CFICU - Unicode and Code Page Support

This Natural profile parameter is required to enable Unicode and code page support for various Unicode settings, for example, if U format variables or the statement MOVE ENCODED are to be used. It corresponds to the NTCFICU macro in the parameter module NATPARM.

See also *Profile Parameters* in the *Unicode and Code Page Support* documentation.

Possible settings	See syntax, below.	Possible subparameter keywords: See *Keyword Subparameters*, below.
Default setting	ON or OFF	Enables or disables the Unicode and code page support. The default value is OFF if profile parameter CP is set to OFF. Otherwise, the default is ON.
Dynamic specification	yes	The parameter CFICU can only be specified dynamically. In NATPARM, use the macro NTCFICU.
Specification within session	no	

CFICU=ON is enforced if profile parameter CP is set to any value other than OFF.

The following topics are covered below:

- CFICU Parameter Syntax

- NTCFICU Macro Syntax

- Keyword Subparameters

- Example of NTCFICU Macro

- Example of Dynamic Parameter CFICU

CFICU Parameter Syntax

The CFICU profile parameter is specified as follows:

```
CFICU=(ON,COLLATE=value,COLNORM=value,CNVNORM=value,LOCALE=value,CPOPT=value,DATFILE=value,BPONLY=value)
```

Or:

```
CFICU=(OFF,COLLATE=value,COLNORM=value,CNVNORM=value,LOCALE=value,CPOPT=value,DATFILE=value,BPONLY=value)
```

Or:

```
CFICU=[ { (ON)  } ]
          (OFF)
```

NTCFICU Macro Syntax

The `NTCFICU` macro is specified as follows:

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
        NTCFICU ON/OFF,                                                *
              COLLATE=value,                                           *
              COLNORM=value,                                           *
              CNVNORM=value,                                           *
              LOCALE=value,                                            *
              CPOPT=value,                                             *
              DATFILE=value,                                           *
              BPONLY=value
```

Or:

```
NTCFICU ON
```

Or:

```
NTCFICU OFF
```

Keyword Subparameters

COLLATE | COLNORM | CNVNORM | LOCALE | CPOPT | DATFILE | BPONLY

COLLATE - Collation Services

Collation is the process of ordering units of textual information (alphabetic sorting). Collation is usually specific to a particular language.

Examples: The character "Ä" is sorted in german locale between A and "B", but in Swedish locale it is sorted after "Z". In Lithuanian, "y" is sorted between "i" and "k".

This subparameter determines the collation service used.

ON	Use Locale ID and ICU's collation services to compare Unicode strings. This is the default value.
OFF	Use ICU's simple Unicode compare.

COLNORM - Normalization Check of Collation Services

Normalization is the process of removing alternate representations of equivalent sequences from textual data, to convert data into a form that can be binary-compared for equivalence.

This subparameter is honored only if `COLLATE=ON` is set.

The ICU Collation Service handles un-normalized text properly, producing the same results as if the text were normalized. This maximizes performance for the majority of text that does require normalization. If Unicode data is known with certainty not to contain un-normalized text, then even the overhead checking for normalization can be eliminated.

`ON`	Check for un-normalized text.
`OFF`	Disable check for un-normalized text. This is the default value.

CNVNORM

The German character "ä", for example, can be represented in Unicode as `U+00E4` or with the use of a combing character as `U+0061`, `U+0308`. Conversion to a codepage considers the combined "ä" (`U+0061 U+0308`) as two code points and produces an "a" and a substitution character, if `U+0308` is no valid character of the target code page. Normalization before conversion creates one code point `U+00E4` from the combined code points `U+0061 U+0308` and the subsequent conversion will deliver the result "ä". The parameter is honored whenever a conversion from U to A format is performed, for example `MOVE U TO A` or `DISPLAY U`, when the output device is a terminal emulation. The additional operation consumes of course additional storage as well as additional CPU time.

If `CNVNORM=OFF`, the `MOVE NORMALIZED` statement can be used to normalize selected strings.

This subparameter activates/deactivates normalization before conversion..

`ON`	Enable normalization before conversion.
`OFF`	Disable normalization before conversion. This is the default value.

LOCALE - Locale ID

The Locale ID is used by ICU's Collation Service to consider language and even region-dependent features of collation. The language code of the Locale ID follows ISO639, and the region code follows ISO 3166.

Possible values	`LLL_RRR`	`LLL` is a 2- or 3-byte language code of lower-case characters. If specified in upper case, it will be translated into lower case automatically. `RRR` is a 2- or 3-byte region code of upper-case characters to classify the language.
Default value	`en_US`	

Examples:

en_US	English (United States)
en_UK	English (United Kingdom)
de_DE	German (Germany)
de_AT	German (Austria)
de_CH	German (Switzerland)
sv_SE	Scandinavian (Sweden)

CPOPT - Fast Code Page Conversion

By default, a conversion from alpha to Unicode format and vice versa is performed by calling ICU functions. Certain code pages are mapping characters to Unicode with 1:1 relationship. In this case, the conversion performance can be increased by using internal translation tables rather than ICU functions.

ON	Use internal translation tables instead of ICU functions, if possible.
OFF	Use ICU functions in any case. This is the default value.

DATFILE - Additional Data Files

Optional data file name. It must be loadable by using RCA technique. The data file contains the converter mapping tables, collation rules, break iterator rules and other locale data. The ICU development kit provides tools to build data files that comply with particular requirements. Refer to the chapter *Data Management* in the *ICU User Guide* for more information.

Possible values	*name*	The data file name can be up to 8 characters long.
	OFF	Removes any data files defined, that is, the default data file is used.
Default value	None	No additional data files are defined. The default data file is used.

BPONLY – Use of ICU Buffer Pool

In thread environments (for example, under CICS) an ICU buffer pool can be used to share ICU data and to reduce the overhead caused by thread compression and roll-out. The keyword subparameter BPONLY determines whether the ICU must use the ICU buffer pool or if it can use thread storage if the ICU buffer pool is not available.

For further information, see *ICU Buffer Pool* in the *Unicode and Code Page Support* documentation.

ON	An ICU buffer pool must be used. Initialization messages NAT3419 and NAT3410 are displayed if the ICU buffer pool cannot be used, and the ICU initialization will be terminated. An error is displayed if ICU is to be used during the session, but ICU is disabled.
OFF	If an ICU buffer pool is not available, ICU will use thread storage. Initialization message NAT3419 will be displayed if ICU uses thread storage instead of an ICU buffer pool. This is the default value.

Example of NTCFICU Macro

```
NTCFICU COLNORM=ON,LOCALE=de_DE,DATFILE=TEST15
```

Example of Dynamic Parameter CFICU

```
CFICU=(COLNORM=ON,LOCALE='de_DE',DATFILE=TEST15)
```