

タスク実施プロセスの構築および記述方法

5Q-6

木村泰己、長崎等、野中誠、東基衛
早稲田大学大学院理工学研究科

1 はじめに

本研究は、DAISY (Distributed and Adaptive Information SYstems) プロジェクト [1] の一環として、組織内でタスクを実施するプロセスを共有化し、エンドユーザが迅速にタスクを実施できる仕組みを構築することを目指している。本発表では、タスク実施プロセスの構築方法、ならびにそのプロセスを記述するモデルを提案する。次にプロセスを構築し、タスクの実施を支援するシステムの概要を示し、その利点について考察する。

2 タスク実施プロセスのモデル化

用語の定義

本研究で使用する用語は、[2] を参考に以下のように定義する。

<タスク>ユーザの目標とする状態を導くためのプロセス全体。またタスクはサブタスクに分割可能な場合もある。(例) 電子メールで連絡するなど。

<タスク実施プロセス>タスクを実施し目的を達成するための一連のタスクの処理手順。

<アクション>タスクを実施するために、ソフトウェアを操作する際の人間のプロセス。(例) センタリングをする、文字を大きくするなど。

<メソッド>アクションに対するユーザの行為の具体的な手順。(例) 文字を選択しコマンドを選択するなど。

<ユーザ操作要素>ソフトウェアを利用しているときにユーザが行える最小単位の基本操作。(例) メニューの選択、マウスの移動など。

タスク実施プロセスのモデル

「状態の遷移を表現することが可能」、「視覚的に表現することが可能」、「条件を記述することが可能」、「階層構造で表現が可能」などの条件を考慮し、タスク実施プロセスを記述するモデルを提案する。

提案するモデルはベトリネットを拡張し、トランジションをタスク、サブタスクおよびアクション、プレースをシステムの状態と捉える。モデルの例を図1に示す。

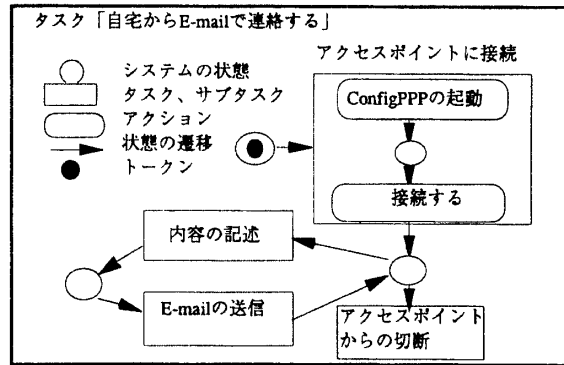


図1 タスク実施プロセスのモデル例

モデルのルール

モデルを記述する際には以下のルールに従う。

- ・タスク、サブタスクの発火前、発火後の状態は常に一つとする。また、どのようなメソッドを用いてアクションを行ったとしても、システムの状態は常に一つとする。

- ・ある状態において、順序に関係なく複数のすべてのタスクを実施することで、目標の状態に導くことができる場合、タスク間を+で結ぶ(図2の左図参照)。

- ・タスクの実施は任意で行う場合、そのタスクをカッコ () 内に記述する(図2の右図参照)。

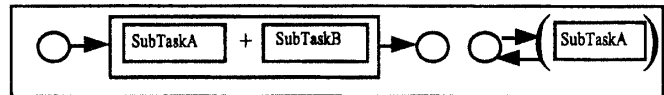


図2 モデルのルール

3 タスク実施プロセスの構築方法

2章で述べたモデルを用いて、タスク実施プロセスを構築していく手順を以下に示す。

タスクの分割

タスクが連続的に進められる場合、ウィンドウやダイアログを表示し、システムの状態に変化をもたらすタスクを意味的に解釈したものがサブタスクの候補となる。また、タスクのアウトプットが明確な

場合、アウトプット内の部品 (図、表、題名など) を作り出すタスクがサブタスクの候補となる。これらのサブタスクの候補によりタスクをサブタスクに分割する。

シナリオの記述ならびにモデル化

サブタスクがどのような順序で行われるかを考え、タスク終了までの主要なシナリオを記述し、そのモデルを記述する。ただしシナリオを記述する際には、例外的なサブタスクは省略する。

シナリオの詳細化

サブタスクに注目し、タスクがさらに分割可能な場合は、分割を行いモデル上に記述する。また、各状態ごとに、他のシナリオを考え、必要に応じてサブタスクを加える。

アクションの記述

サブタスクをアクションと状態を用いて記述する。また、アクションに対する複数のメソッドを記述する。メソッドの記述は、自然言語で行う。

4 支援システム

プロセスを構築し、タスクの実施を支援するシステムのプロトタイプを現在開発中である。その概要を図3に示し、各コンポーネントの機能を述べる。

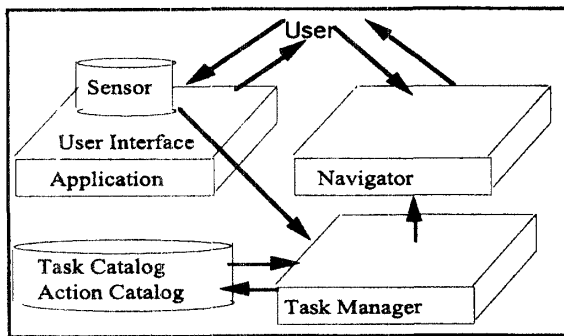


図3 支援システム概要図

4.1 Sensor

Application内のユーザ操作要素に対応するプログラム内に埋め込まれている。Sensorの埋め込まれているユーザ操作要素を Application が獲得した際に、Task Manager にメッセージを送る。

4.2 Task Manager

タスク実施プロセスのモデル化を支援し、タスクの属性をタスクカタログに、アクションの属性を、アクションカタログに登録し管理する。タスクの属性値は {名前、タスク実施プロセスのモデル、アウトプットのプレビュー} である。なお、タスクによってはアウトプットのプレビューが明確でない場合も

ある。また、アクションの属性値は {名前、アプリケーション、メソッド、ユーザ操作要素} である。

また、Sensorからのメッセージをアクションカタログに照らして、該当するアクションを決定する。そのアクションを元にモデル内のトークンを移動させ、Navigator にメッセージを送る。

4.3 Navigator

Navigator はエンドユーザが目的とするタスクを選択する機能を提供する。

また、エンドユーザがタスクを実施してから完了するまで、ユーザの取るべきアクション、ならびにそれらに対応するメソッドを示す機能を提供する。

Navigator の表示方法を図4に示す。図4において点線は選択可能なアクションを表し、アクション4、アクション5はどちらも選択しなければならないことを表す。なお、各アクションのメソッドはすべて表示するものとする。

Action1	Method1-1,Method1-2
Action2	Method2-1
Action3	Method3-1
+ Action4	Method4-1,Method4-2
+ Action5	Method5-1

図4 Navigator の表示方法

5 考察ならびに今後の課題

本論文で示したタスク実施プロセスのモデルを使用することにより、タスク単位のプロセスを一般的に記述することが可能となる。また、エンドユーザの操作を取得し、このモデルと比較していくことができる。システムが、ユーザの状態を把握することにより、これまでのオンラインマニュアル等と比べて、ユーザが次に行うであろう、または行うべきアクションを動的に示すことができる。

今後の課題として、開発段階でアウトプットとして出てくるモデルとタスク実施プロセスモデルの整合性を考えていかなければならない。また、開発段階におけるアクションセンサーの埋め込み方法を考えなければならない。

参考文献

[1] 東基衛;野中誠、長崎等、木村泰己;分散・適応型システム実現のフレームワークと目標、情報処理学会第53回全国大会、1996
 [2] 長崎等、東基衛;適応型ユーザインタフェースの構築、情報処理学会第53回全国大会、1994