

## Data base system for accelerator operation support

Isamu ABE, \*Masakatsu MUTOH and Kazuo NAKAHARA

High Energy Accelerator Research Organization

Oho 1-1, Tsukuba, Ibaraki 305, Japan

\*Tohoku university, LNS

The PF Linac control system is composed of three tiers with 1) Device layer, 2) Middle layer, 3) Human interface (HI/F) layer, and a database (MS-SQL) is running on Windows NT server with multi-Pentium processors at the middle layer. The data base server (with gateway for device and HI/F layer) was built and has been upgraded for J-Linac to offer several functions as 1) Cashing system for a speed up, 2) Standard commands for HI/F layer, 3) Command log, 4) Easy tools for a data base handling, 5) Trigger functions distributed on the networked PC stations. These items will be discussed in this paper.

### 加速器運転支援用データベースシステム構築

#### 概要

電子・陽電子加速器において現在 J 部が建設中であり、合わせて制御システムも増強および改善が行われている。その中で、データベース (DB) について報告する。これまでの 2.5GeV 放射光入射器において、運転支援用 DB を運用してきたが、それらに改良を加え、新しい運用形態を可能にする J 部リニアック用 DB を構築して来たので概要を報告する。

#### 1. データ管理項目

運転支援用 DB は、専らヒューマン・インターフェース層 (HI/F) で運用されているが、HI/F 層で管理すべき加速器運転に必要なデータ項目を以下に示す。

##### 1) スティック・データ

各種の固定的なパラメータや装置データ等

##### 2) ダイナミック・データ、トレンド・データ

各種装置の変化情報等をリアルタイムに保存

##### 3) 画像等に関するデータ

波形情報、画像データ、装置写真などを保存・管理

##### 4) コマンドログ・データ

操作系から送られるコマンド等の記録

##### 5) 運転・故障記録データ、保守記録

加速器運転者 / 保守者が記録する項目

##### 6) マルチメディア情報

Web 等で運用される各種情報

ここで、2) と 4) 項に関しては、リアルタイム性が要求されるものがあり、工夫が加えられている。

#### 2. 運転対象領域と DB による抽象化

本加速器においては、デバイスレイヤーとして電子銃、真空、RF、モニター、ビームトランスポート等の各種の構成要素があるが、運転形態として以下の様に分類する事が可能である。

##### 1) デバイス、又はデバイス間連携操作

- 2) ログ・(相対)トレンド分析・支援による運転
- 3) データベースによる運転支援
- 4) アラーム系による診断支援

これらの要素は、基本的に中間レイヤーにおいて抽象化され、または DB によって装置の抽象化やバーチャル化が行われる。オペレータは、ヒューマンインターフェース層からその操作を行う事で、加速器の具体的操作・運転を行う事になる。

### 3. データベースシステム

#### 1) 全体構成

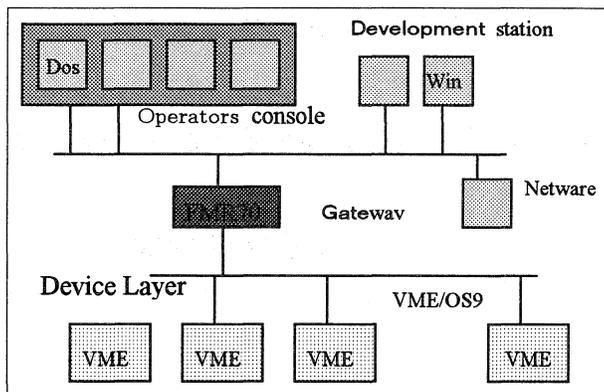
デバイスレイヤーからのデータは、VME や UNIX コンピューターを経由して、全て MS-SQL に定期的に又は情報に変化時に上がってくる仕組みになっている (図.1)。MS-SQL6.0 は、Windows NT3.51 で運用されてきたが、97年6月から Windows NT4.0 に上げ MS-SQL も V6.5 に上げて運用を開始した。

#### 2) ゲートウェイ (G/W) とのリンク

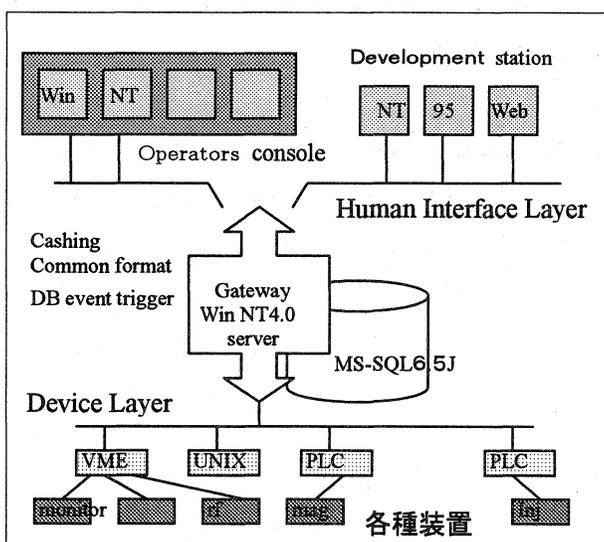
加速器操作・表示系は、クライアント・サーバー方式を歴史的に運用して来たがそのまま継承している。G/W と DB が、中間層の一つのマシンに置かれ (分散運転も可能である)、下位層のデバイスレイヤーと上位層間の抽象化レイヤーとしての機能も果たしている。つまりデバイスレイヤーの全てのデータはこれを通過し、管理され DB へのリンクも G/W によって管理されている。

#### 3) 改造の経過

放射光入射器において、建設当初はエンジンを持つ市販のデータベースは設置されて来なかった。(図.1) 一時、UNIX で SYBASE の試みが行われた事もあった。コンソール系として、クライアント管理の為の G/W をおいてから、データベース・エンジンの運用を試みたのは、Windows NT が採用されてからである。MS-SQL は、比較的安価に導入できる事や他のアプリとの連携も良い事等から導入・運用を開始し、速度改善の為の工夫等を加え、リアルタイムデータに対しても対応を行うなどの改善を加え (図.2) 現在に至っている。OODB の検討も行われたが、当時適当なものが手軽に入手する事は困難だった為、現在も RDB を継続的に運用している。



(図.1 1993年以前のDBを持たないG/W)



(図.2 G/WにDBを強化したミドルレイヤー)

#### 4) トリガー機能

本 DB では、SQL のトリガー機能を運用し、データに条件付けを行い、その条件下でトリガーを発生させ、別の分散化タスクを (ネットワークを超えたステーションでのタスクも可能) 起動する事で、データベース・ドリブ的な運用を行い、オブジェクトの分散を可能としている。

#### 5) キャッシング・システム

これまで、ダイナミックなデータに関して、DB が追従するには無理があった。ましてや PC システムの上では完全に不可能であった。しかし、最近の PC (Pentium CPU) の上で、G/W にキャッシング・システムを全てオン・メモリで付加する事により、これまでの様子を大きく塗り替える事が出来き、実用に至ったと言える。

#### 4. オブジェクト・コマンド

DB に関しては、SQL コマンドなどを発行しなくとも、クライアントが行っている通信体系でも DB とのやり取りを可能にした。式を以下に示す。

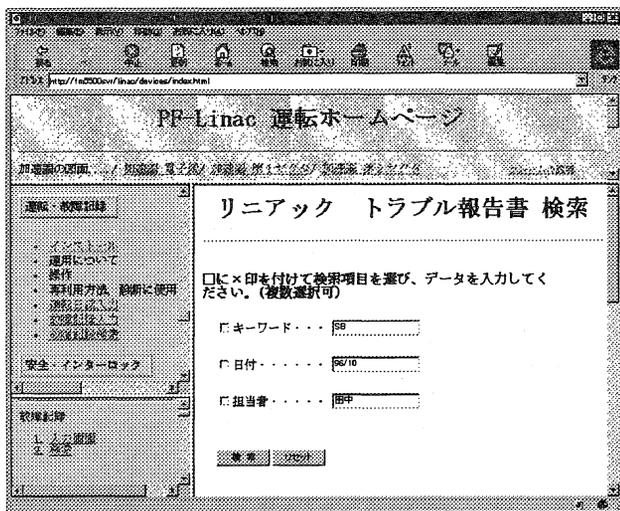
Object name. Method/property. Option <= value >

具体的例として、STC1-1.ADC.Hex と G/W に送る事でステアリングの電流値を ADC で読んで HEX 値でデータをもろう事が出来る。また設定も可能である。この式は汎用化している為、どのレベルでの運用も可能である。SQL コマンドで更に複雑な条件付けをしてプログラムから DB とやり取りする場合、ODBC 接続等で VB からの運用も行っている。

#### 5. データベース運用画面

DB を運用する為には、以下の3つの方法を提供している。最近の WWW ブラウザからの運用例を図.3 に示す。

- 1) アプリからの運用 (専ら制御卓で運用)
- 2) ブラウザからの運用 (何処からでも運用可能)
- 3) ツールからの運用 (開発・保守時などに運用)



(図.3 Web での運用例)

#### 6. 運用形態分析

DB は加速器の運転から見たとき、以下のようなフェーズでの運用が行われる。本来それらに合わせたデータ構造と運用形態を持つ事が望ましい。勿論、マルチスレッドをサポートしている事から、特色のある運用形態とリレーショナルな運用が可能である。

- 1) デザインフェーズ

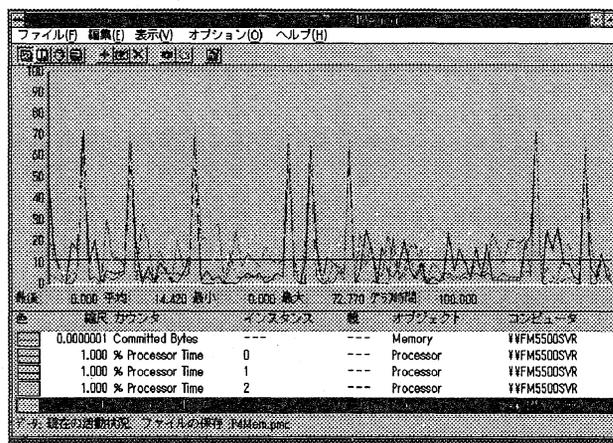
- 2) 試運転フェーズ (ビームコミッショニング)

- 3) 立ち上げ、立ち下げ
- 4) ビーム調整

- 5) 定常運転
- 6) 診断、支援
- 7) 保守の為の運転

#### 7. データベースの性能

実際に運用した時のサーバー CPU 負荷率の状況を図.4 に示す。変化情報が絶えずデバイスレイヤーから上がって来ていて、いくつかのクライアントがアクセスをしている中、DB にまとめて落している様子 (ピーク時) を示す。現実的には、現用のマシン (pentium 133Mhz X 4、128MB) において、G/W の平均負荷率は、15% である



(図.4 サーバーCPU負荷率)

#### 8. 今後と改善策

現状では、モデルベース運転等においては、まだ利用されていないのが実状である。一般ユーザーへの宣伝が必要であると考えている。ブラウジングに関しては、Web ブラウザ等が低学習コストの為、今後ますます専用画面より発展する可能性が大きい。

#### まとめ

97年6月、既存リニアックを使って負荷試験等を行った結果、幾つかの不都合点を解決し、Jリニアックに向けPCレベルのDB運用が可能である確証を得た。

#### 参考文献

- [1] Isamu Abe and Masakatsu Mutoh  
"Data base driven accelerator control using PCs"  
PCaPAC '96, DESY