



# Message from V Java Vendors

## コンポーネントウェアで実現するリッチプロトタイピングツール「SMART PP」 第1回：SMART PPのコア技術，MZ Platformとは何か？

テキスト=柴田 克己  
text by Katsumi SHIBATA



### 完全なコンポーネント開発で生産性を向上

今号から3回にわたり、「SMART PP」のコア技術である、Javaアプリケーション開発・実行環境の「MZ Platform」を紹介していきます。

このMZ Platformは、日本の「ものづくり」強化策として経済産業省が2000年に実施した「デジタルマイスタープロジェクト」において、産業技術総合研究所が開発した技術の名称です。このMZ Platformをコアとして、スマーテックが「SMART PP (Prototype Platform)」という名称で商品化を行っています。

MZ Platformの最大の特徴は、Javaフレームワークをベースにした、完全なコンポーネントベースの開発環境である点です。開発者は、一般的なJava開発ツールのようにエディタ上でソースコードを記述することなく、すでに用意されているコンポーネント（ソフトウェア部品）を選択し、その部品同士をつなぎ合わせていくという作業を繰り返すことによって、短期間で実行可能なアプリケーションを構築することができます。

もともとMZ Platformは、「中堅中小製造業における現場業務のIT化を支援する」という目的で開発が進められた経緯から、標準で用意されているコンポーネントには、一般的なアプリケーションでよく使われるGUIやフォームをはじめとして、各種グラフ、ガントチャート、データテーブル、帳票、3Dイメージビューワなど、製造業における実際の業務アプリケーションに利用しやすいものが、数多く用意されています。また、より高度なカスタマイズを行うための、条件制御、

各種変数、データベースアクセスといった機能も、標準コンポーネントに含まれます。

さらに、MZ Platformの基本コンセプトとして、実際に現場でアプリケーションを使うユーザが、自分の手で業務アプリケーションに必要な修正を加えたり、機能を追加したりといった作業を行う、「EUC (End User Computing)」が強く指向されています。そのため、実際のアプリケーション構築ツール上では、Javaのソースコードを一切記述しません。関係するソフトウェア部品同士を目で見える「線」でつなぎ、その関係は「ガントチャートのデータを設定する」「表示したい文字列を設定する」「アプリケーションを終了する」といった、わかりやすい日本語で定義していきます。

これらの基本となる開発環境、実行環境、約150個の標準コンポーネント群は、すべて無料で利用することができます（図1）。スマーテックでは、今後、MZ Platform上で利用できる、より高度な機能を持ったコンポーネントや、実際の業務システムの開発テンプレートとなるプロトタイプアプリケーションなどを販売していく計画です。

ソースコードを記述せず、完全なコンポーネントベースでアプリケーションの構築を行える開発環境は、これまでもいくつか世に出ていますが、「製造業向けの業務アプリケーション開発が特に容易である」「基本的な開発・実行環境、標準コンポーネントが無料で提供される」「J2SEベースであり、Windows/Linuxの環境で利用できる」といった点が、MZ Platformの特に大きな特徴となるでしょう。

# Message from Java Vendors

## 無料で入手できる開発環境を使う

では、実際にMZ Platformを利用する環境を作ってみましょう。ここでは、Windows 2000/XPでの利用を前提に説明します。

まず、ダウンロード用のWebサイト (<http://www.gihyo.co.jp/ad/download/>) にアクセスして、圧縮ファイルを手に入れてください。解凍するとインストーラ (Setup.exe) とドキュメントフォルダが出現します。インストーラにてMZ PlatformとJ2SE1.4.2\_03, Java3D1.2.1\_04をインストールし、ドキュメントフォルダを好みの場所に配置します。これでMZ Platformが利用可能となります。

MZ Platformのグループには「アプリケーションビルダー」(開発環境)、「アプリケーションローダー」(実行環境)と、各アプリケーションのコンソール、およびReadmeファイルが生成されているはず。実際の開発作業は、「アプリケーションビルダー」上で行います。

## プロトタイピングから実際の運用まで可能

MZ Platformには、詳細なチュートリアルとサンプルアプリケーションが用意されていますので、まずは、これらを参照しながら、作業イメージをつかんでみることをお勧めします。アプリケーションビルダー

を起動し、「ファイル」メニューの「ロード...」から、「Tutorial」フォルダの中にある、サンプルアプリケーションを開いてみてください。電卓や住所録、ガントチャートといったシンプルなアプリケーションが、こういったコンポーネントやイベント処理で成立しているのが、理解できるようになっています。

機能の追加・修正は、アプリケーションビルダーのフィールド上で、右クリックしてコンポーネントを追加。それぞれのコンポーネント間をイベントで接続。イベントによって、どのような処理を行うかを日本語で選択するという作業を繰り返すことで行います。作業中に「画面編集」ボタンを押すと、ユーザインタフェースの調整も可能です。また、開発作業のどの段階でも「実行」ボタンを押すことにより、アプリケーションローダーを呼び出し、その時点でアプリケーションビルダー上に構築されているプログラムの実行状況を確認できます。さらに、その状態でアプリケーションビルダー上加えた変更は、即座に実行画面にも反映されるようになっています。

一般的なJava開発環境を使い慣れた開発者にとっては、「ソースコードを変更して、ビルドして実行」という過程が必要ないため、MZ Platformでの開発作業は、多少独特に感じられるかもしれません。しかし、たとえば、実際にコーディングを始める前のプロトタイピングに、このツールを利用することで、実際に動く

アプリケーションを極めて短期間に作成できるというメリットがあります。もちろん、小規模なものであれば、MZ Platform上のアプリケーションを、そのまま業務に利用することも可能です。次回以降は、実際のアプリケーション構築の例を交えつつ、MZ Platformの技術的な側面や、それを利用することによって、大幅に生産性を向上できる開発場面などをご紹介します。

