Moving to Unicode -- Tips & Tricks

Mark Jamison

IBM

C09

April 29th 1:00-2:00 PM

2008 IIUG Informix Conference

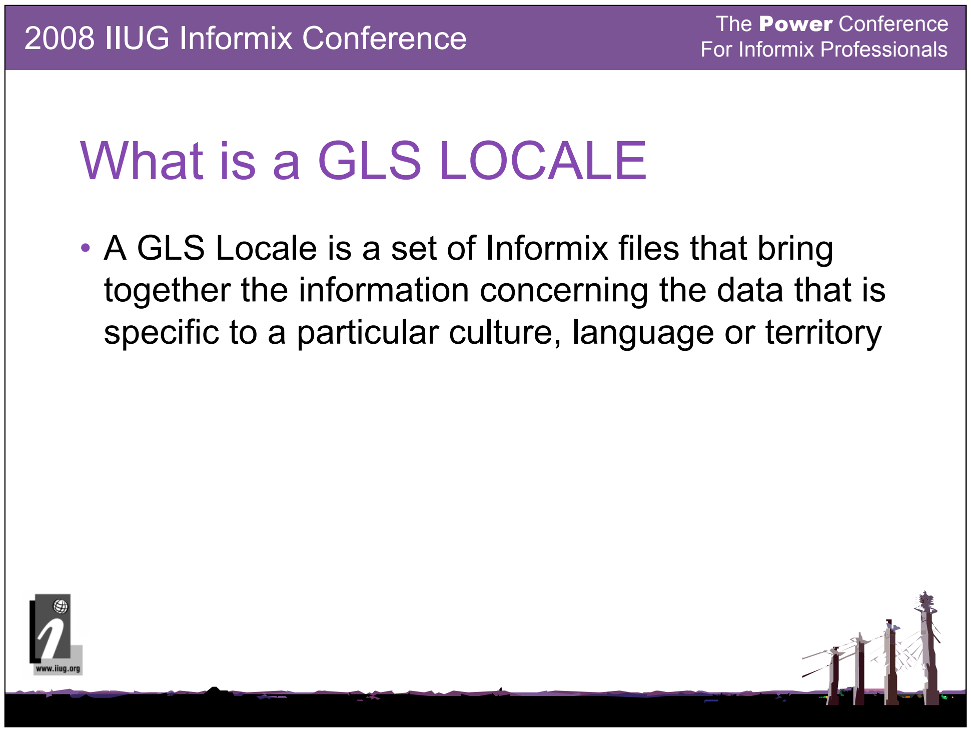
www.iiug.org

What is Unicode?

- The industry standard for computers to represent and manipulate text expressed in most of the world’s writing systems.

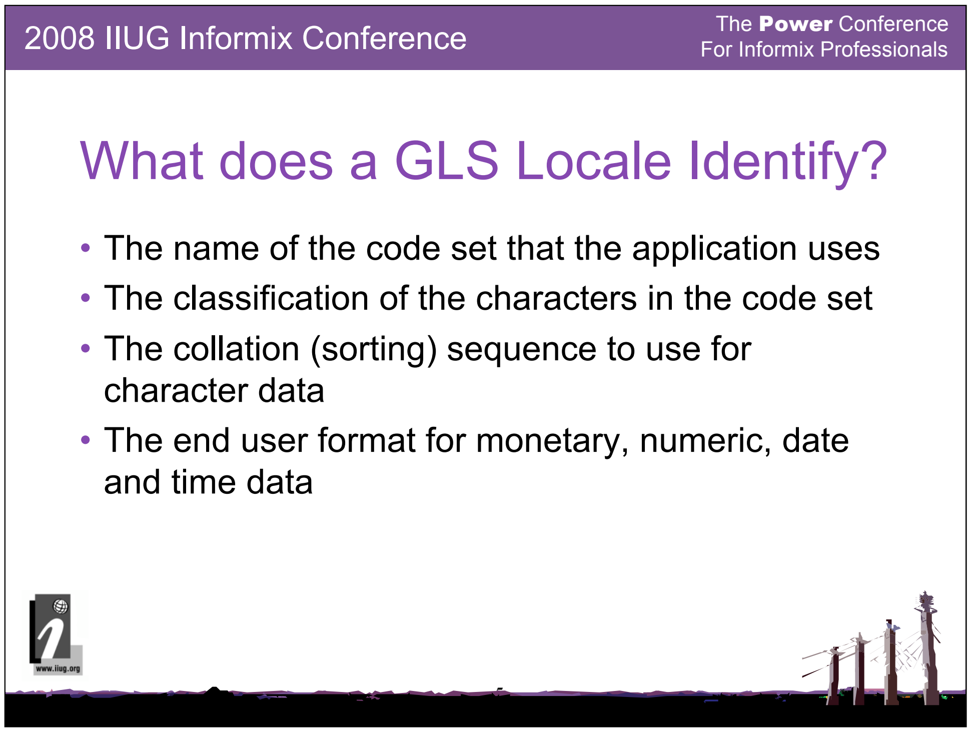
How does IDS handle Unicode?

- IDS uses (G)lobal (L)anguage (S)upport, and Locales to represent Unicode as well as other collation types in the database.



What is a GLS LOCALE

- A GLS Locale is a set of Informix files that bring together the information concerning the data that is specific to a particular culture, language or territory



What does a GLS Locale Identify?

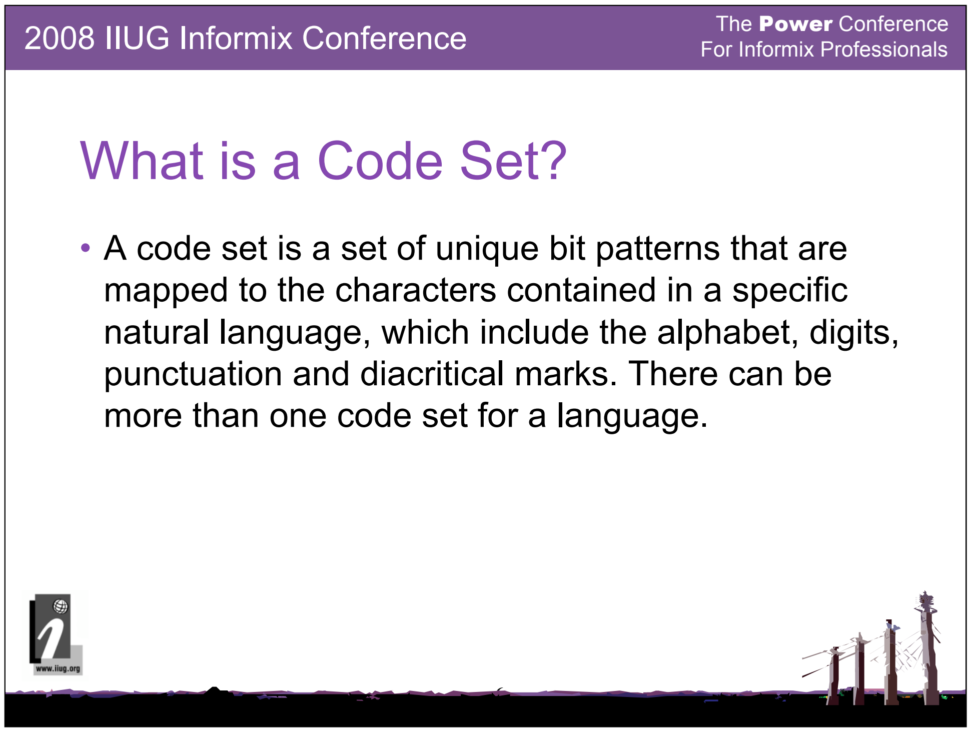
- The name of the code set that the application uses
- The classification of the characters in the code set
- The collation (sorting) sequence to use for character data
- The end user format for monetary, numeric, date and time data

A locale name has language, territory, code set.

For example, en_us.8859-1, where en indicates English language, us indicates the United

States territory and 8859-1 indicates the code set.

Since every database requires a locale for creation, Informix provides a default locale. The default code set is en_us.8859-1 for all Unix based platforms and en_us.1252 for Microsoft Windows.



What is a Code Set?

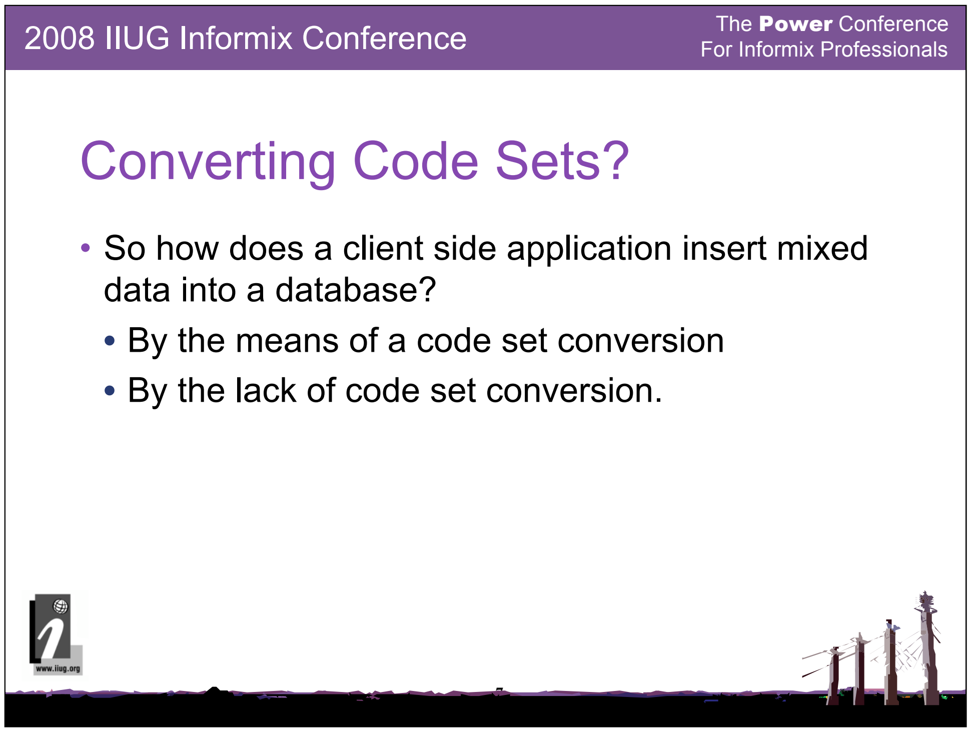
- A code set is a set of unique bit patterns that are mapped to the characters contained in a specific natural language, which include the alphabet, digits, punctuation and diacritical marks. There can be more than one code set for a language.

For Example, English has code sets ASCII, ISO8859-1, Windows Code Page 1252.

The number of unique characters in the language determines the amount of storage that each character requires in a code set. A single byte can store values in the range of 0 to 255, so it can uniquely identify 256 characters.

The ASCII code set is a single byte code set and contains 128 characters. And is the basis for all single byte code sets and most multi-byte code sets that are used by IDS.

If a character set contains more than 256 characters, the code set must contain multi-byte characters. A multi-byte character might require from 2 to 4 bytes. Languages having code sets made up of both single-byte and multi-byte characters are called multi-byte code sets.



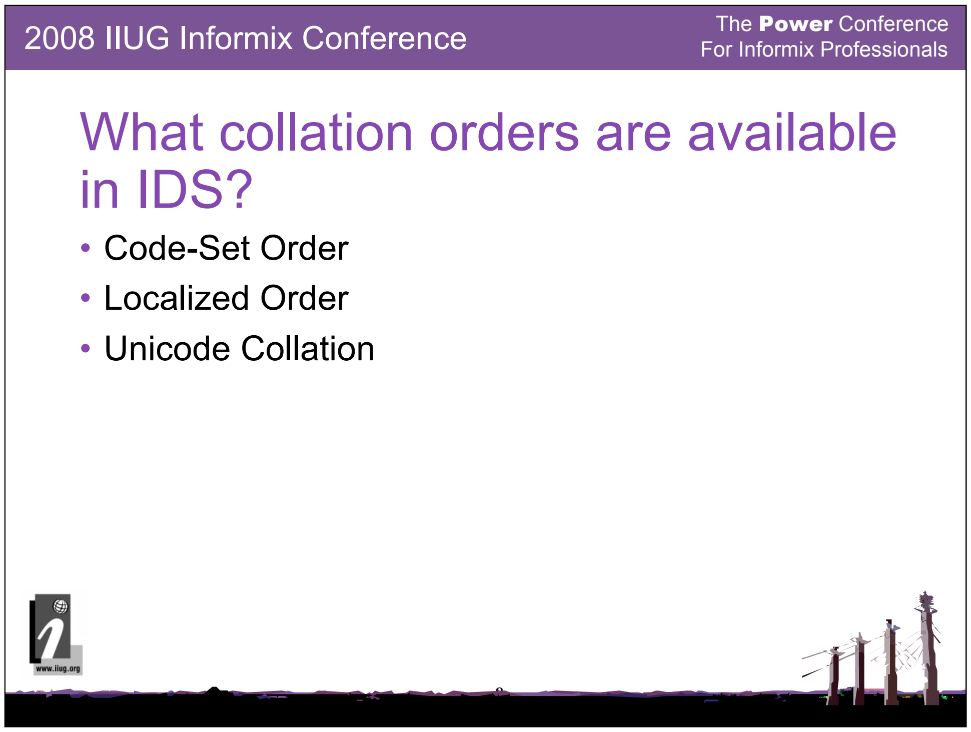
Converting Code Sets?

- So how does a client side application insert mixed data into a database?
  - By the means of a code set conversion
  - By the lack of code set conversion.

In a client server environment, character data might need to be converted from one code set to another if the client and database use different code sets to represent the same characters. The conversion of data from one code set (known as source code set) to another code set (known as target code set). Code-set conversion prevents data corruption when code-sets are different between source and target.

Code set conversion is not language translation. It does not convert between words in different languages. It only ensures that each character retains its meaning when it is processed or written, regardless of how it is encoded. Code-set conversion does not create a character in the target code set if it exists only in the source code set. For each character in source code-set, a corresponding character in the target code set should exist.

If a corresponding character doesn't exist then it will be mapped to a substitution character, which will result in a -202 error on an attempt to insert to the database.



What collation orders are available in IDS?

- Code-Set Order
- Localized Order
- Unicode Collation

•Code-Set Order
Code set order refers to the bit pattern order of characters within a code set. The order of code points in the code set determines the sort order. For Example, in ASCII code set, A=65 and B=66. The character A always sorts before B because a code point 65 is less than 66. Because a=97 and M=77, the string abc sorts after Me.

The data types sorted based on code-set order are as follows:
CHAR
LVARCHAR
VARCHAR
TEXT

•Localized Order
Localized order refers to an order of the characters that relates to a real language.The locale defines the order of the characters in the localized order. The localized order can specify a certain type of collation order.
It might specify sort order based on telephone book as shown

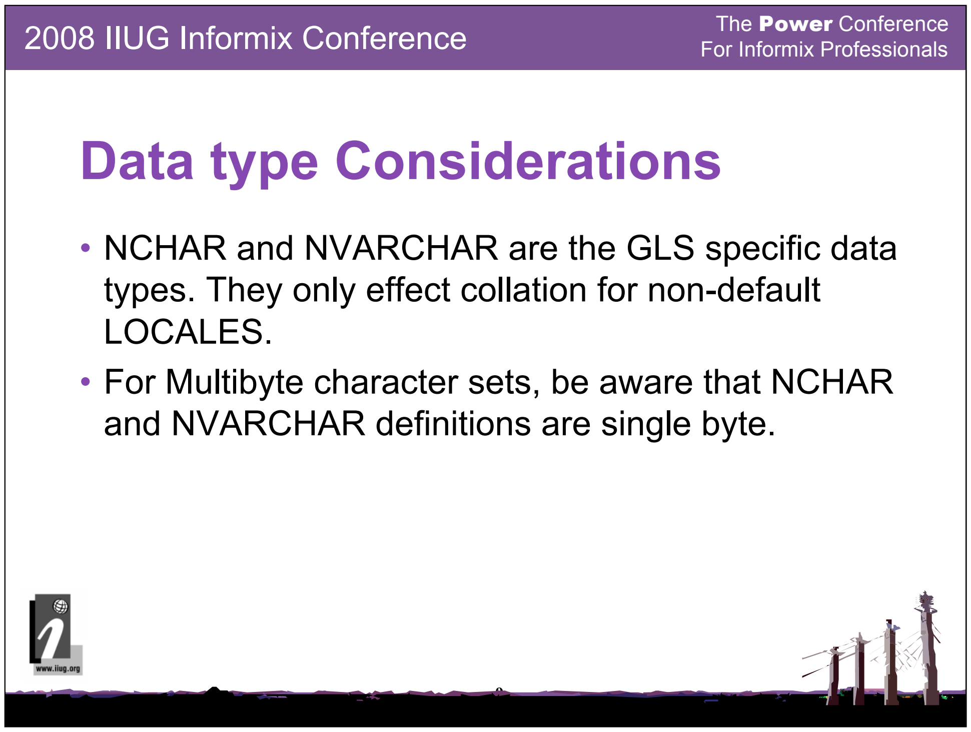
Mabin
McDonald
MacDonald
Madden

Or, it may specify sort order based on a dictionary as shown

Mabin
Madden
MacDonald
McDonald

Localized order, is used for all comparisons operations on NCHAR and NVARCHAR types. It is also used for matches queries on CHAR and LVARCHAR. If the GLS locale doesn't define a localized order then Code-Set Order will be used in place of Localized order.

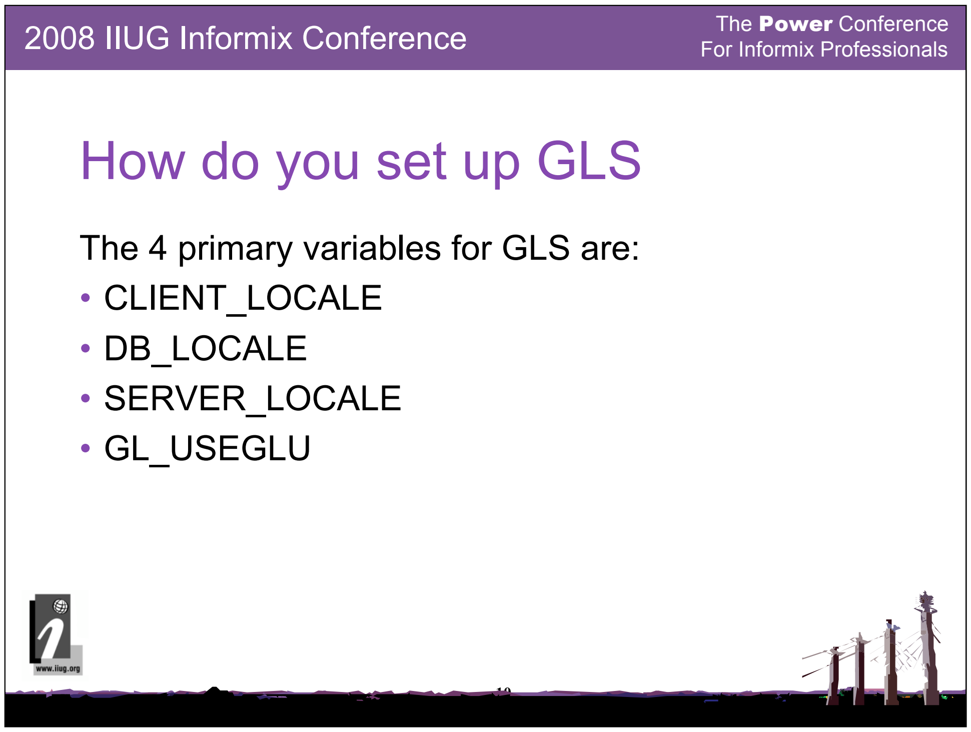
•Unicode Collation
Locale information is collated based on the ICU Unicode Collation Algorithm. For information see the Unicode website's presentation. Unicode collation is only used for NCHAR and NVARCHAR columns, for databases with a locale of utf8 or gb18030-2000



Data type Considerations

- NCHAR and NVARCHAR are the GLS specific data types. They only effect collation for non-default LOCALES.
- For Multibyte character sets, be aware that NCHAR and NVARCHAR definitions are single byte.

For instance an NCHAR(4) can handle 4 characters of single byte data, 2 characters of double byte data, or 1 character of quad byte data.



2008 IIUG Informix Conference

The **Power** Conference For Informix Professionals

How do you set up GLS

The 4 primary variables for GLS are:

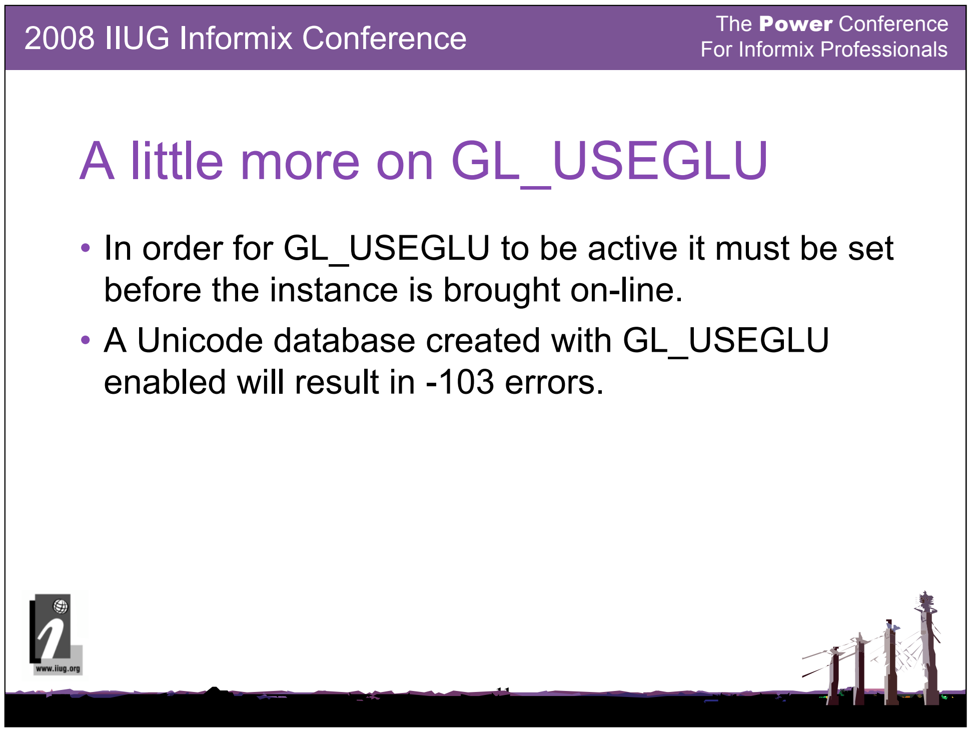
- CLIENT_LOCALE
- DB_LOCALE
- SERVER_LOCALE
- GL_USEGLU

CLIENT_LOCALE - The client locale identifies the locale that the client application uses.

DB_LOCALE - The database locale identifies the locale of the data in the database. The code set specified in the DB_LOCALE determines which characters are valid in any character column, as well as the names of database objects such as databases, tables

SERVER_LOCALE - The server locale identifies the locale that the database server uses for its server-specific files.

GL_USEGLU – The variable used to enable the ICU Unicode collation algorithms.



2008 IIUG Informix Conference

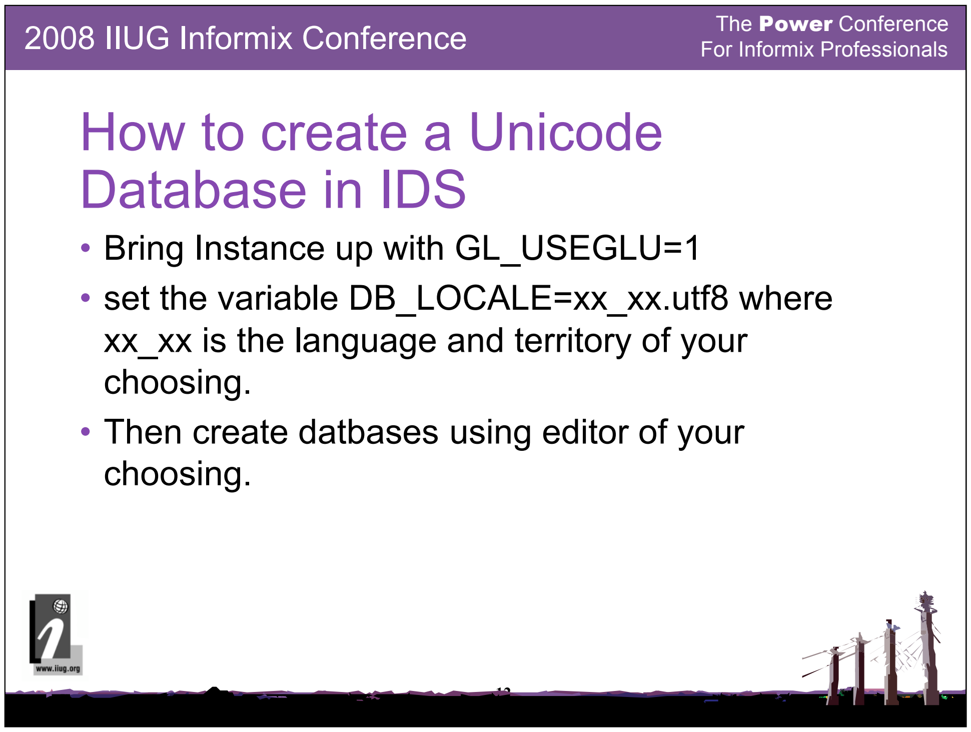
The **Power** Conference
For Informix Professionals

A little more on GL_USEGLU

- In order for GL_USEGLU to be active it must be set before the instance is brought on-line.
- A Unicode database created with GL_USEGLU enabled will result in -103 errors.

A definite word of warning ICU (ie, GL_USEGLU) enabled database will generate the following error if GL_USEGLU is not set at the instance level:

-103 ISAM error: illegal key descriptor (too many parts or too long).


2008 IIUG Informix Conference

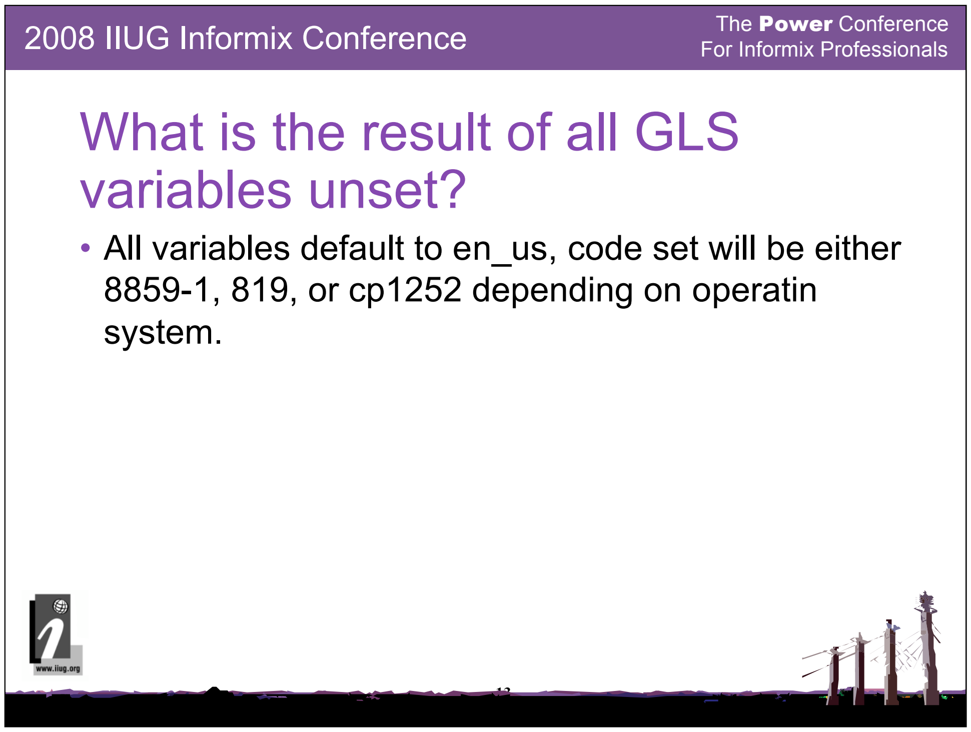
The **Power** Conference For Informix Professionals

How to create a Unicode Database in IDS

- Bring Instance up with GL_USEGLU=1
- set the variable DB_LOCALE=xx_xx.utf8 where xx_xx is the language and territory of your choosing.
- Then create datbases using editor of your choosing.

So let's presume we are creating an American English database. The steps, on an HP Itanium box with KSH, would be as follows:

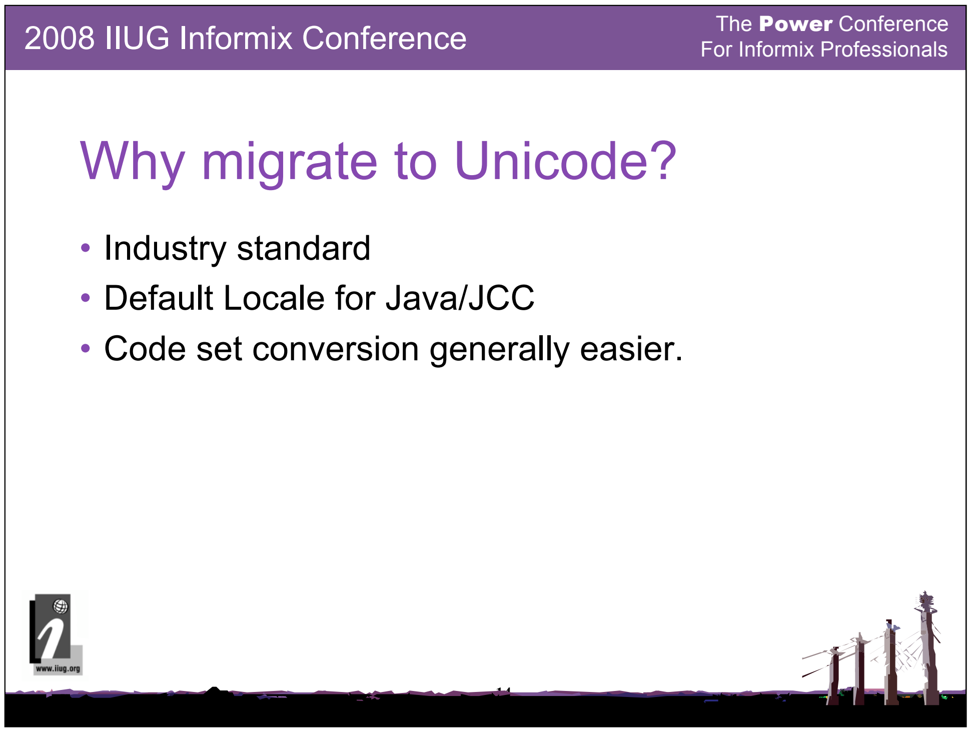
1. export GL_USEGLU=1
2. oninit (presuming a pre-existing instance, and instance currently offline)
3. export DB_LOCALE=en_us.utf8
4. echo "create database uni_example with log" |dbaccess sysmaster



What is the result of all GLS variables unset?

- All variables default to en_us, code set will be either 8859-1, 819, or cp1252 depending on operatin system.

www.iiug.org



Why migrate to Unicode?

- Industry standard
- Default Locale for Java/JCC
- Code set conversion generally easier.

With the move towards JCC, the need for Unicode is more useful. Also to keep in mind that while en_us.819 will give you errors if you try to cross query a ja_jp.sjis database, an en_us.utf8 has no problems with j_jp.utf8

Client-Side Migration

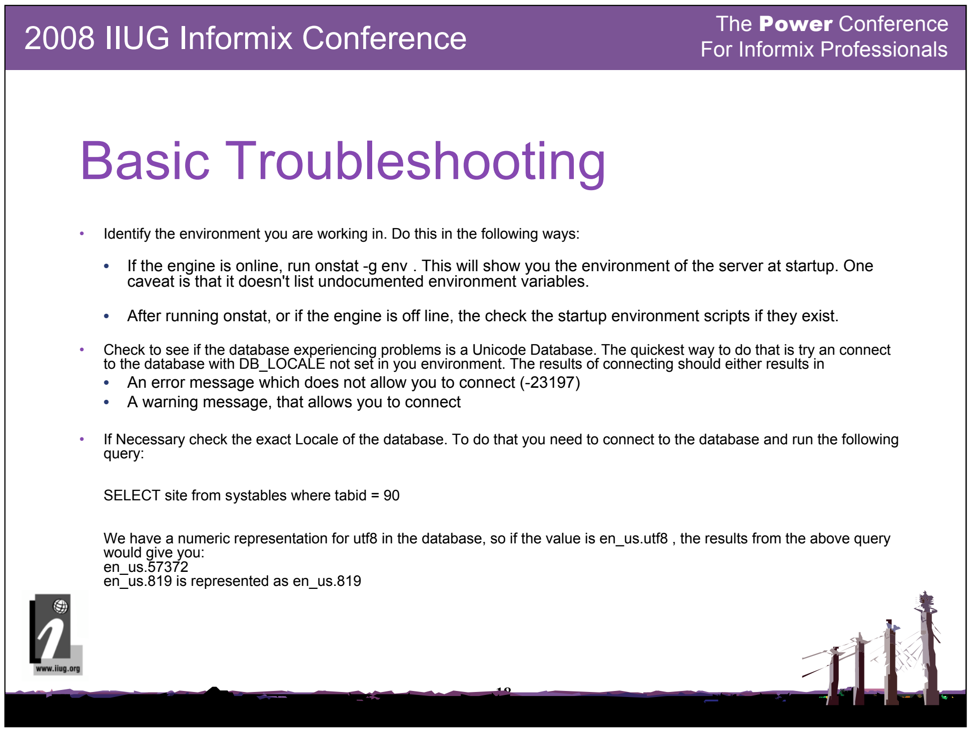
- Recompile apps as GLS aware.
- Set DB_LOCALE to match database.
- Set CLIENT_LOCALE to a valid locale for the given DB_LOCALE

Server Side Migration

- Create new databases with GL_USEGLU=1
- Set DB_LOCALE to correct LOCALE
- Use dbexport or HPL to unload data
- Set CLIENT_LOCALE to desired format when unloading non-unicode database.

Server Side migration Cont.

- Validate unloaded data, contact IBM tech support if necessary
- Load data into Unicode Database.


2008 IIUG Informix Conference

The **Power** Conference For Informix Professionals

Basic Troubleshooting

- Identify the environment you are working in. Do this in the following ways:
  - If the engine is online, run onstat -g env . This will show you the environment of the server at startup. One caveat is that it doesn't list undocumented environment variables.
  - After running onstat, or if the engine is off line, the check the startup environment scripts if they exist.
- Check to see if the database experiencing problems is a Unicode Database. The quickest way to do that is try an connect to the database with DB_LOCALE not set in you environment. The results of connecting should either results in
  - An error message which does not allow you to connect (-23197)
  - A warning message, that allows you to connect
- If Necessary check the exact Locale of the database. To do that you need to connect to the database and run the following query:

  SELECT site from systables where tabid = 90

  We have a numeric representation for utf8 in the database, so if the value is en_us.utf8 , the results from the above query would give you:
  en_us.57372
  en_us.819 is represented as en_us.819

Identify the environment you are working in. Do this in the following ways:

If the engine is online, run onstat -g env . This will show you the environment of the server at startup. One caveat is that it doesn't list undocumented environment variables.

After running onstat, or if the engine is off line, the check the startup environment scripts.

Check to see if the database experiencing problems is a Unicode Database. The quickest way to do that is try an connect to the database with DB_LOCALE not set in you environment. The results of connecting should either results in

An error message which does not allow you to connect (-23197)

A warning message, that allows you to connect

If Necessary check the exact Locale of the database. To do that you need to connect to the database and run the following query:

```
SELECT site from systables where tabid = 90
```

We have a numeric representation for utf8 in the database, so if the value is en_us.utf8 , the results from the above query would give you:
en_us.57372
en_us.819 is represented as en_us.819

Common Errors

- Problem: When inserting into a numeric field I get error -1213 A character to numeric conversion process failed.
- Most Common Solutions: A character value is being converted to numeric form for storage in a numeric column or variable. However, the character string cannot be interpreted as a number. It contains some characters other than white space, digits, a sign, a decimal, or the letter e; or the parts are in the wrong order, so the number cannot be deciphered. Check if the decimal character or thousands separator is wrong for your locale.

Common Errors continued

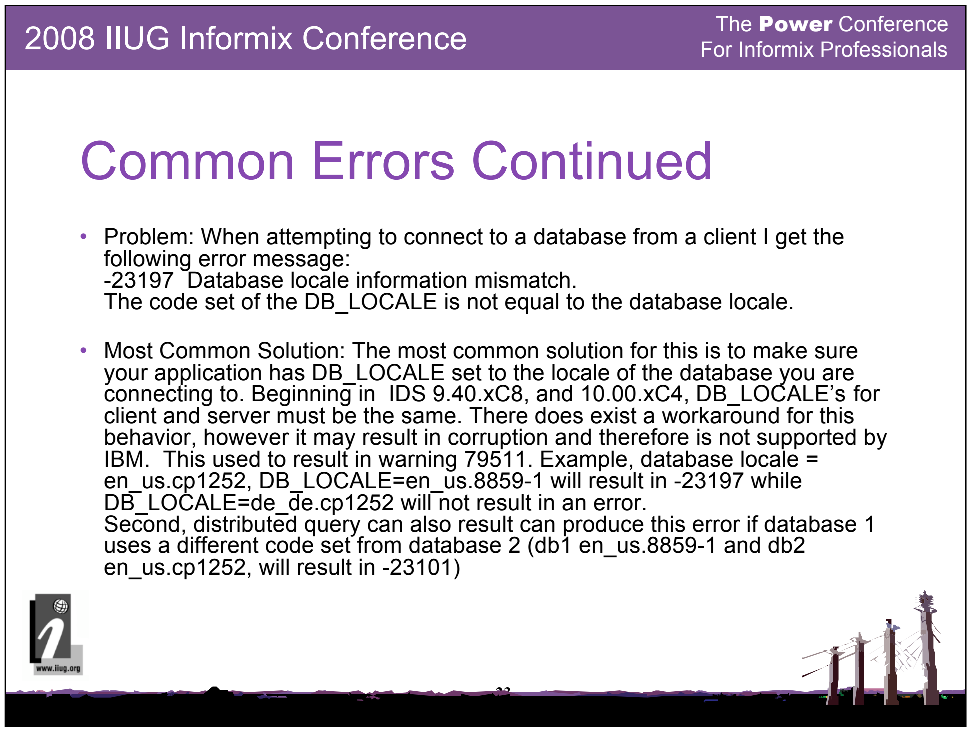
- Problem: When attempting to bring the database engine up I get the following message: -23101 Unable to load locale categories.
- Most Common Solutions: The four most common solutions are
  - INFORMIXDIR is not set or is set incorrectly
  - CLIENT_LOCALE or DB_LOCALE environment variables are invalid.
  - CLIENT_LOCALE or DB_LOCALE environment variables reference locales that are specific to the ILS product. ILS is an (I)nternational (L)anguage (S)upplement that includes additional locales which are not part of the locales you get when you install IDS.
  - GL_USEGLU set in client's environment for application that is not ICU aware.

Common Errors Continued

- Problem: When inserting data I get the error:
  -23103 Code-set conversion function failed due to an illegal sequence or invalid value.
  Illegal or invalid characters occur in the character string. The program could not execute the code-set conversion on the characters that this string contains.
- Most Common Solutions: Reexamine the input string for illegal or invalid characters, and reexecute the program.

Common Errors Continued

- Problem: When attempting to connect to a database the client I get error
  -23104 Error opening required code-set conversion object file. No object code-set conversion (definition) file exists for the two given code sets.
- Most Common Solutions: The environment variables DB_LOCALE or CLIENT_LOCALE might have been set to the wrong value, which caused an error condition to be generated. The files might also be missing, or the information that they contain might be garbled. Check the value of the environment variable CLIENT_LOCALE or DB_LOCALE. The object conversion files usually have the .cvo extension.



Common Errors Continued

- Problem: When attempting to connect to a database from a client I get the following error message:
  -23197 Database locale information mismatch.
  The code set of the DB_LOCALE is not equal to the database locale.

- Most Common Solution: The most common solution for this is to make sure your application has DB_LOCALE set to the locale of the database you are connecting to. Beginning in IDS 9.40.xC8, and 10.00.xC4, DB_LOCALE's for client and server must be the same. There does exist a workaround for this behavior, however it may result in corruption and therefore is not supported by IBM. This used to result in warning 79511. Example, database locale = en_us.cp1252, DB_LOCALE=en_us.8859-1 will result in -23197 while DB_LOCALE=de_de.cp1252 will not result in an error.
  Second, distributed query can also result can produce this error if database 1 uses a different code set from database 2 (db1 en_us.8859-1 and db2 en_us.cp1252, will result in -23101)

www.iiug.org

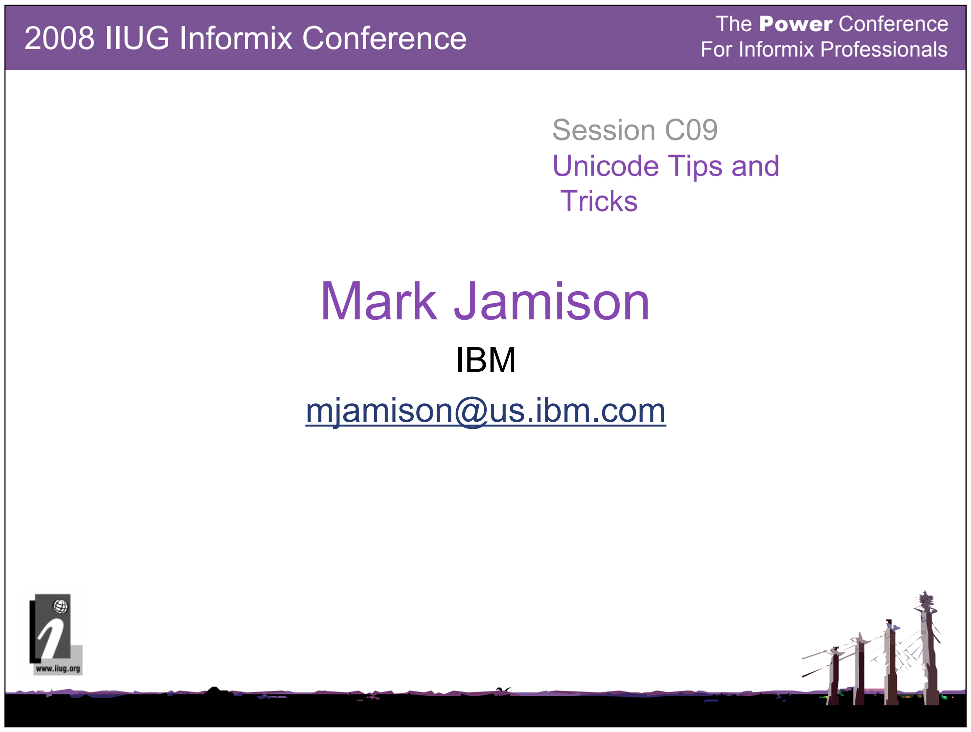
Common Errors Continued

- Problem: When attempting to connect to a Unicode database the application reports the following error:
  -103 illegal key descriptor (too many parts or too long).

- Most Common Solution: This almost always is indicative of a Unicode database that was created in an instance with GL_USEGLU=1, but the database instance has been bounce and GL_USEGLU was unset. Solution is to bring instance offline, set GL_USEGLUE in the environment and bring the instance back on-line.

Questions?

Additional Q&A



2008 IIUG Informix Conference

The **Power** Conference For Informix Professionals

Session C09

Unicode Tips and Tricks

Mark Jamison

IBM

mjamison@us.ibm.com

www.iiug.org