), Somali, Tamil, Tigrigna, and Vietnamese.

Sinhala provided the first hurdle, or more precisely the first stumble. Sinhala is a complex script language that is not currently supported by Microsoft. The Sinhala script has been included in the Unicode standard. Unfortunately, support for the script as not available in Windows. It was necessary to exclude Sinhala from the first set of languages to be developed.

Arabic, Hindi, Persian and Tamil could be displayed on a variety of Windows operating systems using Internet Explorer 5.5. Input was limited to the Windows 2000 operating system.

Amharic and Tigrigna are not supported by Microsoft, but it is possible to display Amharic and Tigrigna Unicode web pages if a Unicode font supporting the Ethiopic script is installed. Input for Amharic and Tigrigna is not available on the Windows operating systems. The Open Road team is currently developing a Unicode keyboard layout for the Ethiopic script using Tavultesosft’s Keyman 5. This will enable the input of Amharic and Tigrigna on a PC running Windows 2000 or Windows XP.

Although it is possible to view a wide variety of writing scripts, support for Unicode input requires Windows 2000 as a minimum. This made the prospect of providing a search function for the site impractical, since no public libraries had adopted the newer operating systems at the time.

The steering committee decided to limit navigation of the site to a browsing model. A search mechanism will be developed for a later version of the site.

Windows text rendering

True type fonts utilise a one to one correspondence between a character in a coded character set and the glyph in a font that represents the character. This model does not work for languages that require complex script processing. A number of scripts supported by Unicode require additional character reordering and glyph processing for correct display of text. This processing is contextual.
The Unicode standard specifically addresses issues relating to text processing. Glyph rendering is handled by the operating system and text processing clients. Complex script rendering on the Windows platforms utilises three components (Hudson, 2000):

- the OpenType font format;
- OpenType Layout services (OTLS) library, and
- Windows Unicode Script Processor (Uniscribe, usp10.dll).

OpenType fonts contain all the information about necessary glyph substitutions and relative glyph positioning. Multiscript fonts can contain specific information in the glyph substitution (GSUB) tables and glyph positioning (GPOS) tables, enabling appropriate layout of each script supported by a particular font. It is also possible to have OpenType layout features associated with specific languages, allowing for typographic variations between languages that share a script.

The OTLS library provides functions that enable text-processing clients to retrieve necessary information from the font that will assist the operating system in rendering the text.


The Devanagari script uses many ligature conjuncts. Figure 1 shows the Hindi word for the Hindi language as it would appear in an application running on Windows 2000 or Windows XP that supported Unicode. The second sample is presented in Unicode logical order without glyph processing.

![Figure 1](image)

Each product (Windows 2000, Windows XP, Internet Explorer 5, Internet Explorer 5.5, and Internet Explorer 6) has different versions of Uniscribe. The following are characteristics of the browsers and operating systems.

- The version that ships with Internet Explorer 5 contains support for the Arabic script. The Windows 2000 version adds Devanagari as well as Tamil.
- Internet Explorer 5.5 contains the same version of Uniscribe as Windows 2000, but does
not automatically update Uniscribe making it necessary to uninstall and reinstall Arabic and Thai language support for Internet Explorer.

- Internet Explorer 6 and Windows XP contain more recent versions of Uniscribe, which provide support for Syriac and additional Indic scripts.
- On the Windows 95/98/NT4 platforms, Uniscribe enables Unicode- and Uniscribe aware programs to view and print Unicode text containing complex scripts supported by Unicode.
- Under Windows 2000 and Windows XP, Uniscribe is required for the input of complex scripts. Input of complex scripts is limited or impossible on Windows 95/98/NT4.

Display of Arabic on Windows 95/98/NT4 is straightforward, and requires the installation of Internet Explorer’s Arabic language pack. Hindi and Tamil are more complex. To successfully display Hindi and Tamil Unicode text, it is necessary to use Internet Explorer 5.5 onwards and have the appropriate version of Uniscribe installed. It is also necessary to install an OpenType font that supports the Devanagari and Tamil scripts.

Currently the only fonts available for displaying Hindi and Tamil Unicode are the Mangal and Latha fonts that come installed in Windows 2000. Due to licensing restrictions it is not possible to copy them and use them on alternative operating systems. The Mangal and Latha fonts do not contain any Latin characters. English language versions of Windows 95/98 require fonts to have the characters necessary for the default code page, in this case Windows-1252.
This would seem to restrict the use of any Hindi or Tamil Unicode web pages to workstations with Windows 2000 or Windows XP installed. To make Hindi and Tamil more widely accessible, we used WEFT3 (Web Embedding Fonts Tool)\(^7\) to generate two embedded OpenType (eot) fonts.

Appropriate entries were then added to the style sheets used by the Open Road directory.

The additions to the style sheet for Hindi were an \texttt{@font-face} declaration indicating the source of the eot font and a class (.hin) to be used for all Hindi text:

```css
@font-face {
  font-family: Mangal;
  font-style: normal;
  font-weight: normal;
  src:
    url(http://home.vicnet.net.au/~andrewc/test/eot/MANGAL0.eot);
}
```

and

```css
.hin {
  font-family: Mangal;
  background-color: white;
  color: black;
}
```

Likewise, for Tamil:

```css
@font-face {
  font-family: Latha;
  font-style: normal;
  font-weight: normal;
  src:
    url(http://home.vicnet.net.au/~andrewc/test/eot/LATHA0.eot);
}
```

and

```css
.hin {
  font-family: Latha;
  background-color: white;
  color: black;
}
```

It is important to note that when search functions are added to the Open Road directory, searching in Hindi and Tamil will require Windows 2000 or Windows XP. This will also be the case for other languages that are only available on Windows 2000/XP platforms.
Language tables

The heart of the Open Road directory is the language table. The Open Road database consists of a range of different tables that store the data required to generate the public directory. The CGI scripts utilise language specific information that is stored in the language tables. The language table is similar in function to localisation resource files used to internationalise software.

The table below (Figure 2) contains data and translations of key terms that are reused by the scripts throughout the site. It contains information about the encoding and text direction to be used, as well as the class to be used from the site’s style sheet. Each language uses a style sheet customised to the language. The style sheets are based on CSS1, and are predominately used for specifying the preferred fonts.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>English Name</td>
<td>Simplified Chinese</td>
</tr>
<tr>
<td>Character encoding</td>
<td>UTF-8</td>
</tr>
<tr>
<td>Text direction</td>
<td>ltr</td>
</tr>
<tr>
<td>Language image</td>
<td>Chinese.jpg</td>
</tr>
<tr>
<td>Instructions image</td>
<td>ChineseResinals.gif</td>
</tr>
<tr>
<td>Go to instructions image</td>
<td>ChineseGoto.jpg</td>
</tr>
<tr>
<td>Native language name</td>
<td>中文简体</td>
</tr>
<tr>
<td>Native OR introduction</td>
<td>通往道路 (Open Road)</td>
</tr>
<tr>
<td></td>
<td>维多利亚州的公共图书馆非英语咨询委员会设立的一个项目。这个项目旨在帮助您附近地区的公共图书馆拥有多种语言的因特网上网设施，通往道路项目设立了这一网站，以帮助您使用您的语言上网“畅游”。</td>
</tr>
<tr>
<td>Welcome</td>
<td>欢迎</td>
</tr>
<tr>
<td>Categories</td>
<td>类别</td>
</tr>
<tr>
<td>Links</td>
<td>链接</td>
</tr>
<tr>
<td>There are no links in this category at present</td>
<td>目前尚没有这一类别</td>
</tr>
<tr>
<td>Language code</td>
<td>zhi</td>
</tr>
<tr>
<td>Class identifier</td>
<td>zhij</td>
</tr>
<tr>
<td>Suspend status</td>
<td></td>
</tr>
</tbody>
</table>

Each record in the table also identifies the language-specific images that are required by the web site. Key terms and translated information is also stored in the records, enabling the Perl scripts to extract and insert the data into the generated web pages.

The use of language tables facilitates the inclusion of additional languages. The CGI/Perl scripts do not need to be modified to accommodate a new language. The first step in adding a new language into the Open Road directory is the addition of a new record to the language table (Figure 3).
The basic structure of the CSS classes for each language is:

```
.zht {
  font-family: MingLiU, PmingLiU, 'Bitstream Cyberbit', 'Cyberbit CyberCJK', 'UWCXMF (Big5)';
  line-height: 150%;
  background-color: white;
  color: black;
}
```

or

```
.pl {
  font-family: 'Times New Roman';
  background-color: white;
  color: black;
}
```
The possibilities and necessities of an embryonic Mark III Open Road

A presentation at an IFLA satellite conference identified the following issues that need to be addressed in any redevelopment of the Open Road’s public web directory:

- Multilingual web accessibility;
- Importing and exporting controlled vocabularies using TMX (an XML DTD for exchanging data between Translation memory software);
- Greater typographic control using CSS3 text module properties;
- Unicode normalisation for data input and search strings;
- Normalisation of output, using a normalisation form suitable for XHTML documents;
- Transparent content negotiation, especially language negotiation;
- Contextual collation based on language;
- More sophisticated use of language control tables and style sheets; and
- Greater use of embedded Open Type fonts for languages using complex scripts, where freely available fonts as not easily available.

These issues incorporate both current and future language technology options, and will form part of the functionality that will be incorporated into the design of the Stage III directory.

Normalisation

One of the functions that should be added to the Open Road directory in a redevelopment of the web site is the ability to search the web site. In order to successfully search a web site it is necessary for the data and the search string to be compatible.

When the Unicode standard was first developed it was necessary that Unicode data could undergo a round trip conversion between Unicode and existing national and international character sets. Each character in those standards was also included in Unicode.

Additionally, each character with a diacritic that exists as a single character can be displayed as a combining character sequence. For example, this would mean that the Unicode character U+00E1 Latin small letter ‘a’ with acute could be represented as U+0061 Latin small letter ‘a’ and U+0301 combining acute accent. The pre-composed U+00E1 is canonically equivalent to U+0061 U+0301, its decomposed form.

The Unicode Technical Committee prefers decomposition. New pre-composed characters will not be added to the standard if a valid combining character sequence can represent the desired character. Existing pre-composed characters were included for round trip compatibility with existing international and national standards when Unicode was initially developed.

Figure 4 illustrates the different ways of representing a Vietnamese string. The character u+1ebf Latin small letter e with circumflex and acute could also be represented as U+00EA U+0301 or U+0065 U+0302 U+0301.
In order to facilitate the processing of data, it is important that input is consistent. One input method editor for a language could be using pre-composed characters, while another could be using combining character sequences. Having both pre-composed and decomposed characters in the data and search strings will have a negative impact on the end user’s chances of locating resources.

The Unicode Standard has specified a set of normalization forms. A redevelopment of the Open Road directory would need to include the normalization of data being entered into the database and the normalization of search strings.

The most commonly supported normalization form is Normalization Form C (NFC). NFC entails a process of canonical decomposition followed by canonical composition. It is also the recommended normalization form to use with web pages.

Collation

An existing problem with the Open Road directory is the lack of sorting of categories and links. The results pages that are generated are unsorted. It is possible to set a locale in Perl and use that locale for appropriate sorting. Unfortunately, some of the languages implemented in Open Road do not contain corresponding locales.

It will be necessary to implement sorting based on the Unicode collation algorithm, an algorithm that takes an input string and uses a collation element table (character mapping data) and produces a sort key. The sort keys for different strings are compared. Collation can be used for sorting, loose matching and string searching.

For the Latin script, the Unicode collation algorithm contains at least three customisable levels:

- alphabetic ordering
- diacritic ordering
- case ordering.

The mapping of characters to collation elements may not be a one-to-one mapping. A fourth level can be added, if the first three levels are insufficient to distinguish between two strings.
Transparent Content Negotiation

In the major web browsers it is possible to specify a list of preferred languages\textsuperscript{13}. If a web page is available in a range of languages, it is possible to configure a web browser to automatically send the web page matching the browser’s preferred language.

Ideally a multilingual public internet workstation within a public library would have no preferred language, and language negotiation would be a superfluous feature. The default setting for language preferences for the major English web browsers is English, strangely enough.

The ‘no preferred language’ feature would be useful for the home user, allowing the user to automatically receive the Open Road directory in their language. This is a server-based extension of the current CGI scripts that enable a user to link directly to a particular language.

For example, if the user follows a link or uses a bookmark to http://www.openroad.vic.gov.au/ he or she goes to the directory’s home page and have to select the language they wish to access. On the other hand, if they follow a link or use a bookmark to http://www.openroad.vic.gov.au/cgi-bin/index.cgi?id=82, the user will open the Open Road directory directly to the Arabic section. Likewise, http://www.openroad.vic.gov.au/cgi-bin/index.cgi?id=79 will directly access the Greek section of the directory.

Language negotiation will enable the user to go to the section they require based on the browser’s preferences. If the browser’s preferred language is English, the optimal solution would be to direct the browser to the directories home page, allowing the user to select a language.

Typographic features

There is a range of CSS attributes in the CSS2 recommendation and the CSS3 Text module draft that need to be incorporated into a redevelopment of the Open Road directory. These attributes improve style sheet support for international layout. The following attributes should be investigated:

- \textit{unicode-bidi} and \textit{direction}. Embed text direction in the style declaration for each language.
- \textit{writing-mode} support would allow the exploration of vertical or horizontal display of CJK text in recent versions of Internet Explorer.
- Internet Explorer 5.5 introduced \textit{text-justify} to provide greater flexibility in support for justification in an international context. It provides newspaper, Kashida, and ideographic justification options to the default inter-word spacing standard in European languages.
- \textit{text-kashida-space} allows the web designer to control the percentage use of Kashida and inter-word spacing used to justify Arabic and Syriac script text.
- \textit{text-autospace} allows control of spacing in CJK text.
Conclusions

We cannot make any assumptions about end-users’ PC configuration. In order to enable the end user to successfully access the site, the web directory provides simple instructions on the configuration of their web browsers. Instructions are provided in the target languages.

Public libraries tend to use older operating systems and older web browsers. The main operating systems in use in libraries are also those operating systems that have more limited language capabilities. As the range of languages provided in the Open Road directory increases, it will be important to find language solutions that are widely accessible.

At the same time, it is must be recognized that choices in browsers and operating systems will have a major impact on the range of languages that a workstation will be able to support.
Bibliography


Endnotes

11 The Open Road project’s URL is http://www.openroad.net.au/

2 MCL-net’s URL is http://www.openroad.net.au/mcl/

3 Those government departments and agencies that do provide non-English language content on their web sites predominately use a resource level access model. This type of access model is based on external agencies providing mediated access to the information resources. The non-English language resources on these sites require an advanced level of English language literacy to access.


5 *Perl, Unicode and i18N FAQ*, http://rf.net/~james/peri8n.html


10 Additionally, it will be necessary to normalize existing data in the database.


13 This list can be weighted.