

ソフトウェア生産を統合的に支援する
列車運行監視システムビルダの実現

3T-6

中田 秀男*

平 克也**

中山 英二郎***

杉本 明*

三菱電機株式会社

* 産業システム研究所

** 情報システム技術センター

*** 伊丹製作所

1 はじめに

システムの計画から構築・実行まで統一した環境で行なう列車運行監視システムビルダをオブジェクト指向方式を用いて実現する。列車運行監視システムビルダを、監視対象設備を表すシンボルの表示、設備属性の管理機能を持ったオブジェクトで構成する。

計画段階のラフスケッチ画面への適用を考えた場合、設計・製作段階に適用した場合は、通常、上記のオブジェクトの取り扱いが異なる。そこで、オブジェクト間に接続機能を持たせ、計画、設計、製作まで統一に取り扱えるようにする。従来、監視単位毎に分かれていた線路間の接続を定義し、実世界通りに1本の線路として取り扱うことにより、ラフスケッチ画面の描画から適用でき、最終的な監視画面の構築まで一貫して行うことができる。

さらに、シンボルの色換え / 数値表示などのオンラインの動作機能をオブジェクトに持たせることにより、エディタ画面が即、監視画面となる。設計、製作、試験、運用を共通の環境で行なうことができる。

2 列車運行監視システム

列車運行監視システムは、列車運行管理における重要なシステムの一つである。列車運行監視システムは、中央指令所にて把握する全線における全ての列車の在線情報を、指令員にCRT画面上で明確に示す。

図1に監視画面例を示す。線路の色換えによる列車の在線表示、信号機の色表示、列車番号の表示、列車の遅延時間表示などを監視画面上で行なう。

従来、計画、設計、製作、試験、運用のソフトウェア生産過程の各段階に、列車運行監視システムの構築を支援する数々のツールが存在していた。ツール間のつながりがなく、ユーザインターフェース、データなどに統一性がなかった。

An Train Supervisory System Builder
Hideo NAKATA, Katsuya HIRA, Eijiro NAKAYAMA,
Akira SUGIMOTO
Mitsubishi Electric Corporation
8-1-1 Tsukaguchi-Honmachi, Amagasaki, Hyogo, 661 JAPAN

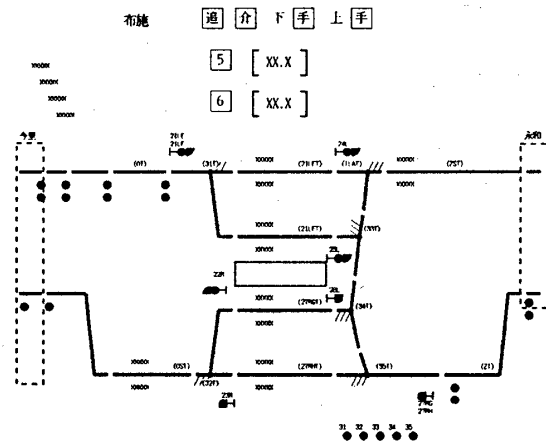


図 1: 監視画面例

本稿における列車運行監視システムビルダでは、データの一貫性を保ち、効率良く列車運行監視システムを構築するために、監視画面の設計・製作を行なうだけでなく、計画段階、試験・運用段階への適用を考える。

計画段階への適用

1. 効率的で容易な作画・編集

計画段階で使用する監視画面のラフスケッチをシステムビルダで作成する。監視画面を容易に効率的に作成するために、監視単位ごとに分かれている線路を、1本の線路として取り扱う。

2. シミュレーションによる実際の動きを実現

計画段階での仕様決定などでは、実際にどのように動作するかを理解を助けるために、シミュレーションを行ない、画面上に表示する方法が有効である。

試験・運用への適用

1. シミュレーションによる検査

試験の初期段階として、シミュレーションを行ない、画面上に表示する方法が有効である。

2. 実際のデータを取り込み表示

試験・運用に適用するためには、ビルダ自身が列車運行監視システムとしての機能を持つ必要がある。

3 システムビルダの実現

3.1 オブジェクト構成

図2に示すように、列車運行監視システムビルダを、監視対象設備を表すシンボルの表示機能を持つビュー、設備属性の管理機能を持つモデルの2種類のオブジェクトで構成する[1]。ビューは、監視画面上の設備シンボルの編集操作機能、および、位置、大きさなどの設備シンボルの属性を持つ。モデルは物理的な設備に対応し、運行監視に必要な設備固有の属性を持つ。

列車運行監視システムでは、監視画面上の線路は、監視単位ごとに分かれて表示される(図1)。在線表示などの監視機能は、監視単位ごとに実行されるため、監視に必要な設備データも監視単位ごとに存在する。従って、監視単位ごとにビュー、モデルを構成する。

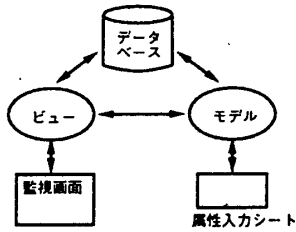


図2: オブジェクト構成図

3.2 オブジェクト接続機能

ラフスケッチの段階で、細かく分割された線路を描くことは少ない。線路を引き、後で線路のカットを行なうユーザインタフェースを採る。ラフスケッチでは概略の分割を行ない、詳細設計時に細かい分割を行なうことにより、実際の監視画面を作成する方式が採れる。また、分岐点は線路と線路を接近させて配置することにより、自動的に設定する。

監視単位ごとに分かれているオブジェクトに接続機能を持たせて、下記の3点の機能拡張を行ない、上記のユーザインタフェースを実現する。

[オブジェクト再編成機能] 線路のカット・カットの取消時に、オブジェクトの分割・統合を行なう再編成機能を持つ(図3-(1))。

[自動接続] 線路同士を近接させて配置する時、線路を自動的に接続し、分岐点を自動的に設定する(図3-(2))。分岐点設定を容易にし、画面上での接続の曖昧さを排除し、ユーザのミスを排除する。

[メッセージ伝播機構] 線路を表す複数のオブジェクトが、1本の線路を表す1つのオブジェクトのように振舞う(図3-(3))。ユーザからは、1本の線路を作画・編集しているように見える。

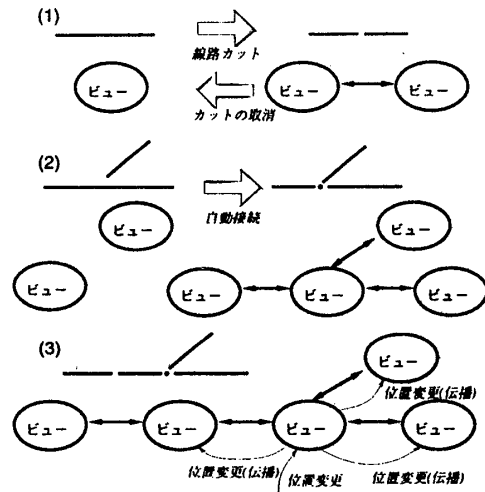


図3: オブジェクト接続機能

3.3 オンライン表示機能

仕様決定、試験のためにシミュレーション結果を画面上に表示する方法が有効である。また、試験・運用のために、実機からの情報を画面上に表示する必要がある。本稿のシステムビルダでは、オンライン表示機能を下記のように実現する(図4)。

1) モデルは、シミュレータや実機との通信機能を持つ。シミュレータや実機から運行監視に必要なデータを受信し、解析を行ない、属性データを変更する。

2) ビューとモデル間も通信機能を持つ。モデルは、自分の持つ属性データの変更をビューに伝える。

3) ビューはモデルからの通信に従い、画面上のシンボルの色換え、数値表示を行なう。

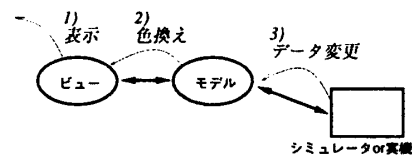


図4: オンライン表示

4 おわりに

現在、実システムに適用し評価中である。

連動制御に必要なデータの抽出機能の追加により、電子連動装置のソフトウェア設計に対応予定である。

参考文献

[1] 中田, 北村, 小島, 杉本: “オブジェクト指向ライブラリ GhostHouse による UI システム構築”, 三菱電機技報 (1993.9)