

バッチ系アプリケーションプログラム生産性向上

3S-2

伊藤 正裕 大江 信宏 杉本 直也 小山 明伸 萱野 重実 安田 智*

三菱電機株式会社 情報通信システム開発センター

三菱電機株式会社 情報技術総合研究所*

1. はじめに

昨今のオープンシステムではクライアントサーバ型アプリケーションの生産性向上のためのCASEツールが多く製品化されているが、汎用機やオフコンからのダウンサイジングではバッチ系や帳表系処理の必要性が高い。

本稿では、このような種類のアプリケーションを効率良く開発するために、パソコンのGUIを用いた操作性の良さと、日本語をベースとしたプログラム設計環境と、プログラム設計と同時に仕様書も自動生成し生産性と保守性を高めたCASEシステムの概要を報告する。

2. 本CASEシステムの特徴

オープンシステムへ移行する際には、ユーザインタフェースを持ったアプリケーションシステムはGUI化の利点を楽しむことができるため、新規にシステム設計を行うケースがほとんどであり、このための多くのCASEツールが製品化されている。バッチ処理系アプリケーションの場合はGUI化、クライアントサーバ化のメリットは特にないためなるべく工数を掛けずに再構築したい。

しかし、オープンシステムでは、従来の設計手法と親和性のあるバッチ処理系開発用CASEシステムは少ない。本稿におけるCASEシステムは、ユーザインタフェース部を有しないバッチ処理システム及びサーバ側ソフトウェアシステムを開

High Productivity CASE

for Batch Programming System

Masahiro Itoh Nobuhiro Ohe

Naoya Sugimoto Akinobu Koyama

Shigemi Kayano Satoshi Yasuda

MITUBISHI ELECTRIC Corp.

発支援の対象としている。本CASEシステムでは項目名の日本語化、処理ロジック入力エディタ部の処理ロジックの日本語表現によりプログラムの高い可読性を実現している。また、仕様書類は、部品仕様定義において、手順に従い、必須項目を入力することにより、プログラミングをすると同時に仕様書作成も可能となっている。これによりバッチ処理系システムを効率よく開発可能とすると共に、生成されたソース/実行可能モジュールと仕様書の一致性を可能とし保守性を高めたCASEシステムとなっている。

3. 設計の流れ

プログラム設計の流れは図1の通りである。

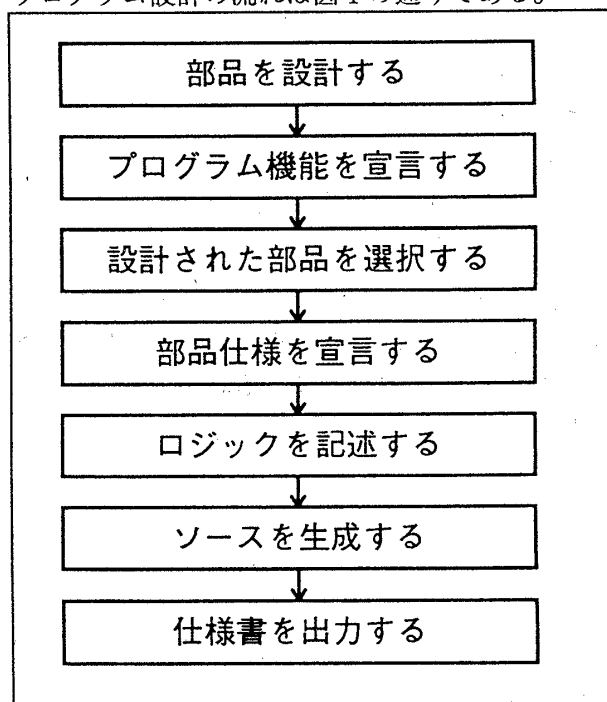


図1 設計の流れ

3. 1 部品設計

部品設計システムにてファイルレイアウト、帳表レイアウトを設計する。

3. 2 プログラム機能の宣言

プログラム機能の宣言を行う。この部分ではロジックとして反映される部分とそうでない部分がある。ソースプログラムに反映されるものは、メインプログラム/サブプログラムの別、引き数情報等である。反映されないものは、概要説明のコメント等である。これは、仕様書の出力時に用いられるばかりではなく、プログラム情報のデータベース化による情報検索用のキーワードとしても用いられる。

3. 3 部品の選択

3. 1で設計した部品はCASEシステムの中でデータベース化される。ここでは、このデータベースからどの部品を使用するかを選択する。

3. 4 部品仕様の宣言

3. 3で選択したファイル・帳表部品に対しての仕様を定義する。

3. 5 ロジックの記述

3. 1～3. 4で行ってきた定義、宣言を利用して処理ロジックを作成する。処理ロジックの作成では、バッチプログラムの代表的な処理フロー^{*1}が用意されているので、不足している処理部分の記述を追加するだけでプログラムを完成させることができる。

このロジックを入力するところ、及び、入力した後のエディタ スクリーン部への表示が本CASEシステムの一つの特徴である。日本語（漢字）で入力させることを実現させるためにエディタにプロンプト（入力ガイダンス）を設けた。（図2）この方式は、従来多くあった、フリーフォーマット形式の一般的な言語処理系に日本語を取り入れたものに比べ、プログラマは命令を選択し日本語項目名入力するだけで、非常に読みやすいプログラムを記述することができる。

3. 6 ソース生成

3. 1～3. 5で行なった結果を用いてソース生成を行うことによりプログラムソース生成が可能である。このソースを生成するモジュール内にて

項目名を通常の半角英数字に変換するため、日本語に対応していない、言語処理系をターゲットとするソース生成も可能となる。

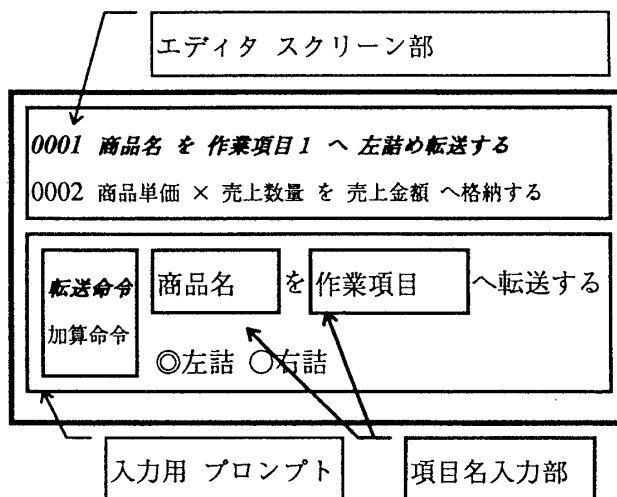


図2 エディタ部

3. 7 仕様書出力

3. 1～3. 5で行なった結果により、プログラム概要、使用している部品情報、部品の詳細、及びプログラムロジックの仕様書が日本語で出力できる。生成されるプログラムソース及び仕様書は同一の情報から生成されるために一意性が保たれる。

4. おわりに

本CASEシステムにより、効率よくシステムが構築可能である。現在のプロトタイプシュミレーションの結果では、仕様書作成、プログラム設計からコーディング、仕様書の手直しを含めて、従来の開発手段である、仕様書作成はワードプロセッサ、プログラミングは部品設計システムとスクリーンエディタといった方法に比べ工数が約半分程度削減できると評価できている。また、プログラミングを行うことと仕様書を作成することが一致するためにソース/実行可能モジュールと仕様書の一致性が得られる。漢字で項目名を付けることも可能となり可読性が高くなった。これにより保守フェーズの作業の効率化も図れる。

^{*1}メインファイルをEOFまで読みこみキーブレイクが発生したら合計処理するフローのこと