

## SuperKEKB の電磁石制御システムのデータ管理

### DATA MANAGEMENT FOR THE SUPERKEKB MAGNET CONTROL SYSTEM

中村達郎<sup>#,A)</sup>, 中村卓也<sup>B)</sup>, 吉井兼治<sup>B)</sup>

Tatsuro Nakamura<sup>#,A)</sup>, Takuya Nakamura<sup>B)</sup>, Kenzi Yoshii<sup>B)</sup>

<sup>A)</sup> High Energy Accelerator Research Organization (KEK)

<sup>B)</sup> Mitsubishi Electric System & Service CO., LTD.

#### Abstract

In the KEKB magnet control system, relational database management system (RDBMS) has been used for the management of the device configuration. Most of the device parameters, such as address of the interfaces, constant parameters of the magnet power supplies, excitation functions of the magnets, limit values for the operation and so on are stored in the RDBMS. These parameters are used mainly for the macro expansion to generate the runtime EPICS IOC database files. They are also used to configure the high level application programs. In the SuperKEKB magnet control system, instead of the RDBMS, the simple configuration files, named TXDB, are going to be used for the management of the device configuration. The configuration data in the RDBMS have to be converted into TXDB format. The stored procedures that generate EPICS parameter files (dbprm files) also have to be converted into Python scripts. The process of the data migration is reported. The compatibility between both methods is also discussed.

#### 1. はじめに

SuperKEKB 加速器の電磁石制御システムでは、制御に必要な各種パラメータなどのデータを管理する必要がある。例えば、インターフェースアドレス、電源の特性定数、電磁石の励磁曲線関数、運転上の制限値、較正定数などである。これらのデータは主に EPICS の IOC database を生成する際に必要になると共に、SADScript や Python といったスクリプト言語で書かれたアプリケーションプログラムからも参照されて利用される。KEKB 加速器ではこれらのデータをリレーショナルデータベースで管理していたが、SuperKEKB 加速器では TXDB と呼ぶ簡素な記法のテキストファイルで管理するよう変更し、データ管理作業の簡素化を図ることとした。KEKB の電磁石や電源の多くは SuperKEKB でも使われるため、それらのデータをリレーショナルデータベースから TXDB へと変換することが必要となる。また、EPICS の IOC database の生成には、リレーショナルデータベース内のストアードプロシージャを使っていたため、これらのプログラムは Python で書き直すこととした。ここではこれら移行作業について述べると共に、移行の際の互換性について議論する。

#### 2. 電磁石制御システムの概要

##### 2.1 SuperKEKB 加速器制御システムと EPICS

SuperKEKB 加速器制御システムは KEKB 加速器制御システム<sup>[1][2][3]</sup>と同じく、骨格となるソフトウェアに EPICS (Experimental Physics and Industrial Control System)<sup>[4]</sup>を採用している。EPICS は世界規模で多くの研究所が共同で開発・維持しているオープンソースのソフトウェアであり、加速器等の大型実験装置

の計算機制御に向けた分散処理を特徴とする。EPICS では IOC (I/O Controller) と呼ばれる計算機に database を常駐させて制御を行なう。(この database は一般に使われる用語としてのデータベースとは異なり、混同を避けるためしばしば EPICS database, runtime database 等の呼び方をされる。ここでは IOC database と呼ぶことにする。) SuperKEKB 加速器では IOC として主に VME のボード計算機、Linux CPU を実装した PLC、Linux サーバ計算機などを使用し、ローカル制御室などに 100 台以上を分散配置する。

IOC database は record と呼ばれる制御単位の集合体である。IOC database は定義ファイルを記述することによって作ることができ、標準的な機能であればほとんどプログラミングすること無しに制御システムを構築することも可能である。

KEKB では SADScript<sup>[5]</sup>, Python など<sup>[6]</sup>のスクリプト言語がアプリケーション開発によく使われる。Python は汎用のスクリプト言語で、文法が簡明で初心者にも扱いやすい。またリスト、タプル、辞書といった構造化に適したデータ型が用意されていて、複雑な処理も分かり易く記述できる。各種ライブラリも充実しており、効率的なプログラミングが可能である。

##### 2.2 電磁石制御システム

SuperKEKB では KEKB の電磁石制御システム<sup>[7][8]</sup>から大きな変更はない。KEKB, SuperKEKB を通じて特徴的なのは、電磁石電源の数が多 (2000 台以上) ことである。その大部分を占めるのはステアリング磁石や補正コイル用の小型電源である。このため KEKB では配線の省力化を狙って IOC と電磁石電源間を ARCNET による通信で結ぶ構成を採用した。

KEKB の電磁石制御のもう一つの特徴は、複数の電磁石電源を同期させた電流設定を可能としたことである。KEKB は加速を行なわない蓄積リングのた

<sup>#</sup>tatsuro.nakamura@kek.jp

め、電磁石電源間での同期は原理的には必要ないが、チューン補正、軌道補正、オプティクス補正など、各種の補正作業がビームを蓄積したまま迅速に行なえるよう、電磁石電源の同期設定機能を導入した。また電流設定時に任意の電流変化曲線（トラッキング・パターン）を指定できるようにしており、柔軟なオペレーションを可能にしている。

KEKBでは同期設定の機能を実装するため、PSICM (Power Supply Interface Controller Module)<sup>[9]</sup>と呼ぶモジュールを開発し、全ての電源にプラグインするようにした。PSICMはまた、ARCNETインターフェースを持ち、IOCとのインターフェースとしての役割も担う。

SuperKEKBでは引き続きARCNETやPSICMを採用している。電磁石電源の増加に伴い、PSICMの増設も進行中である。新しいPSICMには若干の機能拡張が加えられるが、基本的に大きな変更はない。

このほかKEKBではARCNETのほかに、電磁石電源の電流モニターの読み出し系にGPIBを、電磁石インターロックシステムとの情報交換にModbus plusを使用して来た。SuperKEKBではインターロックシステムは全面的に更新される予定なので、その部分とのインターフェースは新たに開発となる。

### 2.3 電磁石制御に必要なパラメータ

電磁石制御には次のようなパラメータが必要となる。(主なものを挙げる。)

- 電磁石名、電源名
- ARCNET アドレス
- GPIB アドレス (電流モニター用)
- PLC アドレス (インターロック情報用)
- 電源の型式・種別
- 電源の定格出力電流
- 電源の設定精度 (DAC ビット長)
- 電源の較正定数
- 電磁石・電源の対応関係
- 電磁石本体と補助巻線の対応関係
- 電磁石の励磁曲線関数
- 電磁石の位置関係
- 運転上の制限条件 (電流値など)
- スタンダーダイズのパラメータ

## 3. KEBB 加速器でのデータ管理

### 3.1 リレーショナルデータベース

KEKBでは制御対象の機器の情報を管理するため、リレーショナルデータベース<sup>[10]</sup>を使用して来た。リレーショナルデータベース管理システム (RDBMS) としてOracleを導入している。このデータベースを (IOC databaseと区別して) 機器データベースと呼ぶ。

リレーショナルデータベースには関係モデルという理論的基礎があり、それに基づき設計・開発される。リレーショナルデータベースではSQLと呼ばれる問い合わせ言語が広く使われており、データ

ベースへのアクセスはSQLを使って行なうのが標準的な方法となっている。

OracleではPL/SQLと呼ばれるSQLの拡張言語が用意されている。これはSQLを手続き型言語に拡張したものである。PL/SQLで書かれたプログラム自体をデータベースに格納することができ (ストアードプロシージャ)、データとプログラムをRDBMSで一元管理することが可能である。

### 3.2 電磁石、電源のデータベース構造

電磁石制御に必要な表の中で中心的な役割を持つのが、次の電磁石に関する3つの表と電源に関する3つの表である。

- MAGNET\_ALL 電磁石の基本情報
- MAGNET\_TYPE 電磁石タイプ別情報
- MAGNET\_ENTITY 電磁石の个体情報
- MAGNET\_PS\_ALL 電源の基本情報
- MAGNET\_PS\_TYPE 電源タイプ別情報
- MAGNET\_PS\_ENTITY 電源の个体情報

タイプ別情報とは、電磁石や電源の型式毎に定まる型式共通の情報である。个体情報とは物理的な電磁石や電源の個別情報である。个体情報は一般には製造シリアル番号で識別される。一方、基本情報は運転で使われる電磁石名や電源名で識別される。通常、基本情報と个体情報は一対一の対応があるが例外もある。(例えば予備品は个体情報しか持たない。)特に数の多い小型電源は、故障の際に予備品と交換されるため、个体情報を基本情報とは独立の表で扱うことでデータ更新がし易くなっている。電磁石と電源の対応は、MAGNET\_ALL表とMAGNET\_PS\_ALL表間の参照で表されている。

電磁石の補助巻線はデータベース上では1個の電磁石として扱われ、独立の名前が付けられている。電磁石本体 (主巻線) と補助巻線の対応はMAGNET\_ALL表間の参照で表されている。

このほか補助的に使われる表が幾つか存在する。中でも重要なのは電磁石の励磁曲線関数に関するものである。励磁曲線関数はいくつかの方式ごとに別々の表で管理されている。励磁曲線関数は電磁石毎に定まるものであるが、実際に制御するのは電磁石電源の電流であるため、同一の電源に結線されている電磁石群の平均値が運転に使われる励磁曲線関数となる。また、近くに設置されている電磁石間 (四極磁石と直近のステアリング磁石など) での干渉による補正も考慮される。

### 3.3 IOC database の自動生成

IOC databaseを定義するファイルは「.db」という拡張子を持つことからdbファイルと呼ばれる。電磁石電源のように機器が多数ある場合には、テンプレートとなるdbファイルとパラメータファイル (その拡張子からdbprmファイルと呼ばれる。) からマクロ展開によりdbファイルを生成するのが一般的である。KEKBでは機器データベースの登録情報を基にdbprmファイルを自動的に生成することでIOC databaseを最終的に生成している。dbprmファイルを生成するためのデータ処理はPL/SQLによつ

て書かれている。

### 3.4 スクリプト言語からの参照

KEKBではSADScript, Pythonなどのスクリプト言語で運転用のアプリケーションプログラムを開発することが多い。このとき機器データベースはスクリプト言語からも参照される。特にリレーショナルデータベースの情報を読んで加工するのにPythonを用いる事が多い。そこでリレーショナルデータベースに特有な表形式のデータを扱うのに適したPythonライブラリの開発も行なって来た。このライブラリはrdbtoolモジュール<sup>[11]</sup>と呼ばれ、それぞれデータベース、表、行に対応するrdb, tdbtab, rdbrowクラスが定義されている。

### 3.5 データ管理の問題点

KEKBではこのようにOracleを使ったデータ管理を10年以上続けて来たが、年を経て改造が加わるにつれ、いくつかの問題が浮かび上がって来た。

KEKB 加速器では性能向上のための改造が絶え間なく行なわれて来た。電磁石や電源の変更もしばしば行なわれて来た。その中には当初想定していなかった変更も少なからずあり、その度にデータ構造の変更(例えば列の追加など表定義の変更)が必要になった。データ構造の変更は、関連するPL/SQLスクリプト、ビュー、アプリケーションプログラムなどの変更に波及することがある。こうした波及を抑える手段としてそれまでの表定義と互換性のあるビューがしばしば導入されたが、複数次に渡る変更の結果重層化したビューがデータ構造の見通しを悪くすることにもなった。また、変更はその時々都合で行なわれ、急を要する場合はその場しのぎ的な対応(例えば特殊な変更の時にPL/SQL内に値が直接書かれた。)をすることもあった。

また、変更作業にあたってはSQLスクリプトを書く必要があり、複雑な変更(たとえばキーとなる名前自体に変更がある場合など。)の時にはSQLスクリプトも長くなり、データの更新には細心の注意が必要となる。

本来リレーショナルデータベースを使う場合、データ構造を頻繁に変更する応用はあまり想定されていない。データ構造が不変のままデータの挿入や更新が頻繁に行なわれるのが想定されている使用法である。これに対して、電磁石のデータの更新頻度はあまり高くない。最も多い変更は、小型電源の故障による予備品との入れ替えであった。これを除くと変更は多い時でも年に数回ほどに過ぎない。データ更新の頻度の割にはデータ構造の変更頻度が高いことが負担となっている。

また別の種類の問題としてOracleのバージョンアップに対応する手間がある。バージョンアップに際してはPL/SQLプログラムなどに変更が必要な場合があり、IOC databaseの自動生成などに影響が無いか動作確認には細心の注意を払う必要がある。さらに言えばPL/SQLはOracle独自の言語であるため、これに依存してプログラムを書いたことは、別のRDBMSに移行しにくくなる結果となった。

## 4. テキストファイルでのデータ管理

### 4.1 TXDB

複雑化してしまったOracleによる機器データベースに代わるものとして、もっと簡単なテキストファイルによるデータ管理ツールTXDB<sup>[12]</sup>を開発した。TXDBはデータベースというよりは、機器パラメータの管理を念頭に置いた記述フォーマットというべきものである。基本的にはデータの作成や更新にはテキストファイルを直接エディタで編集して行ない、データの参照や二次情報への加工などはPythonを使って行なうことを想定している。

TXDB 自体はKEKBの時から開発を行なって来たが、SuperKEKBへのアップグレードに際して電磁石制御システムのデータ管理をOracleからTXDBへ切り替える予定である。

### 4.2 電磁石、電源のデータ変換

SuperKEKBの電磁石や電源のデータは現時点ではまだ準備されていない。一方、KEKBで使われた電磁石、電源の多くがSuperKEKBでも引き続き使用される。そこでKEKBの(運転終了時点での)電磁石や電源のデータをまずOracleからTXDBに変換し、その後SuperKEKBに合わせた変更を行なっていくことを予定している。

データ変換はPythonスクリプトで行なう。その大部分は既にコーディングが終わり、変換の準備がほぼできている。データ変換に際して、データ構造を基本的には維持するが、構造を変更してデータの再編を行なうものもある。また、使用されていない不必要なデータは変換対象から省いた。電磁石のインターロックシステムは全面的に刷新されるため、それに関する情報はデータ変換を行なわない。

データ変換の対象は機器データベースに存在する全ての表ではなく、電磁石制御システムで参照が必要な表に限定される。例えば磁場測定的一次データも機器データベースに収納されているが、これらデータは直接には制御システムからは参照しないためTXDBに変換しない。(TXDBへの変換作業とは別にOracleの利用終了に備えて機器データベース内の表は全てダンプして保存することを予定している。)

### 4.3 TXDBによるIOC databaseの自動生成

TXDBのデータを基にIOC databaseを自動生成するプログラム(dbprmファイルを生成するプログラム)はPythonで用意する。これもほぼコーディングが終了している。今まで使われて来たPL/SQLプログラムのコードをなるべく忠実にPythonで書き直した。

### 4.4 TXDBとPython

TXDBフォーマットで記述されたデータをPythonで扱うためにtxdbモジュールと呼ばれるライブラリが利用可能である。このモジュールではtxdbクラスが定義されていて、TXDBフォーマットのファイルを読み込んで得られるデータはtxdbクラスの

インスタンスとなる。txdb クラスは、リレーショナルデータベースのデータを扱う rdbtool モジュールにおける rdbtab クラスに相当するもので、フィルタリングやソートなどデータ加工に便利な機能が同様のメソッドとして用意されている。このため rdbtool モジュールを利用してリレーショナルデータベースのデータを処理していたアプリケーションプログラムは、txdb モジュールを使うことで同様の処理が可能となる。

#### 4.5 アプリケーションプログラムの互換性

Python で書かれたアプリケーションプログラムに関する限り、前節で述べたように rdbtool モジュールを利用して機器データベースのデータを扱っていたプログラムは、txdb モジュールを使うことで同様の処理が可能となるため、互換性は高い。rdbtool モジュールと txdb モジュールで異なる点は、最初にデータを読み込む時のデータソースの違いになる。機器データベースの特定のテーブルを読み込んでいたプログラムは、TXDB ファイルを読んで対応するテーブルを取り出すことで同じデータにアクセスできる。

しかしながら、アプリケーションプログラムによっては、プログラムの先頭でテーブル全体を読み込むといった単純な問い合わせを行なうのではなく、複雑な問い合わせを行なう SQL 文をその都度動的に作成してデータにアクセスするものもある。この場合、TXDB では SQL を使ったデータアクセスはできないため、SQL で記述されているのと同様のデータ処理を txdb で記述しなおす必要が生じてしまう。

今のところ txdb モジュールで SQL を扱うことはできないが、何らかの RDBMS を中間にはさむことにより、一旦 TXDB のデータをリレーショナルデータベースのデータに変換した上で SQL による問い合わせを実行する方法も検討に値する。このような目的に適した RDBMS として SQLite の使用を検討している。SQLite はデータベースサーバを必要とせず、メモリ上だけのデータベースも扱える上、Python からライブラリとして使えるため親和性が高いと期待される。

#### 4.6 PF-AR 加速器への適用

PF-AR 加速器では 2013 年夏期停止期間中に電磁石電源の更新を行なっている。PF-AR の制御システムは SuperKEKB と共通であり、機器データベースも共通になっている。このためこの更新は TXDB による電磁石・電源のデータ管理の最初の適用事例となる見込みである。

## 5. おわりに

リレーショナルデータベースは最も広く普及しているデータベースシステムである。柔軟性も高く適用範囲も広い。しかし使いこなすにはそれなりの手間とコストが必要である。汎用性が高いため何にでも使えそうであるが、時には簡単な事は簡単な方法

で行なうことが良いこともある。

## 参考文献

- [1] N. Yamamoto et al., "KEKB control system: the present and the future", Proceedings of the 1999 Particle Accelerator Conference, New York, 29 Mar.-2 Apr. 1999, pp. 343-345
- [2] Nobumasa Akasaka et al., "KEKB accelerator control system", Nucl. Instr. and Meth. A499 (2003) 138-166
- [3] Tatsuro Nakamura, et al., "Status of KEKB Accelerators Control System in 2006", Proceedings of the 3rd Annual Meeting of Particle Accelerator Society of Japan, Sendai, Aug. 2-4, 2006
- [4] <http://www.aps.anl.gov/epics/>
- [5] <http://acc-physics.kek.jp/SAD/>
- [6] N. Yamamoto et al., "Use of Object Oriented Interpretive Languages in an Accelerator Control System", ICALEPCS'99, Trieste, 4-8 Oct. 1999, p. 600.
- [7] M. Yoshida et al., "Magnet Power Supply System for KEKB Accelerator", EPAC-98, Stockholm, 22-26 June 1998
- [8] T. T. Nakamura et al., "Magnet Power Supply Control System in KEKB Accelerators", ICALEPCS'99, Trieste, 4-8 Oct. 1999, p. 406.
- [9] A. Akiyama et al., "KEKB Power Supply Interface Controller Module", ICALEPCS'97, Beijing, 3-7 Nov. 1997
- [10] M. Kaji et al., "Relational Database System in the KEKB Accelerator Control System", APAC98, Tsukuba, 23-27 Mar. 1998.
- [11] Tatsuro Nakamura, "Application of the Device Database in the Python Programming", Proceedings of Particle Accelerator Society Meeting 2009, Tokai, Aug. 5-7, 2009, p. 508.
- [12] T. T. Nakamura, "Application of a Simple Text Format as a Device Configuration File", ICALEPCS2009, Kobe, 12-16 Oct. 2009, p. 432.