

16 章 その他

16.1 POSIX(注)のロケールに対応した地域・文化固有機能

本解説の第4章で「漢字等の多オクテット文字処理機能」の説明があります。その部分を読んだ読者の多くは、日本語を扱う上での機能を想定して読んだかと思いますが、この機能は日本だけの為では有りません。日本は勿論ですがアジアを中心とした2バイト文字文化圏を対象としていると共にそれだけではなく、文字コードは1バイト内に収まるが、これまでの標準化COBOLの中では既定の無かった非英語圏のヨーロッパ初め諸国での利用も踏まえた、国際化規格になっています。この節では新規格の、国際化対応、地域・文化固有機能について解説します。

(注) POSIXとは...(Portable Operating System Interface for UNIX)

IEEE(アイトリプルイー、米国電気電子学会)によって定められたものです。後にこの規格はISO(国際標準化機構)によって国際規格として定められています。UNIX標準化に端を発した、システム機能の標準化規格であり、この仕様に準拠する事でポータビリティ確保に効果があると考えられます。米国連邦政府が調達規格に採用した事も有って、各方面で準拠すべき規格と意識、対応されています。

16.2 国際化、他国語対応とは何か

元々国際化、他国語対応とはどのような仕組みの事でしょうか。

- (1) 注釈、定数で各国語が扱える
従来も殆ど問題は無かった。
- (2) プログラミング言語内で使用可能な文字の拡張
開発の生産性・信頼性・保守性の向上
- (3) 国際化プログラム開発が容易化
- (4) 符号系変換機能があり、ファイル変換等が省力化できる

上記は扱える文字コードセットや、符号系に関わる機能が殆どですが、国際化とは下記のような機能も含まれます。

- (5) 文字の照合順序
- (6) 通貨記号、金額の単位

(7) 数字の表現

(8) 日付・時刻の表現など

この様に文化的 (国・地域・言語・習慣による違い) な相違も標準規格において考慮することが重要との認識から、マルチオクテット文字処理機能をより一般化し、"国際化機能"と呼んでいます。

ちなみに新規格の言葉では有りませんが、国際化の事を「i18n」と呼ぶそうです。これは国際化を表す internationalization の先頭「i」と最後の「n」、そしてその間に 18 文字のアルファベットが有る事からです。また関連用語として、多言語化(multilingualization) : m17n、地域化(localization) : l10n と言うのも有ります。これらの言葉は COBOL 言語以外の国際化の話題中でも比較的広く使われている様です。

16.3 ロケール

地域、国際化の指定の為に、言語規格として次の様な機能を持たせています。この指定機能の事を地域固有仕様 (ロケール) と呼びます。

(1) ロケールの分類

ロケールは次の様な分類として定義されています。

分類	意味
LC_COLLATE	比較順序
LC_CTYPE	文字分類と大・小文字変換規則
LC_MESSAGES	情報、対話メッセージと応答の形式
LC_MONETARY	通貨表現
LC_NUMERIC	数値表現、形式情報(小数点文字など)
LC_TIME	日付、時刻表現形式
LC_ALL	上記ロケールの全てを含むカテゴリ

(2) 特殊名段落

特殊名段落で次の指定が出来ます。

LOCALE { 分類名 } IS 地域固有仕様名
 { 定数 }

(3) SET 命令で地域固有仕様の値を設定又は参照できます。

(4) EC-LOCALE... ロケール関連の例外です。

(5) 新設の組み込み関数

BYTE-LENGTH, CHAR-NATIONAL, LOCALE-COMPARE, LOCALE-DATE,
LOCALE-TIME, NATIONAL-OF

(6) 機能拡張された組み込み関数

LOWER-CASE, UPPER-CASE

16.4 国際化の具体例

具体的な例として国際化機能を持った COBOL コンパイラとはどのようなものでしょうか。

例えば貴方は有るユーザ向けに開発済みのシステムを持っていて、これは英語のみを使っていたとします。このシステムを日本語化(例えば SJIS)する場合、どのような修正が必要になるでしょうか。

まず第 1 に画面表示メッセージや、印刷用定数等は書き換える必要が有ります。修正量が多ければその部分はプログラムの外に切り出して、メッセージファイル化しておく事は、後々にも役立ちます。

英語機能の部分も残して置きたければ「コンパイラ指示機能」による、コンパイル時選択実行の指定機能が有用です。また幾つかの場所では日本語(2バイト)データとして扱いたい場合が有るかも知れません。その場合は、データ定義を日本語項目(PIC N)とします。手続き部はどうでしょうか。

実は手続きは基本的に何ら改造を必要としないのです。扱うデータが英語なのか、日本語なのかによって取るべき動作が変わる命令は数多く有ります。例えば次の STRING 命令を例に、説明します。

```
STRING data-1 DELIMITED BY SPACE  
      data-2 DELIMITED BY SIZE  
      INTO data-3  
      WITH POINTER counter-1
```

この時に data-1,2, および 3 が 1 バイト英数字項目(PIC X)の定義で有れば、この命令動作は 1 バイト文字単位での動作になります(POINTER 句で指定されたカウンタ・データは、1 バイト文字としての文字数で扱われる)。

そして data-1,2, および 3 が日本語項目(PIC N)の場合は、2 バイト文字単位での動作、すなわちカウンタ・データは、2 バイト文字としての文字数で扱われる訳です。これにより日本語を扱うプログラムで有っても、1 バイト文字のみ扱っていたものから、手続き部

に関する改造は要らない事になります。

そしてこのソース・プログラムをコンパイルする時の操作として、英語部分ではなく日本語部分を有効にする、日本語メッセージファイルを利用する、また日本語項目の中身は S J I S として扱う、等の指示情報をコンパイラに与える必要が有ります。その為にコンパイラに次の指定を入力します。

```
LC_ALL    Japanese_Japan.932
```

これは現在 Windows で用いられている方式をベースに想定した、1つの例です。下線の前は言語、下線とピリオドの間が国、最後がコードページと言う符号化文字集合・方式を表しています。

それでは次の段階として、上記プログラムを別国語対応させる事を想定します。例えばこのシステムを利用しているユーザー企業が、中国・香港支店を開く事になったとします。同じアプリケーションを、日本だけでなく中国の人でも使いたい訳です。勿論画面表示のための定数や印刷用ヘッダ等は中国語の物に差し替える必要が有ります。上で説明したとおり、メッセージファイル等に分離して有れば、プログラムの修正は殆ど無いでしょう。また PIC N 等の定義は修正不要です。

そしてプログラムをコンパイルする時に、次のロケールをコンパイラに指示します。

```
LC_ALL    Chinese_China.950
```

最後の「950」は代表的な中国語文字セットの1つである「BIG5」を示しています。同時に中国での代表的な通貨記号や日付、時刻表現等も上記指定だけで、自動的に行われると思って良いでしょう。つまり本当に僅かな改修のみで他国語対応のシステム構築が出来る事になります。もちろん具体的な実装においてはもう少し複雑な部分、考慮を要すべき事柄等があるかも知れません。ですが基本的な事は、ここで述べた部分で表現可能です。ますますグローバル化が進んでいる社会、そしてその中で求められる情報技術（IT）の上で、今後国際化の重要性が増加していく事は間違いないと考えられます。その様な要求に応える、規格になっていると言えるのではないのでしょうか。

16.4 国際化のまとめ

国際化の説明の最後に、規格化に携わった方々の論文から引用する形で、国際化規格制定の上でどのような点を考慮し、苦心したのかを国際化の流れ、意義等を汲みながら見ておきたいと思います。COBOL2002 国際規格の中の国際化機能は特に、日本から標準化に参加した方々の努力によって実った成果です。標準化に労された方々に心から感謝致します。

「COBOL 日本語機能の一般化による、国際規格化提案と国際化機能の開発」

< 標準化前の状況 >

日本語化機能は実装が先行していた。各社が未統一のまま実装した為に非関税障壁と海外から批判される危険性が有り日本のみでの標準化は避けるべき状況であった。

このため国内だけの JIS にはせずに、日本から国際規格として提案することにした。

< 主要課題 >

世界各国の賛同が必要である。

日本固有の地域性を排除した公平な汎用仕様にしなければならない。

既存の日本語機能を使用した膨大な資産からの移行性を確保する必要がある。

< 仕様作成方針 >

- (1) 使用可能な文字を世界各国・地域の文字に拡張する。
- (2) 世界の多くの符号系に対応可能な枠組を設定する。
- (3) 既存資産との共存を重視し、符号系切替のシフトコードの存在を意識しない方式とする。
- (4) 文字データ型として各国文字項目を追加し、その機能（定義と操作）は既存の英数字項目と同等とする。
- (5) 各国・地域に適用可能な国際化プログラムの記述が可能な仕様とする。
- (6) 既存の日本語仕様機能の中で使用頻度の高いものは国際規格仕様の核となる構文として採用する。
- (7) 日本特有の機能や名称、時代制約的な機能、ハードウェア依存性の高い仕様を排除する。

1 つは、世界各国の賛同を獲得するために日本という地域性を排除することである。このため、中国や韓国などのマルチオクテット文字を使っている国・地方はもちろんのこと、アルファベット 26 文字以外のヨーロッパ各国文字（ドイツ語のウムラウトやロシア文字など）を含む多種多様な文字とそれらが規定される多くの符号系を対象とする"普遍的な枠組み"を定めることにした。

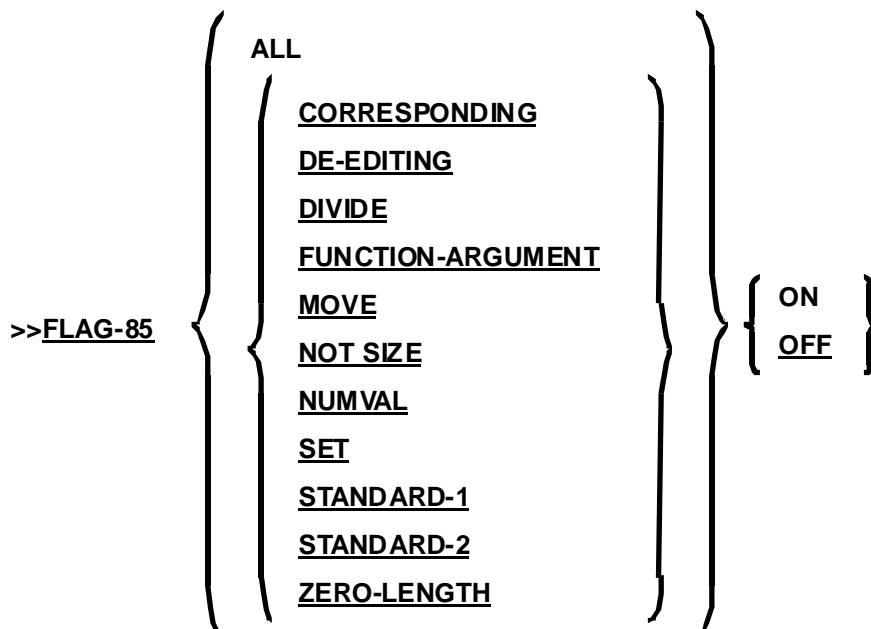
2 つ目の課題は、利用者のために日本語機能を用いた既存資産の移行性を確保することである。

< 結果的に提案した国際化規格案 >

- (1) 対象とする文字を、中国、韓国、タイ、ロシア、ドイツ、北欧、東欧等アルファベット 26 文字以外の世界各国の文字に拡張するとともに、多くの符号系に対応可能な普遍的な枠組を設定する。
- (2) 既存の文字型 (英数字項目) に加え、漢字などのマルチオクテット文字を扱う国別文字 (national character) 項目を新設し、機能は英数字項目と同等とする。特に同じ業務プログラムを各国ごとに適用するときには、データ定義だけを修正しロジックの変更が不要で済むように国際化仕様を設定する。
- (3) 国内で使用頻度の高い日本語処理機能を選定し、それを国際化機能の核となる仕様として採用することにより、既存資産から国際規格仕様への移行性を実現する。

16.5 既存プログラムとの互換

COBOL2002 では「FLAG-85」コンパイル時指示機能によって、前規格との互換性に問題が出るかも知れない機能について、警告メッセージを表示する方法が提供されます。



(1) CORRESPONDING

CORRESPONDING 句指定の ADD, MOVE, または SUBTRACT 命令に於いて、定数以外の添え字付けが有る時、警告する。

(2) DE-EDITING

数字編集項目の逆編集が必要になる MOVE 文を、警告する。

(3) DIVIDE

REMAINDER 付きの DIVIDE 文で、商のデータ項目が符号無しであり、除数または被除数が符号付きの場合警告する。

(4) FUNCTION-ARGUMENT

関数 RANDOM の引数に算術式を指定し、その結果を組み込み関数の引数にしようとする時、関数 RANDOM に警告を出力する。

(5) MOVE

英数字の定数又はデータが数字項目に転記され、その桁数が 31 桁を越えるとき警告を出力する。

(6) NOT SIZE

算術命令で NOT SIZE ERROR が指定され、SIZE ERROR 句の指定が無い場合警告する。

(7) NUMVAL

組み込み関数 NUMVAL と NUMVAL-C は警告する。

(8) SET

条件設定の SET 文が可変長集団項目を対象にしている場合、警告する。

(9) STANDARD-1

特殊名段落の ALPHABET 句に STANDARD-1 の指定がある場合、警告する。

(10) STANDARD-2

特殊名段落の ALPHABET 句に STANDARD-2 の指定がある場合、警告する。

(11) ZERO-LENGTH

長さゼロになりうるデータ定義を警告する。

(12) ALL

上記 (1) ~ (11) を全て含む。