National Character Support

William Keech

Sunbelt Computer Software

OVERVIEW

- Current Support

- Why UTF8

- Language

- Tools

Current Support

- EDITTEXT/RICHEDITEXT methods GetUnicode and SetUnicode

- CONVERTUTF statement

- PRTOPEN flag PRT_FLAGS_UNICODE to detect UTF-16

- SQLIO provides nchar support through ODBC drivers

- CLIENT object methods SetUTF8Convert and SetUTF16Table

Why UTF8

- Fully compatible with 7-bit ASCII

- Standard for HTML

- Represents all 1,114,112 Unicode characters

- String sort order is preserved

- UTF8 is a byte stream

Why UTF8

- No unintentional ASCII control codes would exist in the byte stream

- Support exists across various Operating Systems

- Support in XML

- Support in JSON

- Support in relation databases

Language

- New NCHAR data type

- New N"data" string literal

- Maximum of four bytes need to represent one character

- NCHAR FP, LP, and PL represented as character positions

- Can be moved to/from DIM data type

Language

- No support for AND, OR, NOT, TEST, or XOR

- No support for COMPRESS, ENCRYPT, and ENCODE

- Move to DIM for these verbs

- Supported in most other verbs in a phased approach

- New F14 (Invalid UTF8 string) error

Language

- DIM and NCHAR can be mixed if DIM just ASCII

- UI objects detect NCHAR and switch to UNICODE mode

- Advanced printing support for PDF, direct, and preview

- Disk I/O support on READ and WRITE

Language

- Support for methods using NCHAR

- Support for GETPROP/SETPROP

- No support for old INSERTITEM/GETITEM statements

- LISTVIEW, STATTEXT, BUTTON, CHECKBOX, RADIOBUTTON complete

Tools

- Compiler now handles UTF8 bom on file

- New version of Code Editor that supports UNICODE

- IDE now can use UTF8 bom on file

- Source files that start with .utf8 are treated as UTF8

Tools

- Forms and XML forms can be in UTF8

- Future sunindex and sunaadex will handle utf8 uppercase

Questions?