

# ソフトウェアプロセスガイドツールのための 拡張ワークフロー表現とその実現

6R-2

満田成紀 四ッ柳茂樹 鯨坂恒夫 上林彌彦

京都大学工学部

## 1 はじめに

WWW(WorldWideWeb)におけるナビゲーションは、プロセスのガイドにも適している。しかし、その記述言語HTML<sup>[1]</sup>(Hyper Text Markup Language)はプロセスの構成や条件の記述力に乏しい。

本稿では、IDEF0<sup>[2]</sup>(Integrated computer-aided manufacturing DEFinition 0)をワークフローモデルとして用い、それをHTMLに変換することにより、プロセスを管理しガイドするツールについて述べる。IDEF0でノードからの入出力に用いられるアークに、条件記号を追加するような拡張を行って動的なプロセスの起動・終了条件を記述し、そのプロセスをWWW上に表現してナビゲートするためのツールを作成した。

## 2 IDEF0モデルの動的解釈

### 2.1 IDEF0モデルについて

IDEF0は、1970年代に米国空軍プロジェクトを通して開発されてきた業務分析、システム設計のためのモデリング手法 IDEF の一部であり、機能モデル作成のために使われている。

IDEF0では、プロセスがノードで表され、その関連を表すアークは、ノードに連結している場所によって意味が変わる。ノードに対し連結するアークは、左から入るものは入力、右から出るものは出力、上から入るものは制御、そして下から入るものは機構を表している。

### 2.2 動的な実行のための拡張

IDEF0モデルは、フィードバック要素があるため、静的にプロセスの実行順序を決定できない。そのため、動的な解釈をしなければならない。IDEF0モデルを動的に解釈し、次に実行されるプロセスを決定するためには、あらかじめ要求で与えられている制約をモデルに記述する必要がある。

そこで、制約に基づいてプロセスとつながるアークに条件記号をつけ、その条件が満たされた時にプロセスの実行を開始(あるいは終了)するという方法を考えた。条件記号にはプロセスごとに番号がつけられ、あるプロセス内で、ある番号の開始(あるいは終了)条件が全て満たされていれば、そのプロセスを開始(あるいは終了)できるものとした。使用する条件記号は、次のものである。(nは番号が入る)

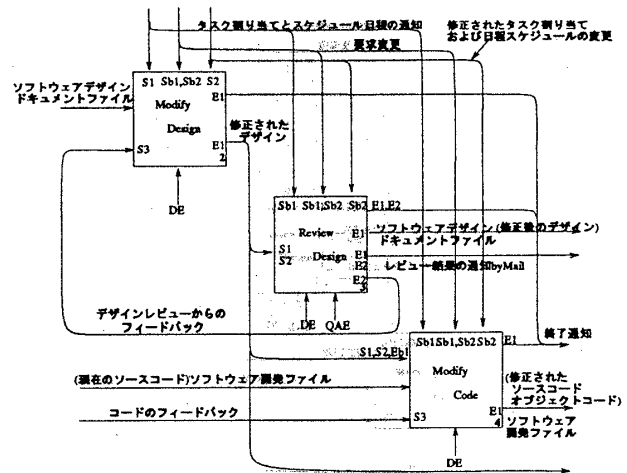


図 1: IDEF0モデルの例

- Sn アーク使用時に条件が満たされ、プロセス終了とともに、リセットされる。開始条件である。
- En アーク使用時に条件が満たされ、プロセス終了とともに、リセットされる。終了条件である。
- Sbn アーク使用時に条件が満たされ、プロセスが終了してもリセットされない。開始条件である。
- Ebn アーク使用時に条件が満たされ、プロセスが終了してもリセットされない。終了条件である。

この条件記号を付加することによって、最初の入力が与えられれば、プロセスの実行順序を動的に決定することができるようになる。この条件記号を適用した IDEF0モデルの例が図1である。例題は[4]を用いている。

もし複数のプロセスが同時に実行できる場合は、機構(箱の下からのアーク)で定義された資源の許す限り、並列に実行することとする。

### 2.3 条件記号による解釈の適用範囲

この条件記号の使用制限は、次のものである。

1. 実行できるのは AND、OR 条件だけである。
  2. 制御(箱の上からのアーク)によってプロセスの実行順序自体が変わる場合は適用できない。
- 1については、NOT条件は全体集合が明確でなければ指し示せないが、一般に全体集合は不確定であることが多いので、使用することはほとんどないと考えられる。2については、無作為に実行順序が変わることは少なく、もし実行順序の変わり方が決まっているならば、これを条件の変更とみなして、条件記号をあらかじめ変更することで対応

できる。

よって、この拡張は十分使用できるものと思われる。

### 3 IDEF0 モデルの HTML への変換

IDEF0 モデルにおいて、プロセスを表わすノードを HTML に、アークをそのリンクとして WWW 上に表現し、動的な実行をすることを考える。このときリンクは、CGI<sup>[3]</sup> (Common Gateway Interface) を用いて制御用のプログラムに必要な値を全て入れることで、起動するツールや IDEF0 を拡張した条件記号を表現する。

その際に重要なことは、どのようなツールをどこで実行するかということと、プロセス内での分岐を考慮することである。

#### 3.1 リンクによる起動ツールの記述

IDEF0 モデルのプロセスを実際に HTML に変換するとき記述する必要があるのが、リンクである。基本的にはリンクとなる部分は、入力・制御に関する部分、出力に関する部分、プロセス自身が実行する部分の3つに分けられる。

入力・制御に関する部分は、条件のチェックだけでよいものと、作業者の参照が必要なものの2つあって、他ツール起動の必要はない。

出力に関する部分は、実際に起った結果を出力するものであるため、ツールの起動を考える必要がある。しかし、どんな情報を出力するかは決まっているので、それに応じたツールを起動すればよい。

プロセスによって実行する部分は、そのプロセスの主要な作業であり、出力に表現されていないもので、具体的にはプログラムテスト等がある。これは、プロセス自体が持つ仕事と考えられる。もし、出力に関する部分にツールが表現されていれば、ここでツールは起動しない。

#### 3.2 成功・失敗のあるプロセスの記述

プロセス内での分岐がある場合は、そのプロセス内のツールが、成功か失敗かを判断している。実際、分岐が起こった時は、他方のリンクは消した方がよい。そのため、成功・失敗のあるプロセスでは分岐を含む HTML ファイルの他に、成功・失敗時のファイルを用意し、分岐した時点でそのファイルに移るようにする。

## 4 プロセスガイドツール

ここでは実際に作成したツールのうち、特に重要なものである、ツール制御とマークによるナビゲーションのプログラムについて述べる。実際のガイドツールの画面例は、図2のようになる。

#### 4.1 ツール制御用プログラム

ツールは2種類に分け実行される。

- ユーザの関与しない命令 (コピー等) は、シェルで実行可能なので、プログラム内で記述する。
- ユーザの関与する命令 (エディタ等) は、普通にシェルで実行しようとする時、ツールを終了するまでブラウザが読み込みを終了しないので、少々危険な方法ではあるが、外部ビューワとしてシェルを起動する設定にし、そこで命令を実行する。

#### 4.2 ナビゲーション用プログラム

表示された HTML の、作業内容が書かれたリンクをクリックすると、その作業に関するフラグ (条件記号にあたる) が全てチェックされ、そのプロセスで以後行う必要のない作業ならば、そのリンクが消える。この動作が繰り返されるが、もしその時点で終了できるプロセスがあれば終了し、資源の残量を考慮して、開始できるプロセスを判断し表示する。

プロセスの開始・終了条件は、IDEF0 でのマークを元にファイルに入れておき、フラグが揃ったときに条件が成り立つようにしている。

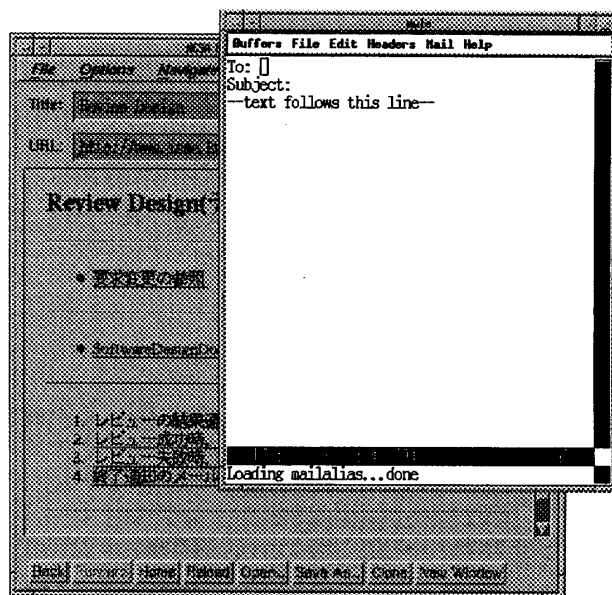


図 2: ガイドツールの画面例

## 5 おわりに

本稿では IDEF0 モデルの動的な解釈をするために拡張した条件記号、およびその IDEF0 モデルの HTML ファイルへの変換について述べた。今後は、IDEF0 モデルから HTML ファイルへの自動変換を実現していく予定である。

### 参考文献

- [1] "Hyper Text MarkUp Language(HTML): Working and Background Materials", <http://www.w3.org/hypertext/WWW/MarkUp/>
- [2] "Announcing the Standard for INTEGRATION DEFINITION FOR FUNCTION MODELING (IDEF0)", Federal Information Processing Standards Publication 183, <http://www.ncsl.nist.gov/standsp/idef0.html>
- [3] "Overview of CGI", <http://www.w3.org/hypertext/WWW/CGI/>
- [4] Marc Kellner: "Software Process Modeling Example Problem", International Software Process Workshop, 1990.