特集

COBOL資産を活用したシステム構築事例

Case Studies of System Architectures That Use COBOL Assets

あらまし

COBOL（Common Business Oriented Language）は，事務処理向けに開発されたプログラム言語で，国際規格により，高い信頼性，互換性を保証し，長年にわたり，基幹システムの主要言語として利用されてきた。

富士通が提供するCOBOL開発環境“NetCOBOL”は，SDAS技術に基づく多くの支援ツール，開発標準スタイルとともに，既存COBOL資産を活用し，Java，VBという先端技術と連携した基幹システムの短期構築，安定稼働を支援している。

本稿では，COBOLと先端技術を融合して構築した事例として，JavaとCOBOLが連携したシステムを構築された株式会社星光堂様の事例と，.NET Frameworkをベースに，VBとCOBOLが連携したシステムを構築された株式会社メイテツコム様の事例を紹介する。

Abstract

The Common Business Oriented Language (COBOL) was developed as a programming language for business transactions. For a long time, it has been a major programming language for mission-critical systems because of its high reliability and compatibility ensured through international standards. NetCOBOL is a COBOL development environment offered by Fujitsu that supports quick construction and stable operation of mission-critical systems. It achieves this by using existing COBOL assets as well as many supporting tools and standard development styles based on technologies that comprise Fujitsu's System Development Architecture & Support facilities (SDAS). In this paper, we introduce two case studies of system architectures that use COBOL assets. The first is at Seikodo Co.,Ltd., which has built a system based on a combination of Java and COBOL, and the second is at Meitetsucom Co.,Ltd., which has built a .NET Framework-based system linked with a combination of Visual Basic and COBOL.

仁藤滋昭（にとう　しげあき）

ミドルウェアコンポーネント事業部
第一開発部 所属
現在，NetCOBOLの開発に従事。

ま え が き

COBOL（Common Business Oriented Language）言語は，COBOL85，COBOL2002の国際規格により高い信頼性，互換性を保証している。COBOL言語は大量データ処理，伝票・帳票処理，そして高速性・高性能などビジネス処理に最適な言語仕様を提供している。

富士通では，メインフレームで動作する“COBOL85”，オフコンで動作する“COBOL-G”，Solaris，Linux，Windowsおよび.NET Frameworkで動作する“NetCOBOL”というソフトウェア製品を共通に開発し提供している（**図**-1）。各製品は，OS固有の機能を除き高い互換性を維持しつつ発展している。10年前，20年前に作成したビジネスロジックが最新環境で動作する。メインフレーム，オフコンの最新機種への入替え，オープンシステムへの移行，旧Windowsシステムのバージョンアップ，SolarisからLinuxへの横展開など，COBOLの高い互換性が短期に様々なシステム形態への移行を実現している。

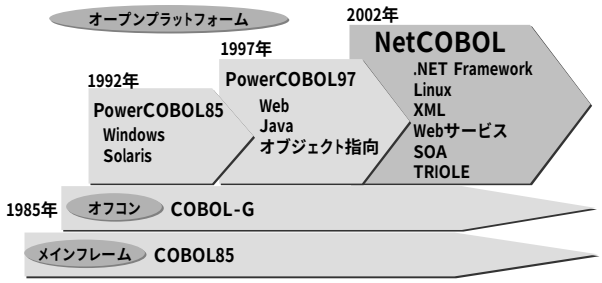

図-1　進化するNetCOBOL
Fig.1-Evolving NetCOBOL.

また，COBOLは，メインフレーム，オフコンと連携したシステム構築も可能にしてきた。例えば，メインフレームとオープンシステムのCOBOLアプリケーションの連携や，メインフレームのデータベースに対し，オープンシステム上のCOBOLアプリケーションからCOBOLのインタフェース“READ/WRITE”でアクセスを可能とした。

さらに，NetCOBOLでは，オブジェクト指向に対応し，富士通が提供するJ2EEに基づいたアプリケーション実行基盤“Interstage”で，COBOLとJavaが連携したシステム構築を可能とした。またMicrosoft社によって開発された.NET環境の実行基盤“.NET Framework”にCOBOL開発環境としては世界で初めて対応し，VisualBasic（以下，VB），C#などマルチ言語連携をサポートすることで，Webフォーム，Webサービス，Windowsフォームなどのシステム構築を可能とした。

これにより，システム形態，既存の資産との関係を考慮し，画面を含むフロント系システムをJavaやVBで，ビジネスロジックなどバック系システムをCOBOLなど，各言語の特性を生かし適材適所に言語を配置した最適なシステム構築が可能になる（**図**-2）。

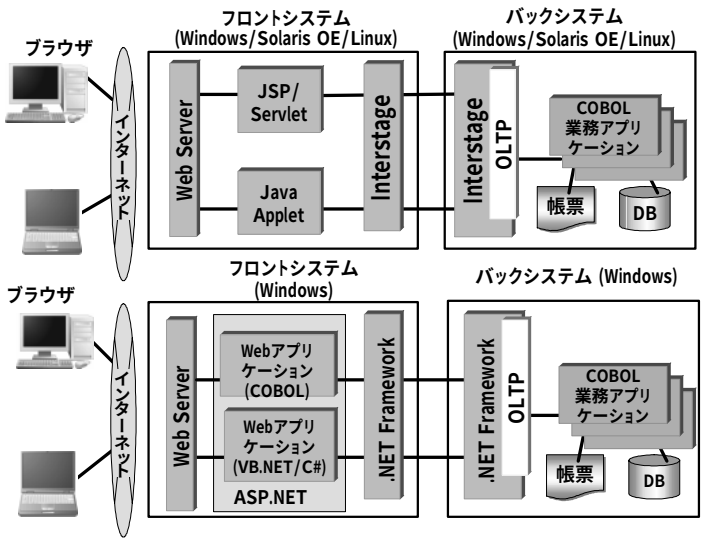

図-2　言語の特長を生かしたシステム構築
Fig.2-System which made the best use of characteristic of language.

COBOLの開発スタイルも進化している。NetCOBOLではSDASの高い生産性と，高品質なもの作り，高い保守性を目指した考えに基づき，設計，プログラミング，テスト，保守に至る開発サイクル全般に対し多くの充実した開発環境を提供している。専用エディタ，ビルダ，デバッガ，画面帳票エディタをはじめ，SDASツールのSIMPLIAシリーズの一部機能を組み込んでテスト支援，ドキュメント作成支援など，生産性，保守性を大幅に向上させることが可能となる。

本稿では，COBOLと先端技術を融合し構築した事例として，Interstage，Javaと連携したシステムを構築された事例，.NET Frameworkで構築した事例を紹介する。

Interstage，Javaと連携したシステム構築事例

本章では，Interstage，JavaとCOBOLが連携したシステムを構築された株式会社星光堂様（以下，星光堂）について述べる。

星光堂は，CD，DVD，ゲームソフトなどの音楽映像関連商品を扱う卸売業である。これまで，受発注在庫管理システム，店頭MD支援/情報提供システム，そして業界独自の戦略的情報システムを構築されてきた。

現行受発注在庫管理システム（基幹システム）は1984年にメインフレームで構築したものをベースに追加拡張を加え，20年間運用してきたものである。そのため，システムは複雑かつ硬直したものになっていた。加えて，個人商店から法人，インターネットなどへの販売チャネルの変化，取扱商品の変化など，取り巻く商環境の急変に対応するための柔軟なシステムが必要となってきた。そこで，オープンシステムで新基幹システムを構築することになった[1]

新基幹システム構築の目的と要件は，以下のとおりである。

(1) 商環境への迅速な対応と安定稼働

星光堂では，全国10,000の店舗に対し24時間365日サービスを提供している。その結果CD約100万枚，DVD約60万枚の在庫を持ち，商品点数60万件，一日のデータ量は平均7万件（ピーク時18万件）に上っている。安定したサービスを提供することが新システムの条件となる。

(2) 段階的な移行とシステムの拡張性

長年利用してきたメインフレームのプログラム総資産は7,500本，その内移行対象COBOL資産は4,500本に達する。そこで，既存のシステムを稼働させながら，オープンシステムとの連携を行う段階的な移行を目指した。

(3) メインフレームで稼働している既存資産との連携

システム構築に当たっては，星光堂の情報部門を担当されている株式会社プラネット様，株式会社富士通システムソリューションズ（Fsol）が，星光堂とともに短期間で，高品質なシステム構築を行った。

● 新基幹システムの構成

新基幹システムは，基幹システムオンライン処理用サーバ（オンラインサーバ），基幹システムDB用サーバ（DBサーバ），基幹システムバッチ業務処理用サーバ（バッチサーバ）で構成している。バッチ業務処理は既存資産を流用してCOBOLで構築し，オンライン処理は新規にJavaで構築した。販売店によっては同時期に新環境への対応が難しいところもあるため，既存のメインフレームを利用したPOSシステムによる入力も，継続して運用している。そのため，メインフレームを経由して連携するシステムを構築した（**図**-3）。

バッチ業務処理システムにCOBOLを採用した理由は，以下のとおりである。

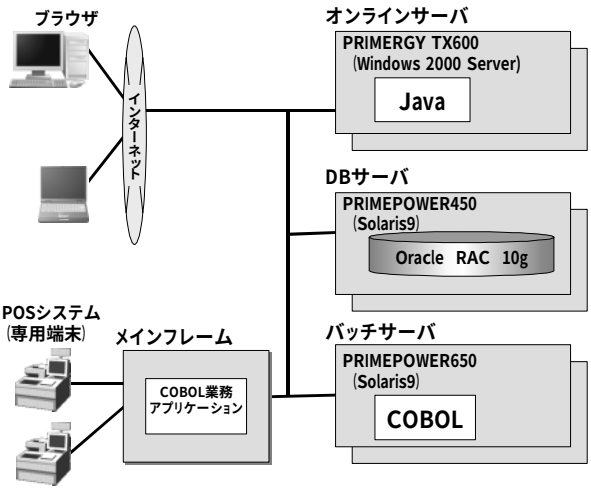

図-3 星光堂システム構成
Fig.1-System configuration of Seikodo.

【開発要員の特性による理由】

(1) 設計の観点

短期再構築のためには，現行システムの分析，新システム設計～開発までの全工程に対応できる要員がいると効率的である。COBOLを採用することにより，メインフレーム資産の読解力を持つ要員を，そのままUNIX環境で活躍させることができる。

(2) メンテナンスの観点

お客様の情報システム部門は，これまでCOBOLで開発してきた。このスキル・ノウハウを活用し，開発，稼働後のメンテナンスも含め対応できる。

(3) プログラムの観点

COBOL言語のプログラマが人的資源として高度でかつ豊富であり，外部要員の投入が容易である。

【技術的な理由】

大量データを処理するバッチ業務処理ではJava言語のオーバヘッドによる性能阻害が懸念され，性能的にCOBOLが優位である。

一方，営業所や販売店のパソコンからWeb経由で発注や在庫引当て処理を行うオンライン基幹端末システムに関しては，柔軟な画面設計，Webシステム構築が得意なJavaを採用した。

● 星光堂システム概要

上述の基幹システムの構成をもとに星光堂で構築したシステム概要について述べる（**図**-4）。

【バッチ業務処理システム構築】

バッチ業務処理システム構築に当たっては，星光堂が資産調査，基本設計およびシステムテスト，運用テスト，データ移行を担当した。インフラの検討，詳細設計，プログラム開発，統合テストはFsolが主体で担当し，既存業務ロジックを継承しつつ，DBなどのインタフェース部分は新規に構築した。

開発に携わった星光堂，Fsol要員はどちらも，長年にわたってメインフレーム，オフコンなどでCOBOL技術を習得してきていた。一方，UNIXの経験は乏しかった。それぞれ今回の開発に当たって，エディタの使い方に不慣れであったり，OS操作はマニュアルを見ながらであったりはしたが，ことプログラム開発という点からは従来のノウハウを十分に発揮し，NetCOBOLであるがゆえの生産性悪化，品質問題は発生することはなかった。その意味でNetCOBOLが持つCOBOLとしての汎用性・互換性が発揮されたと言える。

あわせて，高性能なデータソート・マージユーティリティであるNetCOBOLファミリ製品“PowerSORT”を利用することで，ファイル，データ処理の高速化を実現した。

なお，テストデータ作成，テスト検証にSIMPLIAシリーズを導入すればテスト工程を中心に更に効率化が実現できたものと思われ，今後の課題とされている。

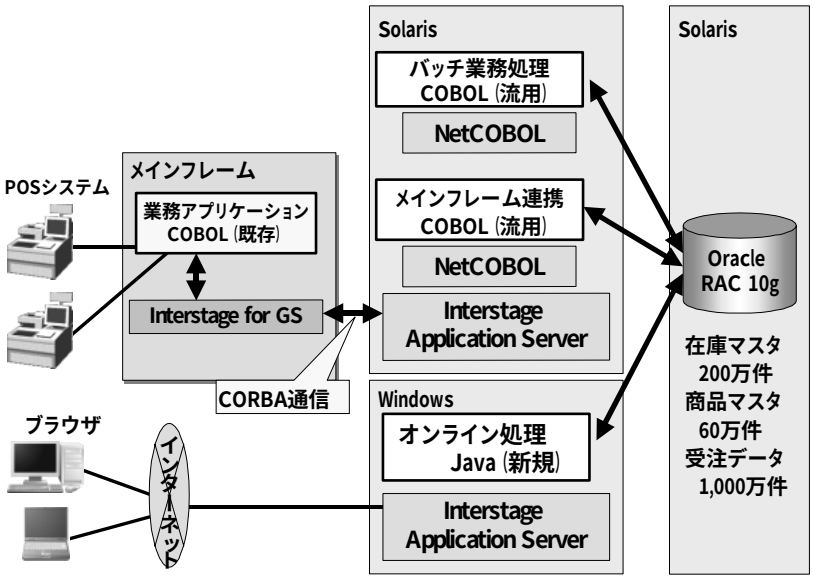

図-4 星光堂システム概要
Fig.4-Outline of system of Seikodo.

【メインフレームと連携したシステム構築】

新基幹システムをすぐに導入できない店舗に対応するため，メインフレームを受け口とする受注業務も継続して稼働している。受注データを受け取るメインフレームのアプリケーションは，そのデータをそのままオープンシステムの受注アプリケーション（COBOL）に中継したり，受注結果をオープンシステムから店舗に中継したりする。

メインフレームとオープンシステムとの連携は，メインフレーム側に“Interstage for GS”，オープンシステム側に“Interstage Application Server”を配置し，CORBA通信により実現している。形態としてはオープンシステムがサーバとなるCORBA通信である。メインフレーム上のCOBOLアプリケーションは表示ファイルインタフェース（READ/WRITE）で，CORBA通信を意識することなく，オープンシステムと連携している。オープンシステム側アプリケーションは，旧メインフレームのCOBOL資産を流用し，DBアクセス部分の変更（SQL化），通信部分の変更（CORBA化）を施した。

【オンライン処理システム構築】

オンライン基幹端末システムは，星光堂がすべて担当し，Interstage Application ServerのJ2EE実行環境機能を基盤として，Javaで再構築し，自由度の高い画面，柔軟なWebシステムを実現した。

● 導入効果

従来環境の性能をクリアし，今後のビジネス展開を考慮した拡張性の高いシステムを構築することができた。

星光堂の事例は，JavaとCOBOLの特長を生かしたシステム構築のパターンである。Web入力部分のみをJavaで，ロジック部分をCOBOLで構築した事例も多い。

また，星光堂では，既存メインフレームとの連携による段階的なシステム移行で，低リスクのシステム構築を実現しているのも大きな特長である。

.NET Frameworkで構築した事例

本章では，.NET Frameworkの上でCOBOLシステムを構築された株式会社メイテツコム様（以下，メイテツコム）について述べる。

メイテツコムは，名古屋を中心に鉄道，自動車，観光，航空代理業など幅広く事業を展開している名古屋鉄道株式会社様の情報システムを担当している。

1986年に社内向け専用端末から鉄道時刻を検索できるシステムをメインフレームで構築した。1998年にインターネットによる一般利用者向けシステムをUNIXで構築し，2004年には，中部国際空港の開港による大幅なダイヤ改正，それに伴うアクセス増加の対応，および画面操作性能の改善に対処するため，.NET Frameworkでシステムを再構築した[2]。.NET Frameworkの採用の理由は，以下のとおりである。

（1） 現行システムに対する優れた性能，価格
（2） マイクロソフト製品の多くのノウハウと.NET Frameworkによる利用実績
（3） VBによるWeb画面，携帯画面の柔軟な開発
（4） 既存COBOL資産の活用
（5） COBOL，VBなど異なる言語が共通の開発環境Visual Studio.NETで効率的に開発可能

従来システムは，富士通ではなく他メーカのメインフレーム，UNIXサーバ，COBOLで運用していた。今回，.NET Frameworkにいち早く対応した富士通のNetCOBOLを評価していただき，採用が決定した。

● 新システムの構成

新システムは，アプリケーション・DBサーバと，Webサーバの2台で構成している。そして，WebサーバはVB.NET，アプリケーション・DBサーバはCOBOLおよびOracleでそれぞれ構成している。Webサーバとアプリケーションサーバ間は.NET Frameworkが提供するWebサービスで接続した。Webサーバとアプリケーションサーバを分けたことで，VBとCOBOL開発の分業が可能となった（**図**-5）。

携帯電話3社の画面はVBで作成した。また，インターネットからの参照画面はマクロメディア社のShockwave Flashにより画面操作性が向上した。

アプリケーションサーバで使用する業務ロジックについては，既存COBOL資産から30キロステップの資産を移植した。メーカおよびOSが異なるCOBOLでも高い互換性によって，5箇月という短い期間でのNetCOBOLによる開発を実現した。

● 新システムの効果

サーバのハードウェア，ソフトウェアの費用を大幅に削減し，システムの単純化により，運用管理者

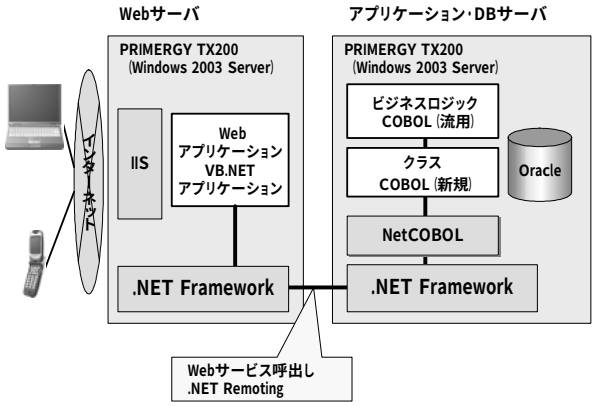

図-5　メイテツコムシステム概要
Fig.5-Outline of system of Meitetsucom.

の負担も軽減できた。また，本システムを利用しているお客様に対し，乗換え案内などきめ細かな情報の提供，携帯電話からの利用など，サービスを向上できた。また懸案だった性能も当初の目標（160件/分）をクリアし，お客様満足度向上につなげることができた。

メイテツコムの事例では，フロントをVB，バックをCOBOLで分業したことが大きな特長である。

もちろん，NetCOBOLでもVisual Studio.NETを利用して柔軟なWeb入力画面を構築することが可能で，COBOLだけでシステムすべてを構築した事例も多くある。

む　す　び

NetCOBOLにより，富士通，他メーカを問わず，メインフレームのCOBOL資産を活用することができる。さらに，新たな技術であるJava，VBとの連携から，既存メインフレームとの連携など，あらゆるシステム形態にも対応可能である。お客様の貴重なCOBOL資産を将来に向け発展，拡大できるのが，NetCOBOLの特長である。

今後もメインフレーム並みの信頼性，互換性，そして先端技術へ対応し，SDASの考えのもと，短期間，高品質なものづくりの開発環境として，次世代の業務システム構築に貢献していく所在である。

参 考 文 献

(1)　大川原冬樹：COBOL資産を活用した星光堂様新基幹システム構築事例．第9回インターネット時代のCOBOL活用セミナー，COBOLコンソーシアム，2005.
http://www.cobol.gr.jp/knowledge/material/050614_report.html

(2)　福間　浩：COBOL資産を活用した.NETによる名鉄時刻表検索システム構築事例．第8回インターネット時代のCOBOL活用セミナー，COBOLコンソーシアム，2005.
http://www.cobol.gr.jp/knowledge/material/050201_report.html