Introduction

This chapter covers the following topics:

- About Code Pages and Unicode
- About Unicode and Code Page Support in Natural

About Code Pages and Unicode

A traditional code page is a list of selected character codes, arranged in a certain order, that support specific languages or groups of languages that share common scripts. A code page can contain a maximum of 256 character codes. For character sets which contain more than 256 characters (for example, Chinese or Japanese), double-byte code unit handling (DBCS) is used: DBCS code pages are actually multi-byte encodings, a mix of 1-byte and 2-byte code points.

Code pages have the inherent disadvantage of not being able to be used to store different languages in the same data stream. Unicode was designed to remove this restriction by providing a standard encoding for all character sets which is independent of the platform, program, or language used to access the data. With Unicode, a unique number is provided for every character.

A single number is assigned to each code element defined by the Unicode Standard. Each of these numbers is called a "code point" and, when referred to in text, is listed in hexadecimal form following the prefix "U". For example, the code point "U+0041" is the hexadecimal number "0041" (equal to the decimal number "65"). It represents the character "A" in the Unicode Standard which is named "LATIN CAPITAL LETTER A".

The Unicode Standard defines three encoding forms that allow the same data to be transmitted in a byte, word or double word oriented format. A "code unit" is the minimal bit combination that can represent a character in a specific encoding. The Unicode Standard uses 8-bit code units in the UTF-8 encoding form, 16-bit code units in the UTF-16 encoding form, and 32-bit code units in the UTF-32 encoding form. All three encoding forms encode the *same* common character repertoire and can be efficiently transformed into one another without loss of data.

In the context of Natural, we are concerned with two of these encoding forms: UTF-16 and UTF-8. Natural uses UTF-16 for the coding of Unicode strings at runtime and UTF-8 for the coding of Unicode data in files. UTF-16 is an endian-dependant 2-byte encoding; the endian format that will be used depends on the platform. UTF-8 is a 1-byte encoding.

For a complete description of Unicode, see the Unicode consortium web site at *http://www.unicode.org/*.

Note:
For obtaining information on Unicode code points, you can use the `SYSCP` utility which is available with Natural for Windows.

About Unicode and Code Page Support in Natural

For Unicode support, the Natural data format U and specific statements, parameters and system variables are used. For details, see the remainder of this documentation.

Most existing data is available in code page format. When converting this data to Unicode, it is required that the correct code page is used. Natural provides the possibility to define the correct code page on several levels:

- The system code page is used if a default code page is not defined in Natural.
- The default code page is used when the Natural parameter `CP` is defined; this overwrites the operating system's code page.
- The object code page which is defined, for example, for a source overwrites the default code page for this object.

When using Unicode strings and code page strings in one application, Natural performs implicit conversions where necessary (for example, when moving or comparing data). Explicit conversions can be performed with the statement `MOVE ENCODED`.

In most cases, existing applications which do not require Unicode support, will run unchanged. Changes can be necessary if the existing sources are encoded in different code pages. For more information, see *Migrating Existing Applications* later in this documentation.

It is not possible to run an existing application and also support Unicode data without any changes to the application. The Natural data format U has to be introduced in the application and it will most probably not suffice to simply replace the A format definitions with U format definitions. All code which assumes a specific memory layout of strings (for example, `REDEFINE` from alphanumeric to numeric format) has to be adapted.

Unicode characters are not permitted within variable names, object names and library names.

Unicode-based data are supported for Adabas.

Natural uses the International Components for Unicode (ICU) library for Unicode collation and conversion. For more information, see *http://icu.sourceforge.net/userguide/*. See also *ICU Library* later in this documentation.