2C-3

# 環境情報データベース向けリアルタイムセンサデータロード方式

竹田　義聡†　　中村　隆顕†
和田　貴成†　　郡　光則†
三菱電機株式会社　情報技術総合研究所†

## 1. はじめに

改正省エネルギー法の施行を受け、環境・省エネ対策への取り組みが重要な経営課題と位置付けられるようになってきた。我々は、大量のビジネスデータや情報セキュリティログの管理で実績のある高性能並列情報検索技術[1]をベースに、電力・温度など省エネに関係するセンサデータをリアルタイムに収集し一元管理する環境情報データベースの開発に取り組んでいる。本稿は、センサデータの特徴を考慮したリアルタイムロード方式の概要と実装について説明する。

## 2. 高性能並列情報検索技術の概要

我々は、時間とともにデータが追加される追記型データベースに適した高性能並列情報検索技術の開発に取り組んでいる[1]。効率的データ配置と高速・高圧縮率データ圧縮によるストレージアクセス技術、データ駆動型の並列処理技術などにより、RDBMS で用いられているメモリへのキャッシュや索引等の効果が得られにくい非構造化データやログなどの高速処理を実現する。
電力・温度などのセンサデータは、時系列に大量に発生し、蓄積に当たっては更新を伴わず追記される。このため上述の高性能並列情報検索技術に適していると考えられる。今回、高性能並列情報検索技術をベースに、環境情報データベースを実現するための技術開発を行った。

## 3. 今回方式の概要

本章では、センサデータの特徴を踏まえ今回方式の概要を説明する。

### 3.1. 機能とテーブルの構成

センサデータは「日付時刻」「センサ ID」「計測値」を項目として持つ。データベースへのロードに当たっては、収集され、レコードとしてまとめられ、蓄積される。この一連の処理に従い、センサデータロードシステムを、データ収集機能、レコード生成機能、データ蓄積機能の 3 機能として実現した。計測値格納テーブルはセンサと同数の列を定義した（図 3-1）。

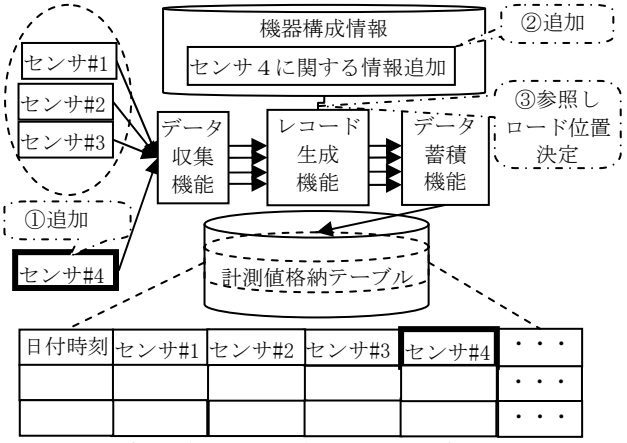


図 3-1　今回方式の構成とセンサ追加への対応

このテーブル構成は、正規化されたテーブル(図 3-2)など他の構成と比べ以下のメリットを持つ。

- 特定のセンサの計測値を抽出・集計する処理が、高性能並列情報検索技術による高速化の効果が特に高い列方向の演算となる
- 各センサの計測値の連続性を利用したデータ圧縮の効果が高くなる[2]。

| 日付時刻 | センサ ID | 計測値 |
|---|---|---|
| | | |
| ： | ： | ： |

図 3-2 正規化された計測値格納テーブルの例

### 3.2. センサの構成変更への対応

データ供給元としてのセンサは、配置や機種変更が容易であり、また故障などによる機器交換が発生しやすいという特徴を持つ。このため、例えばセンサを配置したフロアごとの消費電力集計のように、センサの配置や機種に依存する情報を利用してデータを活用しようとする場合には、センサの構成変更を反映する必要がある。今回、図 3-1に示すように、センサはじめデータロードシステムを構成する機器の構成情報を、計測値格納テーブルとは別のデータベースとして実現し、センサ配置などの変更に柔軟に対応

A Realtime Sensor Data Loading Scheme for the Consolidated Electricity Consumption Database.
† Yoshisato TAKEDA, Takaaki NAKAMURA, Takashige WADA, Mitsunori KORI Information Technology R&D Center, Mitsubishi Electric Corporation

可能とした。機器構成情報は以下の情報を持つ。

- 各センサの ID、機種情報
- 各センサに対応するテーブルの列
- 各センサのデータ属性（型、精度など）
- 各センサの NULL 値（3.3参照）
- 各センサの配置情報

図 3-1の吹き出し①～③は、センサ#1～#3 から構成されているデータロードシステムに対し、センサ#4 を追加する様子を示す。

### 3.3. レコード生成・蓄積

レコード生成機能は、日付時刻の幅が一定時間以内のセンサデータを同一レコードに集約する（図 3-3）。この際、3.2の機器構成情報を参照し、センサに対応する列情報をもとにレコードのフィールド構成を決定する。

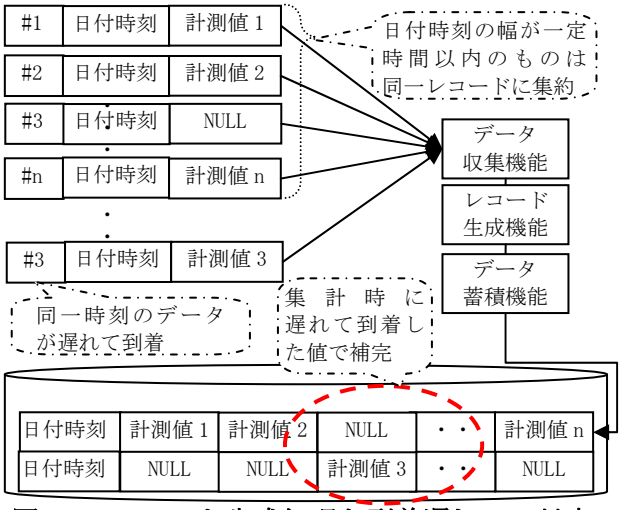


図 3-3 レコード生成処理と到着遅れへの対応

蓄積に当たって考慮すべき点として、センサデータは、センサの故障による欠損、ネットワーク輻輳による到着遅れ、再送による重複が発生する可能性がビジネスデータやセキュリティログなどより高いことが挙げられる。今回は以下のように対処した。欠損は、機器構成情報で設定したセンサ毎の NULL 値で補完する。到着遅れが発生すると、図 3-3中のテーブルのように同一日付時刻に対して複数のレコードがロードされる。この場合、ロード後のデータ利用時においては、後から到着した値により欠損値を補完するようにする。同一のセンサより、同一レコードに集約する時間の幅以内に複数データが重複して到着した場合は、後から到着したデータをレコードに反映し、重複を排除する。

## 4. 実装

### 4.1. リアルタイムロード用 API の必要性

リアルタイムロードのインタフェースとして、ODBC などの汎用インタフェース、或いは大量データ向けのバッチ用インタフェースを利用すると、それぞれの本来の目的と異なり、短い周期で短いデータを単位としてデータベースに書き込むため、オーバーヘッドが発生し書込みスループットが低下することが予想される。これを避けるため、環境情報データベースへリアルタイムにデータをロードする API（Application Program Interface）を実装した。その際、データベースへの書き込み周期とスループットのトレードオフ、データ圧縮に要する時間を考慮し、データ生成から最短 30 秒以内にデータベースへ反映し利用可能にするよう設計した。

### 4.2. 性能評価

この API を用いてセンサ群から 1000Mbps のネットワーク経由でサーバへデータロードする際のピーク時性能を計測したところ、ネットワーク遅延を除き毎秒 20M バイト[1]の環境情報データベースへの書き込みスループットを達成していることを確認した。同構成で ODBC を用いて商用 RDBMS へのリアルタイムロードを行うプログラムを試作したところ、今回方式のほうが約 10 倍高効率であった。測定環境を示す。

- クライアント

| | |
|---|---|
| プロセッサ | 1 個（2 コア）2.66GHz Intel Core2 6700 |
| メモリ | 1.98 GB |
| ディスク | Samsung HD161HJ 20GB×1 本 |
| OS | Windows XP Professional SP3 |

- サーバ

| | |
|---|---|
| プロセッサ | 1 個（2 コア）3.00GHz Intel Core2 Duo E6850 |
| メモリ | 1.96 GB |
| ディスク | WDC WD1600AAJS-19PSA0 144GB×1 本 |
| OS | Windows XP Professional SP3 |

## 5. まとめ

環境情報データベース向けのリアルタイムセンサデータロード方式の概要と実装を説明した。今回実現した技術を、環境 IT システム開発のコアコンポーネントとして活用する予定である。

## 参考文献

[1] 郡光則，他：高性能並列情報検索技術，三菱電機技報，83， No.12（2009）

[2] 加藤守，他：環境情報データベース向け高性能センサデータ圧縮方式、情報処理学会第 73 回全国大会

[1] センサデータのレコードへの集約前のサイズを、データ収集機能のデータ受信から書き込み完了までの所要時間時間で割った値