
シミュレーション支援システムの開発 (日本原子力研究所)
長谷佳明

1 前回からの課題

現在、私の所属している量子シミュレーショングループでは、シミュレーション支援システムの開発を行っている。今月の課題は、ユーザの設定した環境に基づき、シミュレーションのジョブ (仕事) をデータジェネレータ (etc. 超並列計算機) に投入する部位の開発である。

2 ジョブ投入処理の実現

2.1 処理の分割と共通モジュールの作成

シミュレーションのジョブ投入までに必要な処理には、パラメタリストの取得、必要に応じて環境設定ファイルの動的生成、ファイル転送、キューイングシステムへのキューの投入などの処理があげられる。そこでこれら処理を実現するものをシミュレーションで共通して使用可能な共通部として使用可能なモジュールを作成した。これらモジュールを必要に応じ組み合わせることで、複数のシミュレーションコードに対応したシミュレーション支援システムの実現が可能となる。例えば、FTP クライアントの機能を持ったモジュールの開発を行った。しかし、他の共通部として作成する全機能を、ゼロから開発するには、あまりに膨大な時間を費やし、開発者としての負担も大きい。

そこで、インターネット上で公開され、一般にも広く用いられている Java 用の各種ライブラリを用いた。用いたライブラリを列挙する。

Jakarta-oro — Jakarta Project¹において開発されているモジュールである。Perl に似た文字列操作を可能とするライブラリである。

NetComponents — 「savarage.org」にて公開されている、Java 用プロトコルライブラリである。対応しておるプロトコルとしては、Telnet、Ftp、rpc、rshell などである。

例えば、シミュレーションに置いては、設定ファイルの雛形を用いて各ユーザによって、適時変更するという、「検索 置換」という操作が多数発生する。この操作において Jakarta-oro の提供する機能は大変便利であった。

一方、NetComponentns を用いることで、従来は外部コマンドとしてネイティブのコマンドである「rshell」な

どを用いられていた部分を Java のみを用いて実現することが可能となった。一般に外部呼出しを行ったプロセスの管理をすることは容易でなく、エラー発生時の処理も複雑になる。しかしこのライブラリを用いることで、各種プロトコルに対応したモジュールを作ることが可能となった。

2.2 エラー処理の必要性

一般に不特定多数のユーザが使用するシステムにおいては、エラー処理が重要な要素となる。ユーザの想定外の操作にも対応し、できるならばユーザの誤操作をリカバーできるエラー処理対策をシステム側に持たせる必要がある。これはいわば、ユーザサイドのエラー対策である。そしてもうひとつ、必要となるエラー対策が、システムの導入時の設定ミスによるエラー対策である。これは、いわばアドミニストレータサイドのエラーである。私が開発しているシミュレーション支援システムがインストール、そして使用される環境は常に同じでなく、それら変化に対してシステム側が柔軟に対応する必要がある。これら柔軟性を持たせるために、多量の設定ファイルが生まれる。これら設定ファイルが増えれば増えるほど設定は難しいものなり誤った設定パラメータが生じるのである。これら導入時の設定ミスを回復もしくは、インストールユーザをアシストする機能がシステムに必要となる。

2.3 エラー処理

エラーには、回復困難なエラーと回復可能なエラーの二通りが考えられる。回復可能なエラーとして代表的なものには、設定としてデフォルトを用いてもよい場合である。これら場合では、誤った設定パラメータが与えられた場合には、その旨をユーザ、もしくはシステム管理者に対して通知すると共にデフォルトの値を用いることでシステムの稼働を続けることができる。エラー通知に関しては、ユーザー一人一人設定が異なるため、どのユーザによって生じたエラーであるのか知るため、ジョブ ID (シミュレーションを一意に識別するための ID) ごとによる表示を行う機能を設けた。

3 来月への課題

今後は、得られた処理結果のデータベースへの登録、可視化、データバックアップという処理を実現する予定である。

¹Jakarta Project は、Java のアプリケーションサーバの一つである Tomcat を開発していることでも有名である。