TEX の Unicode 対応

ZR

2009 年 7 月 20 日

1　欧文TEX において

Unicode の登場以前から、TEX ではアクセント付のラテン文字を「文字とアクセントのグリフを合成する」ことで表現し、それにより多くの欧州の言語を出力する能力をもっていた。さらに入力文字列をマクロで処理することにより、ラテン文字以外の文字に関しても、TEX のソース上での文字の表現方法（ASCII 文字での翻字、8 ビットのコードページの利用、等）と必要なグリフを収めたフォントさえあれば出力が可能であり、実際、多くの言語を扱う為のLATEX のパッケージが公開され、中には、アラビア語のような複雑なグリフ処理を要するものや、日本語のような巨大な文字集合と多バイトコード入力を扱うものもある。

TEX を Unicode に対応させる試みは、Unicode が広く知られるようになった 1990 年代半ばにまで遡る。Yannis Haralambous と John Plaice により開発されていた Omega（Ω）は Unicode（当時は BMP のみ）の利用を想定して内部処理を 16 ビットに拡張したTEX であった。Omega は多くの言語における組版の慣行を実現するための新しい機構（例えば、日本語を縦書きで出力する、アラビア語のグリフ処理を内部で行う、等）を備えることが予定されていたが、今世紀に入り開発は長らく停止していた。

一方で、従来のLATEX で UTF-8 入力を扱うパッケージ、例えば inputenc の utf8 エンコーディングや、より高機能な ucs パッケージ（utf8x エンコーディング ; Dominique Unruh による）も開発されている。これらを用いれば、LATEX の標準の出力符号化処理（fontenc）の枠内にある文字を、UTF-8 入力により出力できるが、マクロ処理であるが故の制約も多々ある。

放棄されていた Omega の開発は、その後 2005 年に新しいプロジェクトである LuaTEX（開発は Hans Hagen、Hartmut Henkel、Taco Hoekwater による）に取り込まれる形で行われることとなった。LuaTEX は、現在の欧米の標準である pdfTEX（PDF 出力が可能なTEX）に Omega を併合し、さらに Lua スクリプトによる内部処理の調整を可能にした処理系である。これとは別に、既存の Unicode テキスト処理技術とTEX を融合させるというアプローチで Unicode に対応したXETEX（開発は Jonathan Kew による）も開発されている。すなわち、TEX の Unicode への対応は現在のところ着実に進んでいるといえよう。

2　和文TEX において

] 現在日本では、日本語を扱う TeX 処理系として、アスキー社が開発した pTEX（p は publishing の略）が広く使われている。これは、旧来（Unicode 以前）の多バイトの日本語符号系（Shift JIS や EUC-JP）を扱うために内部処理を 16 ビットに拡張したものであり、日本語コードの文字（和文文字）が内部でも「文字」そのものとして扱われる。また、高品位な日本語の組版の為に必要な処理（禁則処理、「四分空き」の自動挿入、

等）が内部処理で行われる。従って、マクロによりこれらの処理を行う方法と異なり、他のマクロ処理と干渉することがなく、欧文TEX 用のパッケージのほぼ全て（8 ビットコード入力を利用したもの以外）を変更なく利用できるという特長をもっている。

しかし、日本でも Unicode の普及が進み、Unicode を基本の文字コードとするシステムが現れるに至って、pTEX が UTF-8（標準的な Unicode の符号化方式）を扱えないことが次第に不便に感じられるようになってきた。また、世界的な流れとして、Unicode を扱わないソフトウェアが「時代遅れ」の烙印を押されて排除される風潮があり、それへの危機感も認識されるようになった。このことへの解決策を示したのが、日本語対応のTEX ディストリビューション ptetex3 の開発者である土村展之氏である。彼は、pTEX の内部漢字コードはそのまま（Shift JIS か EUC-JP）にして入出力部分においてコード変換を施すという方法をとっている。これにより、pTEX のツールの UTF-8 への対応が比較的簡単に行うことができたが、一方で、内部漢字コードの制約の為に、この方法は扱える文字集合が従来と同じ JIS X 0208 に限られるという欠点を持っている。

これとは別に、Unicode の「文字集合」をコード値入力により扱おうとする試みも行われた。齋藤修三郎氏の OTF パッケージは Unicode（あるいは Adobe の CJK グリフ集合）に含まれる「漢字に類する文字（全角幅の文字）」をコード値入力により出力することを実現している。これにより、pTEX において JIS X 0208 にない漢字や異体字を出力したいという要求に応えることが可能になった。

この 2 つの試みおよびそれを巡る議論は、「Unicode 化された pTEX」に対する 1 つの形（目的・仕様・実装方法に関して）を導き出したと考えることができる。仕様に関しては、「欧文部分はそのままで、和文の内部コードを Unicode に変更する」というものである。そしてついに田中琢爾（ttk）氏によって、「Unicode 化された pTEX」が実装された。それが upTEX である。