

デリバティブ・ポジション・リスク管理でのMS-Excelの利用

デリバティブ（金融派生商品）のポジション・リスク管理システム（PRMS）には、高い柔軟性が求められる。本稿では、米Microsoft（MS）社の表計算ソフト“MS-Excel（以下、単にExcel）”をシステムのフロントエンドに用いPRMSの開発を行う場合の利点と課題とについて、当社が実際に行った開発経験を基に述べる。

柔軟性が求められるデリバティブのPRMS

デリバティブでは、顧客やマーケットのニーズに合わせて新しい金融商品が次々と開発されるため、それらのフロントオフィス業務をサポートするポートフォリオ（PF）PRMSには、次のような柔軟性が求められる。

価格評価分析機能（数時間でシステム化）

PRMS機能（数日でシステム化）

すでにシステム対応されている金融商品であっても、マーケットに即した値が、より正確に算出できるように、追加や改善への要求も大きい。

また、マーケットの変化に対応し、ポジションへの影響をタイムリーに分析する機能も不可欠である。

PRMSについては、これまではUNIXワークステーションを利用し、“Motif”などのGUI（グラフィカル・ユーザーインターフェース）を用いた開発が行われてきたが、GUIの変更に時間がかかりすぎることなどがボトルネックになっていた。

そこで最近では、Excelの機能拡張やPC（パソコン）の性能向上を受け、ExcelをPRMSのフロントエンドに利用する取り組みが行われるようになった。

PRMSにExcelを用いる利点

Excelの利点は、コンポーネントを組み合わせるだけで、次のような点を速やかに実現できる生産性の高さにある。

GUI開発

表計算シートやGUIのオブジェクトとVBA（Visual Basic for Application）プログラムの組み合わせで、使い勝手のよいGUIを作成できる。商品やモデルの追加に必要なGUIの変更は、コンパイル（機械語変換）リンクの不要な、インタプリタ（機械語変換・実行ソフト）型のVBAですぐできる。

C++言語による計算手順の開発

目的に応じて、C++やVisual Basicによるプログラム開発ができる。実行にスピードを要求される計算手順については、C++でDLL（Dynamic Link Library）を作成することによってシステムに組み込むことができる。

コンポーネントの利用による開発

MS社やサードベンダーが提供する“COM（Component Object Model）”や“Active X”に基づいたコンポーネントも利用できる。“ADO（Active X Data Object）”などのDB（データベース）アクセスコンポーネントを用い、PFデータやリスク

計算値が保存できる。また、英国のReuters社やTIBCO社などが提供するリアルタイム・マーケットデータ・アクセス・コンポーネントを用いることで、リアルタイムなマーケット状況に基づいたリスク管理機能が、容易に提供できる。

エンドユーザーによるカスタマイズ

最近では、フロントオフィスのユーザーも、Excelに精通している場合が多いので、共通のコンポーネントを利用して、ユーザー独自のカスタマイズを行うこともできる。

整備された開発環境の利用

C++やVBAでプログラムを開発している場合には、Visual C++やExcel 97に組み込みのIDE (Integrated Development Environment) が利用できる。すべてのプログラムをGUIベースのデバッガーやオブジェクトブラウザを利用して開発できる。

テストケースの再利用

計算手順などの検証には、プログラマーがカスタマイズした表計算シートを用いて行える。シートは保存・再利用が可能である。

Excelを用いたPRMS開発の課題

一方、システム化に当たっては、次のような課題があげられる。

計算手順やPFデータの標準化

計算分析機能やリスク管理機能の柔軟性を確保するためには、PFデータやそれを扱う計算手順・結果表示の標準化が不可欠となる。

この点には、C++を利用したオブジェクト指向に基づく設計開発を行うことが望ましい。

データ接続の確保

フロントオフィスからミドルやバックオフィスへのデータの接続は、開発当初から考慮しておく必要がある。PFやポジション・リスクの計算結果の保存DBが不整備では、かえって全体のコストが高くなるからである。

表計算シートの共通化

表計算シートは可能な限り共通のものを作成することが望ましい。ユーザーがカスタマイズを行った機能も、随時共通シートに取り込むことで将来のシステム変更が容易となる。

計算処理時間の最適化

処理に時間がかかるとレスポンスが低下し、使い勝手が悪化する。単純に計算モジュールをExcelに組み込む場合には、モジュール自体が最適化されている必要がある。MS社の分散環境対応のコンポーネント技術“ DCOM (Distributed Component Object Model) ”や“ COM/CORBAブリッジ ”を用い、重い処理をアプリケーション・サーバーに実行させ、レスポンスの向上を図ることが可能である。

計算処理の自動化

Excelは、それ自体がインタラクティブ (対話的) に用いられるため、定時に大量の計算を自動的に行うバッチ処理には向いていない。これを補うためには、バッチ処理機能の開発または導入の必要があろう。

(NRIヨーロッパ 宇治田 聡)