

5A CA Plex Unicode

Support Update:

Java, .NET and IBM i

Presenter Kiyoshi Terasawa

Principal Software Engineer, CA technologies

Date : June 2nd 2011



Kiyoshi Terasawa

- Principal Software Engineer **CA Technologies**
- 19 years experience with 2E and Plex
 - Synon/2E user -> joined Synon Japan -> Support -> QA -> Joined CA US -> Dev
 - Runtime Java and .NET in general
 - Database, Communication or any non-English topic
 - Any other area of Plex product as needed
 - Seasonal 2E localization QA

Legal

This presentation was based on current information and resource allocations as of **May 31, 2011** and is subject to change or withdrawal by CA at any time without notice.

Notwithstanding anything in this presentation to the contrary, this presentation shall not serve to (i) affect the rights and/or obligations of CA or its licensees under any existing or future written license agreement or services agreement relating to any CA software product; or (ii) amend any product documentation or specifications for any CA software product. The development, release and timing of any features or functionality described in this presentation remain at CA's sole discretion. Notwithstanding anything in this presentation to the contrary, upon the general availability of any future CA product release referenced in this presentation, CA will make such release available (i) for sale to new licensees of such product; and (ii) to existing licensees of such product on a when and if-available basis as part of CA maintenance and support, and in the form of a regularly scheduled major product release. Such releases may be made available to current licensees of such product who are current subscribers to CA maintenance and support on a when and if-available basis. In the event of a conflict between the terms of this paragraph and any other information contained in this presentation, the terms of this paragraph shall govern.

CERTAIN INFORMATION IN THIS PRESENTATION MAY OUTLINE CA'S GENERAL PRODUCT DIRECTION. ALL INFORMATION IN THIS PRESENTATION IS FOR YOUR INFORMATIONAL PURPOSES ONLY AND MAY NOT BE INCORPORATED INTO ANY CONTRACT. CA ASSUMES NO RESPONSIBILITY FOR THE ACCURACY OR COMPLETENESS OF THE INFORMATION. TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENT "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT WILL CA BE LIABLE FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENT, INCLUDING, WITHOUT LIMITATION, LOST PROFITS, LOST INVESTMENT, BUSINESS INTERRUPTION, GOODWILL OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

About the session

- CA Plex Unicode support for Java and .NET
- Unicode on IBM i
- CA Plex Unicode Support on IBM i
- CA Plex IBM i dispatcher enhancement

Before start, here is some terminology makes you confuse or refuse

- Codepage
 - Unicode
 - Non-Unicode codepage EBCDIC, ANSI or etc
 - SBCS
 - DBCS
- ASCII
- CCSID

CA Plex Unicode support history

CA Plex Unicode support history

CA Plex r6.1 GA	CA Plex r6.1 PTF	CA Plex r7.0
<p>Java and C# function handles Unicode in program</p>	<p>Unicode field type support for SQL Database</p> <p>New SYS Values Nchar and Nchar_var for FLD SQL format SYS</p>	<p>Unicode support for IBM i</p> <p>New SYS Value UNCIDOE for FLD AS400 format SYS</p>
<p>Java and .NET function</p>	<p>Java and .NET function SQL Database</p>	<p>Java and .NET function SQL Database</p> <p>IBM i Function and Database</p> <p>Java/.NET and RPG Server Communication</p>

CA Plex Unicode support for Java and .NET

Unicode support in Java and .NET

- Java and .NET uses Unicode in program
- However you might not be implemented the benefit
- Because...
 - Your database may store your data with non Unicode codepage

How to use Unicode in SQL Database?

- Set up Database with Unicode encoding
 - By creating the Database storage with Unicode encoding, you can store String data as Unicode
 - It will allow using any character for all String Field including Key Field, probably not the best scenario
- Define Unicode data type at field level
 - With this option you can use ANSI and Unicode fields within one database
 - Use Unicode for customer name or product name fields and keep key field as non-Unicode field

Do you know?

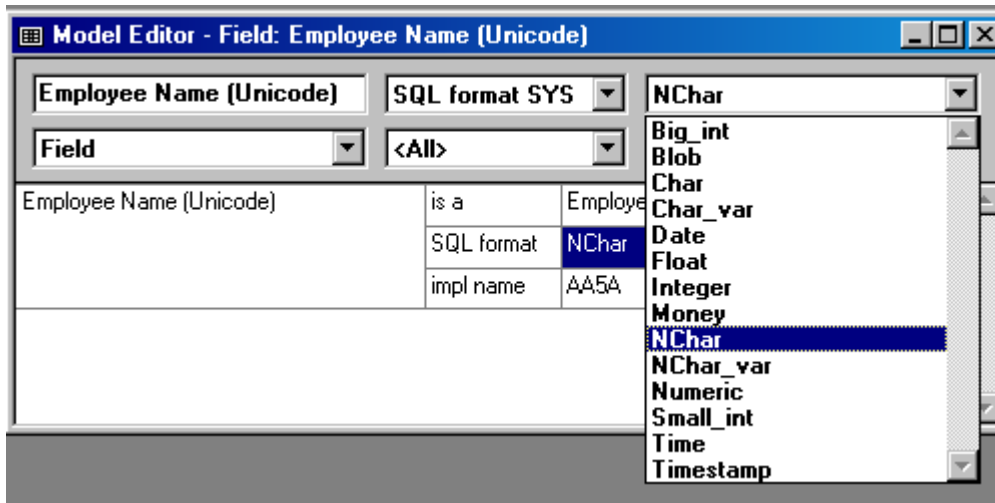
Database Driver automatically convert String data between program encoding and database encoding if necessary

we can



How CA Plex supports Unicode for SQL database?

- FLD SQL format SYS
 - Nchar : Fixed length Unicode String
 - Nchar_var : Varyng length Unicode String



Unicode on IBM i

Character field type

- character field type on IBM i
 - EBCDIC SBCS (one byte per one character)
 - Data type A
 - Support 256 character maximum and per language
 - EBCDIC DBCS (one or two bytes per one character)
 - Data type O, E or G
 - Support more characters than SBCS EBCDIC but still per language
 - UNICODE
 - Data type C or G
 - Support a lot of characters for all language

Unicode on IBM i

– UNICODE on IBM i

- UTF-16 (CCSID 1200) ← This is our pick
 - Always use 2 byte (16 bit) for one character
 - Or 4 byte for Character U+10000 to U+10FFFF
- UCS-2 (CCSID 13448)
 - UCS-2 = UTF-16 – Extension in Unicode version 2.0
 - It means UTF-16 > UCS-2
- UTF-8 (CCSID 1208)
 - One Character = 1 to 4 byte
 - ASCII char is 1 byte (This is why it is said good for Web.)
 - This isn't suitable for Plex byte count base data exchanges

CA Plex supports UTF-16 on IBM i

- Why picking UTF-16 for Plex?
 - UTF-16 > UCS-2
 - WINDOWS, JAVA, .NET use UTF-16
 - It is simple to get byte Count of UTF-16 for client and server data exchange based on Plex Field length

Without Unicode...

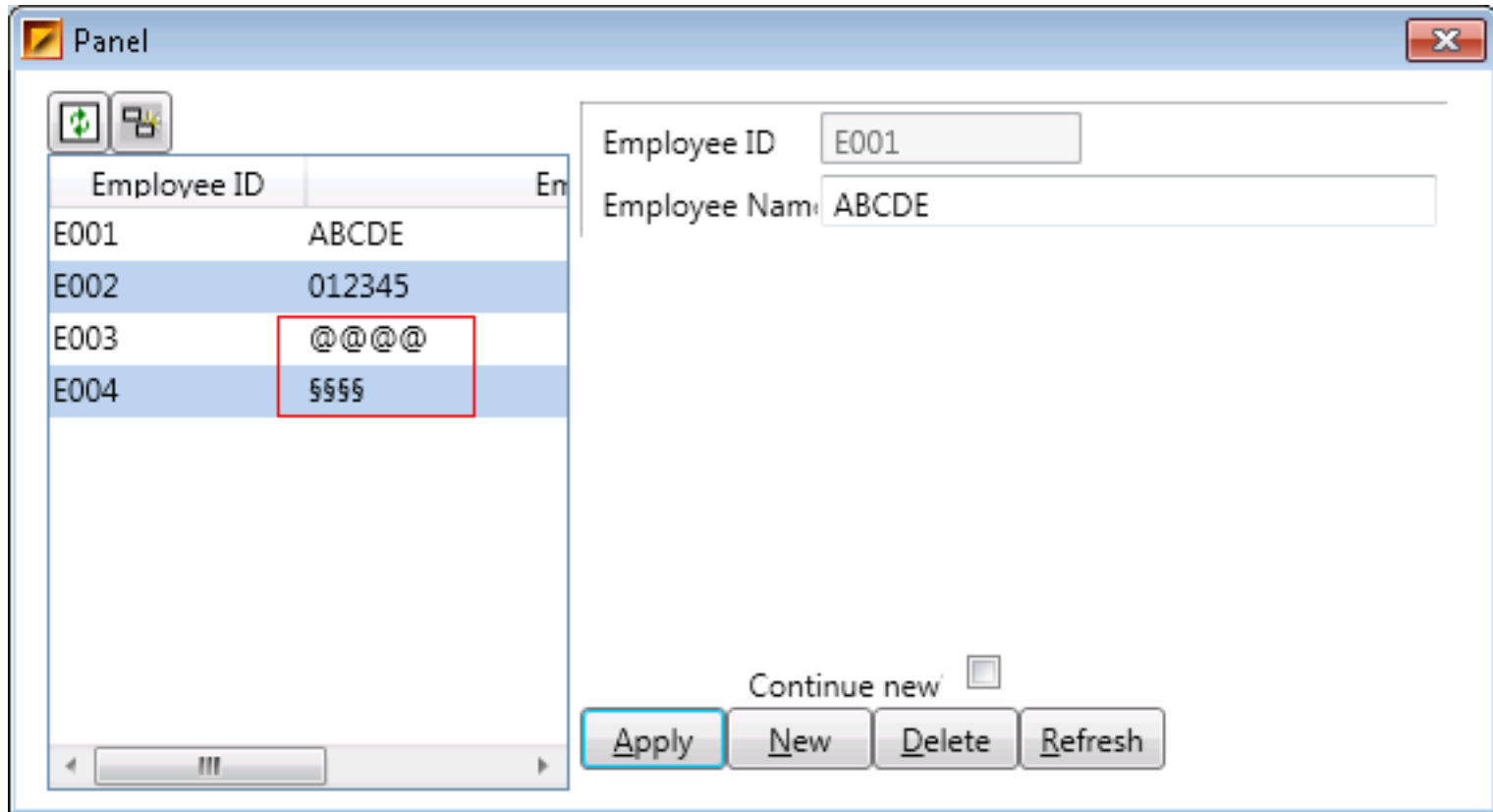
- Without Unicode support you need to
 - Take care of CCSID for JOB, Database or User
 - Set up encoding property for client and server Data conversion
- What you see on client and may not be what someone stored
 - Example :

Between English and German CCSID, character '@' and '§' are mapped to hex value like this

	English CCSID 1140	German CCSID 1141
@ (At sign)	7C	B5
'§' (Section sign)	B5	7C

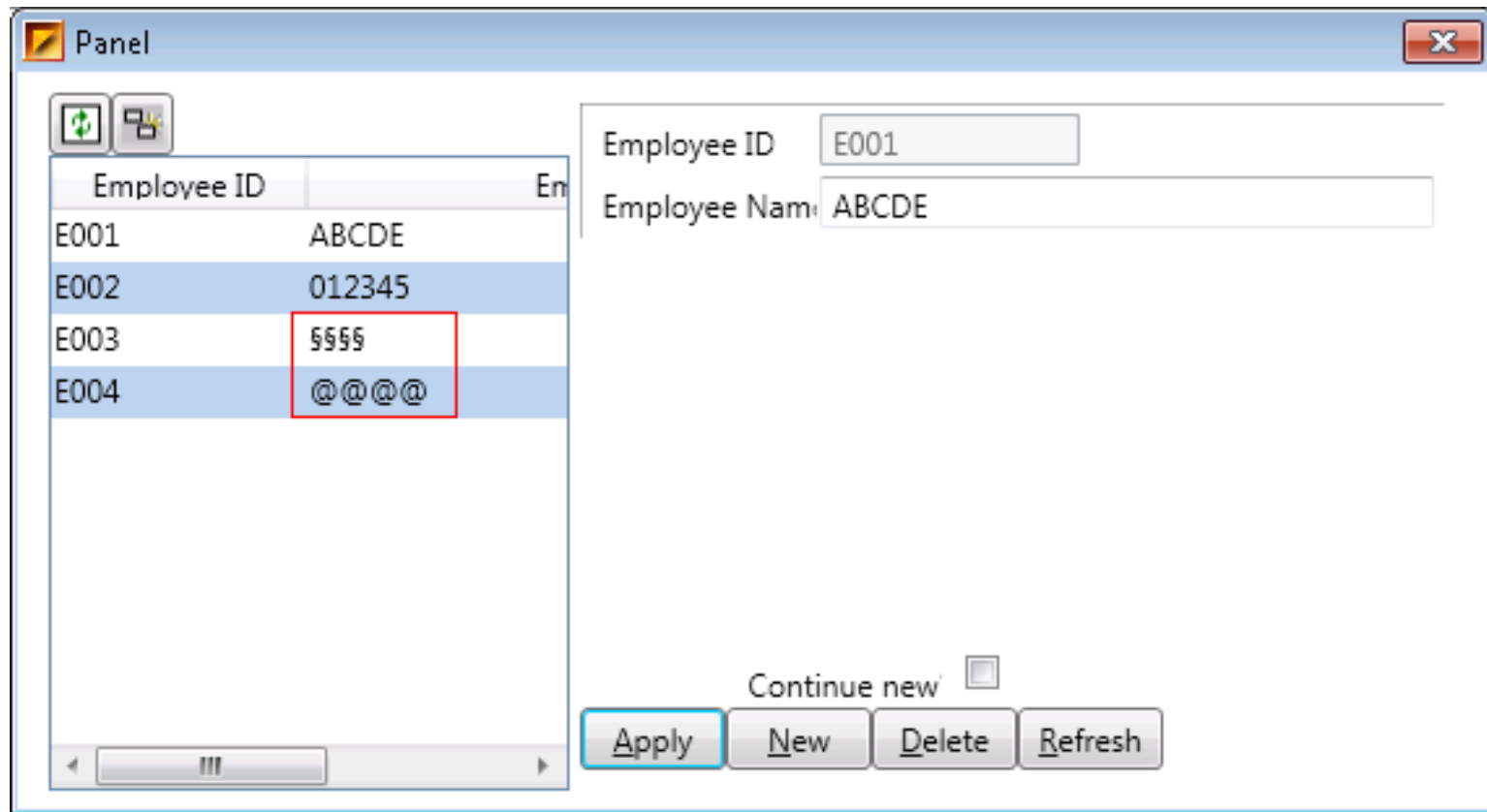
Without Unicode...

— With English 1140



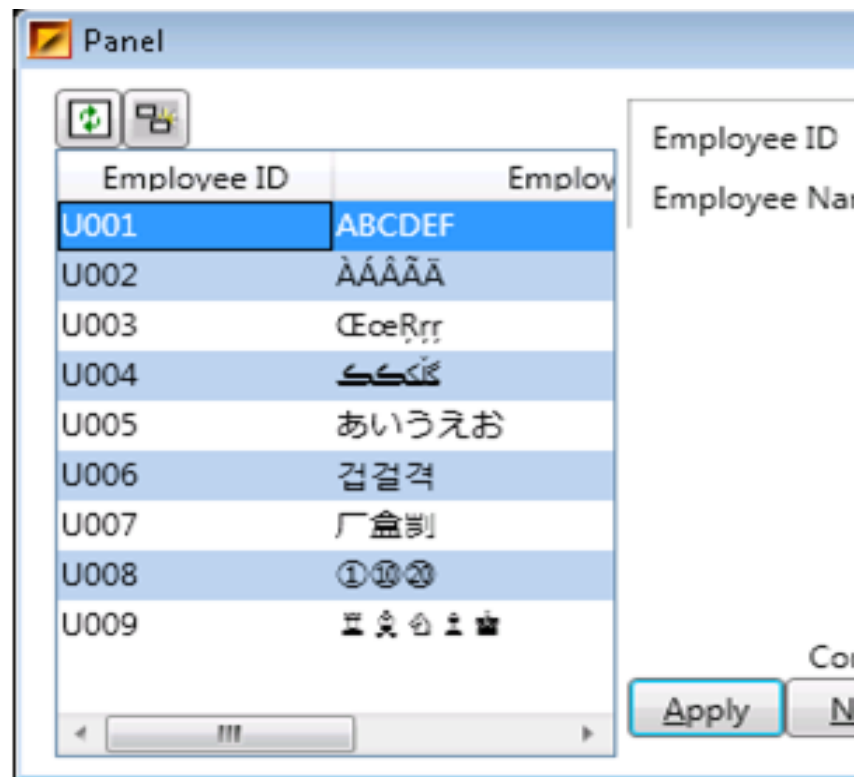
Without Unicode...

- Same data With German 1141



With Unicode you can...

- With UNICODE support you can
 - Store more Character in PF-DTA without changing CCSID
 - Simplify your deployment



How they are stored in PF-DTA

— How Data stored?

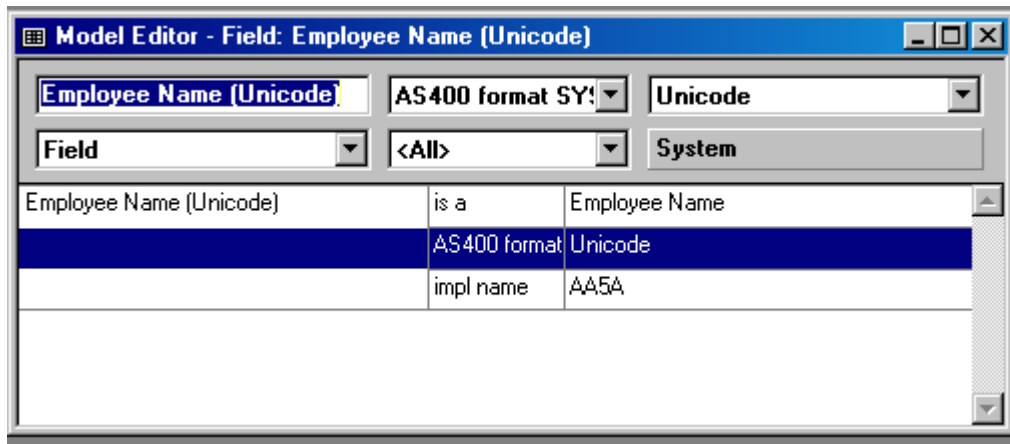
	EBCDIC	UTF-16
A	C1	0041
B	C2	0042
C	C3	0043
D	C4	0044
E	C5	0045
F	C6	0046

```
Session B - [24 x 80]
File Edit View Communication Actions Window Help
Work with File - Single record Display
File. . . . : POWPA02TST/WPFT
Member . . : WPFT
Format : Employee.Physical table Name : WPFT
Heading Length Value Record no. : DISPLAY 3
Counter 4.0
Employee Name 10 ABCDEF GHIJ
CCCCCCCCD
1234567891
Employee Name UNI 0.0
040404040404040404040404
0102030405060708090A
F3=Exit F6=Print F8=Change HEX mode F13=Fast exit F24=More
MF b MW 英数 半角 08/002
I902 - Session successfully started
```

CA Plex Unicode Support on IBM i

Unicode support with CA Plex r7.0

— New FLD AS400 format SYS UNCIDOE



Unicode support with CA Plex r7.0

- What happens with AS400 Format SYS UNICODE?
 - Field definition in PF-DTA DDS Source
Data type 'G' with CCSID (1200) keyword
 - Field definition in RPGIV Source
Data Type 'C' with CCSID (1200) keyword
 - Field definition in Java, C# and C++ source
Generate AS400 format SYS info to RTDI or Field Constructor
Used at Client and Server Data Conversion

Unicode support with CA Plex r7.0

- Support both UNICODE and EBCDIC field in RPGIV
 - Unlike Java or .NET, mixture of EBCDIC and Unicode data co-exist in Program layer
- No impact for Action Diagram Statement
 - Necessary UTF-16 and EBCDIC conversion occurs in RPGIV program
 - Set or Cast
 - Condition
 - Concatenation
 - Substring
 - RTVCND
 - Format Message

limitations

- RPGIV only
- No Unicode support for function with PRTF or DSPF
- No Unicode supports for C++ function

Summary

All you need to do is to set

FLD SQL format Sys Nchar or Nchar_var

FLD AS400 format Sys Unicode

Gen and build Table, View and Functions!

we can



CA Plex IBM I Dispatcher enhancement

CA Plex IBM i dispatcher enhancement

- Connection request Timeout
 - V5R4 feature
 - To avoid the ‘hanging dispatcher’ scenario
- CCSID support for log on to IBM i
 - New Data Area YOBCCSID

Q&A
thank you

appendix

AD statements Unicode handlings

AD Statement : SET R = A

- SET AD statement for Character field is generated as MOVEL
 - The MOVEL supports Data Conversion between EBCDIC and UNICODE

Factor	Result
UNICODE	UNICODE
UNICODE	EBCDIC
EBCDIC	UNICODE
EBCDIC	EBCDIC

AD statement : CAST R, A

- CAST AD statement is for converting data between different data type, i.e. String to String, Number to String or String to Date. Plex generates the CAST as external program call
- Numeric or Date doesn't use Unicode specific character
- Works as expected

AD statement : If A == B (Condition)

- Condition is used for IF, CASE or LOOP block
 - Comparing Character 'A' (EBCDIC hex'C1' and Unicode hex '0041') with '==' operator results the same 'true' for below four cases .

Factor 1	Factor 2
UNICODE	UNICODE
UNICODE	EBCDIC
EBCDIC	UNICODE
EBCDIC	EBCDIC

AD statement : R = A + B (CONCAT)

- If FACTOR 1 or FACTOR 2 are UNICODE call the sub-routine
- The sub-routine has following code (Free format Syntax)

```
/FREE
```

```
YZUTGT= %TRIMR(YZUSTR) + %TRIMR(YZUST2);
```

```
/END-FREE
```

Factor 1	Factor 2	Result
UNICODE	UNICODE	UNICODE
UNICODE	UNICODE	EBCDIC
EBCDIC	UNICODE	UNICODE
EBCDIC	UNICODE	EBCDIC
UNICODE	EBCDIC	UNICODE
UNICODE	EBCDIC	EBCDIC
EBCDIC	EBCDIC	UNICODE
EBCDIC	EBCDIC	EBCDIC

AD statement : SUBSTRING

- If source or target field is Unicode, Plex generates new Syntax
- Use = operator to convert the result between Unicode and EBCDIC
- The Syntax is following (Free format RPG)

```
/FREE
```

```
F00004 = %SUBST(F00001:F00006:F00007);
```

```
/END-FREE
```

F00004 :Target Field, F00001: Source Field, F00006: Index, F00007: Length

AD statement : RTVCND name

- Plex Value PF-Data store data as EBCDIC
- REVCND is handled text in EBCDIC
- Returned text data is in EBCDIC then converted to UNICODE with MOVEL

AD statement : format message

- IBM i MSGF doesn't support UNICODE CCSID
- Formatting occurs after data is converted to EBCDIC
- Formatted Data is in EBCDIC then converted to UNICODE with MOVEL