

## 動的なワークフロー管理と病院診療モデルへの応用

大星 直樹<sup>1</sup> 増井 久之<sup>2</sup> 上林 弥彦<sup>3</sup> 高橋 隆<sup>1</sup>

京大病院 医療情報部<sup>1</sup> 三菱電機株式会社<sup>2</sup> 京都大学工学部 情報工学科<sup>3</sup>

複数の作業による協同作業が行なわれる際には、複数の作業間の関連、物や意思の流れなどを記述するワークフローの管理が重要である。本稿では医療現場におけるワークフローの応用について述べる。ここでは作業フロー記述の際に利用者による表現、理解の容易なIDEF0モデルを取り上げ、それを拡張したものを利用した。災害や事故などにおけるトリアージ(患者選別)を例題としシミュレーションを行った。このとき患者の待ち数により動的にワークフローを変換させるものとしフローの処理能力についての検討を行った。

### Dynamic Workflow Management and Its Application to Triage

Naoki Ohboshi<sup>1</sup> Hisayuki Masui<sup>2</sup> Yahiko Kambayashi<sup>3</sup> Takashi Takahashi<sup>1</sup>

Dept. of Medical Informatics, Kyoto University Hospital<sup>1</sup> Mitsubishi Electric Co.<sup>2</sup>

Dept. of Information Science, Faculty of Engineering, Kyoto University<sup>3</sup>

It is very important to manage and control workflows when multiple parties or individuals are involved. In this paper, we describe an application of workflows to medical environment. We propose a generalized IDEF0 model and present a workflow applied to Triage. The generalized IDEF0 model is intended to express and manage dynamic changes within a workflow. Here we simulate a Triage workflow and discuss the validity of this model.

#### 1 はじめに

ワークフローモデルの構築と記述は、時間的、空間的に分散した作業間での各種協同作業の支援のためのフロー全体の処理能力の評価を目標としている。

本稿では医療現場におけるワークフローのモデル記述を取り上げシミュレーションによるフローの処理能力評価について述べる。ここではワークフローの記述の際に、利用者による表現、理解の容易なIDEF0モデルを拡張したものを利用した。またワークフローの自動的な変更についての考察も行った。

#### 2 ダイアグラムによる作業フローの記述

本章では利用者にとって規則による記述よりも分かりやすい作業フロー表現として、ダイアグラムによる記法の利用を考える。その記法としてここではIDEF0モデル<sup>1)</sup>を取り上げる。

##### 2.1 IDEF0 モデル

IDEF0(Integrated computer-aided manufacturing DEFinition 0)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、業務、生産などにおけるプロセスを正確に表現することができる。また、専門家でない一般の利用者にとっても理解の容易なモデルであるとされている。一般的に使われるワークフローに比べて、データの流れと制御を分離したことや各作業で用いられる資源を陽に定義している点で情報を詳しく表示できる。

IDEF0モデルの基本的な記述は図1に示すような五つの要素からなる。一つ一つの作業は箱によって表され、その作業の主題が箱の内部に記述される。この作業は物の変換、制御変換の双方を含む。

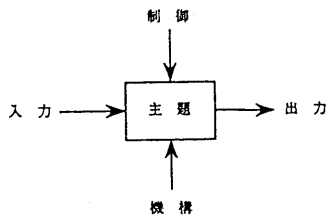


図 1. IDEF0の単位ブロック

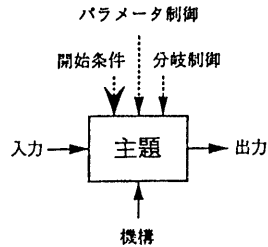


図 2. 拡張IDEF0の単位ブロック

また作業プロセスに関連する事物の流れは以下のように四つに分類され、そのすべてが矢印で表現される。これらの矢印の種類は、作業を表す箱に対する位置によって表される。入力、材料など作業によって処理、加工される物の流れを示す。これは、箱の左から入る矢印によって表される。

機構は、作業を行なう人、作業に用いられる道具などを示す。これは、箱の下から入る矢印によって表される。制御は、作業に関する条件、制約などの制御フローを示す。これは、箱の上から入る矢印によって表される。出力は、出力作業によって作られる物、なされる意思決定など、作業による出力のすべてを示す。これは、箱の右から出る矢印によって表され、矢の先は上記の入力、機構、制御のいずれかとなる。このような箱とそれにつながる矢印の一つの単位とし、それらを結合することによって一連の作業フローを記述する。また各箱の中の活動を詳細化することにより、階層的な記述をすることができる。

IDEF0 モデルの中ではデータフローモデルにおけるデータストアのようにデータや材料、製品などのものの蓄積を明示的には記述しない。

## 2.2 IDEF0 モデルの拡張

本節ではトリガ機能を用いた作業フロー管理を行なうために利用者、計算機双方にとってより詳細な情報が得られるようにIDEF0モデルを拡張することについて述べる。<sup>[1]</sup>

### 2.2.1 制御の流れの分離

IDEF0 ダイアグラムには各作業の結果として得られる出力が、他の作業の入力または機構となる物の流れなのか、他の作業の実行に影響を及ぼす制御の流れなのかの区別がない。したがって階層構造を持ったダイアグラムの中では、ダイアグラムの最終出力が他の活動の制御情報となっているのか入力となっているのかが分からない。

しかし各作業が実際に物を作る作業なのか意思決定のための活動なのかは区別して表示する必要がある。そのため、ここでは作業の制御フローを示す矢印は他と区別して破線で示すものとする。

### 2.2.2 作業開始条件

本来のIDEF0モデルは、誰が何に対しどのような作業を行なうかについて記述しており、いつ、どのような条件の時にその作業を行なうかは一般に明示的には記述されていない。しかしトリガ機能の応用による作業フロー管理のための記述に用いるためには、作業の開始時期について明確な指示が必要である。

作業開始時期についての明示的な指定がない場合には、作業に携わる人やその他必要な資源(ツール類など)が作業開始可能な状態となり、すべての入力が得られた時点、つまり当該作業の機構と入力が揃った時点で開始指示を出すという解釈をすることができる。

しかしすべての入力が揃わない内に、前もって作業の一部を開始できる場合がある。またある種の集計処理のように、入力の有無や量に関わらず一定の時期に定期的に作業を開始しなければならないような場合も考えられる。さらに作業フローにフィードバックループがある場合には、一定の条件が満たされた際に一度終了した作業を再開させる必要も生じる。

上記の場合には先の省略時解釈を用いることができないため、作業の開始時期について他の作業との関係または時刻による明示的な記述が必要である。そこで必要に応じて作業開始時期を示す制御の矢を記述することとする。ここでは利用者にも分かりやすいように作業開始時期に関する制御を他の制御と区別し、太い矢で表すものとする。従来の解釈を行う場合は、入力を制御の両方に入力が入るために冗長に見えることになる。

これらの記法の導入により、ダイアグラムの各要素は図2に示すようなものとなる。この場合にも階層的な記述が可能であることには変わりはない。

### 3 動的変更の実現

プログラム間の協調と人間の協調で大きく違うのは種々の事情によって（作業者が出張していたり仕事が大幅に遅れるなど）、作業のスケジュール自体を変えなければならないという場合が往々に生じることである。ワークフローは、変更が容易であるような機能を持つと同時に変更が容易に必要な作業者に伝達できるようにしなければならない。また、変更時に会議などが必要なら、自動的に必要な人々を召集して、会議を始めるという機能も必要となる。変更には次のような分類がある。

- 1) ダイアグラムの形を変えない変更：条件の変更、縮切の変更など
- 2) 入力、出力以外の変更：機構の変更（作業者の追加）、条件の追加
- 3) ダイアグラムの要素の除去：入力、出力を含めて、使わなくなった部品を消去する
- 4) ダイアグラム自体の変更：作業予定を大幅に変える必要が生じたとき

変更には次のような要素からなる規則が用意される。すなわち規則の適用条件として変更前の部分ダイアグラムから変更後の部分ダイアグラムへの変更が自動化できない場合、利用者の協議を行わせるようなダイアグラムを生成する。

3) および4) の処置は大変であるので、単純な場合は可能な変更をすべて含むような作業のダイアグラムを用意することも可能である。この方法は非常に柔軟ではあるが、利用者が見たときに本当はどの流れで作業が実行されているかはよくわからない。例えば作業中に今後の作業の進行などについて議論するときなどは、これは適切ではない。したがって、実際にどの流れが使われたかを示すようなダイアグラムを得る必要も出てくる。すなわち IDEF0 で作業管理を行う場合は各作業を表す基本的な要素を IDEF0 で用意すると、各基本的な要素に対応するECA規則を用意しておくこと、並びに問題が生じたときの流れを制御するような冗長性を付与する方法を開発すること、および実際の状況に応じて動的に IDEF0 を構成することといった機能が必要となる。

4) のために、メタ規則としてワークフローを条件によって変えるという規則を考える。この方法を用いると、作業フローをその状況に応じて動的に変更することができる。また、動的な変更結果は IDEF0 で表示されるため各利用者にとっても理解しやすいものとなる。

次に利用者協議を起動する変更規則の例（上記4）の特別な場合）を示す。図3はプログラム開発の場合のワークフローの例を示す。この場合、システム設計の後、並行して設計検証とプログラム化が行われる。設計検証の結果がシステム設計とプログラム化の制御として与えられ、もし問題があればシステム設計とそれに対応したプログラム化が修正される。設計検証が縮切に間に合わないとき、条件が満足されるために自動的にワークフローの変更がなされる。また、図4は変更が自動化できない場合に利用者の協議を行わせ、支援を行うかどうか決定するためのダイアグラムを生成する変換規則を示している。ここで、調停プロセスの調停者は新しい縮切と調停する機構の状況を用いて決定する。

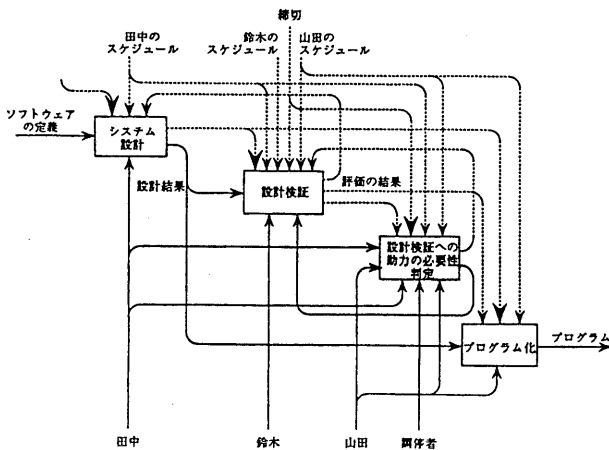


図3. プログラム開発時のワークフロー

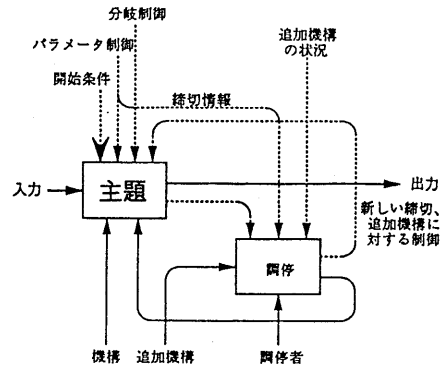


図4. 利用者の協議による IDEF0 ダイアグラム

図4に示すのは、例えばある作業を行なっているときに締切りに間に合わない場合にどうするかという処置を示すような一般的な箱である。すなわち実際の日と作業にかかる予定の日数および締切りから、締切りに間に合わない可能性に相当する出力が出される。そうすると対応する人々のスケジュールを調べて、できるだけその仕事に専任させるとか別の人が助けに行くということが必要である。このためには、その仕事に関与する人々に対するスケジュールを制御とし、実際の人を機構とする。実際にはこれは作業にあたる人ではなくてそれを協議する人々であるので必ずしも実際に仕事をする人と同じであるとは限らない。その結果として新しい作業者が割り当てられるとその信号が制御条件として与えられる。この場合には、作業者としては初めからすべての可能な人が機構として使えるようにしておかなければならない。複数の仕事が行われた場合にそれらをまとめるような箱も必要である。

著者らは、IDEF0のシミュレータを開発した。これによってワークフローの評価、決定を行うことが可能になった。

#### 4 拡張IDEF0モデルによるトリアージ評価

##### 4.1 ワークフローとしてのトリアージ

トリアージとは、集団災害時などに行われる患者の重症度と優先治療順位を決める選別行為である。その判定は、表1のプロトコル<sup>14)</sup>によってなされる。トリアージに当たる医師等はできるだけ短時間に患者の状態を判定、診断し処置、あるいは搬送の決定を行わなければならない。患者はそれぞれの状態に応じて4段階の判定をつけられ、次の処置へと進む。前節で示したIDEF0の拡張は次のような理由でトリアージに適している。1) データのフローに相当するのは患者のフローである。患者がどのような処置を受けるかがワークフローの主な表示内容で、2) 制御としては患者のキューや医師のスケジュール等が考えられ、3) 開始条件はとくに緊急を要するものが重要である。図5に拡張IDEF0モデルによるトリアージのモデル記述を示す。ここでは制御の流れをよりわかりやすくするため、前記の記号を少し変形させている。このトリアージのモデル記述の際に注意することは、(1) 患者の動線方向は1方向で逆行させないこと、<sup>14)</sup> (2) 判定の医師を複数としそれぞれの医師の心理的負担を軽減すること、(3) 比較的判定の簡単な軽傷者に対するトリアージと熟練と経験を要する難しい段階でのトリアージという2段階でのトリアージを行うということである。重傷者は、判定プロトコルによりトリアージ1でP-1, 2群の出力になりトリアージ2の入力になる。一般にワークフローは状況が時間経過に伴い変化し、より適切なものが要求される。このため、フローの動的変換を要求されるがこのモデルではトリアージ1の入力である患者の待ち数によって動的に変化させるものとした。すなわち、待ち数がある一定数を越えたときトリアージ1を並列化し処理能力の向上を図るモデルである。

優先度	処置	色別	疾病状況	診断
P-1	最優先	赤	生命、四肢の危機的状態で直ちに処置の必要なもの	気道閉塞または呼吸困難、重症熱傷、心外傷、大出血または止血困難、開放性胸部外傷、ショック
P-2	待機的	黄	2～3時間処置を遅らせても悪化しない程度のもの	熱傷、多発または大骨折、脊髓損傷、合併症のない頭部外傷
P-3	保留	緑	軽度外傷、通院加療が可能程度のもの	小骨折、外傷、小範囲熱傷(体表面積の10%以内)で気道熱傷を含まないもの 精神症状を呈するもの
P-4	死亡	黒	生命徴候のないもの	死亡または明らかに生存の可能性のないもの

(New York State Department of Health, MCI Manual)

表 1. トリアージのプロトコル

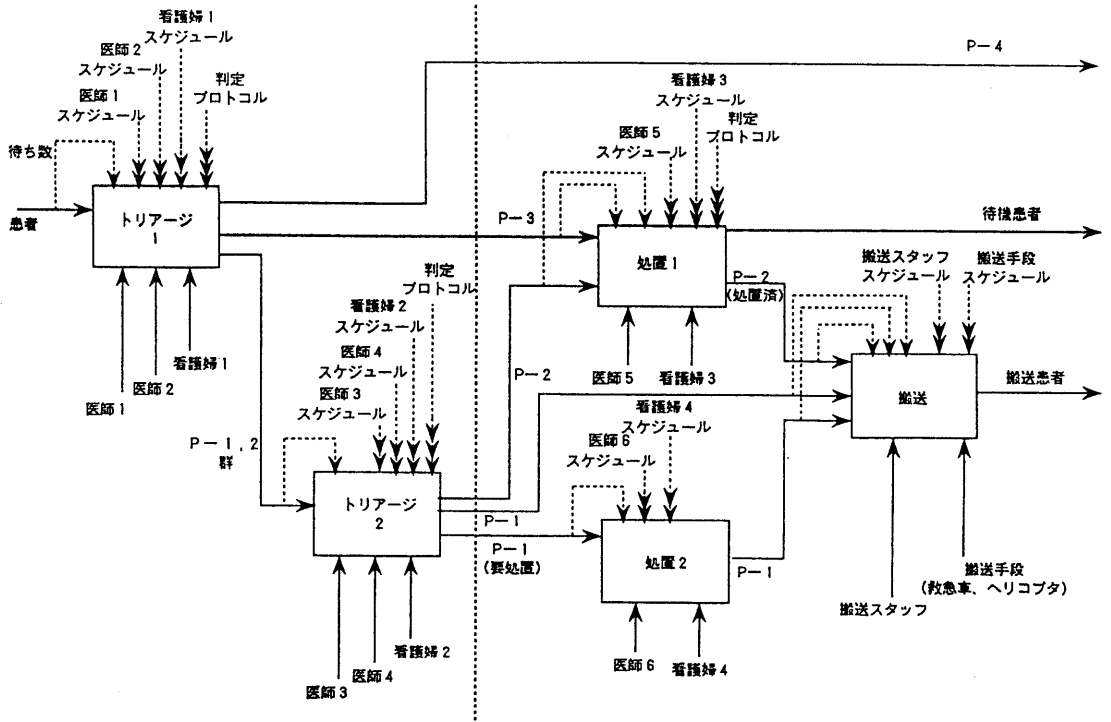


図 5. 拡張IDEF0によるトリアージモデル

#### 4.2 シミュレーション

前記のモデル図に基づいてシミュレーションを行った。患者の到着率は、平均30人/60分のポアソン分布とし、P-1,2の重傷者の割合を到着患者の20%とした。P-1からP-4までの判定がついた患者数の評価を行い、処置、搬送にまで至る患者処理数についての評価は行わなかった。(図5縦の破線までの評価を行った。) トリアージ1に要する時間を変数とし、トリアージ1を並列化する患者の待ち数を5、トリアージ2に要する時間を3分として、到着患者50人、75人を判定するのに要する時間を評価するシミュレーションを行った。この結果を図6のグラフに示す。このグラフによりトリアージ1に要する時間が3分を超えれば並列化により処理時間を減少できると予測できる。また、2分を超えなければ並列化するより単一で処理する方が効率が良い。これは判定に時間のかからない熟練した医師であればトリアージ1は単一で十分であり、判定に時間のかかる医師でも並列化することによりフローの処理能力の低下を防ぐことができることを示している。

Time for Triage  
(divergence of flow : 5patients)  
(Triage2 : 3 minutes)

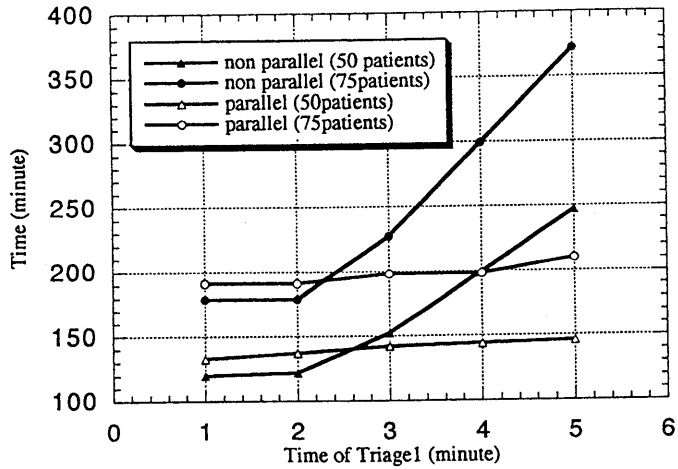


図 6. シミュレーション結果

## 5 むすび

本稿は動的ワークフローの記述モデルとして拡張IDEF0モデルを提案し、応用例として患者選別のワークフローであるトリアージを取り上げ、シミュレーションを行った。このシミュレーションによりワークフローの処理能力の評価を行うことができ、このモデルの有効性を確認できた。拡張IDEF0モデルは、モデル記述の分かりやすさからユーザインタフェース、内部モデル、そしてシミュレーションの統一性のとれたモデルとして期待できる。

## 参考文献

1. D. A. Marca and C. L. McGowan : IDEF0/SADT Business Process and Enterprise Modeling, Electric Solutions, (1993)
2. H. Masui, M. Nomoto and Y. Kambayashi : Representation of Active Rules in Cooperative Work Environment, Proc. FGCS '94 Workshop on Heterogeneous Cooperative Knowledge - Bases, pp.213 - 17, (Dec. 1994)
3. New York State Department of Health EMS Program ; Operational Update, No.87 - 41, Management Kits for MCI Field Response, Albany New York State Department of Health, 1988
4. 青野 允, 災害被災者のトリアージ, 日本医師会雑誌 第110巻, 第6号, (1993, Sep)