

# オブジェクトモデルを適用した リアルタイム・シミュレーション



総合エンジニアリングシステム事業部  
鶴見システム開発部  
チームリーダー

井上 雅久

**Masahisa Inoue**



総合エンジニアリングシステム事業部  
鶴見システム開発部  
ITスペシャリスト

岡本 英子

**Eiko Okamoto**



株式会社 ブリッジ

阿部 智紀

**Tomonori Abe**

ガスパイプラインにおけるオンライン・リアルタイムシステムは、他分野のソフトウェアに比べていくつかの特徴があり、その要求に応えるためには拡張や変更が強固、開発リスクを軽減できるオブジェクト指向技術が必要であった。オブジェクト指向技術を採用するにあたり、まずオンライン・リアルタイムシステムの要件を分析し、各サブシステムで共通なパイプラインモデル情報をいくつかのデザインパターンを用いてモデリングを行った。さらに、適用先に依存しない汎用的な機能を実装していった。このオブジェクトの集まりを「オブジェクトモデル」と称する。

本稿では、オブジェクト指向を用いてオンライン・リアルタイム・シミュレーションを開発するためのポイントを述べる。

## 1. はじめに

JFE日本鋼管株式会社殿は、パイプ製造からパイプラインの設計、施工、検査・診断、補修まで、パイプラインすべてにわたる事業を国際的に展開している。

本稿で述べるガスパイプライン運転支援システムは、パイプラインの計画・設計ツールとして、JFE日本鋼管株式会社が中核となる解析シミュレータの開発および商品企画・販売を担当し、エクサがアプリケーションシステムとして構築してきたものであり、改良を重ねながら活用されてきている。

このシステムは、非定常解析（刻々と変化するガスパイプライン各所の圧力・流量を高速・高精度に解析）計算することを目的としており、1996年に実パイプラインへの適用が開始された。また、JFE日本鋼管株式会社社内の設計用途だけではなく、都市ガス会社やガスパイプライン会社向けに、供給設備や運転計画支援ツールとして販売されている。

ガスパイプライン運転支援システムは、扱う対象の性格から下記が要求される。

- ・ 時間的制約から、処理時間の限度が厳しい
- ・ 監視制御設備から圧力や流量などの計測値を取り込む必要がある
- ・ 外部環境から発生する多くの事象（万一の他工事損傷や設備異常による漏洩検知など）に対応するために、複数の処理を並行して行わなければならない

システムを構築するうえで、これらの要求に応えるためには、拡張や変更に強く、開発リスクを軽減できるようなソフトウェア開発手法の採用が必要であった。本システムでは、その手法としてオブジェクト指向アプローチを採用した。

オブジェクト指向技術は、ソフトウェアの柔軟性、再利用性などを高める手法として、現時点では最も進んだアプローチの一つといえる。しかし、オンライン・リアルタイムシステムに対する実績は、実行効率やリソース制限などの問題から、一部の大規模なシステムを除いて、その適用があまり進んではいない。現在ではCPUの高速化、コンパイラやオブジェクト指向言語の急激な進歩、メモリなどリソースの低価格化などにより、それらの制限は徐々になくなりつつあるが、まだまだオンライン・リアルタイムシステムへの適用事例は少なく、具体的なツールなども十分に提供されているとはいえない状況である。

本稿では、オブジェクト指向を用いてオンライン・リアルタイム・シミュレーションを開発するためのポイントを述べる。

## 2. ガスパイプライン運転支援システム

ガスパイプライン運転支援システムは、5つのシステムから構成される（図1参照）。

### ①PLM（パイプライン・モデラ）

解析対象となるガス導管網モデルをGUIベースで作成することを目的とし、Microsoft Visioをベースにカスタマイズされたアプリケーションである。

### ②ONS（オンライン設定）

RTSで必要となる計測値情報の登録、更新、削除、および計測値情報とガス導管網モデルの関連付けなど、シミュレーションに必要な属性情報をGUIベースで作成することを目的とする。

### ③RTS（実時間シミュレータ）

PLMおよびONSアプリケーションで作成された導管網モデル情報を対象に、適用先（都市ガス会社やガスパイプライン会社）が持つSCADA(Supervisory Control And Data Acquisition)と呼ばれる計測データ収集システムからの計測値送信周期に同期して、最新の計測値をオンライン・リアルタイムで取得し、SCADAシステムの計測周期ごとに、ガス導管網全体の圧力・流量などの最新の流送状態をシミュレーションする。これにより、FCSの初期状態、導管漏洩検知のための基準値を作成することを目的とする。

### ④FCS（近未来予測シミュレータ）

PLMおよびONSアプリケーションで作成された導管網モデル情報を対象に、RTSより最新のシミュレーション結果を初期状態として取得し、Excelシート上で編集された需要・供給シナリオに基づき、最高で2日先までのガス導管網の流送状態をシミュレーションする。これにより、これから行おうとする供給計画の妥当性を検討することを目的とする。

### ⑤PLS（計画支援シミュレータ）

オフラインで将来の需給計画や導管網計画検討、計画工事時や緊急時運用の事前検討、オペレーション訓練機能を行うことを目的とする。

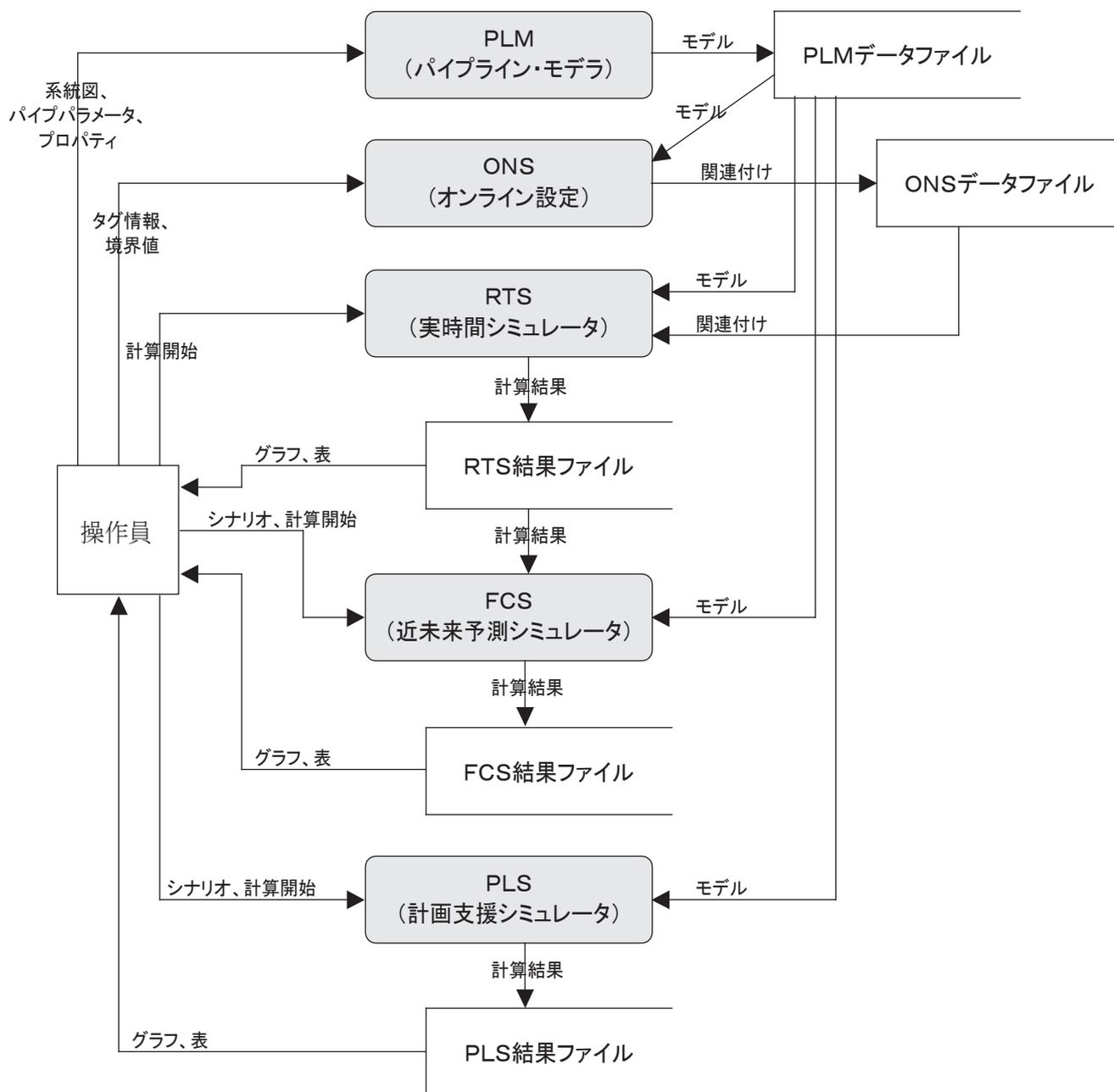


図1 システム関連図

### 3. オブジェクトモデル

パイプラインモデルの静的な情報（パイプ区間長、口径、圧力損失の程度をあらわす係数など）をオブジェクト化したものをオブジェクトモデル（図2参照）と呼んでいる。

各シミュレーションでのデータの統合、およびそのデータを操作するための共通な機能を持つライブラリで、すべ

てのアプリケーションからオブジェクトを参照可能にするため、ActiveX DLLとして作成している。また、ライブラリに含まれる1つ1つのオブジェクトはCOM(Component Object Model)として作成している。

オブジェクト指向技術を採用するにあたり、まずはオン

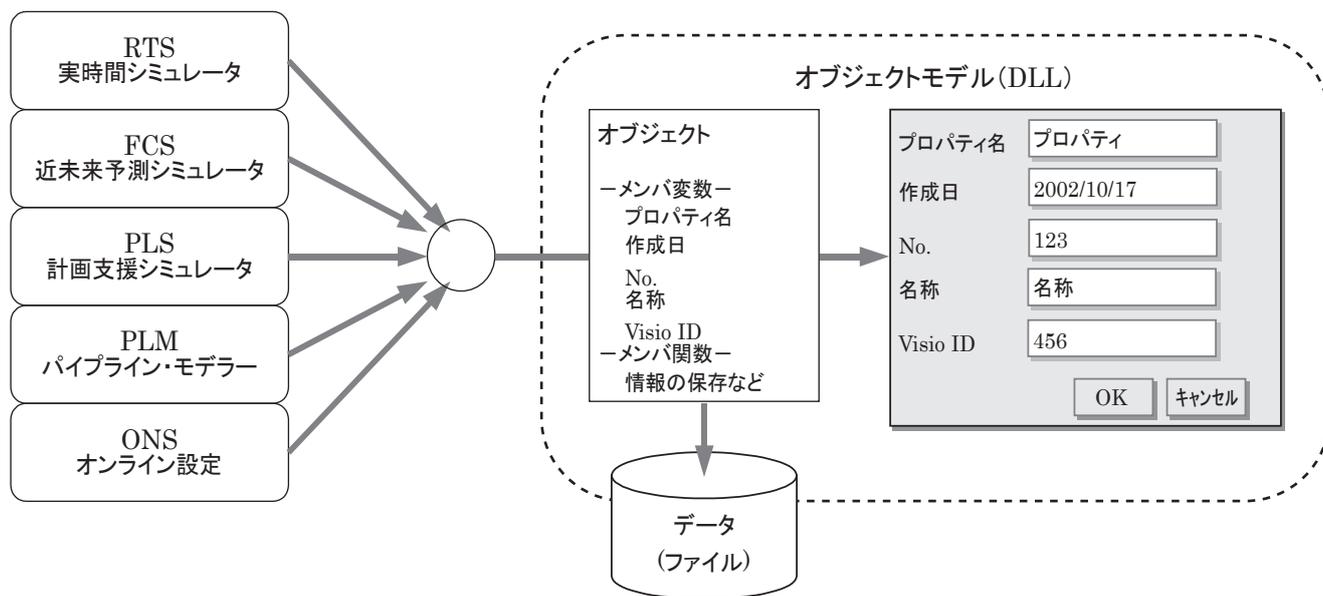


図2 オブジェクトモデル概念図

ライン・リアルタイム・シミュレーションの要件を分析し、各サブシステムで共通なパイプラインモデル情報を、いくつかのデザインパターンを用いてモデリングを行った。さらに、適用先に依存しない汎用的な機能を実装していった。

オブジェクトモデルが保持する情報は、次の4つに分類される。

i) 導管網情報

パイプラインモデルを構成するパイプ、バルブ、ホルダ、ジャンクション（節点）、放散塔といったコンポーネントと呼ばれる要素と、それぞれのコンポーネントが保持する情報（パイプ区間長、口径、標高など）、および各コンポーネントのつながり（入口、出口）の情報。

ii) 計測値情報

各計測地点から送られてくる計測値であり、RTSでは実際の計測値が計算条件（シナリオ）に用いられ、FCSでは近未来の予測を行うための架空の計算条件（シナリオ）に相当する。

iii) 計算情報

非定常計算を行う際の計算条件（計算時間などシミュレータ制御条件）である。

iv) 計算結果情報

非定常計算により算出された結果値で、この結果値を使用して計測点以外の場所のガス流れ状態を把握することができる。

すべてのアプリケーションが、このオブジェクトモデルを使用している。

### 3.1. 実時間シミュレーションの構成

システム構成のうち、実時間シミュレーションを例に説明する。実時間シミュレーションの構成を図3に示す。アプリケーションからは、オブジェクトモデルへの直接的な参照は行わず、RTSEngineと呼ぶ計算制御プログラムを通してオブジェクトモデル内のオブジェクトを操作している。

#### 3.1.1. データロギングサービス

データロギングサービスは、SCADAシステムから定期的に送られる計測値データを監視し、取り込み、格納するものでありNTサービスとして実装している。連続運転によりSCADAの計測値を監視するため、ActiveX EXEとして作成している（図4参照）。

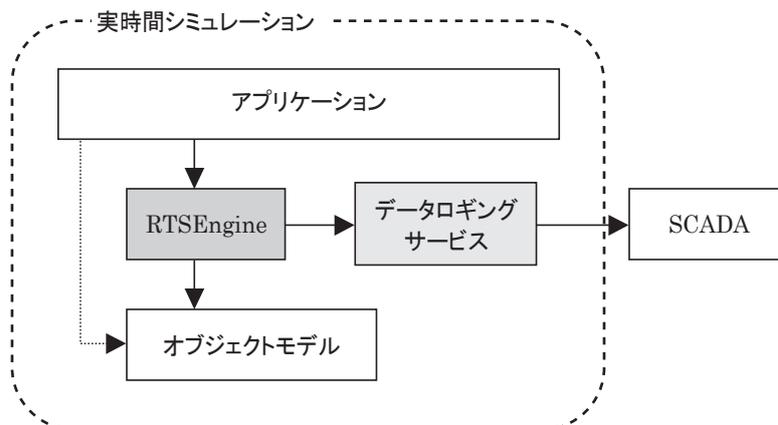


図3 実時間シミュレーションの構成

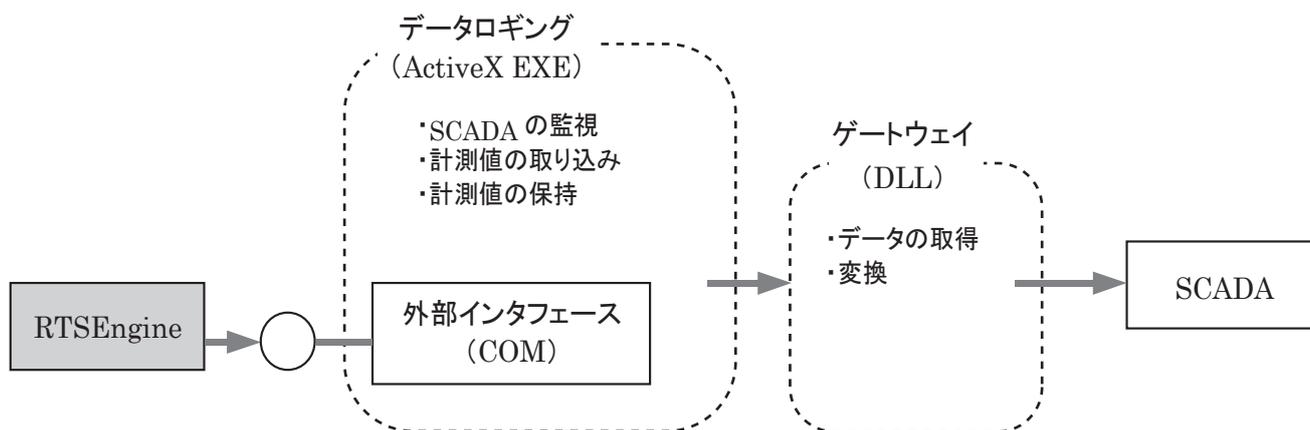


図4 データロギングサービス

データロギングサービスが保持するデータは、外部への提供方法としてCOMオブジェクトを用意している。この外部インターフェースを通して、外部のシステム(RTSEngine)が計測値を取得する。

SCADAシステムは、適用先ごとに異なったフォーマットで情報を転送するため、データロギングサービスでは、SCADAから取得したデータをデータロギングが保持可能な情報へ変換する機能(ゲートウェイ)を切り離れた構成となっている。したがって、適用先用にゲートウェイを置き換えるだけでデータロギングサービスを利用できる構成となっている。

### 3.1.2. RTSEngine (Real Time Simulation Engine)

RTSEngineは実時間シミュレーションの核となるエンジン(図5参照)である。データロギングサービスが格納している計測値データを取得し、桁揃え、単位変換、平滑化といった計測値修整処理、および非定常計算を行う。RTSEngineは、オブジェクトモデルを参照(=オブジェクトモデルはDLLのためRTSEngineのメモリ領域に展開される)し、連続運転を可能とするため、ActiveX EXEとして作成している。

RTSEngineは定期的に計測値修整処理、および非定常計算を行うための「計算ループ」と、計算ループへの設定

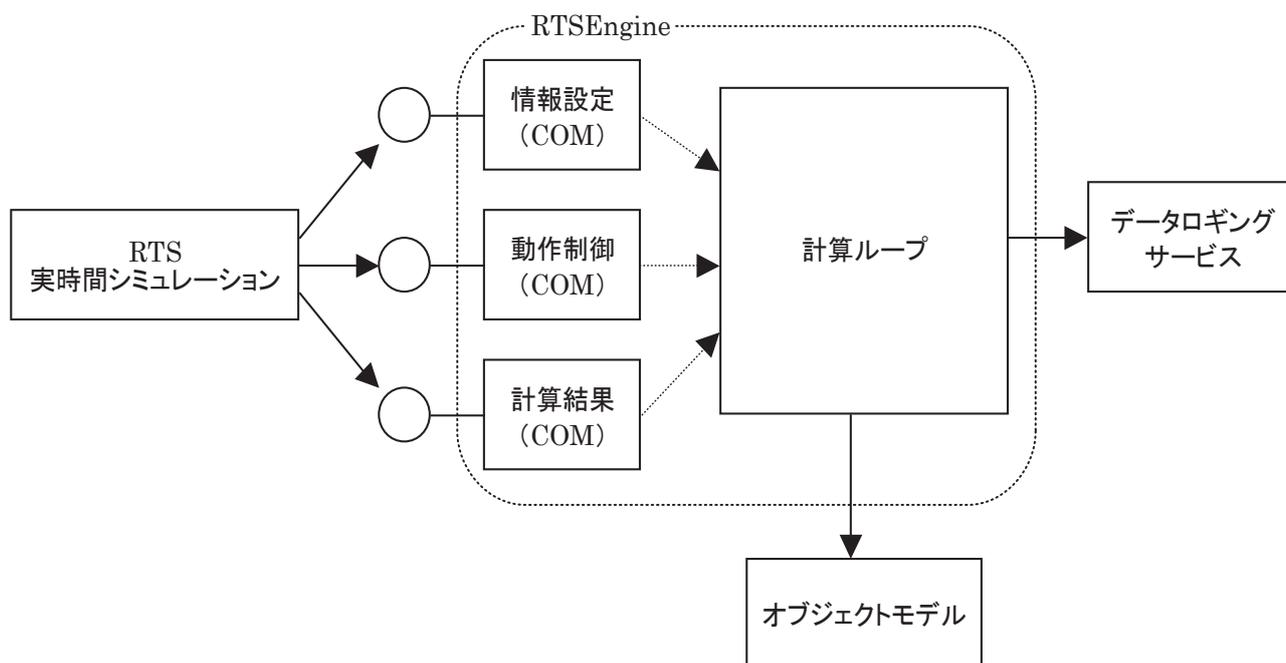


図5 RTSEngineの構成

などの命令を行う外部インタフェースにより構成されている。計算ループ内では、データロギングサービスが持つインタフェースを介して計測値を取得する。外部インタフェースは、次の3つのオブジェクトから構成される。また、リモートマシンからの設定、操作、データ取得を可能とするためにDCOM(Distributed Component Object Model)としての機能を実装している。

- ・ 「情報設定」 オブジェクト  
アプリケーションからRTSEngineへ計算対象となるパイプラインモデルや計算の情報などを設定するためのオブジェクトである
- ・ 「動作制御」 オブジェクト  
RTSEngineの連続運転を開始／終了させるためのオブジェクトである
- ・ 「計算結果」 オブジェクト  
計算ループ内で行われた計算により算出した計算結果を取得するためのオブジェクトである

#### 4. オブジェクトモデルの適用

すべてのアプリケーションにおいて、システム要件の中から共通する要素である静的情報の洗い出しをデザインパ

ターンの使用より行った。この情報は、すべてのアプリケーションが共通で使用可能である。

非定常計算時の基となる動的な情報（RTSで扱う「計測値」とFCSで扱う「シナリオ」）、および非定常計算後の計算結果の値は時間により変化するが、静的情報と同様に保持するデータの分析を行い共通で使用可能となるように工夫した。アプリケーション側は汎用性を持たせるために、各情報ファイルの読み込み、および書き込み処理を設けてファイル内部を意識しなくても利用が可能な機能を実装した。

##### 4.1. ツール

従来の業務アプリケーション開発では、データの流れに着目してDFD (Data Flow Diagram) を用いて機能を分析する構造化分析や、データの要素とその関係に着目してERD (Entity Relationship Diagram) を用いて分析を行う概念データモデリングなどの方法が用いられてきている。しかし、これらは、ソフトウェアの1つの側面だけを捉えた分析手法であるため、システムが複雑になると全体像を正しく把握することが困難である。今回の開発においては、次の3つの視点に着目して、OMG (Object oriented

Management Group) がオブジェクト指向のソフトウェア開発における標準表記法として定めたUML (Unified Modeling Language) を用いた。

- (1) 構造の視点： システムの構成要素とその関係 (誰が・何が)
- (2) 機能の視点： システムの果たす機能とその処理の流れ (何を)
- (3) 振舞いの視点： システムの実行順序とそのタイミング (いつ)

要求分析から設計開発の各アクティビティにおいて、ラショナルソフトウェアが開発したビジュアルモデリングツール「Rational Rose」を積極的に活用した。UMLドキュメントの作成・ソースコードの生成・リバースエンジニアリングといった機能により、ビジネスモデルからプログラムの構造に至るまでソフトウェア開発に伴うさまざまな情報を視覚化し、チーム開発時の生産性、保守性、再利用性を大幅に向上させることができた。

UMLは表記法だけを規定しているので、それを用いてどのように分析するかは、システム開発者に委ねられている。また、オブジェクトモデルや機能別DLLを利用してアプリケーションを構築する際には、UMLのクラス関連図だけでは実行時におけるメモリ展開イメージ (いわゆるオブジェクト階層) と判断し、オブジェクト階層図を作成して技術共有を行った。

## 4.2. 開発プロセス

開発プロセスとしては、ウォーターフォール型の持つ欠点を補うために、要求や仕様の変更・追加を必要な都度随時取り入れ、また開発チームの内外でシステムの出来映え (品質や性能) が開発中にも見えるようにすることを重視して、インクリメンタル・イテラティブアプローチの開発プロセスを採用した。

このプロセスを適用して開発するには、進捗などの具体的な基準を定め、管理する方法を決める必要があり、ウォーターフォール型の開発プロセスに比べ複雑になるので、プロセスとプロダクト (成果物) の両面で厳密な管理を行う必要がある。特に、開発メンバー内のソースコード管理ツールとして、メラント・インターナショナル・リミテッドの「PVCS Version Manager」を有効に活用した。

## 5. オブジェクトモデルの考察

### 5.1. 効果

オブジェクトモデルを使用して効率が良くなったと考えられる点は、次のとおりである。

- (1) オブジェクトモデルでは、適用先用に特別な機能を持たないため、適用先によってコードの修正などが発生せず、そのまま使用することができる。
- (2) オブジェクトモデルが持つ各クラスは、部品として必要な要素と機能を持ち合わせているため、新規適用先への対応時にはほとんどコードを修正することなく対応可能である。
- (3) オブジェクトモデルは、アプリケーションと分離しているため、適用先に合わせたユーザインタフェースを提供しやすくなる。
- (4) 開発時にはユーザインタフェース部とオブジェクトモデル部の作業分担が容易であるため、平行して開発することができ、開発時の効率を高めることができた。
- (5) (オブジェクト指向の考え方に基づき) オブジェクトは、機能を分散させ、自分が行うべき機能のみを持つように作成しているため、不具合発生時には、原因を特定しやすく、また、機能追加時にも対応が行いやすい。

### 5.2. 適用上の考慮点

リアルタイムシステム開発に、オブジェクトモデルを適用する上での注意する点は、次のとおりである。

- (1) オブジェクトモデルは、汎用性、拡張性など柔軟な対応が行え、利用する効果は大きかったが、多量のオブジェクトデータを繰り返し処理すると、著しくパフォーマンスが低下してしまう。配列化することによって処理性能を確保できるような工夫が必要である。
- (2) 今回オブジェクトモデルをVisual Basic (一部VC++) で構築したが、オブジェクトの継承関係が深くなると、メモリを多重消費して使用メモリ量の増加が発生し、リソースやパフォーマンスに影響が出てしまい、オブジェクト構造を構築する際に注意が必要である。

上記以外で気がついた点も、以下に列挙する。

- 効率的かつ十分なテスト作業を実施するためには、テスト・ツールの導入が不可欠である。
- 不具合修正にしても常に全体構成を意識しながら行う必要があるため、今まで以上に関連図書の整備および活用が必要となってくる。

## 6. おわりに

今回のオブジェクト指向によるシステム開発はメンバにとって初めての経験であったが、高い品質と柔軟性に富んだ成果を得ることができたと評価している。今後、この知見を活かして本システムのさらなる増強に取り組む所存である。

最後に本システム開発の機会を頂いたJFE日本鋼管株式会社殿、および実施するに当たって技術コンサル、システム開発方法など多岐にわたり御支援をいただいた（株）ブリッジの方々に感謝の意を表す次第である。

<問い合わせ先>

総合エンジニアリングシステム事業部

鶴見システム開発部 総括チーム

Tel. 045-505-7376 井上 雅久

E-mail: masahisa-inoue@exa-corp.co.jp

-----  
本文中の会社名、製品名、およびサービス名などはそれぞれ各社の商標または登録商標です。  
-----