

# CIMパッケージソフト設計における構造化分析技法の試行とその結果について†

3K-8

○唐島めぐみ\* 土居公司\* 佐藤正道\*\* 福留五郎\*  
 \* オムロン(株)FA統轄事業部コントローラ研究所  
 \*\* オムロン(株)技術本部開発支援センタ

## 1 はじめに

我々はソフトウェアの開発効率向上のための取り組みのひとつとして、ソフトウェア上流設計において構造化分析技法の適用を試みた。特に、我々のCIMパッケージソフト設計への適用を通し、構造化技法を普及する上で明らかになった問題とその対策について考察する。

## 2 構造化分析技法の試行経過

我々は、FA向けのコントローラやコンピュータ、そのコンピュータ上で動くCIMパッケージソフトなどを開発している。91年からソフトウェアの一部分の上流設計にCASEツールを使ってDeMarcoの構造化分析技法を適用し、現在はCIMパッケージソフトの上流設計に適用している。これまでの試行を通して次のような効果があった。

- (1) ソフトウェアの概要がわかりやすく表現でき、データの抜けや矛盾が見つかりやすいので、ウォークスルー用の資料として活用度が高い。
- (2) データ・フロー・ダイアグラム(以下、DFD)においてプロセスやデータなどの数や記述方法に基準を設けることにより、ある程度の設計品質を見込める。

また、次のような問題点が明らかになった。

- (1) 各開発担当者のノウハウに頼る進め方をしたことにより、構造化分析技法の適用方法や開発手順への組み込み方が統一されていない。
- (2) ユーザインタフェースの部分をDFDに記述する方法がないため、表示機能が中心となるCIMパッケージソフトなどのアプリケーションソフトへの適用が広がらない。
- (3) データ中心のトランザクション処理は表現しやすいが、状態遷移中心のリアルタイム処理は表現しにくい。

†Trial of the Structured Techniques for the CIM Package Software Design.

Megumi Karashima\*, Koji Doi\*, Masamichi Sato\*\*, Goro Fukutome\*

\* OMRON Co., Intelligent Controller Lab

\*\* OMRON Co., R&D Support Department

いため、後者が中心のソフトウェアへの適用が広がらない。

それぞれの問題点について、その原因と対策を考察する。

## 3 開発手順への組み込み

我々が構造化分析技法を試行し始めた時には、社内にもほとんど実績がなかったため、各開発担当者ベースでボトムアップ的に適用を進めた。そのため各担当者のノウハウは蓄積されたが、構造化分析技法の適用方法や開発手順への組み込み方は統一されていない。例えば、開発ドキュメントにおけるDFDの位置付けなどには現状ばらつきがある。

今後、試行段階から運用段階へと移行するためには、このようなばらつきをなくし、部門として技法を導入する仕組みを作る必要がある。そのため、現状の適用方法を整理し、次の2つに取り組んでいる。

- (1) 部門標準の開発手順の中に構造化分析技法の適用を取り込む。具体的には、各ダイアグラムを作成するソフトウェア開発の工程、および開発ドキュメント中の各ダイアグラムの位置づけを明確にする。
- (2) 個人のノウハウを部門のノウハウとして活用するしくみを作る。

## 4 ユーザインタフェースの記述方法

我々が開発するCIMパッケージソフトにはユーザインタフェースが中心のソフトウェアが多い。このようなソフトウェアの設計において、我々はユーザインタフェースと内部処理を分けて記述している。内部処理についてはDFDで記述できるが、ユーザインタフェースについてはDFDでは記述できない。しかし、ユーザインタフェースと内部処理との関連がわからないと、ソフトウェアの様子が理解しにくい。

そこで、ユーザインタフェースをDFDで規定されていない別の図形で表現するようDFDを拡張し、内部処理との関連を記述できるようにした。記述例を図1に示す。図1では六角形の図形が画面を表している。なお、ユー

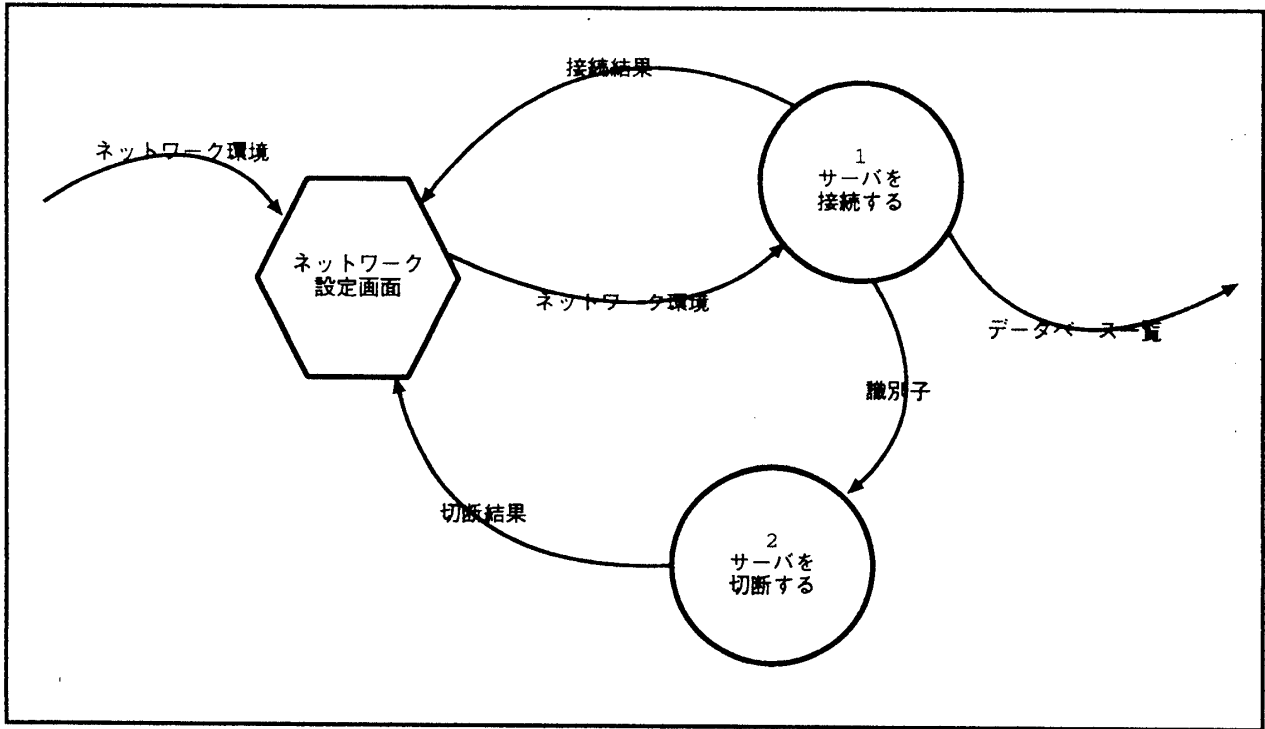


図 1: ユーザインタフェースの記述例

ザインタフェースの詳細な仕様は画面仕様として別に記述した。

このように DFD を拡張したことにより、画面から入力するデータとそうでないデータが区別でき、画面と内部処理との関係が明確になった。さらに、この拡張した DFD から、画面の重要な要件のひとつである表示モード (例えば Motif におけるアプリケーションモジュールまたはモードレス) を決定する手法を検討中である。

## 5 状態遷移や制御の記述

状態遷移や制御の記述方法は従来から問題視されているポイントであり、その対策としては例えば Hatley らによるリアルタイム拡張された構造化分析技法の適用などがある。

しかし、このような制御が中心となるソフトウェア開発者への普及が進まないという現状を考慮すると、構造化分析技法上の問題とは関係なく別の適用上の問題があるのかもしれない。例えば、従来の技法で十分設計上の問題がないのならば強いて構造化分析技法を適用する必要はないだろうと考えている。

## 6 今後の課題

今回の試行結果を踏まえ、我々の部門において今後は構造化技法の運用段階へ移行する。構造化技法をより定着させるには、すでに述べた構造化技法を取り込んだ開発手順の着実な運用や、適用ノウハウの継続的な蓄積がポイントとなる。

## 7 参考文献

- 1) Tom demarco: "構造化分析とシステム仕様", 日経 BP 社 (1988)
- 2) Derek J. Hatley 他: "リアルタイムシステムの構造化分析", 日経 BP 社 (1989)
- 3) 山崎竜二 他: "構造化分析技法を用いたソフトウェア開発", OMRON TECHNICS, Vol.32, No.1 (1992)