

# 電力系統情報管理へのデータベース適用検討

2 X - 7

協園 竜次 畑中 隆洋 土屋 武彦 川村 敏和 田中 立二

(株) 東芝 重電技術研究所

## 1 はじめに

プラント監視制御等のオンライン・リアルタイム・システムにおいては高速応答性を要求される事から、既存のデータベース管理システムを使わず専用のファイル管理システムを用いる事が多い。これに対しオブジェクト指向データベース (OODB) の出現により以下の機能を高速に実現する事が可能になった。

- (1) 複雑・不規則な構造を持つデータベースの構築
- (2) データに付随した操作・処理のデータベース化
- (3) プログラムからのデータベースの容易なアクセス

これらの機能により、CAD、CASE等のエンジニアリング分野や地図情報システム分野等への適用だけでなく、従来は適用の難しかったプラント監視制御等のオンライン・リアルタイム・システムへのデータベースの適用を考慮する事ができる。

電力系統、発電および各種産業におけるプラント監視制御システムにおいては、対象プラントのリアルタイム・データおよび監視・操作画面等の GUI(Graphical User Interface) 関連データのデータベース化およびこれらの統合化が必要とされている [1]。

本稿ではプラント監視制御システムへの OODB 適用の試みとして、電力系統監視・操作画面処理への OODB 適用を検討し、そのモデル化と OODB 適用による効果について述べる。

## 2 情報モデル

本稿では、ターゲットとする電力系統監視制御システムとして、電力系統上のフィーダー線や設備機器に対応した各種情報を必要に応じて検索・表示が出来る構成を考えている。OODB 上でこれらの機能を実現するために用意した電力系統クラスは電力系統の構成とそれを表示・検索・操作するための処理で構成される。今回検討した電力系統情報のモデルを図 1 に示す。

Evaluation of Object Oriented Database for Distributed Electric Power Systems  
Ryuji Wakizono, Takahiro Hatanaka, Takehiko Tsuchiya,  
Toshikazu Kawamura, Tatsuji Tanaka  
Heavy Apparatus Engineering Lab., TOSHIBA Corp.

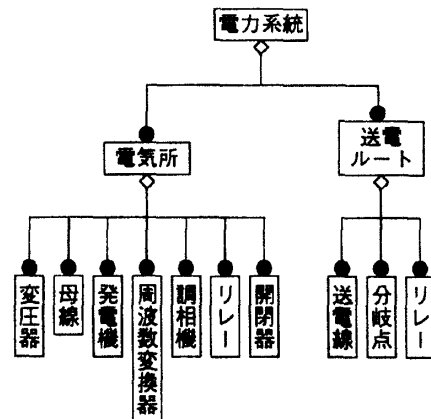


図 1: 電力系統情報のクラス構成

各クラスは、それぞれの設備の名称、管理番号、状態および GUI のための図形情報等との関連などのデータとそれに関連する処理で構成されるものである。

## 3 電力系統監視制御システム上での RDB と OODB の比較

この系統情報を格納するデータベースとして RDB と OODB を用いた場合とで系統情報の検索処理時間の比較を行った。処理の内容は、充電判定(ある箇所での停電が発生した場合にその影響が系統上のどの範囲に及ぶかを求めるもの)で、これには停電の発生箇所から系統の接続情報を辿る処理が必要となる。なお、RDB は市販のものを使用し、OODB は独自開発したものを使用した [2]。

ここでの処理は送電線 (LINE)、開閉器 (CB)、リレー (RELAY)、変圧器 (TRANSFORMER)、発電機 (GENERATOR)、母線 (BUS)、調相機 (RPS)、周波数変換器 (FC) 等の設備機器および分岐点 (POINT) で構成されている。接続リスト (CNLIST) および接続表 (CONNECTION) はこれらの接続関係の情報である。系統は母線を起点として構成されており、接続表は母線に繋がっている機器のリストである。

図 2 と図 3 に、これら系統設備に関する RDB と OODB それぞれのスキーマ定義およびクラス定義の一部を示す。

```
EXEC SQL CREATE TABLE BUS
  (no1 number, name char(36), cnList number,
   voltage number, flag number);
EXEC SQL CREATE TABLE CNLIST
  (no1 number, count number,
   cn1 number, cn2 number, cn3 number,
   .....
   cn19 number, cn20 number);
EXEC SQL CREATE TABLE CONNECTION
  (no1 number,
   cn1 number, eqtype1 number, /* 接続設備 1 */
   cn2 number, eqtype2 number, /* 接続設備 2 */
   onOff number); /* 接続状態 (ON/OFF) */
.....
```

図 2: RDB スキーマ定義

```
class Bus { // 母線
public:
  unsigned short number; // 設備番号
  char name[36]; // 設備名称
  Collection* cnList; // 接続状態
  float voltage; // 電圧
  unsigned short flag; // フラグ
  // コンストラクタ, 他のメンバ関数
  .....
};
class CNLIST {
public:
  unsigned short number; // 設備番号
  Collection* cnList; // 接続状態
  // コンストラクタ, 他のメンバ関数
  .....
};
class CONNECTION {
public:
  Equipment* cn1; // 接続設備 1
  Equipment* cn2; // 接続設備 2
  unsigned short onOff; // 接続状態 (ON/OFF)
  // コンストラクタ, 他のメンバ関数
  .....
};
.....
```

図 3: OODB クラス定義

充停電判定は、図 4 に示す通り母線および発電機を開始点として、接続リストおよび接続表から得られる接続情報を元に各設備を辿りながら行われる。

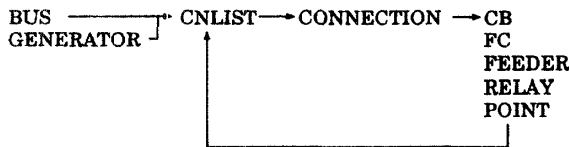


図 4: 検索処理時に辿る系統接続関係

このナビゲーション処理部分に関する RDB および OODB での実装の様子を図 5 および図 6 に示す。

```
/* 母線から設備をたどる処理 */
EXEC SQL DECLARE C1 CURSOR FOR
  SELECT no1, cnList
  FROM Bus
  WHERE power = 1;
EXEC SQL OPEN C1;
EXEC SQL WHENEVER NOT FOUND GOTO end_bus;
for(;;) {
```

```
EXEC SQL FETCH C1 INTO :no1, :cnlist;
  setPowerL(no1, BUSTYPE, cnlist);
}
end_bus:
EXEC SQL WHENEVER NOT FOUND CONTINUE;
EXEC SQL CLOSE C1;
...

```

図 5: RDB での設備検索処理

```
/* 母線から設備をたどる処理 */
for( bus=(Bus*)tBus->first(); //tBus:BUS の集合
  bus;
  bus=(Bus*)tBus->next())
{
  if (bus->power == POWER_OFF) continue;
  Station::setPowerL((Equipment*)bus);
}
...

```

図 6: OODB での設備検索処理

ワークステーション上においてこの設備検索処理にかかる時間を RDB の場合 (CONNECTION テーブルに対して索引を使用) の場合と OODB(索引なし) とで比較した結果、コールドスタートではほとんど両者の差が出なかったものの、ホットスタートでは、OODB が RDB に比べて 10 倍～100 倍高速であることがわかった。そのときの OODB の検索時間は SparcStation2(メモリ 48M) で 0.5 秒以内であり、監視処理に要求されるリアルタイム性能を十分満足できるレベルの結果を得ることができた。

#### 4 まとめ

電力系統自動化システムで重要な位置を占める監視・操作のための系統監視画面の表示情報およびその処理にオブジェクト指向データベースを適用しその機能および性能を評価した。この結果、画面情報を管理するデータベースとしては十分実用になる事が明らかになった。ただし、RDB のテーブル構成が OODB のクラス構成とほぼ一対一となる形で両者を比較したため、RDB に適したテーブル構成も考えて比較する必要もある。また、充停電判定以外の機能では RDB でも十分な場合があり、その場合に OODB と RDB との間でどのようにして情報を共有させていくかも今後の課題である。

#### 参考文献

- [1] 土屋, 田中, 関: 広域ネットワークを利用した電力系統監視システムの柔軟性確保, 情報処理学会第 55 回全国大会, 1997.
- [2] 協園, 土屋, 川村, 田中: オブジェクト指向データベースの開発, 情報処理学会第 48 回全国大会, 1994.