

ソフトウェア国際化入門

国際化 JP 株式会社 代表取締役
末廣 陽一 氏

皆さんは、「国際化」というとどんなことを思い浮かべるでしょうか？原油価格の高騰や円ドルの為替レート、TOYOTA や SONY のように世界的なブランドを確立したグローバル企業、ニュースのスポーツコーナーの最初に出てきた日本人メジャーリーガーの話題など、人それぞれいろいろなところで「国際化」を感じているかもしれません。世界は急激に「国際化」の道を突き進み、今ではそれぞれの国の経済や日々の生活のあらゆるところに外国の影響が及んでいます。影響は双方向で、外国から日本が影響を受けている面もあれば、日本が外国に影響を与えている面もあります。

IT 業界でも「国際化」は着実に進行しています。残念なのは、日本発のソフトウェアで世界中で使用されている、これという目立ったものが見当たらないことです。Windows、SAP、Oracle どれも外国の企業が作ったものです。広くソフトウェアという分野で考えると、日本企業が強いゲームや家電機器向けのものもありますが、これだけ普及してきたコンピュータの主要ソフトがほとんど外国製というのも、なんとも不思議な話です。「日本の会社もがんばれよ」と言いたくなるのですが、「国際化」とはそういうことを受け入れることでもあります。大相撲の外国人力士の活躍を見ながら、「日本人力士ももっとがんばれ」と応援するのとあまり変わらないのかもしれませんが、それこそ土俵が全世界のマーケットとなりつつあるソフトの世界では、同じ機能のものでも、早く、安く、効率的な開発を行うことが大きな課題であり、現在高いシェアを持っているソフトの会社は、早くから「国際化」に取り組んできました。

本稿では、グローバル IT 企業での実務を含め、私が約 20 年近くにわたって関わってきたソフトウェアの「国際化」について、その歴史や手法などを簡単に振り返るとともに、国際化作業における困難とツールによる効率化について説明したいと思います。

IT 分野において「国際化」が広く受け入れられるためには、それを支える技術の進歩と法律などの規制や生活環境やビジネス環境などのマーケットの受け入れ態勢が整うことが必要となります。コンピュータのソフトウェアエンジニアと言うと、英語のマニュアルを読まなければならない時代はそれほど昔のことでもありません。さらに古くは、コンピュータで漢字を満足に扱うこともできていませんでした。当時はコンピュータを使う人も限定されており、マニュアルが英語でも、漢字が扱えなくても十分に役に立っていたのです。今では、日本の文字だけでなく世界中の文字を画面に表示したり、印刷したりすることができるようになりました。デバナガリのこんな文字も「㊦」、ハングルのこんな文字も「ᠠ」日本語の文字と一緒に文書中に入れることができたりします。コンピュータが、複雑な数値計算を行うだけのものではあれば、漢字を使う必要もなかったことでしょう。海外との行き来や、やりとりがこんなに盛んにならなければ、郵便や FAX で十分だったかもしれません。今では、物理的なモノの移動以上と言ってもよいくらい、映像、音声、テキスト、データ、ソフトウェアなどあらゆるものが IT 技術を利用して国境を越えてやりとりされています。

IT 分野の「国際化」において大きな役割を果たしてきたのが標準化です。ISO などの国際標準や各国の国内標準(日本の場合は JIS)、さらに、世の中で広く使われ事実上の標準となっているものもあります。ソフトウェアの国際化で重要な標準といえば、文字コードの標準でしょう。また、近年の国際化の流れに大きな影響を与えた標準と言え、インターネットと WWW でしょう。通信の TCP/IP、メールの SMTP、Web の HTTP など、様々な標準仕様が広く世の中に普及し利用されるようになりました。一般の PC ユーザはこれらを特に意識することはありませんが、Web ブラウザで海外の Web ページを見たり、メールソフトで国内・国外関係なしにメールのやりとりができる

のも、標準化された規格に従って通信したりデータのやりとりができるような仕組みがあつてのことです。

それでは、ここからさらに範囲を狭めて、ソフトウェアの「国際化」について説明します。ソフトウェアの国際化というときに使う「国際化」にあたる米語は、Internationalization です。敢えて米語と書きます。イギリス式英語では、Internationalisation です。スペルの違いにお気づきでしょうか？最初から 15 文字目が「z」と「s」の違いがあります。しかし、長い単語です。国際化の議論をしているときに、毎回このスペルを書くのが面倒になって、縮めて書く方法が考案され、I18N (英語ではアイ エイティーン エヌと読みます。) という表記法が生まれました (参考：共立出版「国際化プログラミング」)。IとNの間に 18 文字あるのです。うまいことにこう表記すると米語も英語も関係ありません。ソフトウェアの国際化技術は、まずヨーロッパにおいて先行して研究され開発が進みました。1980 年代の中盤から後半にかけて欧米のコンピュータメーカーが中心となり議論されました。ISO 8859-1 という国際標準の文字コードを使えば、英語、フランス語、ドイツ語など主要な西ヨーロッパ言語の文字を、環境を切り替える (違う種類の端末を使うとか、違う文字コードを扱うように設定を切り替えるなど) ことなく扱うことができました。ただ、それぞれの国で使われている通貨記号や日付、曜日の表記方法は異なっていました。それで、これらの国ごとに異なる情報を切り替えるための仕組みが考案されました。それが、C 言語標準や後に POSIX などの OS の標準に取り入れられたロケール (locale) の概念です。この仕組みの基本は、同じプログラムが、実行環境に応じて動作を変えることにあります。そして、OS が提供するライブラリ関数などがそのプログラムの動作を変える仕組みを提供し、アプリケーションプログラムの開発者を複雑で面倒な言語や文化・慣習に依存したプログラミングから解放したのです。一つのプログラムが、実行環境の設定を見ながら、英語版として動いたり、フランス語版として動いたりする仕組みができました。同時期に開発されたのが、メッセージ機構 (messaging system) です。プログラムが出す質問やエラーメッセージなどをいろいろな言語で出せるように切り替える仕組みが考案されました。プログラムの動作す

る環境に合わせて、英語、フランス語、ドイツ語など、用意された言語のメッセージが表示されます。注目しているマーケットがアメリカと西ヨーロッパだけのうちは比較的簡単でした。アメリカやヨーロッパのプログラム開発者たちも、それぞれの馴染みのある言語ばかりで、文字やテキストの処理も大きな違いがありません。

「国際化」がホットになったのは、日本を中心とするアジア言語のマーケットを考慮し始めたときです。数千文字を一度に扱わなければならない上、文化・慣習に依存したデータ表現の中にもこれまでの枠組みで対応しきれないものが出てきます。一番苦勞したのが、漢字など 1 文字を複数バイトで表現する必要のあるマルチバイト文字への対応でした。詳細の説明は省略しますが、多くの企業が時間と労力をかけ、これまでヨーロッパ言語に対して実現してきた「国際化」の仕組みをアジア言語にまで拡張しました。¹

新たな「国際化」の仕組みでは、アジア言語を扱うために、文字データを扱うデータ型のサイズが大きくなりましたが、基本的に環境ごとに切り替える点は同じです。例えば、C 言語ではワイド文字 (wide character) を利用したプログラミング手法が開発されました。このころ、Internationalization と対になる概念として Localization が議論されるようになり、I18N と同じルールで L10N (英語ではエル テン エヌと読みます。) と表記されるようになりました。また、対象マーケットが広がったことにより、全世界の文字を表現するための文字コードを決めようとの機運が高まり ISO 10646 や Unicode といった世界中の文字を表現できる文字コード標準制定の動きが活発化しました。「国際化」の枠組みの、アジア言語対応から世界規模への拡張といった動きは、1980 年代の末から 1990 年代の前半に見られ、その後は各社の製品への実装が進みました。標準化活動に積極的に参加した大手コンピュータベンダの UNIX OS をはじめ、マイクロソフト社の Windows OS や、オラクル社のデータ

(1) 私が勤務していた DEC 社 (当時。現在は HP。) は、アジア言語への拡張と同時にヘブライ語やアラビア語など右から左に書く文字を持つ言語もカバーする標準的な手法の確立を提案していましたが、残念ながら受け入れられませんでした。標準化にはその時々世の中やビジネスの情勢が反映されるので、今議論していたら状況は変わっていたかもしれません。

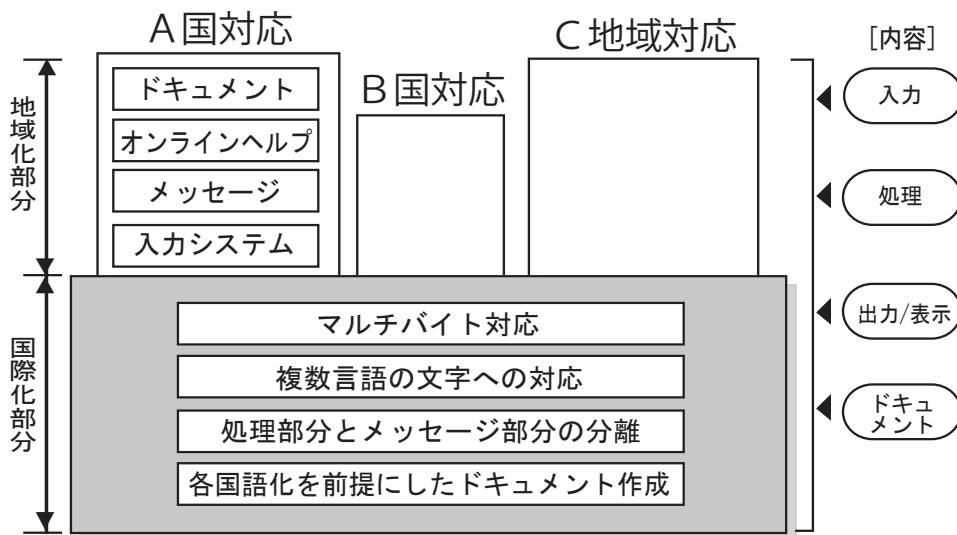


図1: 国際化されたソフトウェアの構成例 (共立出版「国際化プログラミング」より)

ベース製品などグローバル企業の主要製品のほとんどが1990年代後半には国際化対応を済ませました。そして、世界中から収益を上げるビジネスの仕組みを作り上げていったのです。図1に国際化されたソフトウェアの構成例を示しています。

かなり大雑把にソフトウェアの「国際化」の歴史を振り返りましたが、国際化はさらに進化しています。今現在の「国際化」がターゲットとするのは、間違いなく全世界です。それを実現するための標準や技術は揃ってきています。これから作る製品やシステムについては、まずI18Nを実現していることが前提で、必要なきにいつでもL10Nを提供して新しい市場でのビジネスを開始することができるように備えておくことが必要です。一方、国内市場だけを考えている製品であっても「国際化」とは無縁でいられない状況が出てきています。ファイルの文字コードや交換コードなどでUnicodeが利用される場面が多くなってきているためです。Unicodeを処理するだけでは「国際化」は実現できませんが、Unicode対応は「国際化」を考えるよい機会となるはずで

す。I18N(国際化)を実現するためには、ソースコードレベルでの作業が必要です。L10Nが、多くの場合翻訳テキストや現地環境への対応のためのデータの編集等、限定されたデータファイルのやり取りで実現できるのとは異なります。I18N作業の大変さは、実際に経験したエンジニアでないとうわりづら

いと思いますが、ソフトウェアの規模が大きくなればなるほど、元のソフトウェアを開発したエンジニアと同等のスキルが必要となってきます。I18Nが必要な場所というのは多岐にわたっており、そのソフトがどのような機能を持っているか、どのような実装方法をとっているかを把握しないと、I18N作業が必要な場所を特定することは困難だからです。例えば、I18N作業について、以下のような困難が待ち受けています。

■ 見積・調査・立案において

- ソースコードの量が多すぎて、どこでどのようなI18N作業が必要かを把握するのが非常に困難。
- I18N作業の実務経験がないために作業の難しさがわからず、工数の予測ができない。

■ 開発作業において

- I18Nするためにどのような修正を行わなければならないのかがよくわからない。
- I18Nの勉強をするための情報が不足していて系統立てて学習することが困難。

■ テストにおいて

- 元のソースコードが変更されるたびに、人手でチェックするのは困難。
- 勘に頼ったテストだけでは、問題点を発見するのが困難。

■ メンテナンスにおいて

- 国際化して日本語版をリリースすることで目標を達成したので、国際化チームが解散となってしまった。ソースコードだけは残っているが、作業の過程やノウハウが残っていないため、次のバージョンの国際化対応は困難。

これまでのI18N作業は、人に依存して行われてい

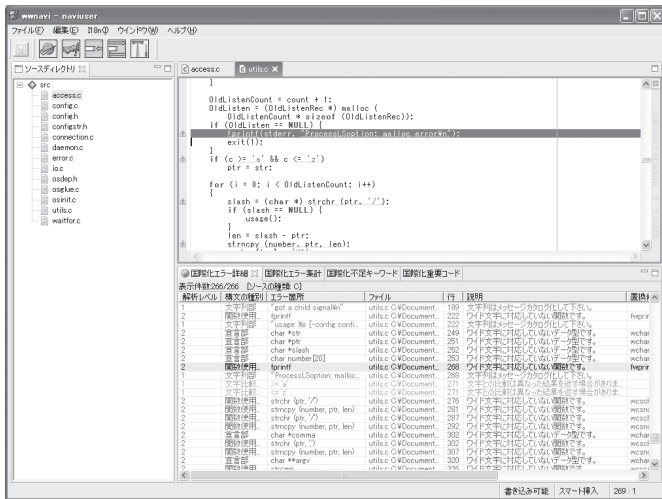


図 2：国際化ツール World Wide Navi の画面

る例が多く、それが大きな問題となっていました。人に依存したままだと、ノウハウの共有ができない、担当がいなくなると対応が困難、効率化が進みにくい、一時的な作業のために人員を確保する無駄が生じやすい、などなど開発の規模が大きくなればなるほどリスクが大きくなってきます。また、必要なスキルを持った人員を確保できないために、「国際化」が実現できないということも発生します。

国際化 J P 株式会社では、このような I18N 作業の困難を軽減するために、世界的に見てもユニークな国際化ツールの開発と、コンサルティング・トレーニング等を通じた国際化のノウハウの提供を行っています。国際化ツール World Wide Navi (ワールドワイドナビ) は、IDE(統合開発環境) ライクな GUI を通して、ソースコードを解析することにより、ソフトウェアの国際化作業の見える化、スピードアップ、効率化をはかります。会社やプロジェクトごとの事情に合わせた解析の設定ができるので、このツール自体に国際化のノウハウを蓄積していくこともできます。(World Wide Navi では、ルール等のデータ保存に XML を利用しています。ソフトウェアの機能拡張やカスタマイズ機能などに柔軟に対応するため、XML が役立っています。)

World Wide Navi は Java で書かれており、Windows でも Linux でも動作しますし、もちろん国際化されています。現在は、日本語と英語のインタフェースが提供されています。スクリプトを組んで作業を効率化し

ているケースも多いかと思われませんが、そのスクリプト自体が作った人に依存しているため、その人の作業を効率化するだけにとどまっていることが多いのが現状です。ソフトウェアの国際化や日本語化あるいは英語化の作業を行う際には、World Wide Navi を利用した作業の効率化を検討してみることをお勧めします。

国際化というとプログラムのソースコードの変更だけを考えてしまいがちですが、実際には製品の企画や出荷、保守、教育など、あらゆることを考慮して計画を練る必要があります。紙面の都合で本稿では詳しく説明しませんが、国際化されたソフトウェアとそうでないソフトウェアでは、保守にかかる手間やコストが大きく違ってきます。ソフトウェアの国際化に取り組む上では、単に機能を実現するだけでなく、開発体制や開発手法などあらゆる面での効率化に目を向ける必要があります。翻訳を含む L10N 作業についても同様です。それぞれの分野のプロは、ツールや作業のやり方を工夫して効率化をはかっていますので、常に最適な方法を考えていくことが大切です。国際化 J P 株式会社では、パートナー企業と協力しながら、I18N、L10N に関する開発作業に関して、様々な観点からアドバイス、作業の支援を行っています。ソフトウェアの国際化を通して多くのお客様のビジネスの拡大に協力できることを願っております。

参考文献：共立出版 清兼義弘・末廣陽一編著「国際化プログラミング」

本稿で登場する製品名、組織名は一般に商標または登録商標です。本稿では、™、® 表記はしていません。

World Wide Navi および国際化 J P 株式会社のソフトウェア国際化サービスについて

- ◆ 製品およびサービスに関する情報 (Web)
<http://www.kokusaika.jp/>
- ◆ お問い合わせ (Mail)
kjp-sales@kokusaika.jp