

マイグレーションとリスク

マイグレーションによるリライト

- COBOLなどで開発されたレガシーシステムを、Javaなどへ変換
- エミュレーションによる「リホスト」でなく、コンバージョンを実行
- システムのセキュリティ強化、効率アップ、速度アップを狙ったデザインが必要
- 新しいビジネスモデルに対応できるかがポイント



文字コードの違い
DB 構造、アクセス方法の違い
オンライン画面の違い
JCL の違い
実行オペレーションの違い
帳票出力の違い

企業が取り組んでいる「IT モダナイゼーション」について、特に注目されているのが、COBOLなどで開発された資産を Java などの他の言語に変換するマイグレーションによる「リライト」です。

従来メインフレームの環境では、SNA に代表される独自のネットワークでホスト中心の基幹業務処理が行われており、以前はインターネットからは独立した仕様で運用されていました。それゆえ、関係者しか侵入できない堅牢なネットワークを構築することができました。

しかしながら、大型コンピュータのホストの特性としてバッチ処理は強いのですが、時代とともにオンライン処理や部門別の分散処理をしたい業務などは、オープン系の別システムで動かし、インターネットを通じてデータを連携するようになりました。それに伴い、メインフレーム上のホストでも、z/Linux のような OS が稼働したり、オープン系でも稼働するような COBOL が採用されたりしています。

但し、メインフレームであってもオープン系であっても、同じ COBOL なので文法上は似ていますが、稼働環境が違うので、多くの相違点が存在するリスクがあるため、オープン系の特性を生かして修正しやすい Java などへマイグレーションする方法が有力視されています。

いずれにせよ、メインフレームとオープン系では環境が違うため、単なるソースコードのコンバージョンのみならず、気を付けるべきポイントが数多くあります。

まずは文字コードの違いですが、メインフレームの英数字は EBCDIC、オープン系は ASCII などが一般的で、Web であれば、UTF-8 が多いです。漢字コードについては、メインフレームではメーカー独自のもので、漢字文字の始まりと終わりにシフトアウト・シフトインの文字コードを挿入するケースが多く、オープン系ではこれも取り除き、Shift_JIS 又は EUC、UNICODE などに変換が必要です。また、メインフレームにあるパック 10 進数の記述方式も、オープン系にはありません。他にも、データベースや、プログラムを制御するための JCL や、コンソールオペレーションも違います。帳票についても、プリンターが異なるため、コーディング方法が違うし、オンライン画面も GUI 前提ではありません。

オープン系は、インターネットが前提条件となったシステムが多く、こうした環境の違いのみならず、サイバー攻撃に対応するためのセキュリティ強化などを考慮して、マイグレーションを行う必要があります。