

4E-5

マルチ・ウィンドウ環境における、任意複数コードセット
の混在可能な、出力機構の一般化田中 忠雄¹ 片岡 裕² 小原 啓義¹

Interoperability および Computerbility の拡張のため、Multi-window 環境下における Multilingual I/O System のモデル化と実証を、X Window System を用いて行なってきた。本研究の今日までの成果は、Multilingual Input Method 及び、コードポイントと出力図形が1対1に対応する場合の Multilingual Output Method が、X11R5 の Sample Implementation としてリリースされている [1]。今日までの研究では、主に入力機構の一般化を行ない、入力機構は、あるコード系の任意のコードポイントの列の生成であり、次の手順での一般化が示された。

1. 有限オートマトンを用いた、表音文字、表音記号の生成
2. context dependent な変換機構による表音文字、表音記号からの表意文字の生成

同様に出力機構は、任意のコードポイントの列に対応する最終表示図形の生成であるが、最終表示図形がコード系のデザインと、文字の構築と表記に依存するため、一般化が困難であった。本研究では、世界の著名な文字とコード系を調査し、表示機構の一般化を行なった。文字の認識は、文字の図形によって行なわれるため、出力機構は、可能な限り正確に文字を表示しなければならない。

文字は、その表記手法によって、表示位置非依存性文字群 (Position independent scripts) と Arabic 等の表示位置依存性文字群 (Position dependent scripts) に分類される [2]。さらに文字は、表示方向に関して、文字自身が表示方向を決定する文字群 (左から右、右から左、上から下)、文字の表示方向が現在の表示方向に従う文字に分類され、各々、表示方向非依存性文字群 (Direction independent scripts)、表示方向依存性文字群 (Direction dependent scripts) と呼ぶ。さらに、表示方向依存性文字群は、表示方向に依存して図形が変化する文字群と、変化しない文字群に分類される (variable form scripts、non-variable form scripts)。従って、文字の出力では、表示の原点を明示する必要があると同時に、出力機構に対し、現在の表示方向 (Current direction) を明示する必要がある。表示の原点は、左端、右端、上端があり、現在の表示方向は、左から右、右から左、上から下がある。

コード系は、図形文字集合と機能文字集合から構成され、最終表示図形集合は、必ずしも図形文字集合のみのコードポイントと1対1に対応しない。ここで、特定のコード系の図形文字集合と機能文字集合の組によって示される文字集合の要素が、最終表示図形集合の要素に写像される関係を明示しなければ、そのコード系が表示し得る文字、すなわち最終表示図形集合を定義したことにはならない。Multi-window 環境では、表示ルーチンは、独立の事象として呼ばれ、必ずしも行の先端から後端までの文字列を表示するとは限らないため、最終表示図形集合を決定するために使用される機能文字群は、次のようにまとめられる。

Generalization of Multilingual Output Mechanism in a Windowing Environment.

Tadao Tanaka, Yutaka Kataoka, Hiroyoshi Ohara

1 School of Sci. & Eng. Waseda University. 2 Center for Informatics, Waseda University.

- 複数のコードポイントから1個の文字を合成するための文字 (BS)
- 文字の位置依存性の接続を切るための文字 (Discontinuity character)
- 1音節文字の区切りを示すための文字 (Disjoint indicator)
- ligature を生成するための文字 (Ligature start、Ligature end)

これらの文字を、特に限定符合文字 (Determinative characters) と命名した。限定符合文字は、他のコードポイントを引数とし、最終図形集合へのインデックスを返す関数であると定義できる。

Arabic 等の表示位置依存性文字は、文字列中の文字の位置により字形が変化する。これらの文字のコードポイントの最終表示図形への写像は、そのコードポイントの直前のコードポイントと、現在及び直後のコードポイントの取り得る結合情報が必要となり、これら3コードポイントの結合情報から最終表示図形集合の要素を求めることができる。当然、正しく表示され得るコードポイントの単位が存在し、その単位以外では、正しく表示されない。

上記の情報を明確に定義し、任意のコードポイント列に加えて出力機構に渡すことによって、正しい最終表示図形集合の要素を求めることが可能となる。すなわち、これら上記の情報を、外部に表として持ち、予め出力機構に渡すことによって、コード系に固有の情報を、出力機構そのものにプログラムする必要はなくなり、一般化が行なえる。

現在、本研究では、上記の情報を記述する定義表の形式を定め、有限オートマトン・インタプリタによって、コードポイントの列と表を対応させ、任意の複数言語の同時出力の実証を行なっている。開発環境は、X11R5 を用い、上記の表には、stateful encoding codeset とそれに対応する任意の stateless encoding codeset 間の相互変換を含めた記述を可能としている。この一般化コード変換機構と一般化出力機構は、現在の X11R5 の対応する機構を置き換える実装方式を採用した。また、この表を元に、入力機構に対し、各コード系のエンコーディング・スキームを渡すことで、入出力間の整合を取り、Interoperability を保証することが可能となる。

[1] A Model for Input and Output of Multilingual Text in a Windowing Environment.
ACM TOIS, Vol. 10, No. 4, Oct 1992, PP 438-451. Y. Kataoka, M. Morisaki, H.
Kuribayashi, H. Ohara