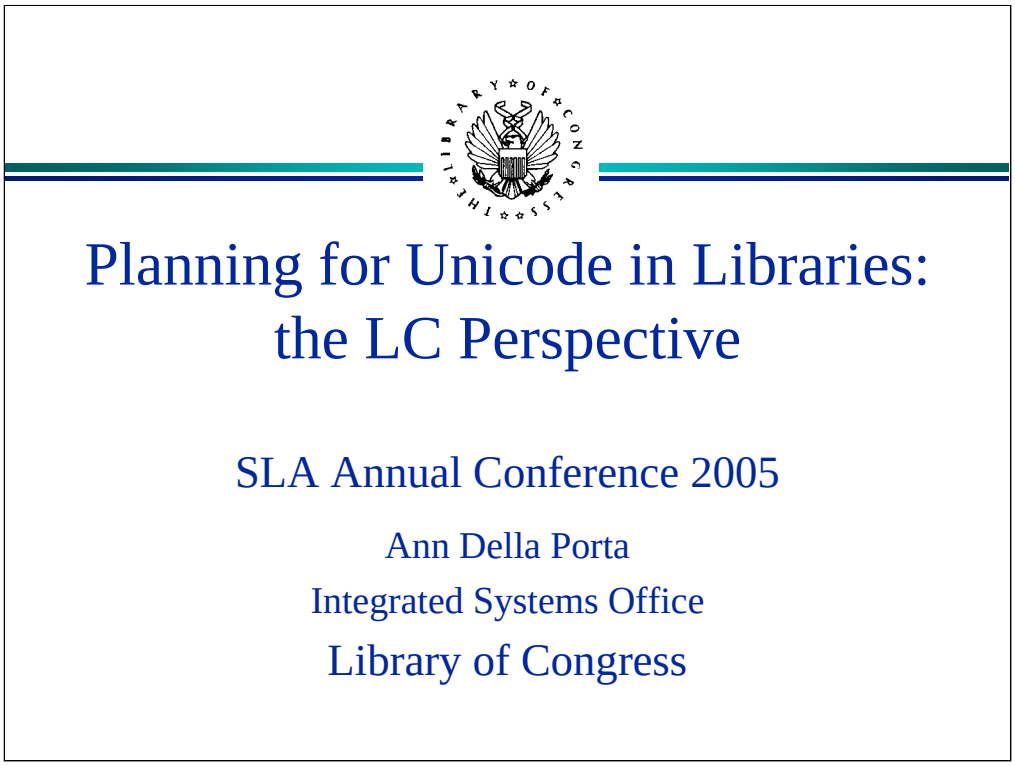
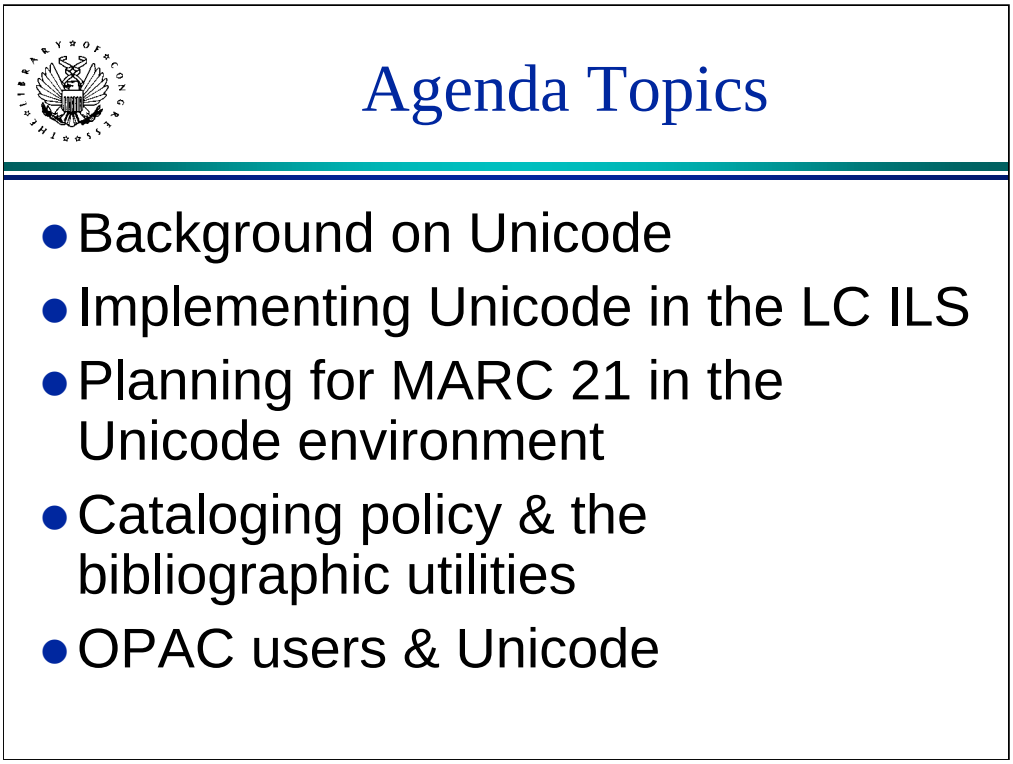


Good morning. Thank you for inviting me to speak with you today. This is my first time at SLA and I'm really pleased to be here. I'd also like to thank Foster Zhang for his advice in developing a presentation for this session. What I'd like to do today is describe what LC has been doing to plan for conversion to Unicode and share with you some the issues we've identified in testing and implementing Unicode-compliant systems. I actually have some answers and suggestions, but I also have some questions that I think need to be considered by librarians.



This is my agenda for today:

First, a bit of background on Unicode, with a focus on libraries and their users;

then, LC's planning for Unicode conversion in our integrated library system and

an update on the MARC standard and Unicode;

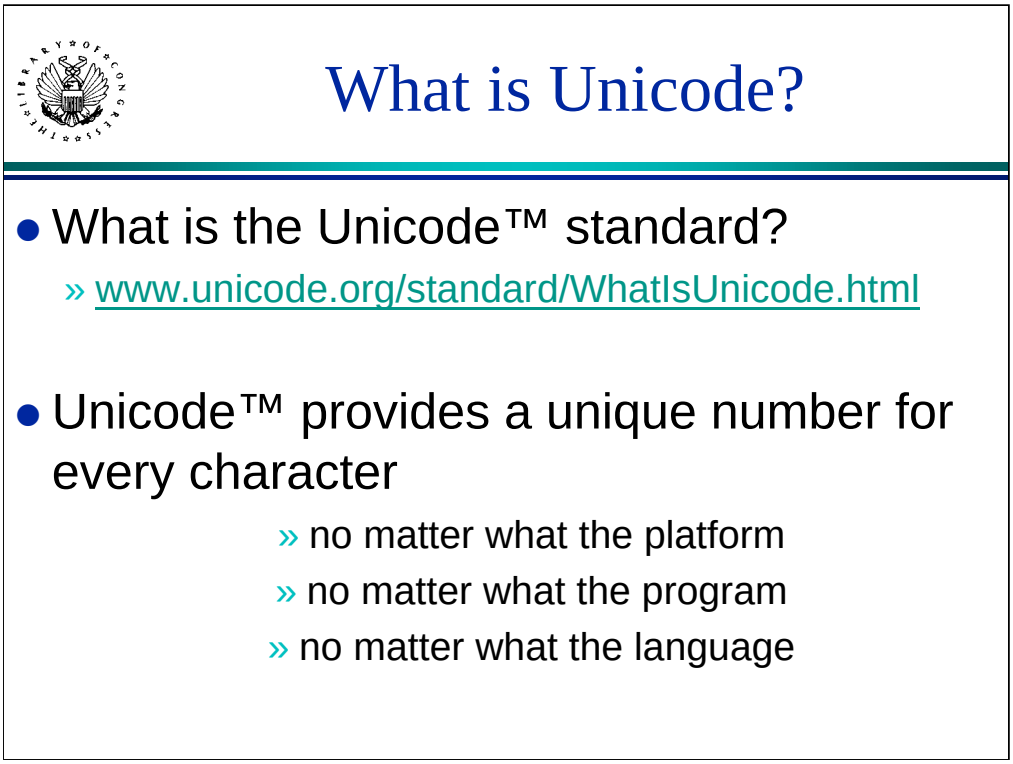
I'll discuss cataloging policy issues and planning with the bibliographic utilities for the Unicode environment; and

I have a few words about OPAC users and plans to support them.

Background on Unicode

- Unicode Mini-tutorial – Unicode “Lite”
  - What is Unicode?
  - Pre-Unicode Environment
  - Unicode Environment
  - Basic Design Features and Syntax
  - MARC-8 and UTF-8
  - Misconceptions

Before I talk about what we’re planning at LC, I’d like to talk about the Unicode standard as it relates to libraries so that we have a common language. I hope this background will help in understanding where we’re coming from and where we’re hoping our migration will take us. Finally, I’d like to clarify some misconceptions I’ve heard about Unicode and our conversion.



The URL on the screen is for the Unicode Consortium's statement: What is Unicode. That site defines the Unicode Standard as a character coding system designed to support the interchange, processing, and display of the written texts of the diverse languages of the modern world. It is a product of The Unicode Consortium. The Unicode Consortium is a group of major computer corporations, software producers, database vendors, research institutions, international agencies, various user groups, and interested individuals.

I took the definition on the bottom of this slide from Mike J. Brown's tutorial on XML and Unicode.

It defines it by stating what Unicode does—

Unicode provides a unique number for every character:

no matter what the platform;

no matter what the program;

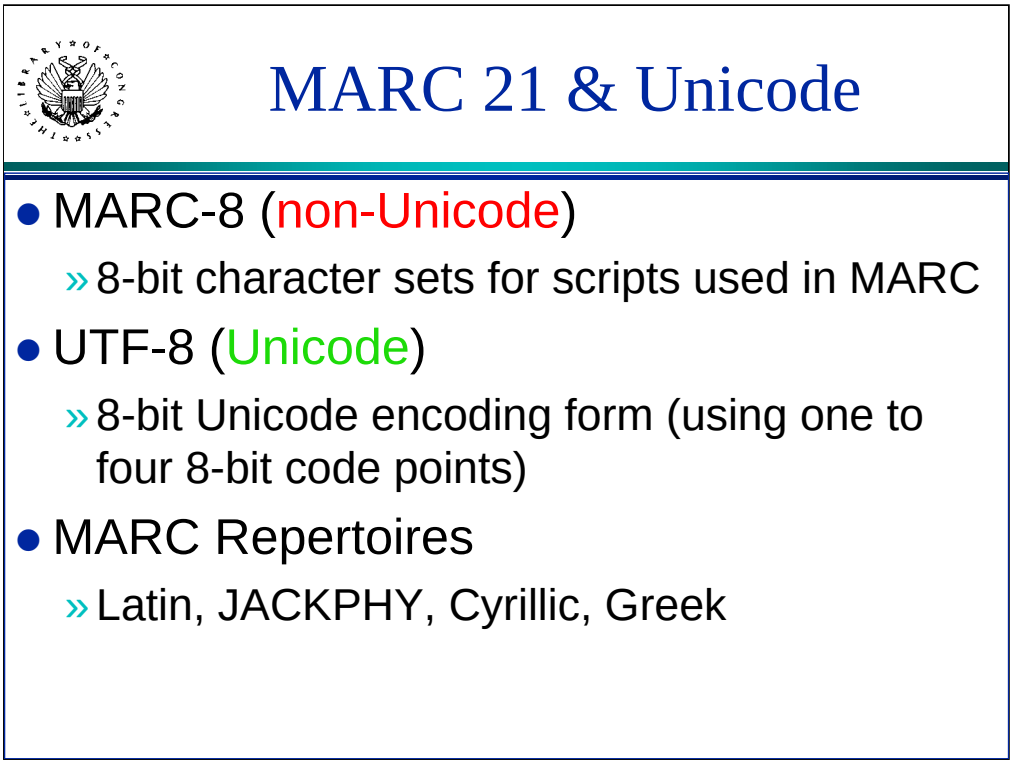
no matter what the language.

From Pre-Unicode (MARC-8)

- Pre-Unicode Environment
  - 8-bit character sets (256 characters max.)
  - Most character sets include ASCII Latin set (128 characters)
  - Variety of characters in remaining 128 (“code pages”)
  - Hundreds of 8-bit character sets in use

Computers at their most basic level just deal with numbers. They store letters, numerals and other characters by assigning a number for each one. In the pre-Unicode environment, we had single 8-bit characters sets, which limited us to 256 characters max. No single encoding could contain enough characters to cover all the languages, so hundreds of different encoding systems were developed for assigning numbers to characters.

As a result, these coding systems conflict with each other. That is, two encodings can use the same number for two different characters or different numbers for the same character. Any given computer needs to support many different encodings; yet whenever data is passed between different encodings or platforms, that data always runs the risk of corruption. That’s important to remember when we talk about MARC-8 and UTF-8.



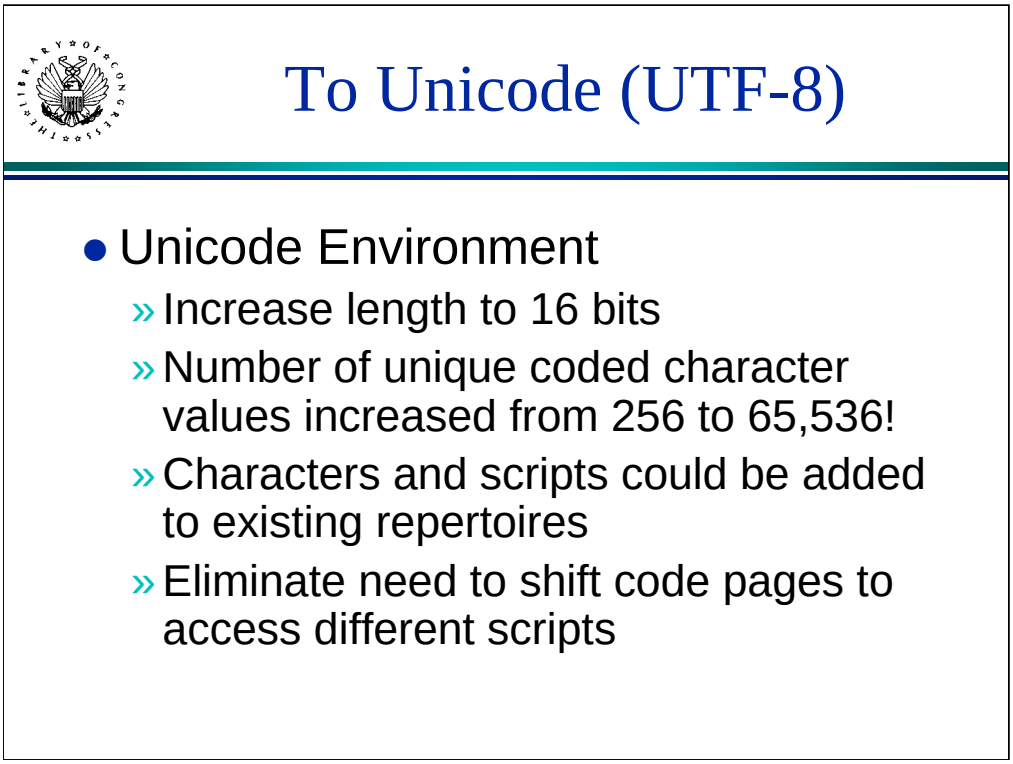
MARC-8 refers to the collection of 8-bit character sets for different scripts used in MARC records up to this point; that is, prior to conversion to Unicode.

UTF-8 refers to the Unicode transformation format which allows any Unicode character to be encoded using a sequence of anywhere from one to four 8-bit code units. (A code point is an abstract numeric value representing a character.)

In LC's database, now-- prior to conversion-- our data are stored in MARC-8.

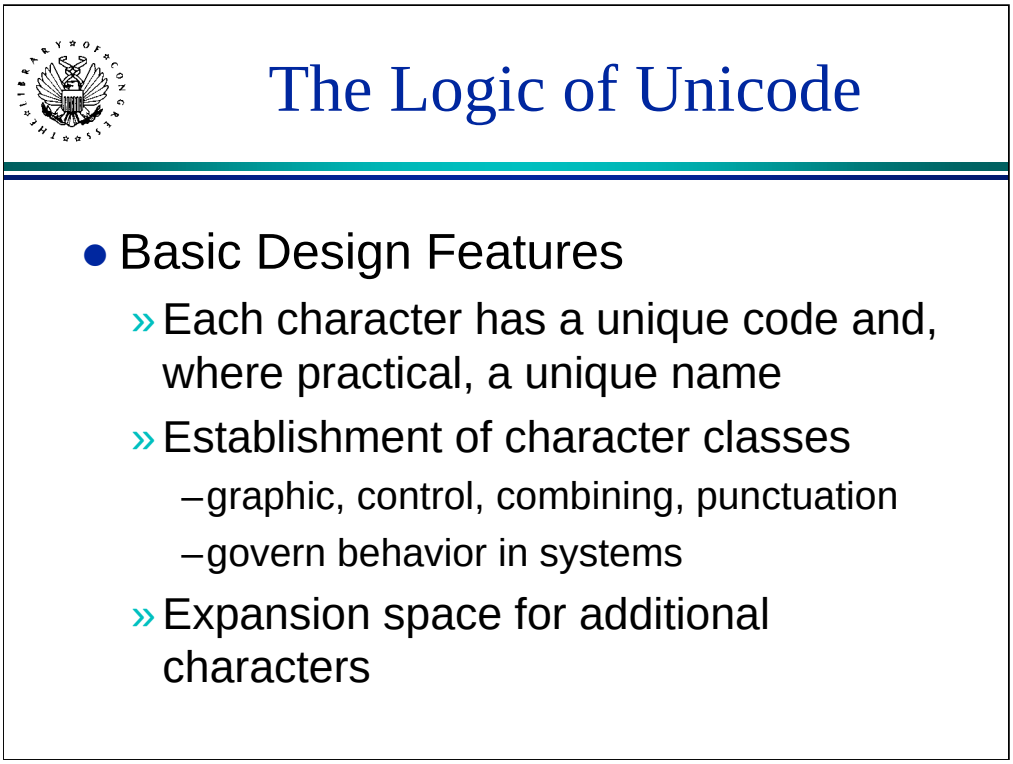
In its implementation of Unicode, Endeavor Information Systems, Inc., has made the decision to store MARC data as UTF-8.

The MARC 21 repertoires are a subset of the Unicode character set. The MARC repertoires include the Latin, Cyrillic, and Greek scripts, as well as the JACKPHY scripts; they do not include Armenian or Thai, for example. At least up to now. I'll talk more about that later.



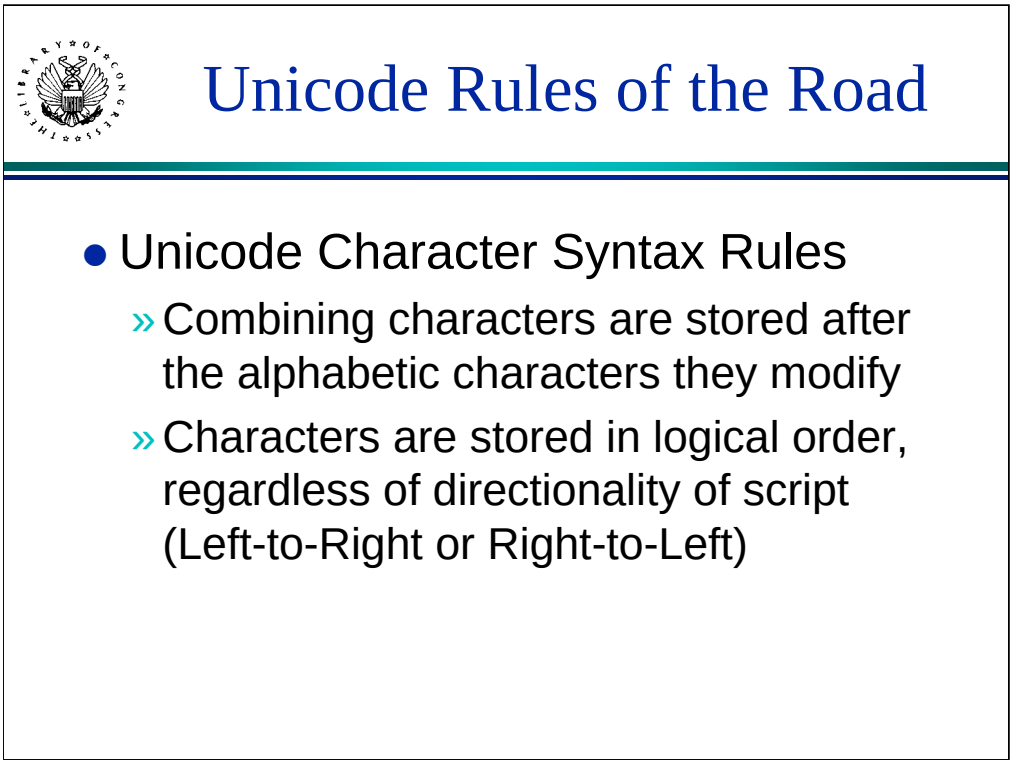
So what is the Problem? The MARC-8 standard is limited with its 8-bit "byte," that is, we're limited in what characters we can use. We also have a need to share bibliographic data more broadly and in more languages and scripts.

The solution is to increase the length of the 8-bit "byte" to allow the encoding of more characters. With Unicode, we increased the length to 16 bits. This allowed the number of unique coded character values to increase from 256 to over 65,000! Badly needed characters and scripts could be added to the existing repertoires, and the need to shift code pages to access different scripts could be eliminated. Everyone could implement a single standard.



In Unicode each character has a unique code and, where practical, a unique name (Han ideographs (Chinese) are not named, because there are too many).

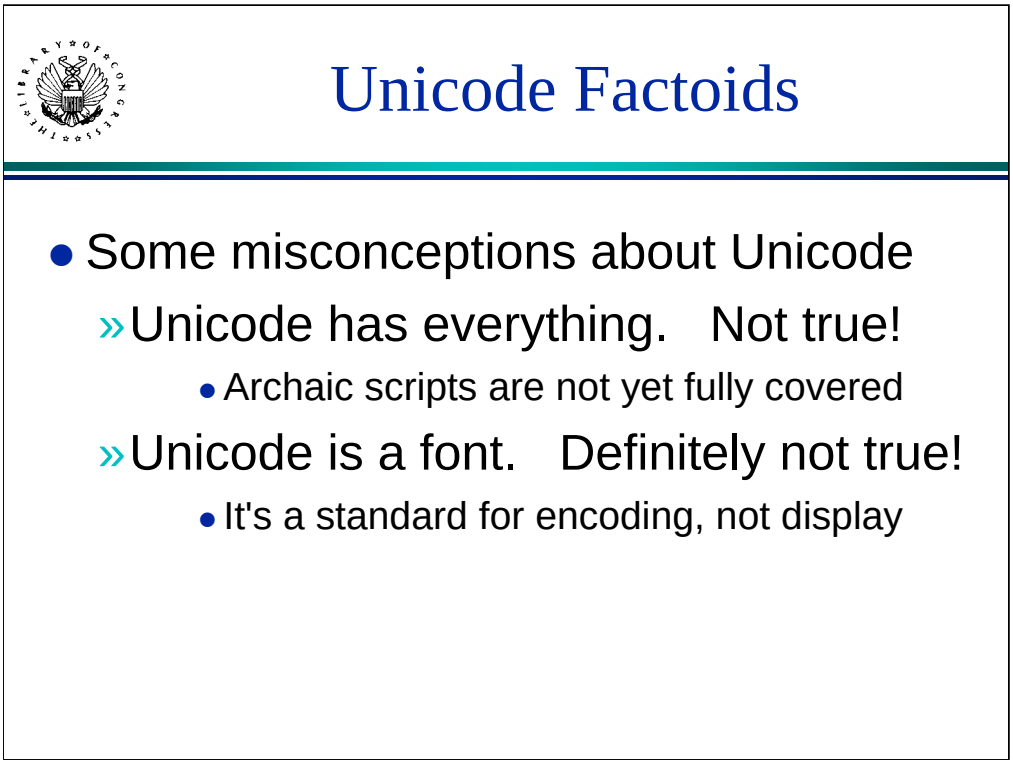
The Unicode standard established these character classes: graphic, control, combining, and punctuation, which govern behavior in systems. It also provides for expansion space for additional characters.



These are two Unicode Character Syntax Rules to keep in mind as we discuss the conversion of library data and the MARC standards and cataloging policy in the Unicode environment:

Combining characters are stored after the alphabetic characters they modify

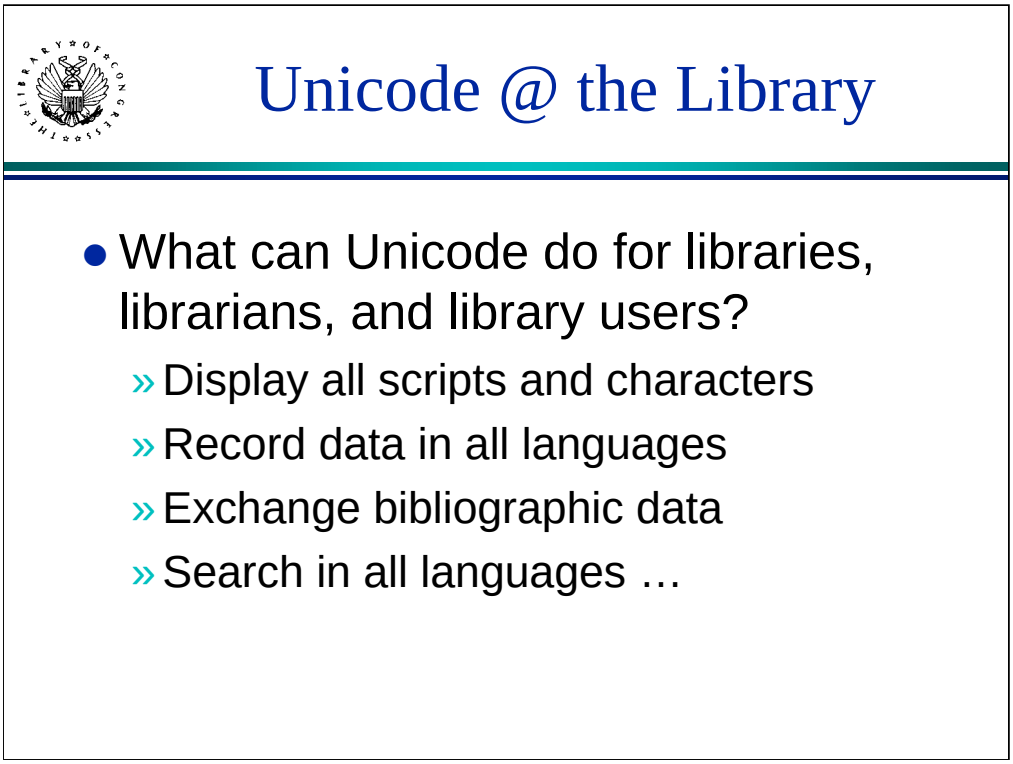
Characters are stored in logical order, regardless of directionality of script (Left-to-Right or Right-to-Left)



A few reminders:

Unicode does not yet encompass all scripts, although the Unicode Consortium is working on it.

Unicode is **not** a font. It's a standard for encoding, not display. Several fonts have been defined for displaying Unicode characters. In testing at LC, we have been viewing our data using Arial Unicode MS. As a federal agency, the Library of Congress is licensed to use the Andale font, which we've also used in testing. Those two fonts have proven to be the most useful so far, but I'll say more about fonts later.



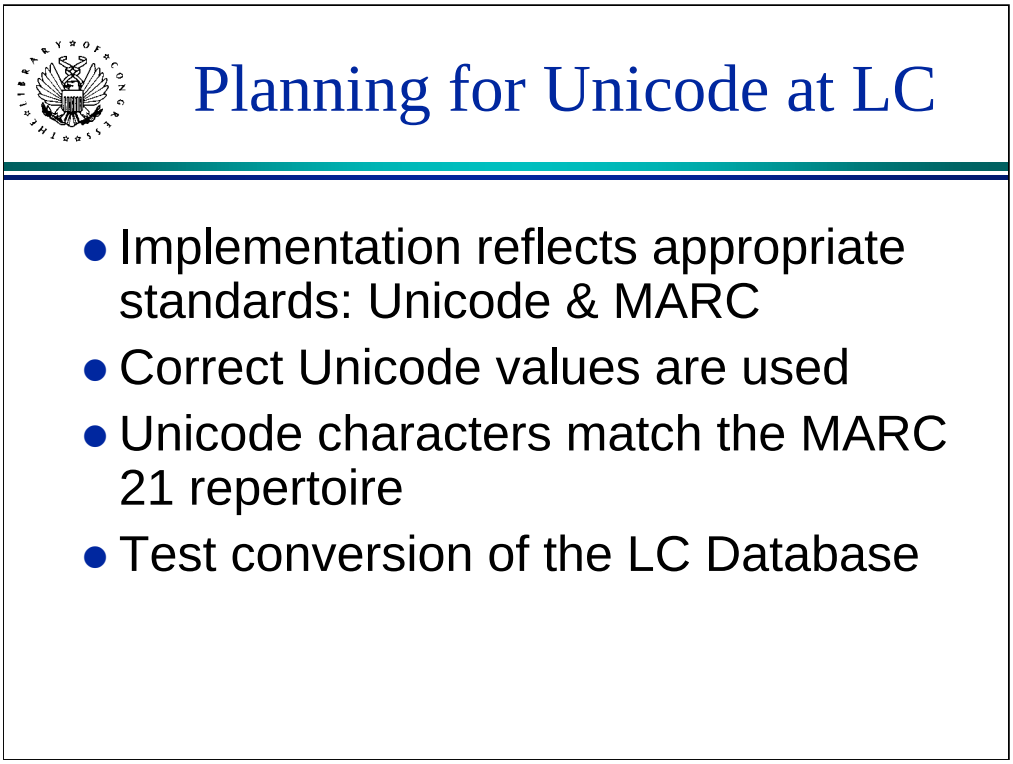
So what can Unicode do for libraries and their users? Finally we have a means to display all the languages represented in our catalogs. The Library of Congress regularly catalogs in over 70 languages on any given day. We have materials in hundreds of languages in our collections. In fact, over half the titles in our catalog are in a language other than English.

For LC, we'll be able to deliver linguistically accurate displays to users of the LC Online Catalog. Our users will see the scripts and characters that they know and read.

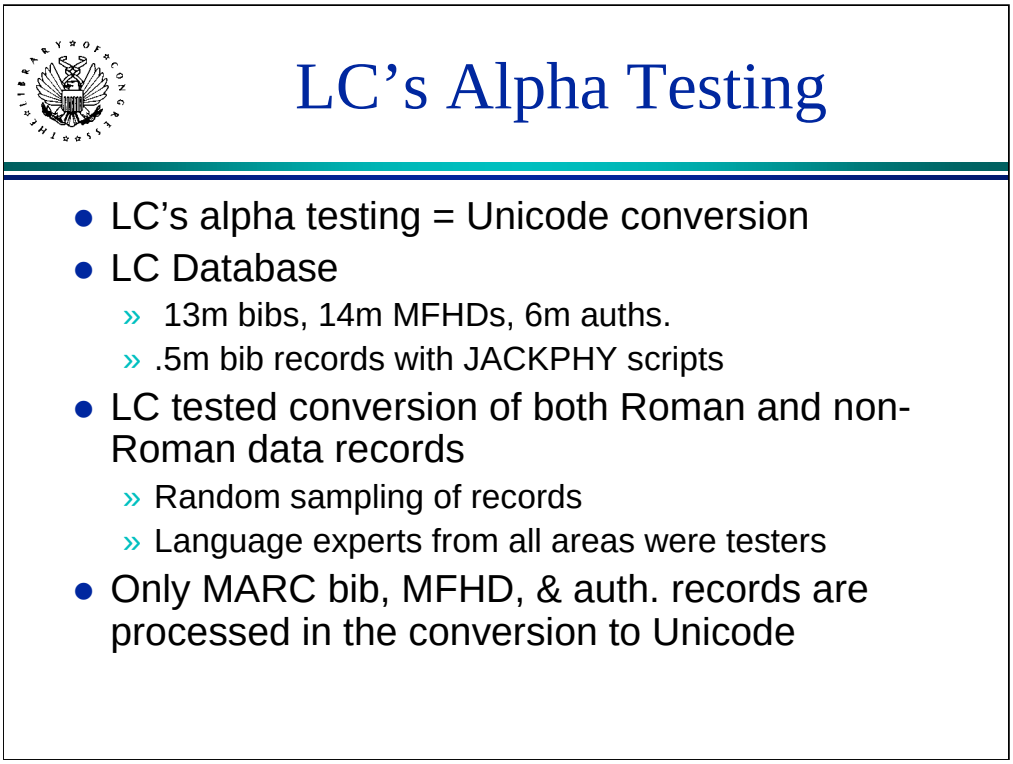
We'll be able to record the scripts and characters that appear in the materials we're cataloging.

We'll be able to exchange data (both import and export) in a standard format used throughout the world.

Our users will be able to search in those scripts and characters, too.



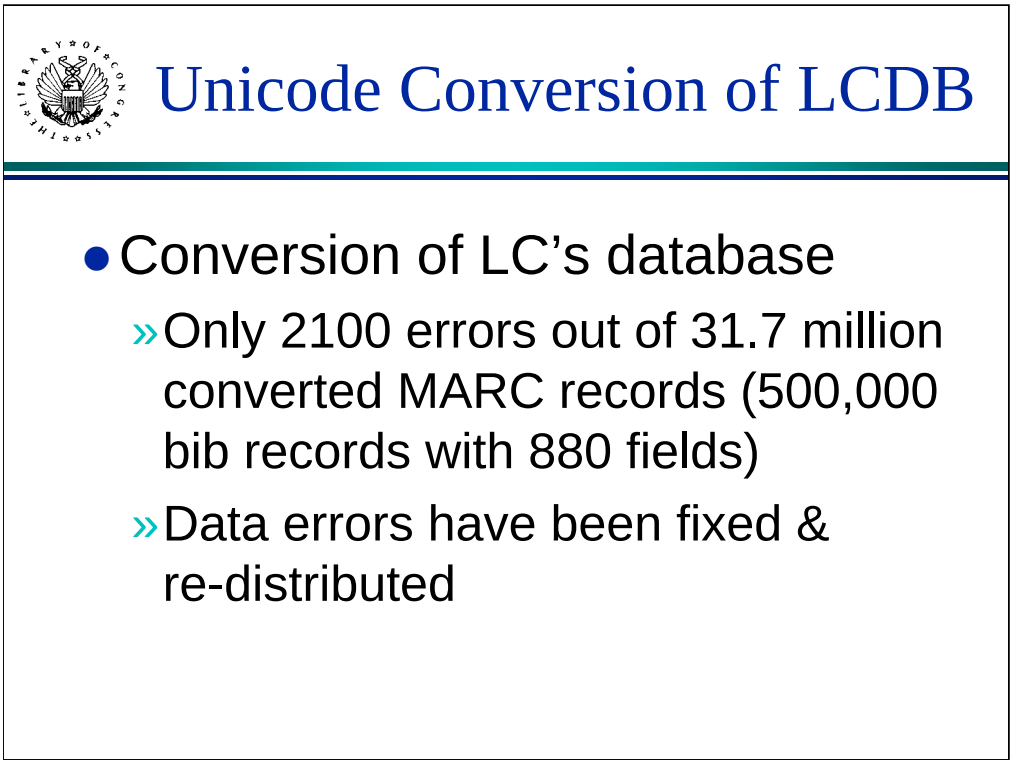
We were pleased that our software vendor, Endeavor, invited LC to participate in testing its Unicode release. We viewed our role as helping to ensure that the implementation reflects the appropriate standards. From a Unicode point of view it meant that the correct Unicode values were used. And from a MARC point of view it meant that the Unicode characters used match the MARC 21 repertoire. We also think that conversion of LC's database was an excellent environment for testing.



When Endeavor first approached us about testing Unicode conversion, we quickly agreed. We thought that conversion of a copy of LC's production database, with 13 million bibliographic records, almost 14 million holdings records, and over 6 million authority records would be an excellent test. We have over half a million bib records with non-Roman data in the JACKPHY scripts, but in our testing we really focused on the conversion of all MARC records. We wanted to be sure that our Roman data were converted correctly, including Vietnamese, Indic languages and scripts that we routinely romanize, like Russian.

The Voyager with Unicode Release converts MARC records. Data in bib, holdings, and authority records are the only data that are touched in the conversion. Our testing methodology was to select 150 records per language by **random sample** from about 500,000 non-Roman script records. We found that this random sampling technique worked well and we found a lot of interesting stuff.

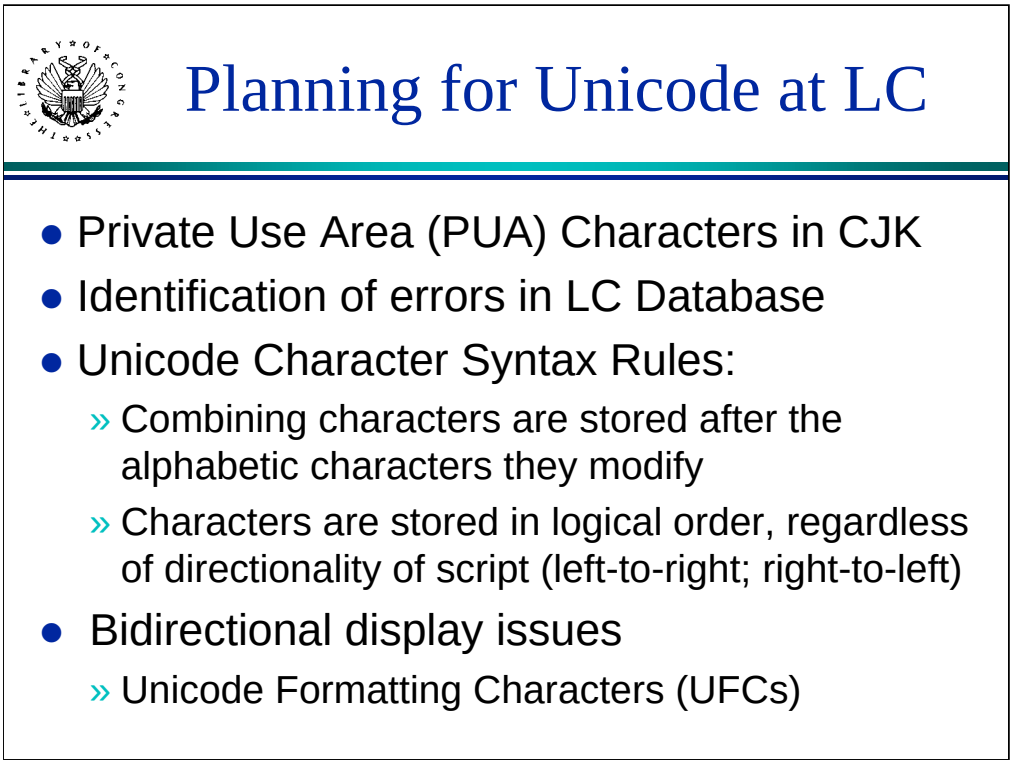
LC's language specialists from public services and technical services were part of the team of testers. They looked at records in the JACKPHY languages, as well as Vietnamese, Hindi, Sanskrit, and Cyrillic records. Each tester was provided with printouts of records from RLIN or OCLC with non-Roman script for comparison as they reviewed each record in the converted database. LC technical experts also reviewed the structure of the converted MARC records, the error logs, and data import and export.



We were pleased with what we saw in our test conversions. Out of the 31 million MARC records in our database, we only had about 2100 conversion errors the first time our database was converted. That's less than .01% error rate. Endeavor delivered conversion logs that identified both error conditions and warning conditions. We used the logs to identify conversion problems and reported the conversion errors to Endeavor.

The logs also pointed out many of our own data errors. Yes, the Library of Congress has errors in its data. We've been correcting those errors and re-distributing the records. As a result, we've seen fewer errors in the subsequent test conversions of our database.

So what are those errors? Some are invalid characters, things like carriage returns or tabs that are not in the MARC 21 repertoire or valid Unicode characters. Those are being fixed. The Cataloging Distribution Service is re-distributing all the records that we correct. Just a note to those of you who are contemplating this conversion-- We've found that many of the tabs and hard returns were introduced into records by catalogers using copy and paste.



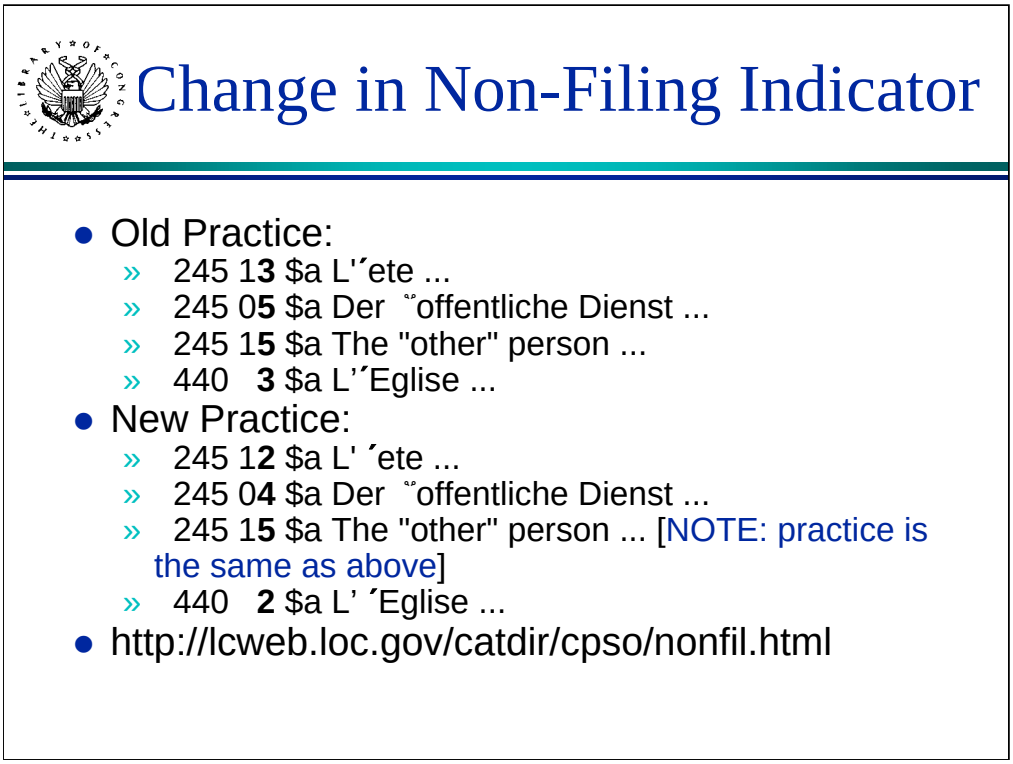
Based on our random sampling and conversion logs that we received, we found that approximately 10% of our CJK records had Private Use Area characters. What are PUAs? Unicode provides for Private Use Area characters to enable local creation and use of certain characters that are not defined in Unicode. A problem arises when sharing Unicode text across areas where PUAs are defined differently at the local level. An example would be a symbol like the Nike swoosh. The Nike Corporation could define a PUA for that symbol and use it throughout its organization, but there are no guarantees when sharing documents or other resources outside the organization, that the symbol would be retained.

Initially librarians identified about 258 CJK PUA characters to preserve their integrity when mapping to the East Asian Character Code (also known as EACC). These PUAs are used in the MARC-8 environment. They are **not** represented in Unicode, but most are similar enough to existing Unicode characters that they could be mapped to them. Early on LC decided not to maintain the use of PUAs in our database, because we could not guarantee the integrity of PUAs across systems. Endeavor programmed the conversion to convert PUAs to the equivalents to which they were similar. LC supplied the list of PUAs and their equivalents for conversion. The few PUAs without equivalents– I think there are about two dozen-- will be converted to the geta. I would describe the geta as two short horizontal strokes. It tells the reader that there is a character to be read, but it could not be rendered. This was one of the first policy decisions that we made. And the good news is that RLG followed that decision in its conversion and I expect OCLC will do the same.

A few important points for those of us who are charged with maintaining library databases: In Unicode, combining characters are stored after the alphabetic characters they modify. Translation: diacritics come after the characters they modify. That's part of the Unicode conversion process. That's going to be part of the learning curve for catalogers.

I mentioned before that characters are stored in *logical* order. I've found it helpful to think of characters being stored from front to back and *not* to think of the data being in right-to-left or left-to-right strings. This will be important when we look at bi-directionality in scripts. One of the benefits of having a full conversion of LC's database was the ability to see how bi-directionality affects the display of right-to-left scripts together with left-to-right scripts in MARC records. Especially because our cataloging rules require that we interpose left-to-right and right-to-left scripts in bib records. Think of the 260 field for place of publication, publisher and date– often those pieces of information are in different scripts.

RLG has addressed the bi-directional display issue by using the Unicode formatting characters. The UFCs force punctuation and surrounding characters to display correctly in mixed right-to-left and left-to-right text. The bad news is that the UFCs are not in MARC-8. We're still grappling with this issue and examining how RLG is using the UFCs in RLIN21. At some point we may get all our HAPY records from RLG in UTF-8 with the UFCs so that we can get the proper displays in our catalog. Today I have no concrete decision for you about this aspect of bi-directionality.



I just said that combining characters are stored after the characters they modify. That means the letter E with an accent grave will display with the diacritic correctly "floating" over the E, but in the data will be stored with the grave following the letter E.

In January 2003 LC changed its practice for counting non-filing characters in MARC 21 records when definite and indefinite articles are present and the first filing word following the article begins with a diacritic. Under the new practice, the non-filing zone does not include any diacritics associated with the first filing character, but does include any diacritics associated with the definite or indefinite article. It also includes spaces and other alphanumeric characters that precede the first filing character.

This new practice will facilitate our move to Unicode where the diacritics will follow the characters they modify. This slide has some examples of what I'm talking about, along with the URL for the statement on this policy change from CPSO. We've already fixed records with non-filing indicators and distributed them.

Logic to Identify & Fix Non-Filing Indicators

- Select by language
- List of def. & indef. articles:
  - » http://www.loc.gov/marc/bibliographic/bdapp-e.html
- IF character in position defined in indicator is a diacritic, the count is reduced by 1

Several people have asked us for the logic we used to make the changes in our database:

CDS did the selection from its database for us and delivered files of records by language. Having the records segregated by language allowed us to spot indicators that were totally wrong. For example, there should be no indicator value “5” in English records. By the way, there’s a list of Initial Definite and Indefinite Articles in the MARC Format at this address.

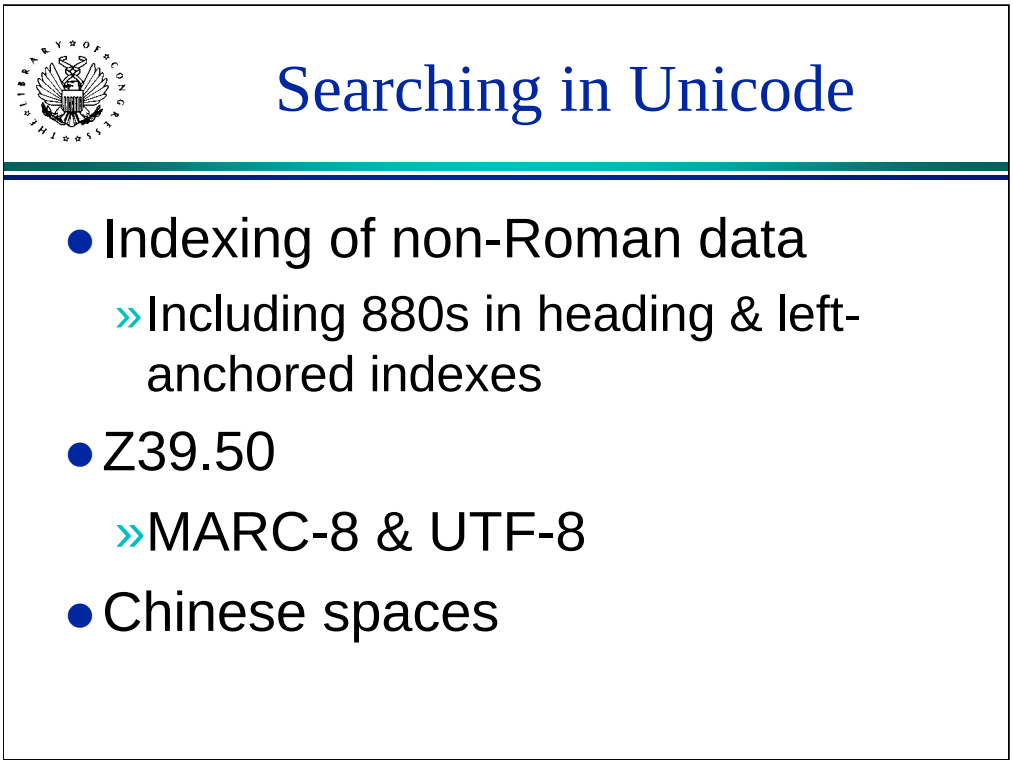
After selecting by language, we selected all records where definite and indefinite initial articles were present and the first filing word following the article in 245 and 440 fields began with a diacritic.

We then looked at the indicator value in 245/440, if the character in that position was a diacritic, the indicator count was reduced by 1 so that the non-filing count ends on a space instead of the diacritic. After conversion to Unicode, the first letter of the title will be correctly displayed.

We used a global change feature to correct the records, which were re-distributed by CDS.

Records changed under this project are being redistributed by CDS via their regular MARC Distribution Service.

[Count the article, diacritics associated with the article, any blank space, an alif, an ayn, or any mark of punctuation preceding the first filing character. Do *not* count a diacritic associated with the first filing character (the alif and ayn are not diacritics, they are special spacing characters not considered for filing).]

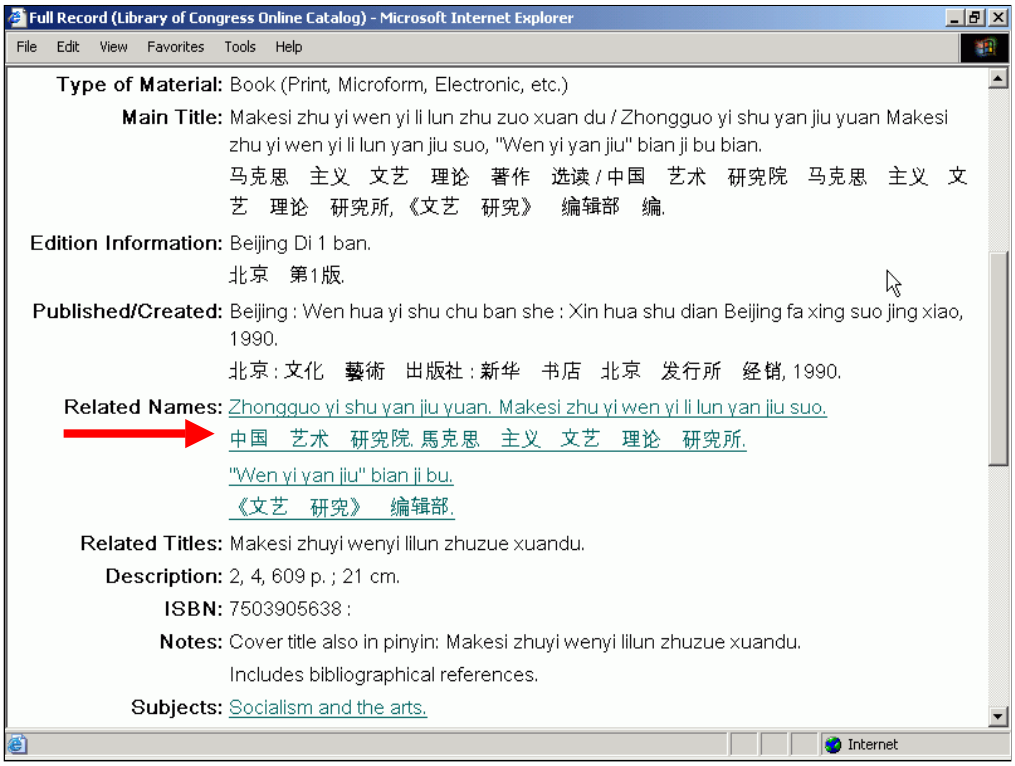


What about the basic functions of an ILS: search and retrieval? We were very pleased to see all non-Roman data indexed in our database, even the 880s are indexed both left-anchored and keyword.

The Library of Congress will offer both MARC-8 and UTF-8 records via Z39.50.

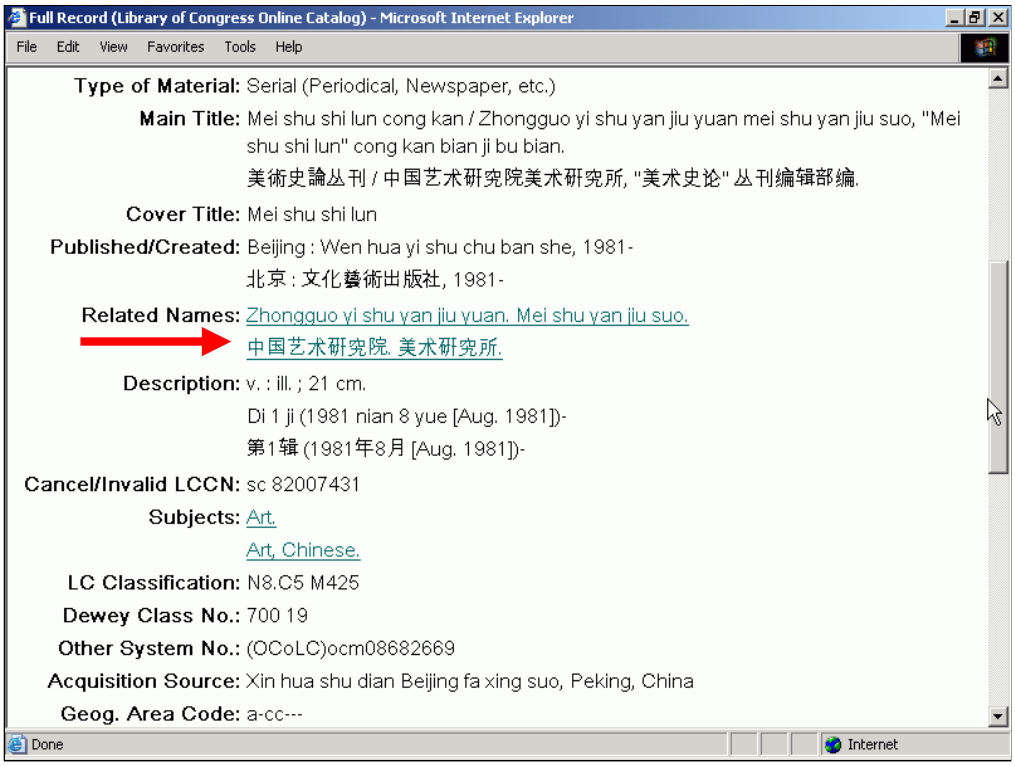
There are some things that only became apparent with the ability to see our records. One of those was spacing in Chinese data. We recommended to our vendor that the Chinese space be treated in the same way a Roman space is treated for indexing purposes.

But beyond that is the expectation for culturally appropriate display for our readers. RLIN requires Chinese data to be input with spaces. This is to enable aggregation in RLIN searching. But this is a tremendous issue for our Chinese patrons, as the spacing is not appropriate to a Chinese reader. Some of you know that our Chinese serials are created in OCLC, as part of CONSER, where spaces are not used between characters.



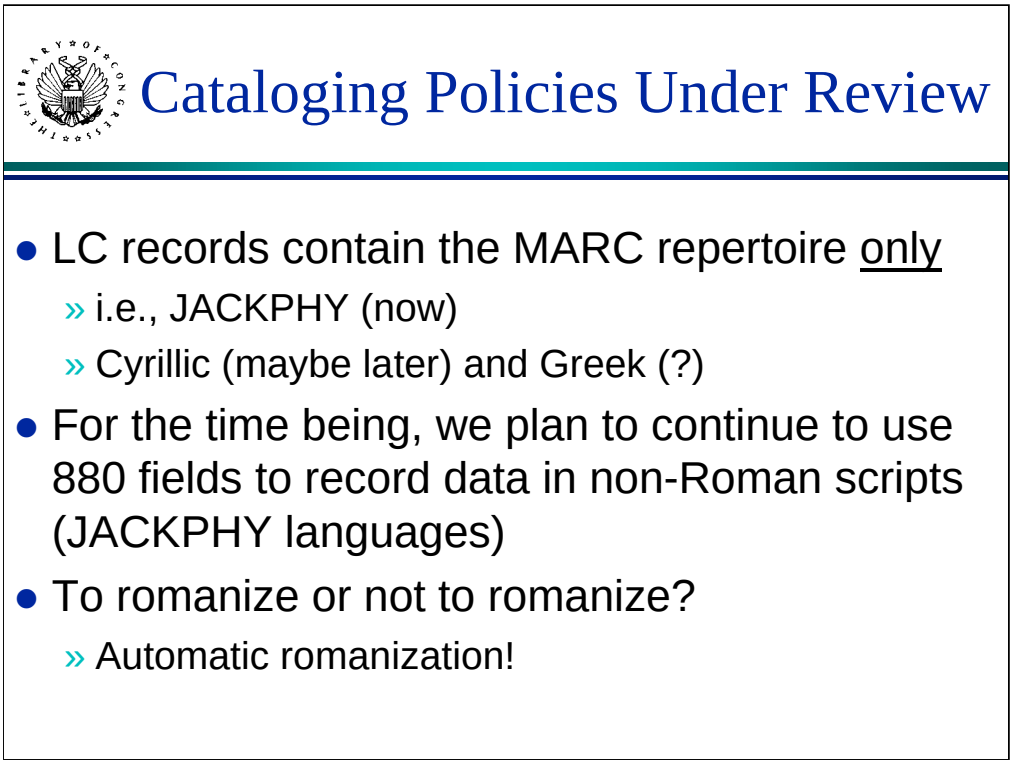
I have a few slides to illustrate ..

This is a record for a Chinese monograph. Even if you're illiterate in Chinese like me, you can see the spaces. Note especially the spacing in that access point.



This is one of our serial records, which was cataloged in OCLC. This is the spacing that Chinese readers find acceptable. And there is an access point, this time without the separating spaces. Think for a minute how this will affect co-location, sorting or filing, and authority control in our databases.

As a matter of fact, when I get back to DC we're having a meeting to discuss whether we can get rid of the spaces in Chinese in RLIN. Of course we'll also discuss this with folks at RLG.



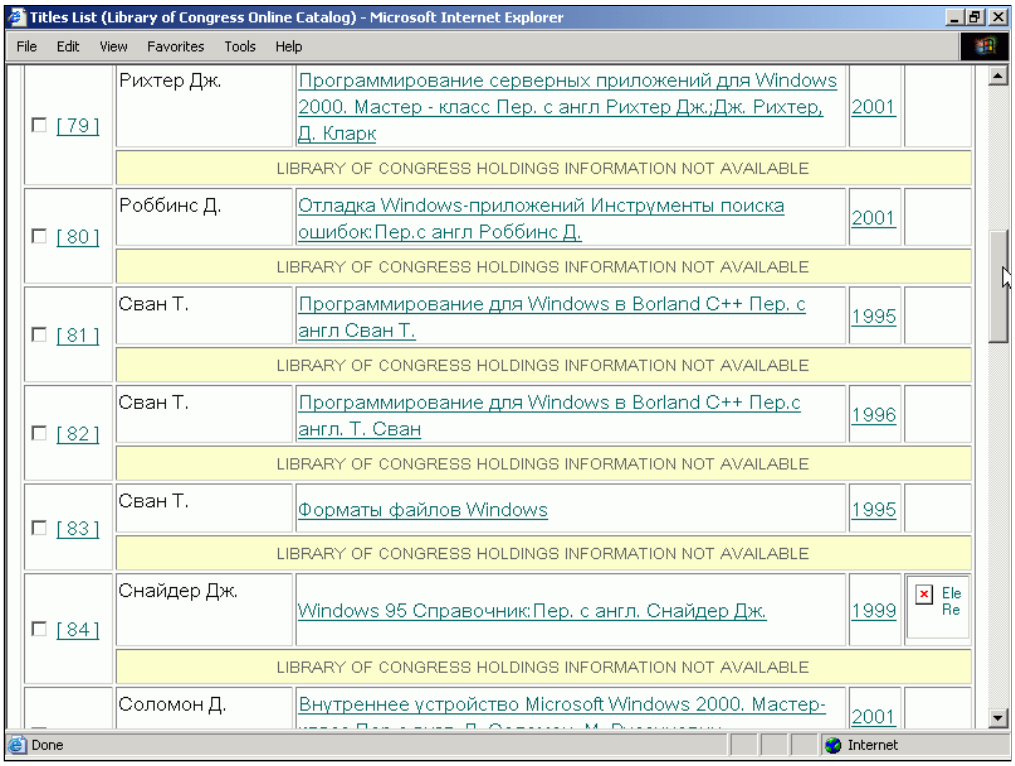
There are many aspects of our cataloging policies that will be reviewed as we plan to implement Unicode. As I said, it's wonderful to have an opportunity to see a conversion of the full LC Database as we undertake our planning for Unicode.

We plan to continue our current JACKPHY workflows immediately after our conversion to Unicode. But the Cataloging Policy and Support Office, headed by Barbara Tillett, has already started to review our cataloging policies and our workflows to see where they might be changed or adjusted to take advantage of our implementation of Unicode. And of course you probably won't be surprised to hear that we're thinking of adding Russian data, that is, in Cyrillic script, in our bibliographic records.

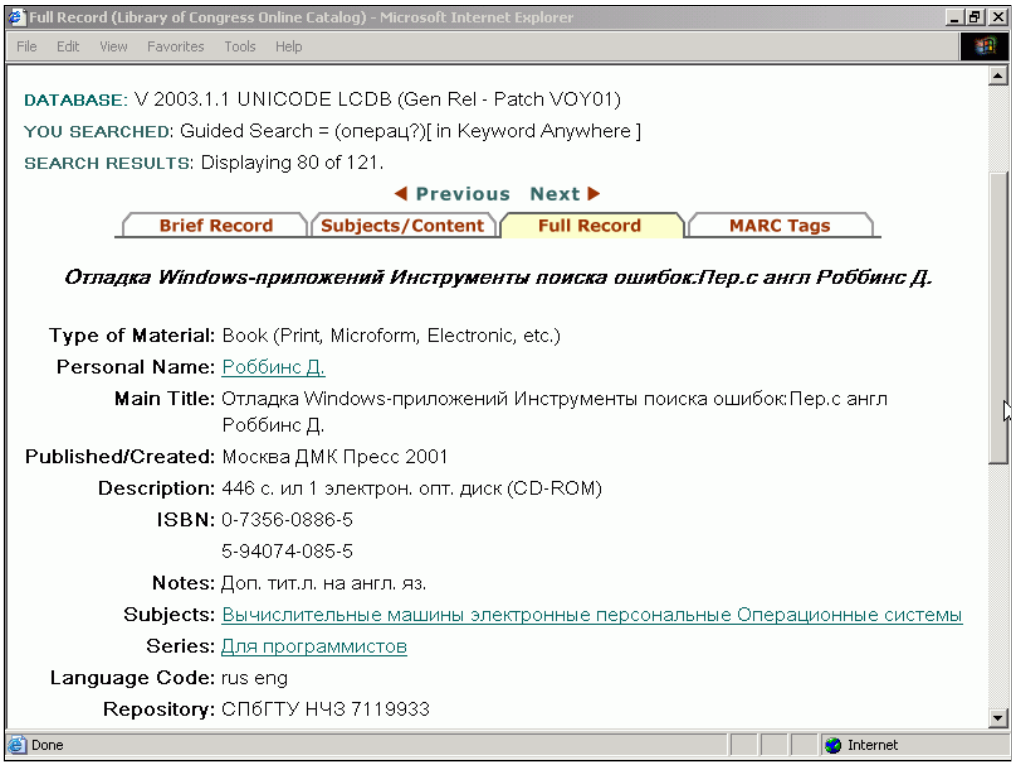
LC records will contain only those Unicode characters that are recognized by the MARC 21 repertoires, that is the Roman and JACKPHY languages, Cyrillic (maybe later), and Greek (a more remote possibility).

For the time being we will continue to record non-Roman scripts in 880 fields (for alternate graphic representations), even though our software will enable us to record non-Roman scripts anywhere in a MARC record. But this will be another decision for librarians– Should we record **only** the scripts on the materials we're providing access to in our catalogs? Should we abandon romanization in favor of the native scripts? At a time when libraries are hard-pressed for resources, can we afford to transcribe both the native scripts and their romanized forms for our users? What about those of us who can't read all those scripts? How are we to manage our catalogs?

Technology may save us here, because automatic transliteration does exist. I know that OCLC is planning to use it for Cyrillic. We need to pursue automatic transliteration functionality with our ILS vendors.

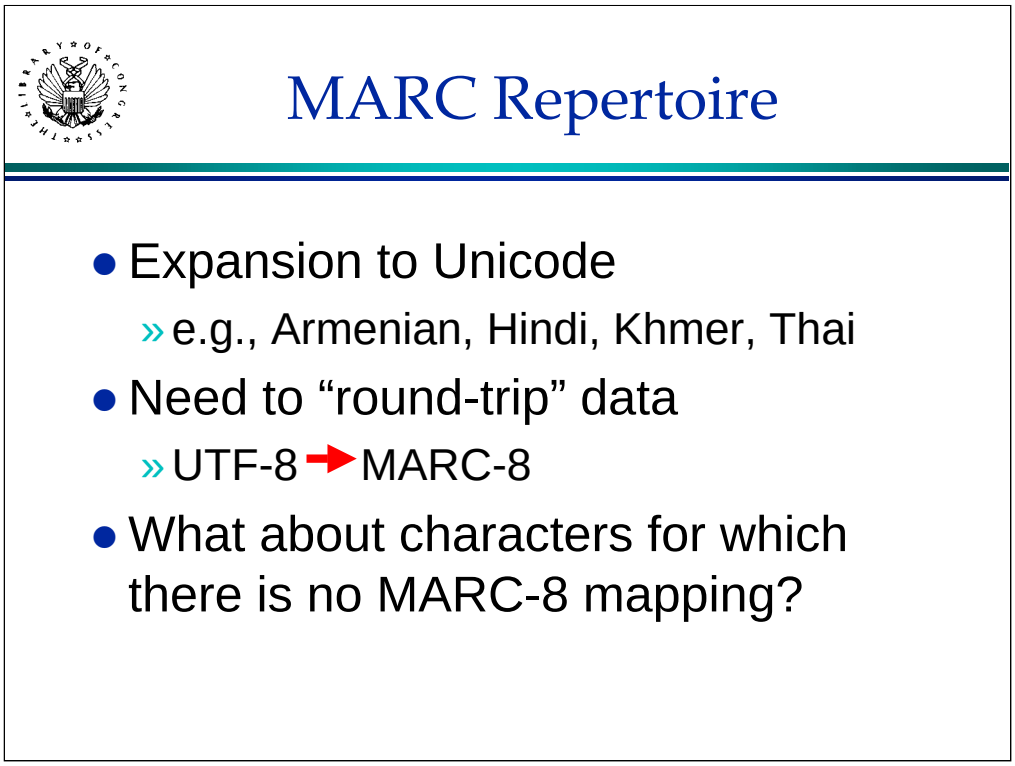


Here's an example for us to think about. This is from our test database, where we loaded a couple of hundred Russian records. That is all Cyrillic (except for the words Microsoft and Windows, which appear in all the titles). These records have no romanization. This is the only title display that a user would see in searching.



Here's one of the records from that list of titles. We could probably all agree that at least the subject access points should be LCSH and therefore in English. And the main entry should be romanized. But what about the series?

This is just an example of the issues that will face us as we seek to take advantage of the abilities that Unicode will provide to us.

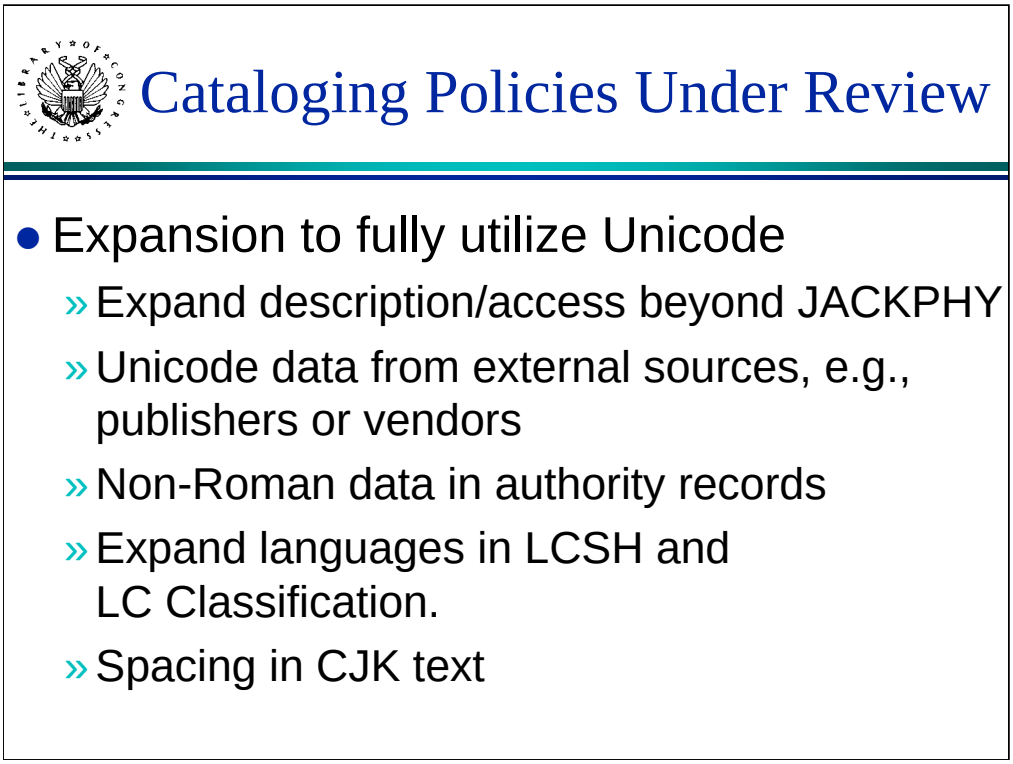


So still talking about expanding our universe …

LC's Network Development & MARC Standards Office has been investigating the expansion of MARC to all of Unicode. Recently I found out that NetDev's position is that MARC already accommodates all of Unicode. That is, any Unicode encoding can be used in a MARC record in UTF-8. Each library will have to make a decision about whether it should limit its data to characters that have MARC-8 mappings. What this means is if we input Thai characters in a MARC record in UTF-8, we will not be able to export a record in MARC-8 with those characters, nor would these characters be delivered via Z39.50 in MARC-8.

For LC, we'll have to take into consideration what our customers need. Any library that has not yet converted to Unicode will still need to get MARC-8 records from the Cataloging Distribution Service. I think the real question will be, how long we'll have to provide MARC-8 records. I'll remind you that LC stopped selling printed cards just 8 years ago.

This is a recent decision, so the Cataloging Policy & Support Office and the Cataloging Distribution Service have just begun to consider the ramifications for our bibliographic data and what steps we will take to provide MARC-8 records.



Here are some other policy issues we're considering.

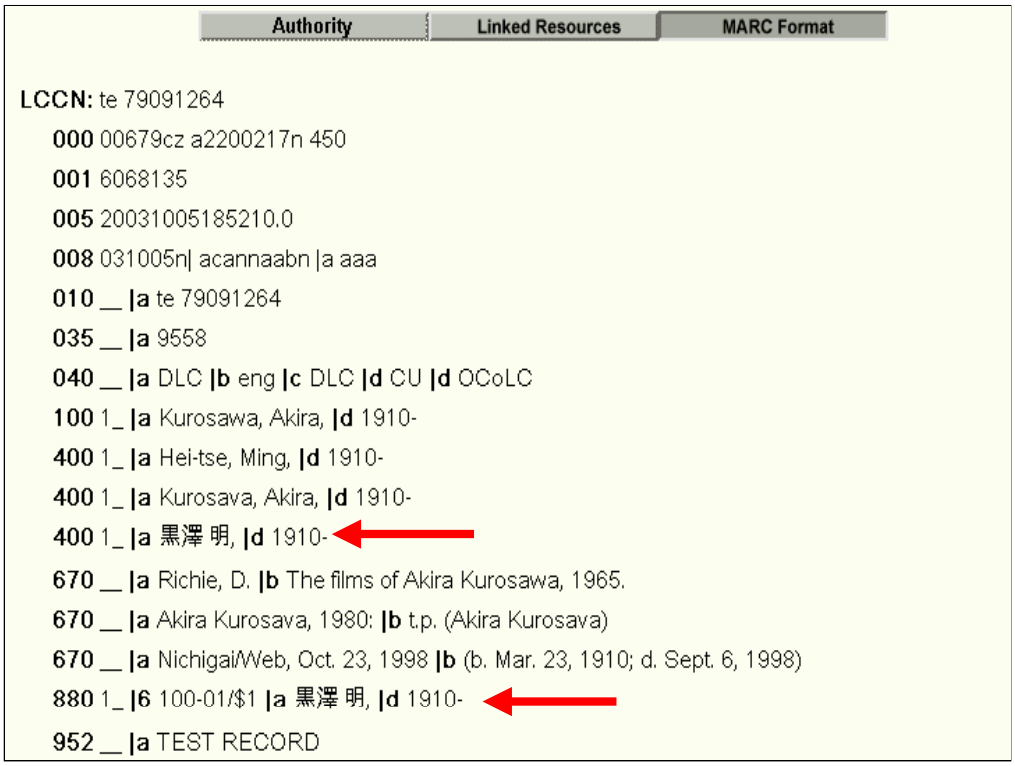
-We'd like to explore possible expansion of bibliographic description and access to languages beyond JACKPHY based on a phased approach. I've already hinted that we may expand to Cyrillic. But we need to consider which languages would be given priority, accommodation for record distribution, staffing and training, etc. And of course there that pesky round-tripping issue for MARC-8.

-We want to determine when and where we'd use non-Roman characters in MARC records imported from other sources, for example from publishers or vendors. There are potential savings there, but what will we sacrifice?

-We'd like to consider expanding the languages that we would accommodate in LCSH and LC Classification. In fact, we're in discussions with Minaret Software, whose product we use for editing and publishing the LC Classification schedules, about our requirements for including non-Roman scripts in that library tool.

-For a long time we've wanted to include non-Roman characters in authority records. We'll need to discuss what fields to use for recording non-Roman data, that is 4XXs or 880s. I'll demonstrate this in a minute. Obviously LC will work with the NACO nodes, RLG, OCLC and the British Library, on this expansion.

-So I thought I'd show you some examples of what I've been talking about

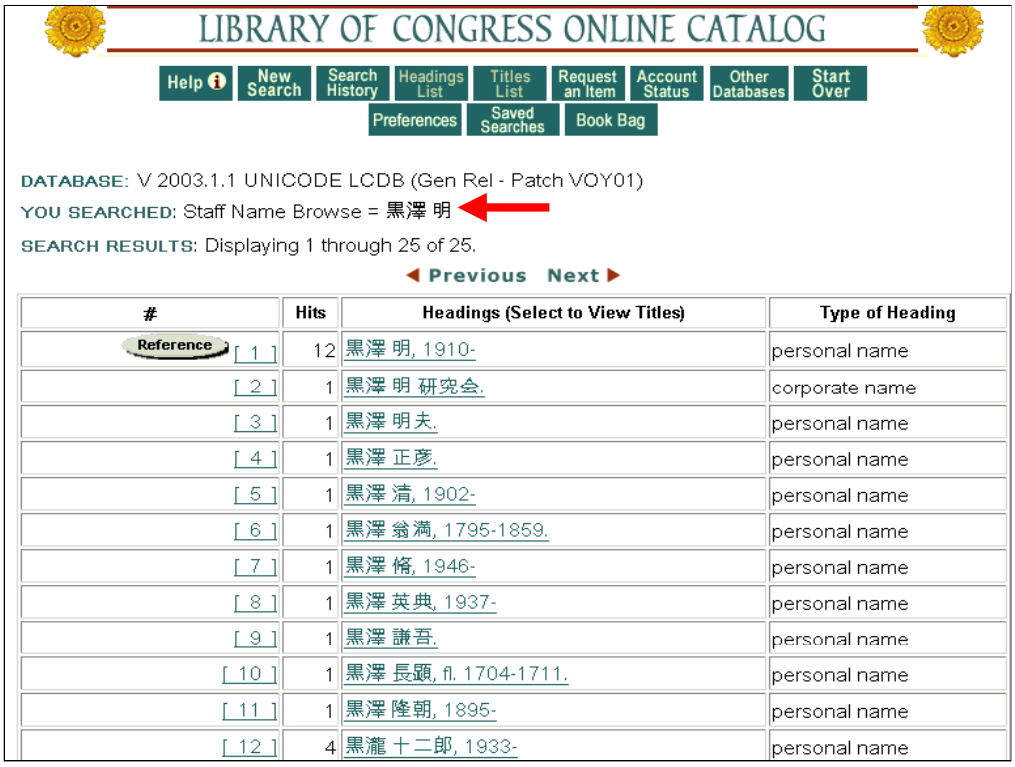


LC has not yet converted to Unicode, so what I'm showing you today is from a full copy of the LC Database that has been upgraded to the Voyager with Unicode release in a test region. Furthermore, the examples I'm using do not reflect current LC policy. They're just experiments in what Unicode functionality can do.

This is a test authority record for Akira Kurosawa. You can see that there are two places where Japanese characters appear– a plain 400 reference and an 880 field that is paired with the 100 heading on the record. This is one of the policy decisions we'll have to make as we decide on how to incorporate non-Roman scripts in authority record. Today there are no non-Roman data in the LC Name Authority File or in LC Subject Headings. This provides us with a clean slate and we have the luxury of deciding how we want to utilize Unicode in our authority data.

This slide shows you what the authority record would look like in the Unicode version of LC Authorities.

Here I'm performing a heading search on the Japanese characters of the director's name.



LIBRARY OF CONGRESS ONLINE CATALOG

Help | New Search | Search History | Headings List | Titles List | Request an Item | Account Status | Other Databases | Start Over

Preferences | Saved Searches | Book Bag

DATABASE: V 2003.1.1 UNICODE LCDB (Gen Rel - Patch VOY01)

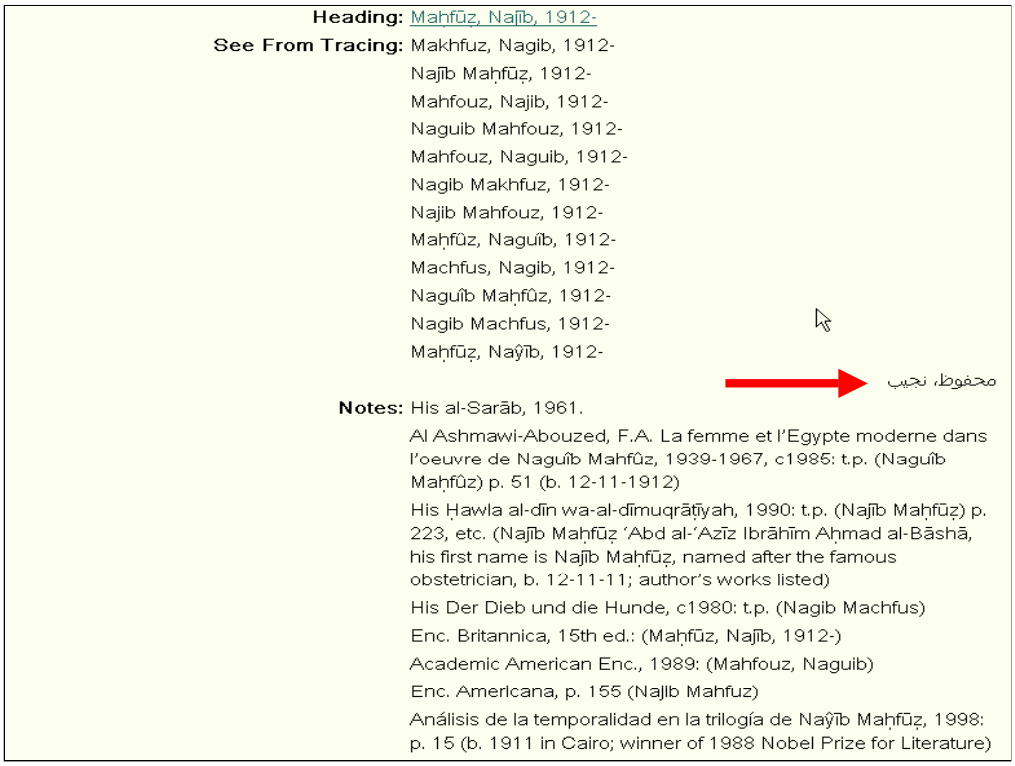
YOU SEARCHED: Staff Name Browse = 黒澤 明

SEARCH RESULTS: Displaying 1 through 25 of 25.

◀ Previous Next ▶

#	Hits	Headings (Select to View Titles)	Type of Heading
Reference [1]	12	黒澤 明, 1910-	personal name
[2]	1	黒澤 明 研究会.	corporate name
[3]	1	黒澤 明夫.	personal name
[4]	1	黒澤 正彦.	personal name
[5]	1	黒澤 清, 1902-	personal name
[6]	1	黒澤 翁満, 1795-1859.	personal name
[7]	1	黒澤 脩, 1946-	personal name
[8]	1	黒澤 英典, 1937-	personal name
[9]	1	黒澤 謙吾.	personal name
[10]	1	黒澤 長顕, fl. 1704-1711.	personal name
[11]	1	黒澤 隆朝, 1895-	personal name
[12]	4	黒瀧 十二郎, 1933-	personal name

And these are my search results. Remember that the Japanese characters appear in 880 fields on bib record. The heading indexes in our database will include the 880 fields that are paired with headings.



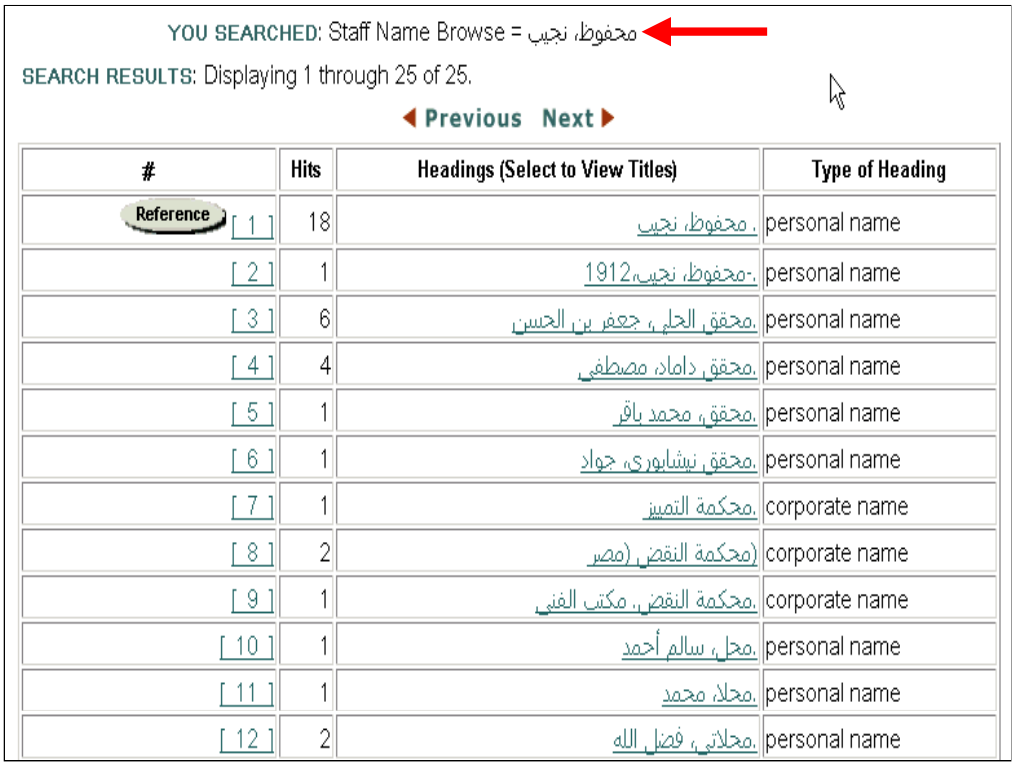
Heading: Maḥfūẓ, Najīb, 1912-

See From Tracing: Makhfuz, Nagib, 1912-
Najīb Maḥfūẓ, 1912-
Mahfouz, Najib, 1912-
Naguib Mahfouz, 1912-
Mahfouz, Naguib, 1912-
Nagib Makhfuz, 1912-
Najib Mahfouz, 1912-
Maḥfûz, Naguíb, 1912-
Machfus, Nagib, 1912-
Naguíb Maḥfûz, 1912-
Nagib Machfus, 1912-
Maḥfūẓ, Naŷīb, 1912-
محفوظ، نجيب

Notes: His al-Sarāb, 1961.
Al Ashmawi-Abouzed, F.A. La femme et l'Egypte moderne dans l'oeuvre de Naguíb Mahfûz, 1939-1967, c1985: t.p. (Naguíb Maḥfûz) p. 51 (b. 12-11-1912)
His Ḥawla al-dīn wa-al-dīmuqrāṭīyah, 1990: t.p. (Najīb Maḥfūẓ) p. 223, etc. (Najīb Maḥfūẓ ʻAbd al-ʻAzīz Ibrāhīm Aḥmad al-Bāshā, his first name is Najīb Maḥfūẓ, named after the famous obstetrician, b. 12-11-11; author's works listed)
His Der Dieb und die Hunde, c1980: t.p. (Nagib Machfus)
Enc. Britannica, 15th ed.: (Maḥfūz, Najīb, 1912-)
Academic American Enc., 1989: (Mahfouz, Naguib)
Enc. Americana, p. 155 (Najib Mahfuz)
Análisis de la temporalidad en la trilogía de Naŷīb Maḥfūẓ, 1998: p. 15 (b. 1911 in Cairo; winner of 1988 Nobel Prize for Literature)

I also want to show a right-to-left language, so here is the OPAC display of the authority record for Najib Mahfuz.

You can see the reference in Arabic script in the authority record.



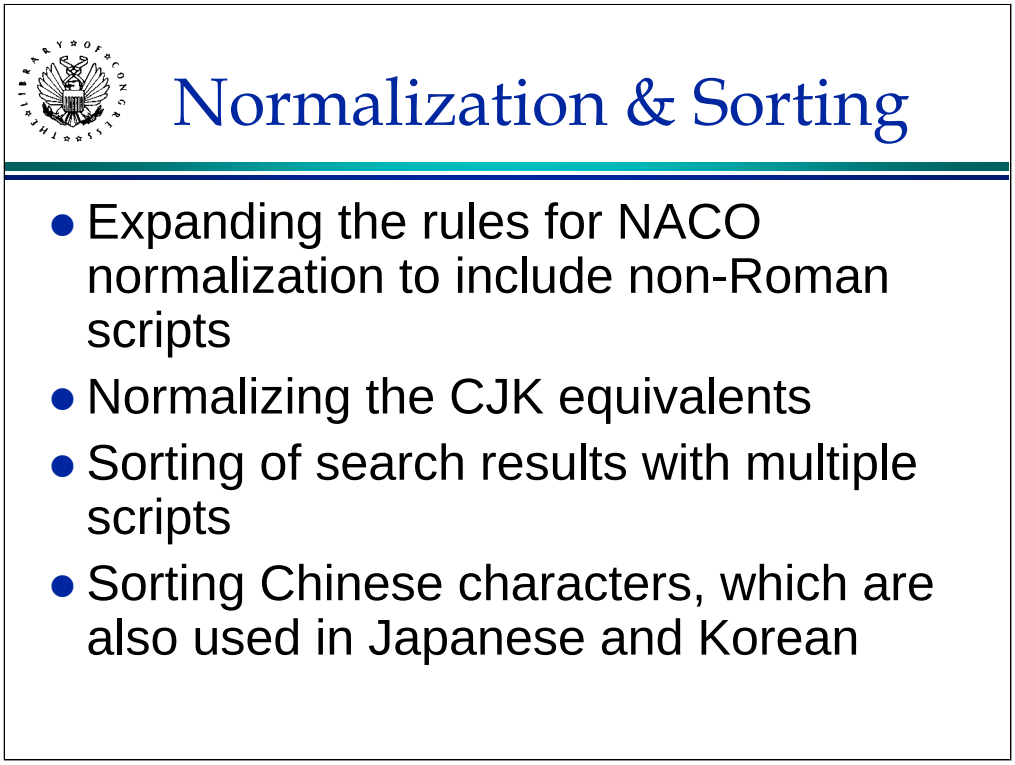
YOU SEARCHED: Staff Name Browse = محفوظ، نجيب

SEARCH RESULTS: Displaying 1 through 25 of 25.

Previous Next

#	Hits	Headings (Select to View Titles)	Type of Heading
Reference [1]	18	محفوظ، نجيب .	personal name
[2]	1	محفوظ، نجيب،1912- .	personal name
[3]	6	محقق الحلي ، جعفر بن الحسن.	personal name
[4]	4	محقق داماد، مصطفى.	personal name
[5]	1	محقق، محمد باقر.	personal name
[6]	1	محقق نيشابوري، جواد.	personal name
[7]	1	محكمة التمييز.	corporate name
[8]	2	(محكمة النقض (مصر	corporate name
[9]	1	محكمة النقض. مكتب الفني.	corporate name
[10]	1	محل، سالم أحمد.	personal name
[11]	1	محلا، محمد.	personal name
[12]	2	محلاتي، فضل الله.	personal name

You can see the headings display, again the headings are coming from 880 fields.



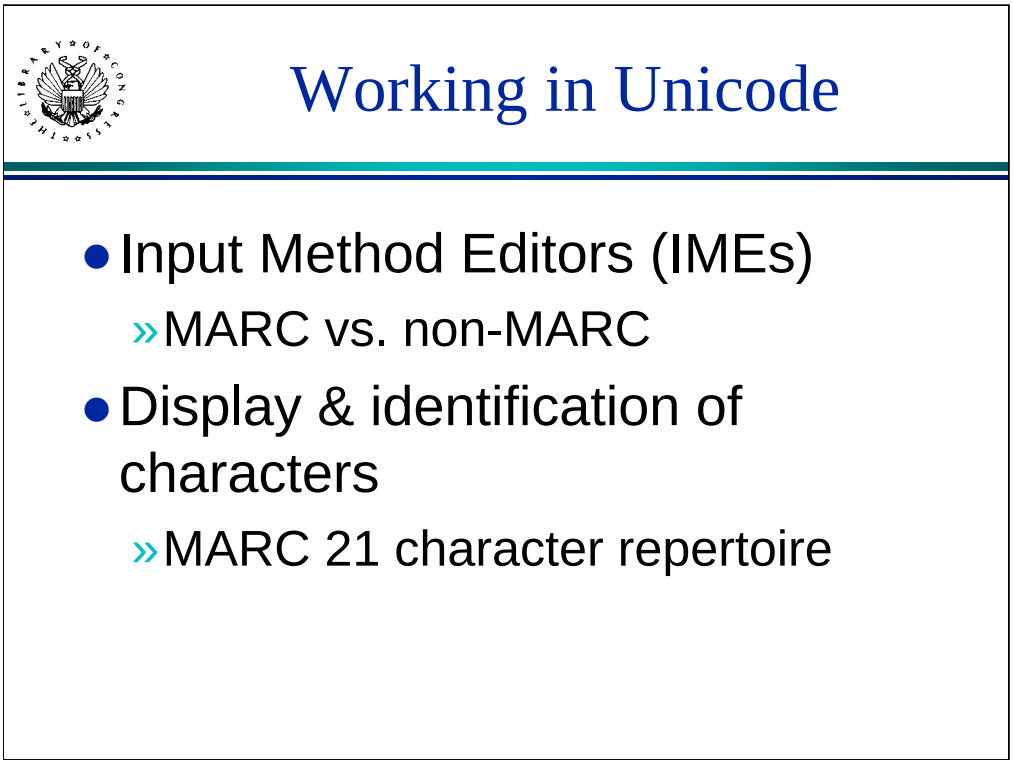
One requirement in adding non-Roman scripts to authority records is to consider the impact on the NACO normalization rules. Remember that normalization, or folding as it's called in Unicode, is a process that takes character strings and simplifies them to eliminate punctuation and diacritics to enable comparison. For example, to compare a 1XX field to a 4XX to make sure that they are unique and don't normalize to the same form. This normalization is a part of the daily exchange of authority data between LC and the utilities as NACO participants and LC staff create and update authority records, to ensure uniqueness of headings and references and accurate matching of headings and see-also references. And remember, spacing is important in normalization, so we have to think about that as we consider spacing in our Chinese data.

One interesting aspect of our testing with both our ILS vendor and RLG is the realization that the standard input method editors or IMEs, enable the keying of CJK characters that are not in the MARC Repertoire and worse, the MARC equivalents of these characters are not always available in the IMEs. It took us a long time to analyze this problem and identify a possible solution.

What we've done is develop a database that maps the non-MARC CJK characters to their MARC equivalents. We're planning to make a Web interface available to staff so that they can search on a non-MARC CJK character to find the correct MARC equivalent. After testing, we hope to make the table of equivalents available publicly. This was a very thorny problem, and it took us several conference calls and meetings with RLG to explain the problem to them. I think they've now endorsed our approach to providing access to the equivalents via the Web. We'll make an announcement when it's available outside LC.

But getting back to normalization, we still need to think about how to get our users from those non-MARC CJK characters to their MARC equivalents. We believe that this is a normalization issue that should be discussed by systems vendors, the utilities, and their customers.

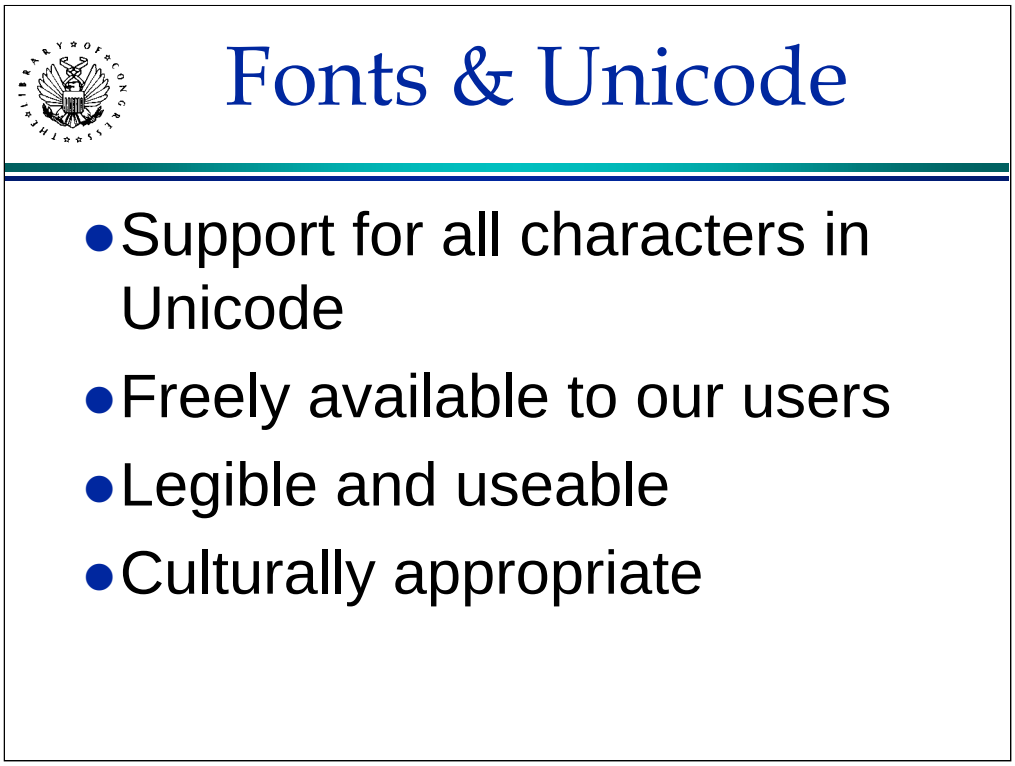
We're very concerned about the sorting of Chinese characters. In the Voyager with Unicode release Chinese characters are sorted by code point. That's right, the order of characters is determined by the Unicode encoding. Unfortunately, Chinese readers do not know the encodings and the resulting sort order is not obvious or logical to them. In effect search results are returned in an order that is not understandable to them. And remember that Chinese characters are used in Japanese and Korean. This makes navigation of all CJK search results difficult.



We're already trying to plan our workflows for the Unicode environment. We want our staff to have all the tools they need to take advantage of the capabilities in Unicode. We'll have to rely on standard Windows functionality for Unicode.

So, for example, we've identified Input Method Editors (IMEs) that provide staff with the ability to input non-Roman scripts on the standard PC keyboard. We've have some very practical experience with these IMEs, since RLIN21 also relies on the standard MS tools. They are generally quite usable. Part of the problem though, is that they offer CJK characters that are not in the MARC21 character repertoire and in *a few* cases, they do not provide the MARC equivalents. Said another way, if a cataloger sees a Japanese character on the page, they search the Japanese IME for the proper character to input in the bib record. The character that looks to be the appropriate character in the IME turns out *not* to be in the MARC character repertoire. Furthermore, the cataloger has no way of knowing this from looking at the character. Worse, the appropriate MARC character is not available in the Japanese IME. This is important to us, because as the maintenance agency for MARC, we are obligated to follow the standard.

To help our catalogers use the right characters, we've developed a table of the non-MARC characters along with their MARC equivalents, which we've mounted on our intranet. Once we've tested it here, we plan to make it available on the Web. I understand that MS Office2003 has better IMEs, but we have not tested it.



A caveat: fonts are my hobby horse and here's why-- In my opinion, this is what we need:

> a font that supports all characters in all scripts in Unicode MARC repertoires (Arial Unicode MS does not have all characters in Unicode)

> freely available to libraries and their users

> useable by librarians, e.g, clear distinctions between zero and upper cased O; or lower cased L and Arabic number 1; unambiguous display of diacritics in the ALA character set, such as the ligatures in romanized Cyrillic or double diacritics in Vietnamese

We asked the Digital Library Federation to consider this issue, thinking that perhaps they had the wherewithal to obtain my dream font. Unfortunately DLF came to the same conclusion as OCLC, that it's too expensive. Here's the history: Many fonts have been defined for displaying most Unicode characters. In our testing at LC, we have been using the MS Arial Unicode font. We like it because it's readable and it has (almost) all characters. But Microsoft has ceased to make the Arial Unicode MS font available as a free download, forcing users to find a legal and usable Unicode font. At LC all staff workstations have MSOffice installed, therefore we can use MS Arial. But we think that our public users and other sites will have to deal with Microsoft's decision not to offer the MS Arial font as a free download. How will libraries provide a font that will enable users to view metadata in all scripts in our catalogs? OCLC and RLG recommend MS Arial Unicode font for their clients. OCLC gives this rationale: "OCLC cannot provide this font directly to our member libraries due to exclusive licensing agreements between the font developer and Microsoft. While researching this issue, we found that some local systems already use this font, and many member libraries have access to it. OCLC attempted to purchase a different font, but due to the excessive ongoing cost, we determined that spending such a large amount of money would not be in the best interests of our members."

Fortunately, MS Arial Unicode accommodates the JACKPHY languages adequately. But just adequately. I mentioned clarity to distinguish similar characters, like oh and zero. But beyond that we have a responsibility to present culturally appropriate displays for our users.

Fonts & Unicode

- Arabic
- Nastaliq
  scripts.sil.org/cms/scripts/page.php?site_id=nrsi&item_id=Nastaliq
- Burmese? Amharic? Others?
- MS Longhorn !?
  - » ClearType
  - » www.poynterextra.org/msfonts/index.htm

One example is the Arabic script. The Arabic script as rendered in the MS Arial font is childish to most Muslims. Arabic is considered to be the language of God, therefore a beautiful, calligraphic script is prized most highly. The simplicity of MS Arial is seen as insulting.

Continuing with the Arabic script, there are other Arabic script languages, such as Urdu, that use the ornate Nastaliq style. They require a different font in order to be acceptable to readers. For example, in Nastaliq, the second time a "noon" appears in a word it has a different shape from the first; the third time it's different again from the second (each time it gets bigger and hangs lower beneath the line). If you'd like info from a source more authoritative than I, try that address: http://scripts.sil.org/cms/scripts/page.php?site_id=nrsi&item_id=Nastaliq

I've been careful to say that MS Arial covers most of the scripts in Unicode. What doesn't it have? Burmese or Myanmar, Amharic, and other languages and scripts that I know we have in our collections. We'll have to address this before we start to work in these languages.

A colleague at LC has contacts at Microsoft and he reports that the ClearType team is working on fonts that are legible on screen as well as on print. ClearType will function ideally on LCD screens. You can view "Longhorn Readability Fonts" at that address http://www.poynterextra.org/msfonts/index.htm That font has six type families that contain Latin, Greek and Cyrillic script developed at the same time and harmonized to work in multilingual documents. Apparently the plan is for Cambria to be the default font in Office 2006(?). I should note that fonts can reside on workstations or be delivered dynamically from servers when appropriate licenses are purchased. Ascender will handle licensing of numerous fonts developed by independent contractors for Microsoft. Ascender can also customize font versions and formats.

At LC we'll be able to deliver a configuration that will support 99% of our staff. But I we have yet to address the needs of our external users. We believe that we'll have to craft extensive help pages for OPAC users to give them info on configuring their PCs so that they can read the JACKPHY scripts that are in our records today. We will not be able to supply a font. We'll have to move carefully as we expand beyond the JACKPHY scripts, as in the case of Burmese.

Unicode in LC's Web OPACs

- LC's Web OPAC will:
  - Display all scripts
  - Enable download of records in MARC-8 and UTF-8
- LC will implement these features in the LC Online Catalog & LC Authorities
  - catalog.loc.gov
  - authorities.loc.gov
- Z39.50 may offer MARC-8 as well as UTF-8
- CDS will distribute in MARC-8 & UTF-8
- LC records will only contain characters in the MARC 21 repertoires

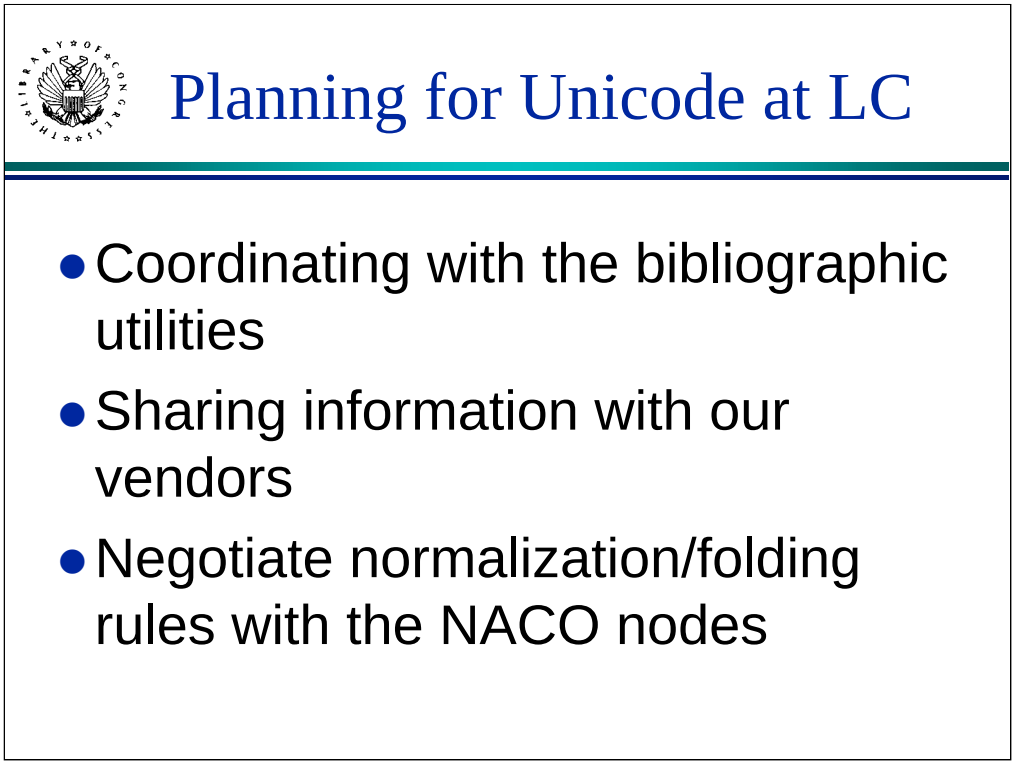
To wrap up, I'd like to remind you what features LC will offer to our public users once we've converted to Unicode.

Users of the the LC Online Catalog be able to see the correct non-Roman scripts displayed and they will be able to save records in MARC-8 and UTF-8 from the Web OPAC. Remember that at the moment the only non-Roman scripts we have are the JACKPHY languages. We will make these features available in the LC Online Catalog and LC Authorities (those are the addresses on the Web).

We will offer MARC-8 and UTF-8 records via Z39.50. We'll provide instructions for configuring Z39.50 search attributes on our Web site.

The Cataloging Distribution Service (CDS) will continue to provide MARC 21 bib and authority records in MARC-8. Once the Library's data are converted to Unicode, CDS will offer records in UTF-8 as well as MARC-8. As I mentioned before, LC records will contain only those Unicode characters that are in the MARC 21 repertoire.

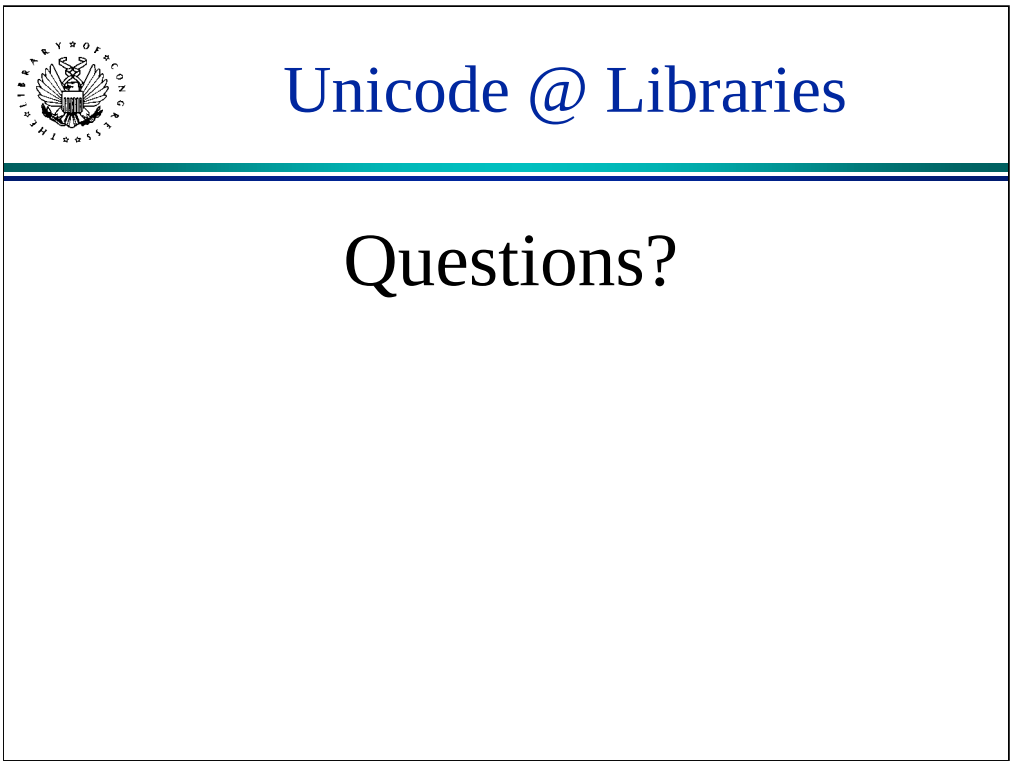
So if your database is not yet converted to Unicode, you'll want to get records from LC encoded as MARC-8; if your database is in Unicode you'll want to get records in UTF-8.



Another aspect of LC's conversion to the Unicode standard is our work with other organizations to ensure a smooth transition.

We've been testing and exchanging information with RLG and OCLC as each of us move to Unicode to ensure that our implementations are as seamless as possible. Of course we'll need to time any changes in our practices with the bibliographic utilities and participants in the Program for Cooperative Cataloging (PCC). We'll be talking to the vendors that supply us with bibliographic records to help move them in the direction of Unicode.

And we'll need to negotiate normalization or folding rules with the NACO nodes, that is, OCLC. RLG and the British Library.



Thank you for your attention. I'll be happy to answer any questions.