

# 製造ラインにおけるトラッキングと履歴管理のための 時間/場所/物体モデル

高田 秀志<sup>†</sup> 島川 博光<sup>†</sup>  
井戸 譲治<sup>†</sup> 堀池 聡<sup>†</sup>

本論文では、製造ライン監視制御システムにおけるトラッキング機能、および、製造履歴管理機能を実現するための時間/場所/物体モデルを提案し、それに基づくミドルウェアの開発と実ラインへの適用結果について述べる。本モデルでは、実対象を製造物である物体と、物体に対して作用する場所で表現し、物体の場所間での移動を形式化するために、場所に対して5種、物体の形態に対して3種の基本要素を定義する。また、移動にともなって起動される処理をECAルールで表現し、移動前/移動時/移動後/不整合時の起動論理で場合分けする。また、時間の経過にともなって収集された製造履歴を、時間、場所、物体の各視点で集約して表現する。本モデルに基づいて実現したシステム構築環境を実ラインの監視制御システムに適用した結果、10人月相当のソースコードを自動生成できた。また、設計途中の仕様変更や実装中の設計誤りに対しても迅速な対応が可能となった。

## The Time/Place/Object Model for Tracking and History Management in Manufacturing Lines

HIDEYUKI TAKADA,<sup>†</sup> HIROMITSU SHIMAKAWA,<sup>†</sup> GEORGE IDO<sup>†</sup>  
and SATOSHI HORIIKE<sup>†</sup>

In this paper, we propose the Time/Place/Object model to realize tracking and history management functions in manufacturing line controlling and monitoring systems. The model represents a target line as objects corresponding to manufactured products and places corresponding to manufacturing processes. It is shown that places and objects can be classified into respectively five and three types for tracking. In addition, the ECA model is applied to represent behavior of objects' migration and ECA rules are invoked according to pre-move, on-move, post-move and invalid-move logic. Furthermore, manufacturing history is provided in three views from time, place and object. We have implemented a system development tool based on the model and applied to a real manufacturing line. As a result, the tool can generate source codes of approximately 10 person/month development volumes.

### 1. ま え が き

製鉄や食品、化学などのプラントでは、水処理などのように連続的に物が流れ、定常状態に保つことを目的とする監視制御とは対称的に、各製品ごとに原料の種類や分量、各工程の制御パラメータなどの製造条件を変化させ、いわゆる多品種生産を実行しなければならない。このような製造ラインの監視制御システムでは、各製造単位を原材料から最終製品に至るまで設備や工程ごとに追跡し、製造条件に応じて各製造装置の設定をしたり、製造実績データを収集したりするト

ラッキングと呼ばれる機能や、品質管理や設備保守などのために収集したデータを検索・提供する製造履歴管理機能が中核として必要となる。

現在、このようなシステムは、その大部分を対象ごとの個産品として構築しており、ソフトウェアの生産性が大きな問題となっている。その理由としては、製品ごとに発行される製造指図に基づく各制御装置の設定や製品の追跡処理が複雑であるために、異なるプラント間で汎用的に使えるモデルが構築されていないことや、計算機の能力が十分でなかった頃の名残として、汎用的なソフトウェアライブラリやツールを利用するとオーバーヘッドをとめない、個別にチューニングをしないと要求される性能を満足できないと考えられていることなどがあげられる。しかし、システムに求

<sup>†</sup> 三菱電機株式会社産業システム研究所  
Industrial Electronics & Systems Lab., Mitsubishi Electric Corp.

められる機能の増大にともない、このようなシステム構築をしていたのでは採算がとれなくなってきた。一方で、近年の計算機性能は飛躍的に向上し、共通なソフトウェアモジュールを用いても十分な性能の確保が期待できるような環境になってきた。このような状況では、製造ラインの構成や挙動、制御を表現可能なモデルを定義し、それに基づいた開発環境や実行環境を構築することが、ソフトウェアの生産性を大きく向上させると考えられる。

本論文では、製造ラインにおけるトラッキング、および、製造履歴管理機能を実現するために、製造ラインの構成と挙動をモデル化し、計算機上でこれを表現可能にする時間/場所/物体モデルを提案する。本モデルは、以下のような特徴を持つ。

- 実対象を、追跡の対象となる物体と、物体に対して作用する場所で表現する。
- 物体の移動の追跡を形式化するために、場所に対して5種、物体の形態に対して3種の基本要素を定義する。
- 物体の移動にともなって計算機中で実行すべき処理を、ECAルール<sup>1)</sup>の形式で記述し、移動前/移動時/移動後/不整合時の4つに場合分けをして起動する。
- 製造履歴を、時間、場所、物体の3つの視点で表現する。

さらに、システムを構築するための環境として本モデルに基づくミドルウェアを開発し、ソフトウェア生産性を向上させるためのツールとして具現化した。本ミドルウェアは、開発環境と実行環境から構成される。開発環境は、GUI(Graphical User Interface)により実対象の構成や動作を定義するビルダと、その定義に基づいて対象に特化されたソースコードを自動生成するジェネレータで構成される。実行環境は、物体の実績値などのデータを格納する基盤となるデータベースシステム、物体を追跡して移動にともなうECAルールを発行するトラッキングマネージャ、格納したデータの参照などを行うためのライブラリ関数群から構成される。本ミドルウェアを、実際の製鉄ラインに適用した結果、約10人月相当のソースコードを自動生成できた。また、製作途中の仕様変更へも容易に対応できた。

以下、2章では、製造ライン監視制御システムの構成を概観し、トラッキングや実績管理に要求される機能を整理する。3章では、2章で整理した機能を実現するためのモデルとして、時間/場所/物体モデルを定義する。4章では、時間/場所/物体モデルに基づいた

ミドルウェアの実現と、実プラントへの適用結果について述べる。5章では、関連する研究との比較を行い、6章では結論を述べる。

## 2. 製造ライン監視制御システムの分析

本章ではまず、本論文で対象とする製造ライン監視制御システムの一般的な構成について述べ、トラッキングシステムに要求される機能について整理する。

### 2.1 製造ライン監視制御システムの構成

鉄鋼などの製造ライン監視制御システムの代表的なシステム構成を図1に示す。最上位層には、操業管理や在庫管理、設備管理などを行ういわゆるビジネスシステムが構築され、需要予測や受注状況により立てられた生産計画に基づいて製造指図が発行される。製造指図は、製造ライン内のオペレータステーションやコントローラを包括して管理するプロセス計算機に渡され、プロセス計算機では、コントローラに対する設定値などの制御コマンドの発行や、オペレータステーションの画面表示などが行われる。コントローラでは、プロセス計算機から発行される制御コマンドにより、アクチュエータの操作が行われる。また、コントローラではセンサから製造ラインの状態を獲得し、それがプロセス計算機によって収集される。プロセス計算機では、収集したデータを製造実績として最上位層のビジネスシステムに渡す。

### 2.2 トラッキングと履歴管理

本論文で扱うトラッキング機能と履歴管理機能は、製造ライン中のコントローラを包括した処理が必要となるため、プロセス計算機により実現される。図2に、トラッキング処理の一例を示す。たとえば、実際の製造現場において、操作員がボタンを押すなどの作業によりロット1が工程1に投入されると、工程1に備え

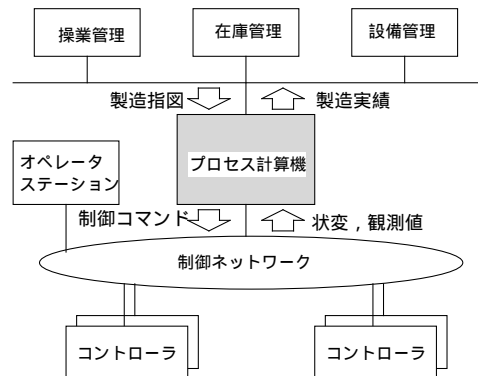


図1 製造ライン監視制御システムの構成

Fig. 1 Organization of a manufacturing line control system.

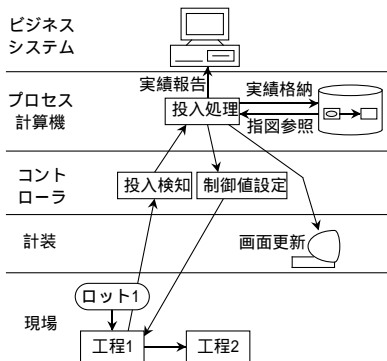


図2 トラッキング処理の例

Fig. 2 An example of a tracking process.

付けられているセンサの測定値が変化し、その変化をコントローラが検知してプロセス計算機における投入処理が起動される。プロセス計算機では、データベース中に保持しているラインの状態に対してロットの投入に矛盾がないかの検査、オペレータステーションなどの計装設備への画面更新、データベースからの指図の参照とそれに基づくコントローラへの制御値の設定、ビジネスシステムへの投入実績の報告、投入量などの実績値のデータベースへの格納、データベース中のライン状態の更新などが行われる。製造ラインの監視制御は、このような処理が各設備や工程間の製造物の移動などのイベントに対して起動されることにより実行されている。

履歴管理機能は、このようなトラッキング処理によって収集された実績データを、ビジネスシステムやオペレータステーションに提供するものである。

### 2.3 我々のアプローチ

上記のようなトラッキング機能や履歴管理機能を実現するために必要なプロセス計算機の機能をまとめると、以下ようになる。

- (1) 物の追跡  
製造対象となる物品が、どの装置（あるいは工程）に存在しているかを追跡する。
- (2) 物への作用の設定  
製造指令によって指定された各物品の各装置での製造条件を、物品の追跡状態に応じて各装置に設定する。
- (3) 物や設備の状態の収集  
設定された製造条件による製造結果として観測された値を実績値として収集し、格納する。
- (4) 製造履歴の提供  
物品の製造履歴を、単に時系列的に収集されたセンサ値の形式で提供するだけでなく、設備や

製品などの単位に集約して提供する。

このような機能を持つシステムの構築環境を提供するためには、製造ラインの構成と物の流れ、プロセス計算機での処理の起動、物の流れに対応した製造履歴を表現可能なモデルが必要である。我々は、製造ラインを直観的に「時間軸上で、製造物が、製造設備（工程）間を移動する」ととらえ、それに基づく時間/場所/物体モデルを定義する。時間/場所/物体モデルは、設備や工程を場所、製造物を物体として表現し、物体の場所間の移動とそれとともになう処理を表現するトラッキングモデルと、時間、場所、物体の各視点から製造履歴を集約して表現する履歴管理モデルにより構成される。また、このモデルに基づく開発ツールを提供することにより、製造ライン監視制御システムの構築コストの削減を図る。

## 3. 時間/場所/物体モデル

### 3.1 基本的定義

#### 3.1.1 時間/場所/物体

先に述べたように、製造ラインは直観的に、「時間軸上で、製造物が、製造設備間を移動する」と表現できる。トラッキングや履歴管理を行うためには、この製造物と製造設備の識別可能な最小単位を定義する必要がある。本モデルでは、製造設備の最小単位を場所 (Place)、製造物の最小単位を物体 (Object) と呼ぶ。

時間 (Time) は、計算機で扱う場合には、用いる計算機のハードウェアや OS の性質に依存して、表現できる最も過去の時刻から最も未来の時刻まで、時計の粒度で等間隔に刻まれた離散的な時刻の並びとなる。時刻間には大小関係が存在し、より過去を表す時刻の方が小さい。

場所は、対象の管理方針に依存して、物理的な各製造設備に対応したり、論理的な製造工程に対応したりする。物体は、一般的に、上位システムから発行される製造指図の単位に対応する。

物体はいくつかの物体クラスに分類される。たとえば、「1次原料」や「最終製品」などの物体クラスが存在する。各物体のインスタンスは、いずれかの物体クラスに属する。

本論文では、 $L$  個の時刻からなる時間、 $M$  個の場所、 $N$  種の物体クラス、および、 $i$  番目のクラスに属する  $N_i$  個の物体を、以下のように表記する。

時間 時刻の列  $t_1, \dots, t_L$

場所 対象内の場所  $p_1, \dots, p_M$

物体 対象内の物体のクラス  $c_1, \dots, c_N$ 、クラス  $c_i$  の物体  $o_1^i, \dots, o_{N_i}^i$

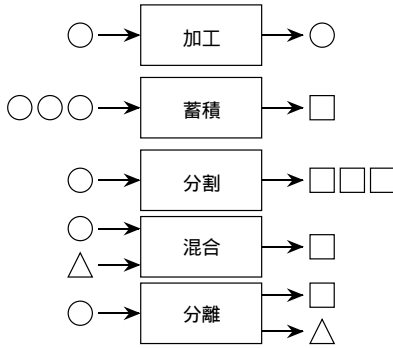


図3 場所の分類  
Fig. 3 Primitives of places.

3.1.2 場所の分類

場所は、それが物体に対して行う作用の違いにより、図3に示すように5つに分類できる。

加工 入る物体と出る物体が同一物体で、その性質を変更するもの。

蓄積 同一クラスに属する複数の物体が入り、1つの物体になるもの。

分割 1つの物体が入り、同一クラスに属する複数の物体になるもの。

混合 それぞれ異なるクラスに属する複数の物体が入り、1つの物体になるもの。

分離 1つの物体が入り、それぞれ異なるクラスに属する複数の物体になるもの。

この分類は、入る物体と出る物体に対して、それが同一であるもの、入るものが複数であるもの、出るのが複数であるもの、複数のものが入り出すときにそれぞれの物体のクラスが同一の場合と異なる場合のすべてを網羅している。

3.1.3 物体の形態

物体の形態は、図4に示すように、以下の3つに分類できる。

点型 物体はある1つの場所にしか存在できない。

線離散型 物体は複数の場所に存在できる。

線連続型 物体は複数の場所に存在でき、かつ、他の物体とつながっている。

この分類は、物体が1つの場所にしか存在できないか、それとも、複数の場所に存在できるか、また、複数の場所に存在できる場合には、他の物品と連続しているかないかのすべての場合を網羅している。

3.2 トラッキングモデル

製造ラインにおける製造物は、原材料としてラインに投入され、いくつかの製造装置を経て、最終製品としてラインから出ていく。物体の挙動は、イベント

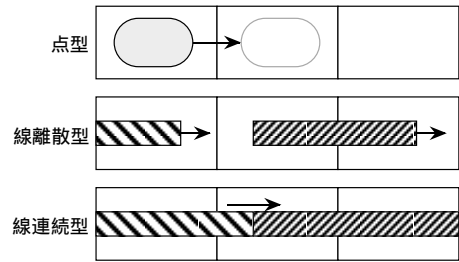


図4 物体の形態  
Fig. 4 Object forms.

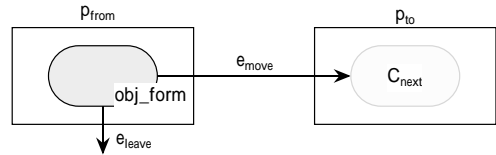


図5 物体の移動  
Fig. 5 Object movement.

(Event)が発生したとき、条件(Condition)を評価して、それが満足されれば動作(Action)を実行するというECAモデルとして表現することができる。ここでは、計算機中で現実のプラントでの物体の移動を追跡し、移動にともなう処理を起動するためのトラッキングモデルを定義する。

3.2.1 物体の移動

物体クラス  $c_j$  に属する物体の挙動は、 $M_j$  個の移動の集合  $\{move_1^{c_j}, \dots, move_{M_j}^{c_j}\}$  で表される。各移動  $move_i^{c_j} (1 \leq i \leq M_j)$  は、組

$$(e_{move}, e_{leave}, p_{from}, p_{to}, obj\_form, C_{next}, pre\_ECA, move\_ECA, post\_ECA, inv\_ECA)$$

で表される。各要素の意味するものは、以下のとおりである(図5参照)。

移動イベント  $e_{move}$  物体が実際の製造ラインで移動したときに処理を起動するトリガである。

出イベント  $e_{leave}$  移動元場所の種別が加工で、かつ、物体の形態が線離散型、あるいは、移動元場所の種別が分割の場合に、物体が移動元場所から排出される処理を起動するトリガである。

移動元場所  $p_{from}$  移動イベント  $e_{move}$  によって移動させる物体が存在している場所である。特別な場合として NULL 値を許し、この場合は、物体の製造ラインへの投入を表す。

移動先場所  $p_{to}$  移動イベント  $e_{move}$  によって物体を移動させる先の場所である。特別な場合として NULL 値を許し、この場合は、物体の製造ラインからの排出を表す。

物体の形態  $obj\_form$  移動元場所の種別が加工の場

```

移動前ECAルール;
if(移動時ECAルール == 成功) {
  移動後ECAルール;
} else {
  不整合ECAルール;
}

```

図 6 移動時の ECA ルール起動論理

Fig. 6 Invocation logic of ECA rules involved with object movement.

合について、移動する物体の形態（点型，線連続型，線離散型）を表す。

移動後物体クラス  $C_{next}$  移動元場所の種別が加工以外の場合について，移動元場所から移動先場所へ移動する物体のクラスを表す。

移動前 ECA ルール集合  $pre\_ECA$  物体を移動させる前に起動する ECA ルールで，物体が移動元場所から出ていく前の処理を記述する。

移動時 ECA ルール集合  $move\_ECA$  物体を移動させる ECA ルールで，実際に物体を移動させるかを判定し，物体を移動させる処理を記述する。

移動後 ECA ルール集合  $post\_ECA$  物体を移動させた後に起動する ECA ルールで，物体が移動先場所に移動した後に実行する処理を記述する。

不整合 ECA ルール集合  $inv\_ECA$  物体の移動時に不整合が発見されたときに起動される ECA ルールで，例外処理を記述する。

物体の形態が線離散型の場合には，物体の尾端が場所から出ていく判定が必要なため，出イベントが必要となる。また，移動元場所の種別が分割の場合には，分割される物体の区切りを管理するために，出イベントが必要となる。

物体の場所間での移動にともなって実行されるべき処理は，移動前，移動時，移動後，不整合の各 ECA ルールで記述される。ある移動イベント発生に対する ECA ルールの起動論理を図 6 に示す。移動前 ECA ルールで記述される処理としては，物体が場所に存在していた間の実績値のデータベースへの格納などがある。移動時 ECA ルールで行われる処理としては，イベントに対応して実際に移動させるべき物体が存在するかどうかのセンサ値による判定などがある。移動後 ECA ルールとしては，移動先場所での物体に対する制御パラメータの設定などがある。不整合 ECA ルールは，移動元場所に移動させるべき物体が存在しない場合などに起動され，現実の製造ラインと計算機中の製造ラインの状態を合致させるための処理などが行われる。

### 3.2.2 物体の移動処理

物体の移動処理は，移動元場所からの物体の出処理，および，移動先場所への物体の入処理で達成される。物体の出処理や入処理は，移動元場所や移動先場所の種類，および，移動する物体の形態によって異なる。

#### 3.2.2.1 物体の出処理

場所  $p_i$  からの物体の出処理は以下のとおりとなる。

- (1) 移動元場所の種別が加工のとき
  - (a) 物体の形態が点型のとき  
場所  $p_i$  が移動元場所となっている移動イベント  $e_{move}$  をもって物体の移動元場所からの出処理を行う。
  - (b) 物体の形態が線離散型のとき  
場所  $p_i$  が移動元場所となっている移動の出イベント  $e_{leave}$  をもって移動元場所からの出処理を行う。
  - (c) 物体の形態が線連続型のとき  
場所  $p_i$  が移動先場所となっている移動イベント  $e_{move}$  をもって物体の出処理を行う。
- (2) 移動元場所の種別が蓄積/混合/分離のとき  
場所  $p_i$  が移動元場所となっている移動イベント  $e_{move}$  をもって移動元場所に存在する物品の出処理を行う。
- (3) 移動元場所の種別が分割のとき  
場所  $p_i$  が移動元場所となっている移動の出イベント  $e_{leave}$  をもって移動元場所に存在する物品の出処理を行う。

#### 3.2.2.2 物品の入処理

場所  $p_i$  が移動先場所となっている移動イベント  $e_{move}$  をもって物体の入処理を行う。

#### 3.2.3 物体移動の特殊処理

現実の製造ラインを直観的に表現するために，場所の種類として，

- 先入れ先出しキュー
- 先入れ後出しキュー
- コンベヤ

を加工の拡張として定義する。キューは，複数の物体が存在できる加工種別の場所であり，移動の際に移動元場所から移動する対象となる物体を，先入れ先出し/先入れ後出しの規則に則って決定する。コンベヤは，複数の加工種別の場所の並びであり，1つのイベントで複数の物体が移動する。コンベヤには，基準位置に対して4つの動きが定義でき，図7に示す例（基準位置は場所2, 3, 4）を用いると，以下ようになる。前へ前進 場所2, 3, 4の物体がそれぞれ場所3, 4,

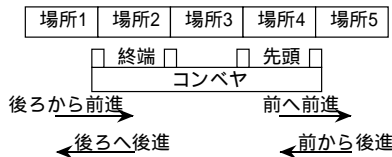


図7 コンベヤの動作

Fig. 7 Actions of a conveyor.

5に移動する。

前から後進 場所3, 4, 5の物体がそれぞれ場所2, 3, 4に移動する。

後ろから前進 場所1, 2, 3の物体がそれぞれ場所2, 3, 4に移動する。

後ろへ後進 場所2, 3, 4の物体がそれぞれ場所1, 2, 3に移動する。

### 3.3 履歴管理モデル

本節では、トラッキングモデルによって追跡された履歴を管理するためのモデルについて述べる。履歴を管理するために、まず、場所や物体の状態を定義し、次に、時間、場所、物体の間で決まる写像を定義する。最後に、これらの定義を用いて、履歴に対する時間、場所、物体のそれぞれの視点を定義する。

#### 3.3.1 場所、物体の状態

一般に場所や物体は、センサなどで測定され、時間によって変化する状態を持つ。場所や物体の状態を表現するために、以下の記法を用いる。

場所の状態 場所  $p_i$  の時刻  $t_l$  に対する状態を  $\sigma_{p_i}(t_l)$  とする。

物体の状態 物体  $o_k^{c_j}$  の時刻  $t_l$  に対する状態を  $\sigma_{o_k^{c_j}}(t_l)$  とする。

#### 3.3.2 写像

時間  $t_l$ 、場所  $p_i$ 、物体  $o_k^{c_j}$  の間に、以下の写像を定義することができる。

場所・物体写像  $Time(p_i, o_k^{c_j})$

場所  $p_i$  に物体  $o_k^{c_j}$  が存在していた時刻の集合を得る。

場所・時間写像  $Object(p_i, t_l)$

場所  $p_i$  に時刻  $t_l$  において存在していた物体の集合を得る。

物体・時間写像  $Place(o_k^{c_j}, t_l)$

物体  $o_k^{c_j}$  が時刻  $t_l$  において存在していた場所の集合を得る。

#### 3.3.3 視点

前項で定義した写像を基に、時間、場所、物体のそれぞれに対する視点を定義する。

時間視点  $V_{time}(t_s, t_e, t_p)$  は、3つ組  $(t_l, p_i, o_k^{c_j})$

の集合である。ここで、 $\forall t_l \in \{t_s, t_s + t_p, t_s + 2t_p, \dots, t_s + (\lfloor \frac{t_e - t_s}{t_p} \rfloor) t_p\}$ ,  $\forall p_i \in \{p_1, \dots, p_M\}$ ,  $\forall o_k^{c_j} \in Object(p_i, t_l)$  である。

場所視点  $V_{place}(p_i, t_s, t_e, t_p)$  は、4つ組

$$(t_l, o_k^{c_j}, \sigma_{p_i}(t_l), \sigma_{o_k^{c_j}}(t_l))$$

の集合である。ここで、 $\forall t_l \in \{t_s, t_s + t_p, t_s + 2t_p, \dots, t_s + (\lfloor \frac{t_e - t_s}{t_p} \rfloor) t_p\}$ ,  $\forall o_k^{c_j} \in Object(p_i, t_l)$  である。

物体視点  $V_{object}(o_k^{c_j}, t_p)$  は、4つ組

$$(t_l, p_i, \sigma_{p_i}(t_l), \sigma_{o_k^{c_j}}(t_l))$$

の集合である。ここで、 $\forall t_l \in Time(p_i, o_k^{c_j})$ ,  $\forall p_i \in \{p_1, \dots, p_M\}$  である。

時間視点は、 $t_s$  から  $t_e$  までの  $t_p$  ごとの時刻  $t_l$  において、各場所  $p_1, \dots, p_M$  にどの物体が存在していたかを得るものである。

場所視点は、 $t_s$  から  $t_e$  までの  $t_p$  ごとの時刻  $t_l$  において、場所  $p_i$  にどの物体が存在し、また、場所や物体の状態はどうであったかを得るものである。

物体視点は、物体  $o_k^{c_j}$  が各場所に存在していた各時刻において、場所や物体の状態はどうであったかを得るものである。

#### 3.3.4 物体の系列

加工以外の場所では、混合や分離などの作用により、場所に入る物体と出る物体は異なるものとなる。物体  $o_k^{c_j}$  が加工以外の場所に入り、物体  $o_{k'}^{c_{j'}}$  として出てきたとき、物体  $o_k^{c_j}$  は物体  $o_{k'}^{c_{j'}}$  を導出したと呼び、 $o_{k'}^{c_{j'}}$  を導出した物体の集合を  $parent(o_k^{c_j})$ 、物体  $o_k^{c_j}$  から導出された物体の集合を  $child(o_k^{c_j})$  で表す。

ある物体  $o_k^{c_j}$  を初期要素として含む集合に対して、集合中の各物体  $o_{k'}^{c_{j'}}$  の  $child(o_{k'}^{c_{j'}})$  および  $parent(o_{k'}^{c_{j'}})$  中に含まれる物体もその集合に含まれるように再帰的に要素を追加していくと、物体  $o_k^{c_j}$  を導出するに至った物体、および、物体  $o_k^{c_j}$  から導出されるに至った物体の集合を得ることができる。このようにして構成された集合を物体  $o_k^{c_j}$  の系列物体集合と呼び、 $lineage(o_k^{c_j})$  で表す。

$lineage(o_k^{c_j})$  中のすべての物体  $o_{k'}^{c_{j'}}$  に対して物体視点  $V_{object}(o_{k'}^{c_{j'}}, t_p)$  を求めることにより、製造ラインにおいて原材料から中間生成物を経て最終製品になるまでの過程を追跡することができる。

#### 3.4 例

本節では、これまで述べた時間/場所/物体モデルによる製造ラインの表現を、図8に示す例に基づいて紹介する。

例にあげる製造ラインは、4つの場所  $p_1, p_2, p_3,$

$p_4$  から構成される．このうち  $p_1, p_2, p_4$  は、場所に入れられた物体と出ていく物体が 1 対 1 に対応する加工工程であるが、 $p_3$  は、 $p_1$  と  $p_2$  からの物体を混ぜ合わせて  $p_4$  に排出する混合工程である．

物体のクラスとしては  $C_1, C_2, C_3$  が存在する．混合工程  $p_3$  では、 $C_1$  の物体と  $C_2$  の物体から  $C_3$  の物体を生成する．なお、各物体とも、トラッキングの形態は点型であるとする．

各物体クラスに対する移動

$$(e_{move}, e_{leave}, p_{from}, p_{to}, obj\_form, C_{next})$$

の集合は、以下のとおりである．なお、ここでは移動時に起動される ECA ルールについては例示しない．

- $C_1$ 
  - $(e_1, NULL, NULL, p_1, \text{点型}, NULL)$
  - $(e_3, NULL, p_1, p_3, \text{点型}, NULL)$
  - $(e_5, NULL, p_3, p_4, \text{点型}, C_3)$
- $C_2$ 
  - $(e_2, NULL, NULL, p_2, \text{点型}, NULL)$
  - $(e_4, NULL, p_2, p_3, \text{点型}, NULL)$

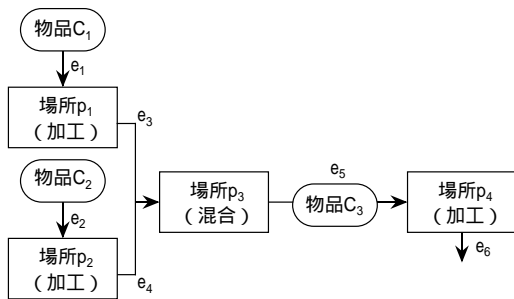


図 8 製造ラインの例

Fig. 8 An example of a manufacturing line.

$$(e_5, NULL, p_3, p_4, \text{点型}, C_3)$$

•  $C_3$

$$(e_6, NULL, p_4, NULL, \text{点型}, NULL)$$

移動イベント  $e_5$  が 2 つの物体クラス  $C_1, C_2$  の両方に現れるのは、それぞれの物体クラスについて、移動を完結させる（出イベントにより場所から出ていく、あるいは、他の物体クラスになって移動する）必要があるからである．

次に、この製造ラインにおいて、以下のような順序で物体の移動が発生したとする．

- (1) 時刻  $t_{l_1}$  に場所  $p_1$  に物体  $o_1^{c_1}$  の入．
- (2) 時刻  $t_{l_2}$  に場所  $p_2$  に物体  $o_1^{c_2}$  の入．
- (3) 時刻  $t_{l_3}$  に場所  $p_2$  から場所  $p_3$  へ物体  $o_1^{c_2}$  が移動．
- (4) 時刻  $t_{l_4}$  に場所  $p_1$  から場所  $p_3$  へ物体  $o_1^{c_1}$  が移動．
- (5) 時刻  $t_{l_5}$  に場所  $p_3$  から場所  $p_4$  へ物体  $o_1^{c_1}$  と物体  $o_1^{c_2}$  が混合され、物体  $o_1^{c_3}$  となって移動．

このような物体の移動の履歴に対する写像の例を以下に示す．

場所・物体写像  $Time(p_2, o_1^{c_2}) \rightarrow \{t_l | t_{l_2} \leq t_l < t_{l_3}\}$

場所・時間写像  $Object(p_3, t_{l_4}) \rightarrow \{o_1^{c_1}, o_1^{c_2}\}$

物体・時間写像  $Place(o_1^{c_2}, t_{l_2}) \rightarrow \{p_2\}$

また、時間、場所、物体の各視点を直観的に理解するために、その概念を 3 次元的に表現したものが図 9 である．時間視点は、図のように時刻を固定して切った平面により得られる場所と物体の対応関係を、固定する時刻を変化させて時系列的に並べたものである．場所視点は、ある場所（図の場合は  $p_2$ ）を固定して

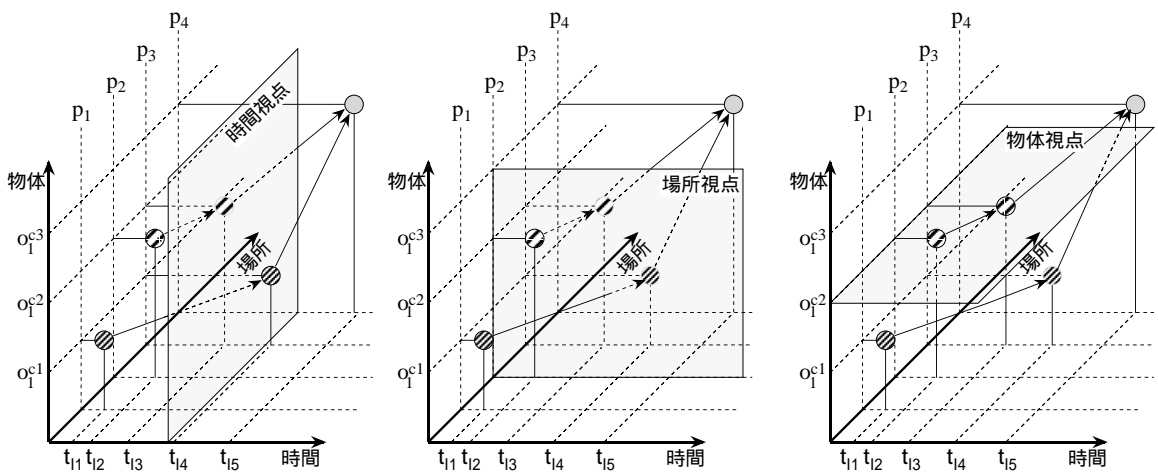


図 9 視点の 3 次元的表现

Fig. 9 Three-dimensional representations of each view.

切った平面により得られる時間と物体の対応関係を、視点の各組に対応する各時刻について時系列的に表したものである。物体視点は、ある物体（図の場合は  $o_1^{c2}$ ）を固定して切った平面により得られる時間と場所の対応関係を、視点の各組に対応する各時刻について時系列的に表したものである。

$lineage(o_1^{c3})$  は、 $\{o_1^{c1}, o_1^{c2}, o_1^{c3}\}$  となる。このそれぞれの物体に対して物体視点を得ることにより、原材料  $o_1^{c1}$ 、 $o_1^{c2}$  から最終製品  $o_1^{c3}$  までの製造履歴を追跡することができる。

#### 4. ミドルウェアとしての実現と適用

前章で述べた時間/場所/物体モデルに基づき、トラッキングおよび実績管理機能構築環境をミドルウェアとして実現した。本ミドルウェアは、図 10 に示すような以下の要素により構成される。

##### (1) 開発環境

###### ● ビルダ

GUI 画面により、制御ネットワークのアドレス割り付けや取り込みの設定、イベント検出条件と報告形式の設定、製造ラインの構成、物体の移動条件などを設定する。

###### ● ジェネレータ

ビルダにより定義された情報に基づき、対象システムごとに特化されたソースコードを自動生成する。

##### (2) 実行環境

###### ● 能動型実時間時系列データベース RTDS (Real-Time Data Server)<sup>(0)</sup>

制御ネットワークからのデータ獲得と格納、イベントの検出、上位系へのデータの提供を行う。

###### ● 時間的オブジェクト指向データベース RTVS (Real-Time View Server)<sup>(1)</sup>

データの永続化、RTDS で提供される時間的データのオブジェクト化、トランザクション管理などを行う。

###### ● トラッキングマネージャ

開発環境のジェネレータにより生成されたプラント依存ソースコードを用いて、物体の追跡、ECA ルールの起動を行う。

###### ● アプリケーション開発ライブラリ

トラッキングマネージャから起動される ECA ルールや、操作画面制御、上位計算機との接続などのアプリケーションを構築するための機能を、プラント依存ソースコー

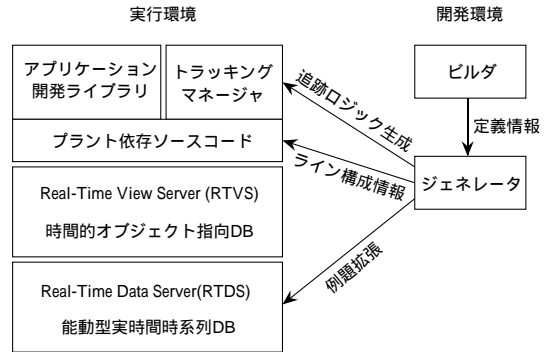


図 10 ミドルウェアの構成

Fig. 10 Organization of the middleware.

ドを用いて提供するライブラリ群である。次節より、これらの内容について述べる。

##### 4.1 開発環境

開発環境は、Microsoft 社の Windows95/NT 上で動作し、開発者が GUI 画面で対象システムの構成を入力できるビルダと、ビルダで入力された定義を基にして、実行環境で実行されるソースコードを出力するジェネレータから構成されている。

実行環境のうち、対象システムごとにより処理が異なるものは、データの獲得/格納/提供やイベント検出を行う RTDS、および、物体の追跡や ECA ルールの呼び出しを行うトラッキングマネージャである。ビルダやジェネレータは、その定義内容の性質の違いや、他分野への応用を考慮して、RTDS とトラッキングマネージャとで独立して構築した。

開発者が対象システムに応じて RTDS に対するビルダで入力すべき情報は、以下のとおりである。

**データの獲得定義** コントローラが接続されている制御ネットワークから獲得すべきデータのアドレス割り付けと名前付けを定義する。制御ネットワークからのデータは、分散共有メモリの周期読み出しと、メッセージ型の非周期読み出しがある。

**イベント検出条件の定義** 獲得したデータに対してトラッキング処理などを起動するためのイベントを検知し、報告するための条件を定義する。条件としては、ビット信号の立ち上がりや立ち下がりを含む線形一次不等式が定義できる。

**提供データ構造の定義** 獲得データの周期的な転送や要求に応じた時系列データの転送、および、イベント検出時に報告されるデータの構造を定義する。

一方、トラッキングマネージャに対するビルダで入力すべき情報は、以下のとおりである。

**対象システム構成の定義** 対象システムに存在する場



所を定義する．場所は階層化して定義することができ、製造ラインをいくつかに分割して管理できる．

対象システムの物体クラスの定義 対象システムで扱う物体の定義を行う．これには、物体に対する作用を制御するためのデータや、物体に関して収集する実績データの構造の定義も含まれる．

対象システムの物体挙動の定義 物体がどのようなイベントをトリガとして、どのような場所間を移動するかを定義する．また、物体の移動に対応して起動されるべき ECA ルールを定義する．

ジェネレータではこれらの情報に基づいて対象システムにカスタマイズされたソースコードを生成する．RTDS とトラッキングマネージャでは、その動作の性質の違いにより、ジェネレータの方式も異なっている．

RTDS のジェネレータでは、最小で、かつすべての機能を包含した例題システムのプログラムを、ビルダの定義に基づいて膨らませる例題拡張<sup>13)</sup>と呼んでいる方式を新たに開発した．データの獲得処理、イベント検出処理、提供データの合成処理などが、ビルダの定義に基づき、ジェネレータにより例題プログラムから拡張される．

一方、トラッキングマネージャでは、トラッキング処理を対象システムに依存しない部分と依存する部分を整理して分離し、固定的なソースコードとなる対象システムに依存しない部分から、ジェネレータで生成される対象システムに依存する部分を呼び出す方式をとった．イベントによる物体の移動処理や ECA ルールの起動、対象システムの場所構成、物体に関するデータ構造に応じたデータの格納や参照などを行う関数がジェネレータにより生成される．

#### 4.2 実行環境

実行環境は当社製産業用計算機 MELCOM MR シリーズ上のリアルタイム UNIX で動作する．本モデルウェアは、我々がすでに開発を行った 2 つのデータベースサーバ RTDS と RTVS を基盤としている．これらの詳細については各文献に譲る．

本モデルにおける時刻は、UNIX における時刻印型である timeval により表現され、場所および物体は、RTVS 上で永続化されたインスタンスとして実現されている．場所および物体のデータ構造を図 11 に示す．場所に対応するインスタンスには、現存する物体へのリンクを格納するための属性が存在している．蓄積場所のように、1 つの場所に複数の物体が存在できる場合には、複数の物体がリンクされる．また、物体に対応するインスタンスには、3.3.4 項で述べた物体の導

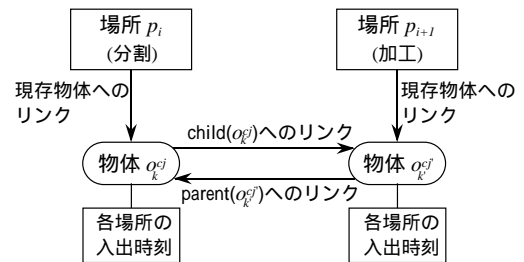


図 11 時間/場所/物体モデルを実現するデータ構造

Fig. 11 A data structure realizing the Time/Place/Object model.

出関係を保持するためのリンク属性が存在している．また、物体には、ビルダにより定義された情報に基づき、通りうる各場所に対して、その場所に入った時刻と出た時刻を保持する属性が存在している．この時刻により物体と場所、時間が関連付けられ、写像や視点の導出が可能となる．

トラッキングマネージャは、RTVS からイベントを受信し、イベントの種別に応じて物体の移動処理や ECA ルールを起動する．トラッキングマネージャは、場所に対応するインスタンスと、物体に対応するインスタンスをリンクで関連付け、物体の追跡を行う．また、加工以外の場所で物体に対して作用が行われたときには、作用前の物体のインスタンスと作用後の物体のインスタンスの関連もリンク付け、系列物体集合の取得を可能にする．

アプリケーション開発ライブラリは、アプリケーションが、トラッキングマネージャの追跡状態の取得や変更、実績データの格納、時間、場所、物体の各視点の取得を行うためのインタフェースを、約 100 個のライブラリ関数として提供する．

#### 4.3 実システムへの適用

本モデルウェアは、製鉄におけるプロセスラインのトラッキングシステムに適用された．本システムは、上位計算機からの製造指図データの管理、オペレータステーションへの画面表示データ提供、コントローラへの設定値発行、実績収集、上位計算機への実績報告など、鉄鋼プラントの管理として中核となる機能を実現している．

製鉄におけるプロセスラインは、1) コイル状に巻き取られた鉄板をコンベヤにより搬送し、払い出しリールに装着するまでの入側セクション、2) 払い出しリールから鉄板を巻き戻し、先行する鉄板と溶接して連続的な化学的処理を行った後、製品単位で切断し、再びコイル状に巻き取る中央セクション、3) 巻き取られた鉄板をコンベヤにより倉庫などに搬送する出側セク

表1 実システムへの適用例  
Table 1 An application result of the middleware.

場所数	物体 クラス数	ECA ルール数	生成ソース コード行数
68	4	563	31352

ションの3つのセクションに分けられる。本モデルによる表現を適用すると、入側セクションから中央セクションに物体が移動する段階で、1枚の鉄板に製造仕様変更点が発生するため、払い出しリールが分割種別として定義される。また、中央セクションでの化学的処理の後、製品単位で切断するシャートと呼ばれる設備は分割種別として定義され、複数の製造仕様変更点を含んで巻き取る可能性のある巻き取りリールは、蓄積種別として定義される。セクション間での物流速度差を吸収するためのループと呼ばれる設備は3.2.3項で述べた先入れ先出しキューとして実現されている。その他、化学的処理をするための槽などは加工種別の場所である。入側セクションおよび出側セクションでは、物体はコイル状の鉄板であるので、その移動は点型として定義される。一方、中央セクションでは、物体は長い板状に巻き戻され、かつ、先行する物体と後行する物体と溶接されているので、線連続型として定義される。

ECAルールでは、下位コントローラへの設定値の送信、下位コントローラからの実績値の格納、オペレータステーション画面の更新、上位ビジネスシステムとのデータの送受信、コンベヤ上の在荷信号による現実のプラントと計算機中の状態との整合性の検査と異常処理などが行われている。

本適用におけるシステムの規模を表1に示す。本適用に対するビルダへの入力に要した時間は2日程度であった。ジェネレータによるソースコード生成に要する時間は、数秒程度である。一方、生成されたソースコード量は約31k行(コメント行も含む)であったが、一般的にいわれている1人月で達成できる作業量である3k行で換算すると、約10人月に相当する。また、ビルダとジェネレータを提供することにより、製作の段階で生じた仕様変更や設計の誤りに対しても迅速に対応することができた。

本件は本モデルウェアの第1次適用であり、システムエンジニアのノウハウの構築も必要だったが、第2次適用以降は、本件のノウハウを用いて、さらに開発効率の向上が期待できるものと考えている。また、システムの実現段階で、並行して動作する複数ECAルール間でデータの施錠の競合が生じてデッドロックが発生したため、施錠順序の詳細な調査が必要になっ

た。このようなECAルール間での整合性の検査機構については、どのような整合性が必要とされ、ツール化が可能かの検討も含めて、今後の課題である。

## 5. 関連研究

本論文で述べた内容の成果は、製造ラインの構成や挙動のモデル化と、製造過程の制御や製造結果の集約を行うシステムの構築ツールを提供したことである。

我々はすでに文献12)において、バッチ型プロセスにおける製造情報履歴管理の高機能化のために、製造履歴を時間に着目したプラントビュー、製造工程に着目したユニットビュー、製造物に着目したバッチビューで多次的に提供するためのモデルとその実現機構について述べた。本論文では、文献12)で述べたモデルに対して、以下のような一般化と拡張を行っている。

- トラッキング機能に必要な物体の追跡を可能とするため、文献12)で定義した場所に対する5種の基本要素だけでなく、物体の形態に対しても3種の基本要素を定義した。これにより、製鉄のように物体が長さを持ち、複数の工程にまたがって存在するようなラインに対しても適用を可能とした。
- 物体の移動にともなって実行されるべき処理をECAルール<sup>1)</sup>の形式で記述し、移動前/移動時/移動後/不整合時の4つに場合分けをして起動する。これにより、物体の移動に対して製造装置に設定値を出力するなどの制御機能を取り込むことができた。
- 製造履歴管理に対して、時間、場所、物体の関係を写像を用いて表現することにより、文献12)で示した多次元ビューを一般化した。

米国計測制御学会(ISA)では、バッチ型プロセスを監視制御するためのモデルと用語を標準として定義し<sup>2)</sup>、すでに本標準に基づく市販のパッケージツールがいくつか販売されている。本標準では、プラント内の設備や、製造指図であるレシピの管理、各製造工程における処理フェーズのモデル化に重点が置かれており、本論文で述べたようなトラッキングや実績管理のためのモデルについては議論されていない。

物の流れを扱うものとしては、ワークフロー管理がある。ワークフロー管理では、人間が行う事務作業の支援を目的としているために、手順の変更や作業員間の協調作業などに柔軟に対応するためのモデルが提案されている<sup>3),5)</sup>。これに対して、本論文で述べたモデルでは、製造ラインの定型的な処理を対象とすることで、各処理工程(場所)や物体の形態を形式化し、ECAルールによってトラッキングすることを可能に

している。また、ワークフローの履歴管理手法についての提案<sup>4)</sup>もあるが、本論文で述べたような時間、場所、物体の各視点から履歴を集約して管理しているものは見当たらない。

製造データを扱うデータモデルとしては、SAM\* (Semantic Association Model) を適用したもの<sup>6)</sup>や CIM における生産管理とデータ管理を統合するためのモデル<sup>7)</sup>などが提案されている。しかし、これらは部品構成の表現や検査工程でのワークフローなどが対象となっているため、本論文で対象としたトラッキングのような物体の移動にともなうルールの起動などは扱えない。また、製造業における業務間の情報のやりとりをペトリネットを用いたルールベースで実現する手法<sup>8)</sup>が提案されているが、トラッキングのような製造工程間でのデータや処理のやりとりには、本論文で述べたような ECA ルールの 4 種の起動論理のような、現実のプラントにおける物体の移動と強い関連を持った単位での処理の起動が必要となる。

製造ラインの多様性や設計変更に対応するため、実時間データベースにおけるスキーマ進化を扱ったもの<sup>9)</sup>がある。これにより、システムの機能的な変更が行われたときの変更が容易になっているが、我々が開発環境として提供したようなツールを提供するには至っていない。

## 6. あとがき

本論文では、製造ライン監視制御システムにおけるトラッキングおよび実績管理機能を実現するための時間/場所/物体モデルと、それに基づくミドルウェアの構築と適用例について述べた。本ミドルウェアの適用により、システム構築コストの削減、構築途中の仕様変更への柔軟な対応が達成できた。

今後は、上流設計であるシステム設計や、設備保全などの作業フェーズに適用できるように、開発環境の拡張を行いたいと考えている。

謝辞 本研究に対して有益なご議論をいただいた京都大学情報学研究所上林教授に感謝いたします。

## 参考文献

- 1) McCarthy, D.R. and Dayal, U.: The Architecture of an Active Data Base Management System, *Proc. International Conference on Management of Data*, pp.215-224 (June 1989).
- 2) ISA S88.01 Batch Control Standard Part 1: Models and Terminology, Instrument Society of America (1995).
- 3) Lomet, D., et al.: Special Issue on Work-

flow Systems, *IEEE Data Engineering Bulletin*, Vol.18, No.1 (1995).

- 4) Koksai, P., Arpinar, S.N. and Dogac, A.: Workflow History Management, *SIGMOD Record*, Vol.27, No.1, pp.67-75 (1998).
- 5) Kunishima, T. and Yokota, K.: Flexible Workflow Frameworks for Supporting Collaborative Work, *Proc. International Symposium on Cooperative Database Systems for Advanced Applications (CODAS '97)*, Kyoto, pp.501-508 (Dec. 1996).
- 6) Su, S.Y.W.: Modeling Integrated Manufacturing Data with SAM\*, *IEEE Computer*, Vol.19, No.1, pp.34-49 (1986).
- 7) Adam, N.R. and Gangopadhyay, A.: Integrating Functional and Data Modeling in a Computer Integrated Manufacturing System, *Proc. International Conference on Data Engineering '93*, pp.302-309 (1993).
- 8) Harhalakis, G., Lin, C.P., Mark, L. and Muro-Medrano, P.: Implementation of Rule-Based Information Systems for Integrated Manufacturing, *IEEE Trans. Knowledge and Data Engineering*, Vol.6, No.6, pp.892-908 (1994).
- 9) Zhou, L., Rundensteiner, E.A. and Shin, K.G.: Schema Evolution of an Object-Oriented Real-Time Database System for Manufacturing Automation, *IEEE Trans. Knowledge and Data Engineering*, Vol.9, No.6, pp.956-977 (1997).
- 10) 島川, 水沼, 竹垣: プラント履歴データの実時間獲得・提供システム; 電子情報通信学会論文誌 (D-I), Vol.J78-D-I, No.8, pp.798-806 (1995).
- 11) 高田, 島川, 浅野, 井戸, 竹垣: 時間的正当性を考慮したオブジェクトモデルに基づくプラント監視向けデータベースシステムの構築, 電子情報通信学会 (D-I), Vol.J79-D-I, No.10, pp.853-862 (1996).
- 12) 高田, 島川, 浅野, 竹垣: 能動機構と多次元ビュー生成に基づくバッチ製造情報管理システムの実現, システム制御情報学会論文誌, Vol.10, No.11, pp.575-584 (1997).
- 13) 島川, 井戸, 高田, 堀池: 例題拡張による実時間制御システムの開発, 情報処理学会論文誌, Vol.40, No.5, pp.2468-2477 (1999).

(平成 11 年 7 月 12 日受付)

(平成 12 年 3 月 2 日採録)



高田 秀志(正会員)

1968年生．1993年京都大学大学院工学研究科情報工学専攻修士課程修了．同年三菱電機(株)入社．プラント監視制御システムにおけるデータベース環境，エンジニアリング環境の研究開発に従事．1998年システム制御情報学会論文賞受賞．1993年第46回情報処理学会全国大会奨励賞受賞．システム制御情報学会会員．

環境の研究開発に従事．1998年システム制御情報学会論文賞受賞．1993年第46回情報処理学会全国大会奨励賞受賞．システム制御情報学会会員．



井戸 譲治(正会員)

1969年生．1994年大阪大学大学院工学研究科通信工学専攻修了．同年三菱電機(株)社．実時間システム，データベースの研究開発に従事．



島川 博光(正会員)

1961年生．1986年京都大学大学院工学研究科情報工学専攻修士課程修了．同年三菱電機(株)入社．平成5年マサチューセッツ大学客員研究員．1999年京都大学大学院工学研究科情報工学専攻博士後期課程修了．工学博士．データベース，実時間システムに関する研究に従事．IEEE，ACM，システム制御情報学会，日本ソフトウェア科学会各会員．

科情報工学専攻博士後期課程修了．工学博士．データベース，実時間システムに関する研究に従事．IEEE，ACM，システム制御情報学会，日本ソフトウェア科学会各会員．



堀池 聡

1957年生．1982年京都大学大学院工学研究科数理工学専攻修了．同年三菱電機(株)入社．1986年マサチューセッツ工科大学客員研究員．工学博士．分散実時間システムの研究に従事．IEEE，電子情報通信学会各会員．

究に従事．IEEE，電子情報通信学会各会員．