

システム要求仕様化のための状態遷移システム

4 J-5

表示インターフェースの開発

山田 将弘 石井基樹 金指文明 富樫 敦†

静岡大学情報学部理工学研究科‡

1. はじめに

一般的に、ソフトウェア開発には多大な労力と時間が必要とされ、開発者を悩ませている。その理由は、ソフトウェアの実行は自動または半自動であるのに対し、ソフトウェアの開発作業は手作業だからである。また、一度作ったソフトウェアを改良する時には、その古いソフトウェアを完全に理解する必要があると言う再利用性の困難さもある。これらの問題を解決するために、多くの企業が多くの開発環境を製品化している。これらの開発環境では、手作業コーディングを GUI コーディングで、再利用性については狭義のコンポーネント指向で開発の支援をしている。しかし、これらはコーディングレベルでの開発支援であり、ユーザ要求中のモデルの正しさを前提としている。つまり、現在のソフトウェア開発環境ではソフトウェア開発の上流工程の支援が十分ではない。

本統合型支援環境では、この問題を解決するために要求分析などのソフトウェア開発における上流工程の支援をする。ただし、対象とするのはリアクティブシステムである。この統合型支援環境ではユーザ要求を整合性がとれるようにし、最終的にはシステムのプロトタイプであるプログラムを出力する。

本論文では、この統合型支援環境を成すツールの1つである内部動作表示システムの設計を行う。このツールは表示だけでなく、図に対しての編集も行う。

以下にこの開発支援環境の全体構成図を示す。

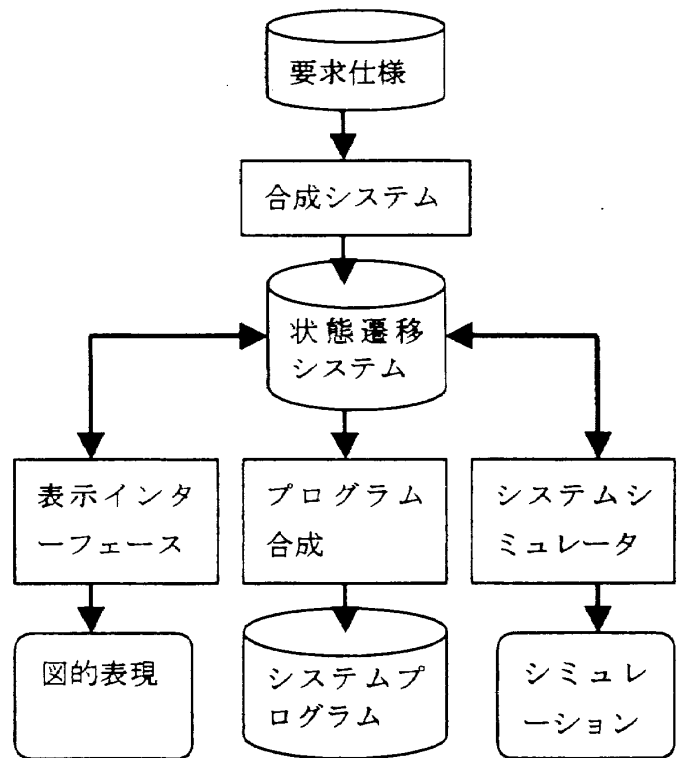


図1. 開発支援環境の全体構成図

2. 表示インターフェース

2. 1. 状態遷移システムの表示

本統合型支援環境は、支援対象がリアクティブシステムなので上位システムからの本ツールへの入力状態遷移システムが適当である。この状態遷移システムのフォーマットは本統合型

“Development of Graphical Interface using State-Transition-System for System requirement Specification”

† Masahiro Yamada, Motoki Ishii, Fumiaki Kanezashi, Atsushi Togashi

‡ Graduate School of Science and Engineering, Shizuoka University.

支援環境固有のものであるとする。

本ツールは受け取った状態遷移システムをグラフ表示する。グラフの表示形態としては、一般的な状態遷移図や、3D 表現を考えている。ここで 3D 表現を考える理由は、状態遷移図が複雑になると 2D の場合、各状態の配置が非常に困難になってしまうからである。ただし、現在この 3D 表現では、状態遷移図全体を表示するのではなく、その時の着目状態を中心とした一部分の表示を行うように考えている。

## 2. 2. 状態遷移システムの編集

本ツールで表示されるものは状態遷移システムであるので、編集作業は表示される状態と遷移に対して行う。

考えられる編集作業としては、状態追加、状態削除、遷移追加、遷移削除、各要素の名前変更、状態のグループ化などがある。

この編集によって、表示されているグラフが変更されていくわけであるが、この編集内容を統合支援環境の最初の入力である要求仕様に反映させたい。したがって、状態遷移システムの編集後このツールでの編集内容を上位システムへ伝えるようにする。

## 2. 3. 上位システムへの統合

本ツールは統合支援環境を成すツールの内の 1 つである。当然、本ツール作成後には統合支援環境へ組み込まれる事になるのだが、理想としてはその組み込みを容易に行いたい。また、これに取って代わる新しいツールができた場合はこの取り替え作業も容易にしたい。さらに、このツールを他の目的で使えるように再利用性についても考えたい。

以上の要求から、このツールを狭義のコンポーネント指向におけるコンポーネントとして作成する事とする。本統合支援環境は、プログラミング言語として Java 言語を使う事になってい

るので、Java Beans API に基づき、このツールを Bean 化 (コンポーネント化) する事とする。統合支援環境を Bean を扱えるようにしておけば、ツールの組み替えなどが容易に行えるようになる。従って、本ツールとその上位システムとのメッセージ交換はイベントによって行う事となる。

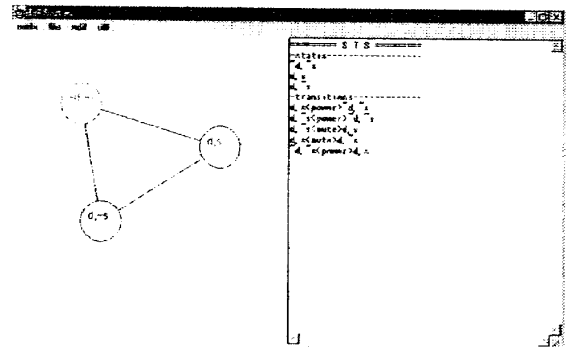


図 2. 表示インターフェース試作品

## 3. 結論

本ツールの設計において、考えなければならない事は 2 つある。1 つは、表示編集ツールとしての扱い易さである。とくに表示形態には気を使う必要がある。もう 1 つは、1 ツールとしての扱い易さである。最終的なシステムとの統合や再利用性はツール作成後に行うのではなく、ツール設計時に考慮すべき事である。

## 謝辞

本論文は、一部文部省科学研究費 (基盤研究 (C)08680343, 重点研究 09245214), 電気通信普及財団, 東海産業技術新興財団, 栢森情報科学新興財団の援助による。

## 参考文献

- [1]Togashi, A., Kanezashi, F. and Lu, X.: A Methodology for the Description of System Requirements and the Derivation of Formal Specifications, In FORSE/PSTV'97, pp.383-398(1997).