

## 層別化及び可視化による情報処理システムの構築

5H-6

出口 博章  
[三菱電機ビジネスシステム]

富沢 研三  
[三菱電機]

田村 幸子  
[福岡女子短期大学]

魚田 勝臣  
[専修大学]

### 1. はじめに

情報システムの立場から企業活動を業務特性に基づいて層別化し、それに合致した開発方法を選択することを考えた。その中で個別企業固有の業務処理のシステムの開発については、ユーザーの深い参画が不可欠であるとの認識から、基本設計を含めてエンドユーザーの参画を求めるリユースモデルを提案した。今回はそれを更に進めソフトウェアを可視化することを考え、プログラムの内容をダイアグラム部品と手続部品によって表現する。こうした可視化によってエンドユーザーは設計からプログラムに至るまで、立ち入ることが出来、システム完成前に、最終システムのイメージを描くことが出来る。

### 2. 企業活動層別化と情報システム実現方法の選択

企業の活動を情報システムの視点から、大きく3つに層別化した(図1)。そしてそれぞれの特性に応じて次のように開発手法(下線)を選択することを提案する。

我々の経験ではこの考え方で、個別開発を40%にすることが出来る。

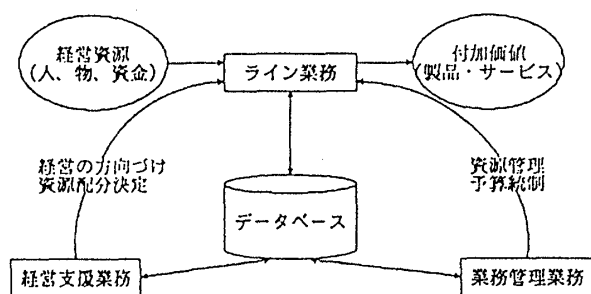


図1 企業活動のモデル

#### (1)ライン業務

企業の人、物、金、情報などの資源を活用して付加価値を生み出す業務である。リアルタイム性を持った

トランザクション処理が多く、企業の基幹データベースはこのライン業務で構築される。企業の戦略や固有の業務ルールを最も反映し、業務の変化に柔軟に対応出来る事が要求される。従って個別開発による構築法が適している。

#### (2)業務管理業務

配分された経営資源の予算を統制管理し、付加価値である製品、サービスの産出状況を計数的に把握、管理する業務である。企業間で業務ルールの類似性が高いのでパッケージソフトウェアによる実現が適している。

#### (3)経営支援業務

企業の長期的展望に立って意志決定を支援する業務である。一般的に年度計画、戦略の企画、商品計画などの業務が多い。ライン業務、業務管理業務が作り出すデータベースや外部情報をインプットとして、資源配分や、経営目標の設定を支援する。データベースを加工し幹部に対する報告者を作成することが多く、EUCによる方法が適している。

### 3. 可視化技法による情報システムの構築

#### 3.1 可視化技法の必要性

個別開発によるライン業務のシステム開発方法としてはウォーターフォールモデルが適用される。しかしエンドユーザーが要求仕様書を作成する段階で開発に参画し、後はコンピュータの専門家にまかされる。どんなシステムが出来上がってくるかは開発途上でエンドユーザーは目でチェック出来ない。この事が次のような問題を引き起こし、生産性を悪化させている。

(1)要求と出来上がったシステムにずれがあり手戻り作業が発生する。

(2)設計変更、システム変更時に非能率になる。

(3)プログラム流用が困難である。

System Development Method Using Visualization

Hiroaki DEGUCHI[Mitsubishi Electric Business System Corp.], Kenzo TOMISAWA[Mitsubishi Electric Corp.], Sachiko TAMURA[Fukuoka women's Junior College], Katsuomi UOTA[Senshu University]

このようにプログラムの中味がブラックボックスになり開発の生産性を極端に悪化させているため、プログラムの可視化が重要な課題であると考えた。

### 3.2 プログラムの構造

業務システムのプログラムの記述要素を次のように区分して考える。

#### (1) データ項目に関する記述

データの項目名、属性、値、計算式、条件式などを記述する部分。

#### (2) 入出力レイアウト、ファイルレイアウトに関する記述

データ項目やファイル、画面、帳票のレイアウトに関する定義。

#### (3) 処理手続に関する記述

処理の流れを記述する。この部分は更に以下の4種類に分類出来る。

##### (a) プログラム形式

プログラムの処理を「見出し処理」「明細処理」「合計処理」などまとまった単位にブロック化し、その組合せでプログラムの構造を示す。

##### (b) 入出力処理

ファイルや帳票、画面に対する入出力処理。

##### (c) 入出力項目、作業項目に対する処理

ファイルや画面から入力された項目の内容を判定したり計算したりする項目に対する処理の記述である。

##### (d) コンピュータ特有の処理

合計項目の初期化、用紙の改ページなどコンピュータ特有の処理の記述である。

### 3.3 可視化プログラムの構造

プログラムの記述要素を次の要点で可視化する。

#### (a) プログラム内での記述項目を減らす。

データ項目に関する記述、入出力、ファイルレイアウトはシステム内で共通部品として扱い、独立した外部定義とする。処理手続のうち画面処理、帳票処理はあらかじめ準備したパターンを選択するようにする。

#### (b) データ項目の部品化

項目の名前、意味、属性、計算式や値など項目の処理を含めて、カプセル化し、部品化する。

#### (c) 入出力レイアウト部の図式表現

ファイルレイアウトのイメージが一目で理解できる図式表現とする。

以上の考察からプログラムの構造を可視化したのが図2である。

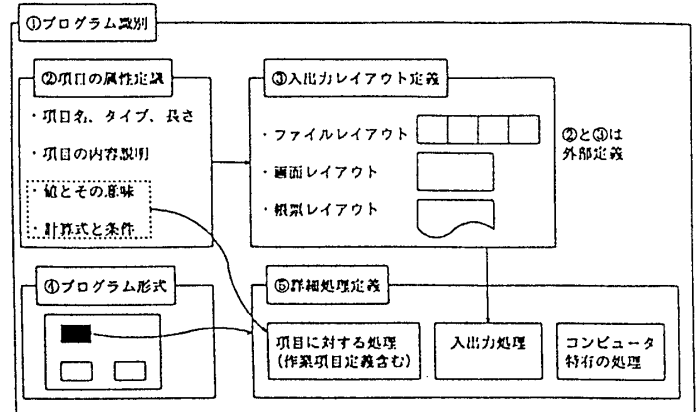


図2 可視化プログラムの構造

### 3.4 可視化を実現するシステムの構成

この開発環境は3種類の機能要素から構成する。

#### (1) 情報定義部

項目の属性定義、入出力レイアウト定義、プログラムのパターン定義から構成する。定義されたものはプログラム作成部で参照する。

#### (2) プログラム作成部

情報定義で作成されたプログラム部品を参照しながらプログラムを組み立てる。

#### (3) ソース生成部

組み立てられたプログラム情報をもとにソースプログラムを生成する。

### 3.5 可視化技法の効果

可視化技法の効果として次の事柄が期待できる。

- (1) エンドユーザーが理解しやすいため業務と一致したプログラムができる。
- (2) プログラムの修正は部品単位で行うことが出来、全体へのインパクトを少なくできる。
- (3) プログラムの流用が容易となる。
- (4) エンドユーザーコンピューティングが容易となる。
- (5) 生産性や品質が向上する。

### 4. むすびに

企業の業務を3つの階層に分類して、それぞれの性質に適合したソフトウェアの実現手段を選択することと、個別開発するソフトウェアについて可視化によってエンドユーザーに理解できる表現方法を提案した。今後このシステムの実用化を検討している。

### 参考文献

- (1) 魚田勝臣：“ソフトウェアリユースモデルと開発環境” 専修大学 経営学論集 Vol. 57, PP. 88-117(1993. 10)