

## ビジネスコンピュータにおける新開発環境の開発

5 K-8

菅野幹人<sup>†</sup> 森山令子<sup>†</sup> 吉村礼子<sup>†</sup>

大江信宏<sup>†</sup> 北川克也<sup>††</sup>

<sup>†</sup>三菱電機株式会社 情報システム研究所

<sup>††</sup>日本ユニシス株式会社

### 1. はじめに

パソコンをクライアントとしビジネスコンピュータをサーバとする新しい開発実行環境 HYPERPRODUCE を開発した。パソコンの持つ GUI (Graphical User Interface) やマルチメディア機能と、ビジネスコンピュータの持つ基幹業務処理機能を組み合わせ両者の特徴を合わせ持つシステムを効率よく構築することができる。

本稿では、特にクライアント側開発環境を中心に説明する。

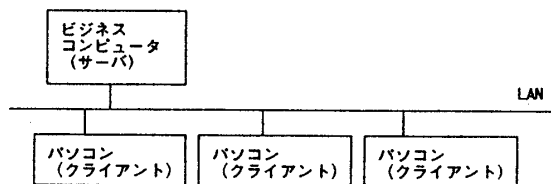


図-1 システム環境

### 2. 構成

図-2は新開発環境の構成図である。新開発環境はMicrosoft® Windows™ Version3.1上に構築されており、次の2つの部分から構成される。

#### ●サーバプログラム開発部分

ビジネスコンピュータ上で動作する事務処理用簡易言語の設計開発環境

#### ●クライアントプログラム開発部分

クライアントプログラムの画面、通信などのデバッグ支援環境、運用支援環境

サーバプログラム開発部分、クライアントプロ

A Development of New Programming Environment for A Business Computer

Mikihito Kanno

MITSUBISHI Electric Corporation

5-1-1 Ofuna, Kamakura, Kanagawa 247, Japan

グラム開発部分ともにパソコン上に実現されており、パソコン上で設計開発を行う。

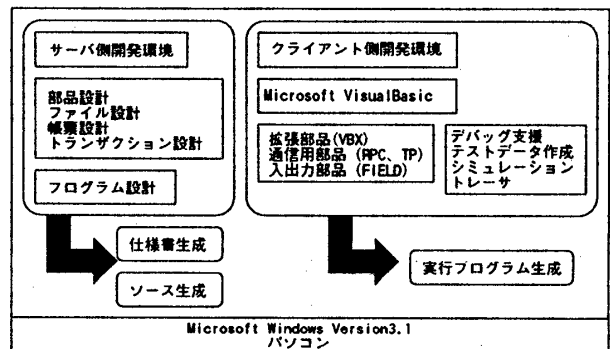


図-2 新開発環境の構成

### 3. サーバ側プログラム開発

サーバ側のプログラム開発は事務処理用簡易言語の専用開発環境となっており、Windows環境でダイアログから必要なパラメータを設定することによりプログラム設計を行う。プログラム設計を支える機能として入出力部品設計があり、ファイル、帳票、トランザクションなどの入出力部品を対話的にかつビジュアルに設計可能である。

### 4. クライアント側プログラム開発

クライアント側で動作するプログラムの開発はMicrosoft社のVisual Basicで行うが、Visual Basicに対して以下の観点から拡張を施している。

(1) GUI画面を用いるが、サーバ側で培われた入出力項目の概念をクライアント側のプログラミングに活かす。

(2) クライアントサーバ型のプログラム開発において相互の通信インタフェースを合わせることやデバッグが難しい。

これらの拡張を実現するためにVisual Basic用の入出力部品および通信拡張部品を開発した。

### ●入出力部品

入出力部品は、基幹業務アプリケーションにおいて頻繁に使用される部品であるので、この部分のロジック記述を減らすことが生産性の向上につながると考えられる。そこで、入出力部品に対して入力タイプ、長さ自動チェックや数値の日付時間、通貨編集のような独自追加することによりロジックを記述することなくビジネス専用の入出力を簡単に行えるようにした。

### ●通信部品

従来のアプリケーション開発環境から、クライアントサーバ型の開発環境に移行する場合に通信部分の設計が大きな問題となる。そこで通信部品を提供することでビジネスコンピュータにおける通信の標準化と通信ロジックの削減を目的として通信部品の開発を行った。通信部品はサーバと通信を行うための通信タイプとしてRPC(Remote Procedure Call)とTP(Transaction Processing)の2つの通信タイプを持つ。RPCは小規模な業務アプリケーションの構築用でありTPは大規模なシステム向けを想定している。通信部品は以下の特徴を持つ。

- ①サーバ側開発環境で設計するトランザクション設計パターンを取り込み、通信におけるデータ長などの整合性をサーバ側とクライアント側で簡単に同期させることを可能とした。
- ②データ型の差異やコード体系の変換をトランザクションパターンによりパソコン側の通信部品で解析することを可能とした。
- ③通信部品の中にデバッグ機能などを組込むことが容易となった。

### 5. クライアント側デバッグ支援環境

分散型アプリケーションのデバッグ支援環境では以下の機能が求められている。

- (1)連携プログラムがない環境でのデバッグ

- (2)実際に通信しているデータの把握

これらの環境を提供することが、デバッグ作業の効率化やひいては生産性を向上させることから次のツールを開発した。

### ●デバッガ

このツールは、トランザクション設計パターンに基づいてあらかじめ入力されたテストデータパターンにより、シミュレータとクライアント通信側通信部品との間でデータ交換を行う。通信部品内部にデバッグモードかどうかを自動的に判断する機構をとりいれているためプログラムを変更しなくてもシミュレーションが可能となっている。

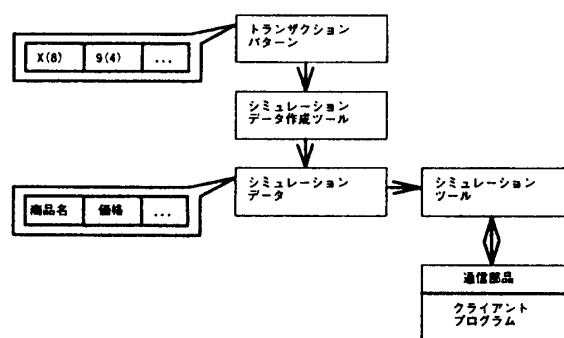


図-3 デバッガイメージ

### ●トレーサ

サーバプログラムとの通信データを実際にトレースするツールである。これにより、通信データをログファイルに採取し、プログラムしたとおりのデータが通信されているかどうか解析することができる。将来的には、このツールをトランザクション量などの解析ツールとして用いる予定である。

### 6. おわりに

現在本開発環境はビジネスコンピュータにおけるプログラム開発で実際に使用されており、大規模なパッケージプログラム等も本開発環境を使用して構築されている。今後さらに改良を行い、分散開発運用支援やプログラムの自動生成機能などを付加していく予定である。