

3L-7

SA手法へのユーザインタフェース情報の追加

寿原 則彦、堀川 博史、高野 彰
三菱電機㈱ 情報電子研究所

1. はじめに

近年CASE(computer aided software engineering)ツールと称する、ws上で設計情報をグラフィックに記述するエディタおよび、入力された情報を検索、分析する解析ツールが米国を中心に広まりつつあり、日本でも注目を集めるようになってきた([1])。こういったツールで主に採用される手法はSA(Structured Analysis:構造化分析)/SD(Structured Design)と呼ばれるものである。そこで本報告では、実際にツール作成に携わった経験等([2],[3])も踏まえ、特にSA手法について、問題点等を考察する。

2. SA/SDにおける標準的な作業

SA手法は、設計するシステムを、機能とその間のデータの流れに着目し、それらを図的に表し、機能をブレイクダウンして行くことにより機能を明確化し、設計情報の抜けを極力押えていく手法である。具体的には次に示すような作業を実施する。

- (1)データフロー図の編集
- (2)ミニスペック、データディクショナリの編集
- (3)入力した設計データの解析、検索

SD手法は、SA手法の適用後、実際のプログラムのモジュール構成、および個々のモジュールのインタフェースの決定を図的表現などを用いて行なう手法である。SA/SDが終了すると、今度はモジュールのアルゴリズムの決定、コーディングと作業は進んでいくことになる。

3. 問題点

SA手法の目的は、設計の前段階で図などを用いることによってデータや機能の抜けをできる限り押えることである。ツール化により手法を適用できる環境はかなり整ってきた。しかし実際に適用してみるとこの段階までに得られた情報だけでは、設計情報としては不十分であることが分かった。外部仕様書に記述すべき項目で、SAで扱わない情報を記述量から順に3つ上げると、

- (1)ユーザインタフェース
- (2)エラー処理
- (3)実現方式についての制限事項

ユーザインタフェースに関しては、今までSAという枠の中ではほとんど議論されてきていない。最近のソフトウェアはボタン、メニューなどを活用して高度なユーザインタフェースを提供するものが増えてきている。ユーザインタフェースはいわばツールの共通イメージをつかさどる部分であり、一般的には外部設計段階で決められるものであり、SAの枠組の中で記述されるのが自然である。そこでSA手法にこの情報を付加する方法を考えてみた。

4. 設計手法の考察

ユーザインタフェースの情報を付加する方法を考える際に注意すべき点としては、特徴的な機能とユーザインタフェースとの対応関係を明らかにする必要があることである。ユーザインタフェースの設計は画面形式を決定することである。そこで、SA手法の適用と並行して画面設計を行なう。画面イメージとの対応がはっきりしている機能についてはその間にリンクを張る。これにより一つの機能からそれと関係が深い画面イメージを、その機能に対するミニスペックを見るように参照できる。

5. まとめ

ここではSA手法への、ユーザインタフェース情報の取り込みのための一つの案を示した。今後はここで述べた手法の適用を通して問題点の明確化を進めていきたい。

参考文献

- [1] IEEE Software, March 1988.
- [2] 情報処理学会第34回全国大会4T-4.
- [3] 情報処理学会第36回全国大会2M-8.