

ワークフローモデルにおけるビュー機能*

3E-7

国島丈生†

奈良先端科学技術大学院大学情報科学研究科

上林弥彦‡

京都大学工学部高度情報開発実験施設

1 はじめに

オフィスにおける定型的協同作業を支援するグループウェアとして、業務の流れを計算機によって支援するワークフロー管理システムが近年注目されている。我々は、業務の進行によって発生してくるさまざまな資源をも含めたデータ管理環境としてワークフローをとらえることができると考え、そのためのワークフローモデルを研究している [3,4]。

本稿では、現在研究中のワークフローモデルに対するビュー機能の実現について述べる。一つのワークフローは対応する作業全体の手順・構造を表すため、ある条件に合致した作業のみを取り出してワークフローとして見たい、といった要求は自然なものであると考えられ、ビュー機能が必要であると考えられる。

2 活動オブジェクト階層によるワークフローモデル

ここでは、[3,4]で提案したワークフローモデルの概略について述べる。このモデルでは、図1に示すように、ひとつの単位となる活動を活動オブジェクトとして表現し、活動オブジェクトの集約階層として一つのワークフローを表現する。

活動オブジェクト o は、次の7つ組で表現される：

$$o = \langle I, O, SO, E, ATTR, AG, C \rangle$$

ここに、 I はその活動に必要な入力オブジェクトの集合、 O は活動の結果生じる出力オブジェクトの集合、 SO は o の副活動を表す活動オブジェクトの集合、 $E \in \{AND, OR, EXOR\}$ は副活動オブジェクト間の実行モード、 $ATTR$ はその活動の属性集合、 AG はその活動オブジェクトを実行するエージェントの集合、 C は $I, O, SO, ATTR, AG$ に関する制約集合をそれぞれ表す。 o と $o' \in SO$ には親子関係があるという。 $I, O, SO, E, ATTR, AG, C$ などは動的に変更され得る。

活動オブジェクト o, o' について、 $x \in o.O$ かつ $x \in o'.I$ が成り立つ場合、 o' は o に依存するといひ、 $o \rightarrow o'$ と書く。依存関係はこのように暗示的に定義する他に、明示的に定義することも可能である。

2.1 活動オブジェクト階層上での属性・制約の継承

親オブジェクトから子オブジェクトへの属性・制約の伝搬を表現するため、集約階層上で親子関係にあるオブ

*A View Function on Workflow Model

†Takeo Kunishima, NARA Institute of Science and Technology

‡Yahiko Kambayashi, Kyoto University

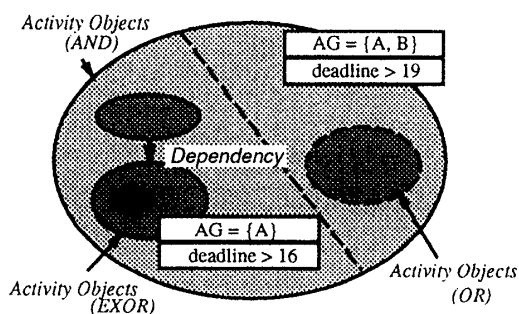


図1. 活動オブジェクトによるワークフローの表現

ジェクト間での継承機構を導入している。 $ATTR, C$ については、親オブジェクトから子オブジェクトへ継承され、子オブジェクトで上書きされないもののみが子オブジェクトでも有効となる。また、 AG については、子オブジェクトで未定義であれば無条件に親オブジェクトから継承する。

2.2 実行手順に関する意味の概要

活動オブジェクトに対応する活動が始められると、活動オブジェクトは活性化される。また、対応する活動が終了すると、活動オブジェクトは非活性化される。

活動オブジェクト o の副活動オブジェクト間の実行順序は、その実行モード E と相互の依存関係により規定される。すなわち：

- $E = AND$ の場合、 $o' \in SO$ のうち活性状態にあるものがすべて並列に実行される。
- $E = EXOR$ の場合、 $o' \in SO$ のうち活性状態にあるものが排他的に、かつすべて実行される。
- $E = OR$ の場合、 $o' \in SO$ のうち活性状態にあるものが1つ以上並列に実行される。
- $o_1, o_2 \in SO$ において o_1 が o_2 に依存するとき、 o_1 は o_2 の実行終了後に実行が開始される。

3 活動オブジェクト階層のビュー

ある活動オブジェクト階層に対して、条件を満たす活動オブジェクトのみを選択し、包含関係、実行順序、制約等を保存しつつそれらを階層的に構成した活動オブジェクト階層を、元の活動オブジェクト階層のビューと呼ぶ。本章では、このビューの構成方法について述べる。

3.1 推移的包含関連・依存関連

例えば、活動オブジェクト o_1, o_2, o_3 ($o_2 \in o_1.SO, o_3 \in o_2.SO$) において o_1, o_3 がビューに含まれる活動オブジェクトとして選択された場合を考えると、ビュー上では o_3 が o_1 の子孫活動オブジェクトになっていなければならない。また、 $o_1 \rightarrow o_2, o_2 \rightarrow o_3$ であってビューに o_1, o_3 が選択された場合には、 o_3 が o_1 に依存しているという関連が成立していなければならない。さらにこの場合、 o_1 と o_3 との間に別の作業が入り、時間的に連続して作業が行われることはない、ということもビュー上で表現できる方が望ましい。

そこで、活動オブジェクト間の関連として新たに推移的包含関連、推移的依存関連を導入する。

定義 1 活動オブジェクト o, o' 間の推移的包含関連 ($o \in^* o'$) を以下のように定義する:

1. $o \in o'$ ならば $o \in^* o'$
2. $\exists o''$ s.t. $o \in^* o'', o'' \in o'$ ならば $o \in^* o'$
3. これ以外に $o \in^* o'$ が成立することはない。

また、 o, o' 間の推移的依存関連 ($o \stackrel{\pm}{\rightarrow} o'$) を以下のように定義する:

1. $\exists o''$ s.t. $o \rightarrow o'', o'' \rightarrow o'$ ならば $o \stackrel{\pm}{\rightarrow} o'$
2. $\exists o''$ s.t. $o \stackrel{\pm}{\rightarrow} o'', o'' \rightarrow o'$ ならば $o \stackrel{\pm}{\rightarrow} o'$
3. これ以外に $o \stackrel{\pm}{\rightarrow} o'$ が成立することはない。

□

元の活動オブジェクト階層上で成立していた $\in^*, \rightarrow, \stackrel{\pm}{\rightarrow}$ については、ビュー上でも成立するようにビューを構成する。

3.2 実行モードの保存

活動オブジェクト間の実行順序の意味をビュー上でも厳密に保存しようとするならば、それらの祖先の活動オブジェクトをすべてビューに選択しなければならない。しかし、これでは、最悪の場合元の活動オブジェクト階層とビューとが完全に一致してしまう。

活動オブジェクト o, o' 間のビュー上での実行順序は、元の活動オブジェクト階層上でのこれらの least upper bound (以下 lub) となる活動オブジェクトの実行モードによって決定されると考えると、 o, o' の lub のみをビューに含めれば良く、また lub の実行モードが AND の場合は意味が完全に保存される。

ただし、lub の SO 中にビューに選択された活動オブジェクトを持たないものがあり、かつ lub の実行モードが OR や EXOR である場合、これだけでは保存されない可能性がある。例えば、 $o + o' + o''$ で o, o' が選択された場合、 o, o' がともに実行されないようなケースが抜け落ちる。また、 $o \oplus o' \oplus o''$ で o, o' が選択された場合、 o'' の実行中は o, o' のどちらも実行されないが、この意味が保存されない。以上のような問題は、ビュー上での lub の SO に新たに活動オブジェクトを一つ追加することにより回避できる。

3.3 継承の展開

上位の活動オブジェクトがビューに含まれない場合に、元の活動オブジェクト階層上でその活動オブジェクトから継承されている属性や制約がビュー上では継承されなくなる。

この問題は、ビューを構成する際に、元の活動オブジェクト階層上で継承されてくる属性や制約をすべて展開し継承関係を解消することにより解決できる。

3.4 ビュー構成アルゴリズム

1. 包含関連に基づく活動オブジェクト階層の構成
元の活動オブジェクト階層から、与えられた条件を満たす活動オブジェクトを選択する。それらを元の活動オブジェクト階層上での推移的包含関連を保存するように、活動オブジェクト階層として構成する。
2. 継承される属性・制約の展開 (3.3節)
3. 実行モードの処理 (3.2節)
4. 依存関連・推移的依存関連の保存
元の活動オブジェクト階層上で $o \rightarrow o'$ があり、 o, o' がともにビューに含まれる場合、ビュー上でも $o \rightarrow o'$ をつける。また、元の活動オブジェクト階層上で依存関連 $o \rightarrow o_1 \rightarrow o_2 \rightarrow \dots \rightarrow o_n \rightarrow o'$ があり、ビューに o, o' のみ含まれるような場合、新たに o'' という活動オブジェクトをビューに含め、 $o \rightarrow o'' \rightarrow o'$ とする。

4 おわりに

本稿、および [3, 4] で述べたワークフローモデルをシステムとして実現し、有効性を確かめる必要がある。

また、モデルに関する問題点として、データベースビューにおけるビュー更新の問題がある。本稿で提案したビュー構成方法においてビュー更新の問題が起ころうのか、どの程度の更新ならば許されるのか明らかにする必要がある。

参考文献

- [1] H. Ishii and M. Ohkubo. Message-Driven Groupware Design Based on an Office Procedure Model, OM-1. *Journal of Information Processing*, Vol. 14, No. 2, pp. 184-191, 1991.
- [2] M. Suzuki and T. Katayama. Meta-Operations in the Process Model HFSP for the Dynamics and Flexibility of Software Processes. In *Proc. 2nd. Intl. Conf. Software Process*, pp. 202-217, Feb. 1991.
- [3] 国島, 上林. 集約階層をなす協調作業上における作業間の属性の継承. 情報処理学会第 47 回全国大会, 第 6 巻, pp. 283-284, Oct. 1993.
- [4] 国島, 上林. 活動オブジェクトに基づく作業フローモデルの実行手続き. 情報処理学会第 48 回全国大会, 第 6 巻, pp. 241-242, Mar. 1994.