

Gaussianポータル・サーバー構築に関する一考察

A Study on the Construction of Gaussian Portal Servers



日本アイ・ビー・エム株式会社
公共サービス事業部
公共サービスコンピテンシー
コンサルティングITアーキテクト
工学博士

大澤 暁

Gyo Ohsawa, Ph.D.

Consulting IT Architect
Public Sector Services Competency
Sector Services-Public
IBM Global Services Japan

Gaussianは、計算機センターをはじめとする多くのサイトに導入され、計算化学者だけでなく専門家以外にも活用されているアプリケーションです。ただしGaussianジョブは、入力条件によっては計算実行時間が長時間に及ぶこともあり、現状のジョブ実行方法の生産性は必ずしも高くはありません。

本論文は、グリッド技術をベースに複数のシステムをバックエンドに備えたGaussianポータル・サーバーを構築することを目的に、ポータル・システムを構築する際の課題点や今後の検討事項を論じます。さらにGaussianポータル・サーバーの設計や実装例を紹介し、今後のアプリケーション・ポータル・サーバーを構築する際のグリッド・ミドルウェア関連技術の向上につなげようとするものです。本論文で紹介するシステムにより、計算化学者は個々の計算環境の仕様を知らなくても、高価な計算機リソースを簡単かつ効率的に使用できるようになります。

なおGaussianポータル・サーバーは、独立行政法人 産業技術総合研究所グリッド研究センターにて設計されたものを基に、IBMがソフトウェア作成請負契約に基づいて作成したものです。

Gaussian has been incorporated into many sites, especially those belonging to computer centers, and it is being used not only by computer scientists but also by a large number of non-specialists. But, depending on the input parameters, execution of Gaussian jobs may take a very long time and the productivity of current job execution methods is by no means high.

In this paper I take a look at the various problems involved in constructing portal systems with the aim of constructing Gaussian portal servers equipped at the back end with several systems and based on grid technology, and I then go on to look at the matters that will have to be studied in this regard in the future. I then take a look at the design of Gaussian portal servers and actual examples of their implementation in an attempt to raise standards of grid middleware technology to be used to construct application portal servers in the future. The system introduced in this paper will make it possible for computer scientists to make easy and effective use of costly computer resources even without being aware of the specification of individual computing environments.

The Gaussian portal server described in this paper was designed by Grid Technology Research Center, National Institute of Advanced Industrial Science and Technology, and IBM implemented its software.

1. はじめに

従来、大規模なスーパーコンピュータを中心に行われることの多かった大規模科学技術計算も、最近では比較的安価なPCクラスターやワークステーション上で行われるようになってきています。その結果、さまざまな地域やサイトに散らばる計算リソースやデータベースなどを有機的・合理的に接続して協調動作させることにより、これらを有効活用するグリッド・コンピューティング技術が注目されるようになってきました[参考文献1]

グリッドとはもともとパワー・グリッド(電力網)のアナロジーであり、その応用分野や研究分野は計算グリッド、データ・グリッド、アプリケーション・グリッド、インダストリアル・グリッドなどの多方面にわたっています。計算グリッドは、複数のロケーションに分散配置されている計算リソースを統合して高効率に共有することを狙いとしています。またデータ・グリッドは、散在する大規模データベースなどの共有または共同利用することを、アプリケーション・グリッドは、ある特定のアプリケーションを複数のロケーションにて効率良く実行してサービスを提供することを第一の目的としています。

計算化学の分野でよく利用されるアプリケーション・プログラムの一つにGaussian[参考文献2]があります。Gaussianは非経験的(ab initio)および半経験的分子軌道計算を行うプログラムであり、化学分野から機械分野まで幅広く研究者に利用されています。ユーザーは簡単なキーワードと分子構造を入力することで、気相中あるいは溶液中の分子に関する物性反応など、種々の物理量を必要な精度で計算することが可能です。しかしGaussianの計算には多大な時間を要することが多く、また研究者はキーワードや分子構造などの入力値を微妙に変化させて計算を繰り返すことが少なくありません。このような状況から、複数の計算プラットフォームをバックエンドに備えたGaussianポータル・サーバーを構築して研究者の用に供することは、研究の生産性向上に大きく寄与するばかりでなく、サーバー構築に関する基本概念やミドルウェア、セキュリティ、データベース関連技術など、今後のグリッド・ミドルウェア技術開発の源となると考えられます。

本論文では、グリッドに基づいたポータル・サーバーを構築する際の考慮点を概説し、Gaussianポータル・サーバー実証システムについて報告します。次章からはグリッド・システムにおける要件とGaussianポータル・サーバーとの関連について述べ、3章ではGaussianポータル・サーバーにおける設計の概要と設計・実装例を、それを受けて4章において今後の展望を論述します。

なおポータル・サーバーの構築は、独立行政法人 産業技術総合研究所 グリッド研究センターとの共同プロジェクトによるものです。お客様が設計されたものを基に、日本アイ・ピー・エム

がソフトウェア作成請負契約に基づいて作成しました[参考文献3~8]

2. グリッド技術を用いたポータル・システム

2.1. グリッドの階層構造

グリッドは、地理的に散在している計算リソースやデータ・リソースはもとより、広義には実験装置やヒューマン・リソースなどの各種リソースを対象とし、高速ネットワークを通じて地理的なリソースの差異を意識させずに、高効率に使用できるインフラストラクチャーを提供することに意義があります。その実現に当たってグリッドは、通常次のような階層構造で考えられます。

• アプリケーション層

実際のコンピューティング・アプリケーションであり、流体・構造・量子力学はもとより、最近ではライフ・サイエンスやバイオテクノロジー系のアプリケーションも考えられます。本論文では量子化学を扱います。

• PSE / ツール層

グリッドを構築するには、次に挙げる基本ミドルウェア層が必要であり、PSE / ツール層は、アプリケーション層と基本ミドルウェア層を仲介する役割を担います。特定のアプリケーションをWebブラウザ経由でバックエンドに依頼するポータル・システムなどがこの代表例です。

• 基本ミドルウェア層

バックエンドの各種リソースと連携を取るために、リソース情報の共有や、スケジューリング、各種アロケーション、セキュリティなどを実現する層です。各地に散在するさまざまなリソースの差異は、この基本ミドルウェア層で吸収することになります。

• ハードウェア層

実際のリソースです。

これまで科学技術計算ではアプリケーション層とハードウェア層の二つの層を中心に研究が行われてきましたが、グリッドにおいてはPSE / ツール層と基本ミドルウェア層が重要になることは明らかです。

2.2. PSE / ツール層の要件

グリッド・システムを使用するユーザーの場所・手段などを考慮すると、PSE / ツール層は前節で述べたようにポータルの機能として提供されることが好ましいでしょう。主な機能を以下に挙げておきます。

• ユーザー管理

- アプリケーション・インターフェース
- 可視化

2.2.1. ユーザー管理

最近ではWebブラウザ上でユーザーIDとパスワード(またはパスフレーズ)を入力して認証するケースが増えてきていますが、グリッド・システムも例外ではありません。ただし地理的に散在するシステムを扱うことからPSE / ツール層だけではなく、基本ミドルウェア層にも関連する項目です。またユーザーは不特定の場所からアクセスする可能性があることを考慮する必要があります。

一般にユーザー管理といえば、ユーザー認証(User Authentication)を指すことが多く、簡単なものならSSLを用いてローカルのブラウザからユーザーIDとパスワードをサーバーに送付することで実現可能です。グリッドの場合にはさらに、ユーザー認可(User Authorization)が重要になります。ポータルへの入り口としてのユーザー認証が終了しても、そのユーザーがどんな権限で、どれだけの時間、どのシステムを、どのように使用できるかといったいわゆるシステム使用権限は、各システムで細かく規定されるのが一般的です(図1)。Globus Toolkitには、GSI(Grid Security Infrastructure)コンポーネントにおいてこれら認証と認可のAPIが用意されています[参考文献9]。具体的には次のようなものがあります。

- SSL/TLS上での公開鍵暗号方式の使用
- CA(認証局)によるデジタル署名
- X.509証明書によるプロキシの遠隔生成と権限の委譲

ユーザー管理において最も重要なのは、秘密鍵やプロキシなどの機密情報をだれが、どこで、どのように管理するかということです。例えばセキュアなイントラネットによる閉じたシステムと、VPN(Virtual Private Network)などを経由せずにインターネットにまたがるシステムとでは、おのずとその考え方が異なっ

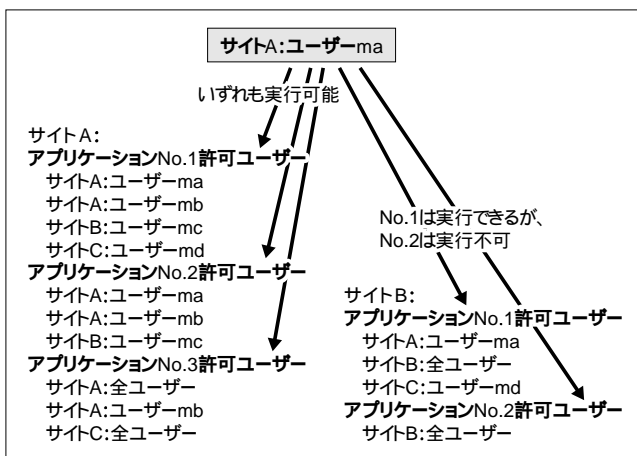


図1. システム使用権限例

てきます。Globus Toolkitでは、バックエンドの計算リソースはインターネット上にあっても、ユーザーが直接かかわるポータルの部分はイントラネット上に存在するものと仮定しているケースがあり、本論文が想定している不特定の場所からのユーザー認証/認可では問題になることがあります。この点については2.4節と3.1.1節で論述します。

2.2.2. アプリケーション・インターフェース

アプリケーション・ポータルとして考えた場合、操作性に統一感を持たせることは、ユーザーに使いやすさを提供し、ひいてはその普及に資することになります。

グリッド・システムは対象とするアプリケーションが多岐にわたるため、全般にわたって統一的なインターフェースを構築することは不可能です。しかし本論文が対象とする量子化学アプリケーションには、Gaussianのほかにも例えばGAMESS[参考文献10]やNWChem[参考文献11]などの有名なソフトウェアがあり、これらに対しても統一した入力指示・結果表示などが行えることが望まれます。

2.2.3. 可視化

計算結果の可視化も重要です。Tele-immersion[参考文献12]をはじめとする各種可視化の技術革新は進んでいますが、そのほとんどはローカルに閉じたシステムを対象としていました。画像の可視化には膨大な帯域を必要とするからです。しかし昨今のブロードバンド化に伴い、インターネット経由でも高速に画像を配信できるインフラストラクチャーが整いつつあり、グリッド・システムにとっても今後重要な機能の一つとなると考えられます。

2.3. 基本ミドルウェア層の要件

基本ミドルウェア層はいわばグリッド・システムの根幹を成す部分でもあり、主に次の機能が必要です。

- セキュリティー
- 並列実行ライブラリー
- システム管理
- データ / 課金管理
- リソース管理

2.3.1. セキュリティー

ユーザー管理としてのセキュリティに関しては2.2.1節で述べましたが、そのほかにも考慮すべきポイントが幾つか存在します。

- ジョブ実行結果やデータベースに保存されている貴重な情報をイントラネット / インターネットを問わずにセキュアに配信する仕組みが必要です。内容によっては認証が必要なもの

や課金につながるもの、データの公開にレベルを設ける必要のあるものもあります*1。

- 一般にセキュリティを確保しようとする、パフォーマンスが犠牲になる可能性があります。従ってシステムを構築する際には、総合的な観点からどこまでセキュリティを確保すべきかを検討する必要があります。
- セキュリティの技術は日進月歩です。よってセキュリティ関連ルーチンをパーツ化し、後から変更できるように設計することが重要です。

*1: 例えば、イントラネットにはすべての情報を公開し、インターネットに対してはデータの公開を制限する場合などが該当します。

2.3.2. 並列実行ライブラリー

ジョブが複数のCPUを用いて並列実行される場合には、何らかの通信ライブラリーを用いることになります。科学技術計算ではMPI[参考文献13]を使用することが少なくありませんが、グリッドに対応したMPICH-G2[参考文献14]なども登場してきています。あるサイトの閉じたシステムで実行される場合には、MPICH-G2に限らずにそのサイトで用意されている並列実行ライブラリーが使用できます*2。

*2: 例えばGaussianでは並列実行ライブラリーとしてLindaを使用します。

2.3.3. システム管理

グリッド・システム全体の管理という観点では、次のポイントがあります。

- 複数のシステム間における管理プロトコルの確立
- 管理に伴うフローの最適化
- RASおよびパフォーマンス

Globus Toolkitでは各サイトのシステムに関する情報を共有・管理する仕組みが備わっており、この機能を使用すれば問題はありません。しかし今後はグリッド・システムそのものが複雑になる可能性があり、例えばシステムおよび管理の階層構造も考慮する必要があります。その場合には、Globus Toolkitが管理するシステムの子システム・孫システムなどどのように情報連携し、効率良く管理していくかを考慮しなければなりません。

またシステム管理のもう一つの要素としてI/Oがあります。

Globus Toolkitでは、NexusやGlobus-ioなどが用意されていますが、階層構造の場合には上記検討項目がそのまま当てはまります。

2.3.4. データ / 課金管理

スケーラビリティおよびパフォーマンスが重要なポイントになります。データ管理に関しては、分散データベースのレプリカ運

用と、複数サイトにおけるデータ転送がキーとなります。サイトが異なる場合にはデータ転送をセキュアに行う必要がありますが、Globus ToolkitにはGridFTPプロトコルが用意されており、これを使って解決することになるでしょう。しかしGridFTPはパフォーマンスの問題を抱えており、セキュリティとスピードのトレードオフとなっています。両者を考慮した新しいプロトコルの開発が待たれるところですが、例えば分散データベースのレプリカ・アルゴリズムを使用したり、SSL上で独自にデータ転送 / コピーの仕組みを構築することも有効です。

課金管理に関しては研究途上の段階であり、明確なライブラリーやツールはほとんど存在しません。複数のサイトを接続するグリッド・システムであることが課金管理を難しくしており、以下の点を総合的に判断する必要があります。

- 各サイトにおける課金ポリシーの相違点の明確化、およびその相違点を吸収する場。
- 課金単位(サイト全体 / グループ単位 / 個人)、複数のサイト間でジョブの相互実行をした場合には、通常、サイト間で課金が発生します。この費用の管理方法として、サイト合計による累積でよいのか、個人明細が必要なのかなどによりデータベースの設計やその管理方法が異なってきます。
- ジョブが長期にわたり実行される場合の課金のタイミング*3。

Globus Toolkitにはこうしたツールが用意されていないので、グリッド・システムを構築する際には、それぞれを設計・開発する必要があります。

*3: ホテルに長期滞在する場合に、例えば毎週会計を求められるケースと、毎月あるいはチェックアウトのときのみでよいケースがあるのと同じです。

2.3.5. リソース管理

リソースとは、グリッド・システムに含まれるIPアドレスやシステム管理者名にはじまって、オペレーティング・システムの種別、システム特性、負荷状況、実行可能アプリケーション情報、課金、ネットワーク特性など多種多様なものがあり、これらを効率良く統合的に管理する必要があります。Globus Toolkitは、LDAPを用いてリソースを管理する仕組みが構築されています。グリッド・システムを構築する際のポイントを以下に挙げておきます。

- ダイナミック・アロケーション

ジョブをどのサイトのどのシステムに投入するかを、動的かつ効率的に決定するアルゴリズムが重要です。これは分散システムにおけるジョブ投入アルゴリズムと同じであり、ギャング・スケジューリングなどの手法も考慮する必要があります。しかしサイトが異なる場合には、ジョブを投入するために必要なプログラムやデータの転送*4、課金ポリシーなどを含めて総合的に検討する必要があります。

• 各サイト内のジョブ管理

Globus Toolkitには、2.3.3節で述べたようにサイト単位での情報交換の仕組みは備わっていますが、例えばあるサイトに大規模なクラスター・システムがあった場合、そのクラスターにおけるジョブ管理はクラスター内で行わなければなりません。例えばRS/6000® SP™システムではAIX® LoadLeveler® [参考文献15]やLSF [参考文献16]などのバッチ・キューイング・システムが、LinuxクラスターではPBSPro® [参考文献17]などが使用可能です。しかしこのジョブ管理情報はダイナミック・アロケーションの観点からサイトを越えて管理される必要があり、それぞれのシステムにおける情報の整合性やデータ・フォーマットの相違点の吸収などを考慮しなければなりません。

* 4: 実行するジョブが汎用アプリケーションの場合、サイトによってはプログラムが既に導入されているケースもあります。

2.4. Gaussianポータル・サーバーの要件

本論文が対象としているポータル・サーバーは、1章で述べたように量子化学の分野で重要なGaussianアプリケーションを高速かつ高効率に実行できるグリッド・システムです。最終的には図2に示すシステムを想定しており、サーバーの要件をまとめると次のようになります。

- 一つのサイトにおけるバックエンドの計算サーバーも、種類・用途に応じて複数で構成できること。
- 複数のサイトを相互に接続できること。
- ユーザーはインターネット経由でサイト外部よりセキュアに使用でき、またWebインターフェースが使用できること。
- 過去に実行したジョブ情報を計算結果を含めてデータベースに蓄積し、同じ入力条件の実行要求に対して、データベースに登録してある結果情報を即時表示できること。これによりシステムの効率的な利用とターンアラウンド・タイムの最少化を図ります。
- 計算情報は機密性にランク付けを行い、ユーザーごとにアクセスおよび閲覧などの制御ができること。

上記システムを構築するに当たり、現状ではグリッド・システムの基本ミドルウェア層におけるデファクト・スタンダードであるGlobus Toolkitを使用するのが、将来的にも他サイトへの拡張性にも富んでいることから有効だと考えられます。しかしGlobus Toolkitに用意されていなかったり、想定されていない以下の機能が問題となりました。

- ユーザーがサイト外部よりセキュアに利用する仕組みがGlobus Toolkitにはありません。グリッドではプロキシ・クレデンシャルにより認証と認可を行うことは2.2.1節で述べた通りです

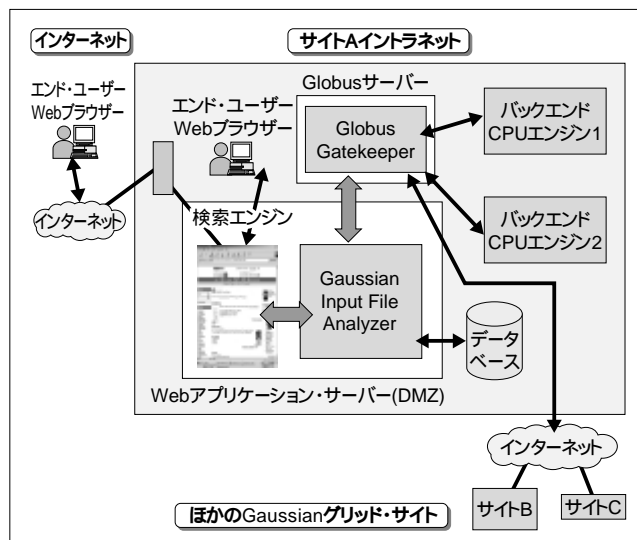


図2. Gaussianポータル・サーバーの概要

が、これらを作成するための秘密鍵を含めた機密情報をどこに格納するかという問題が発生します。プロキシ・クレデンシャルを安全な第三者サーバーに保持し、必要に応じてユーザーIDとパスフレーズを用いて取得するMy Proxy方式は、GlobusのGSIで保護されるばかりでなく秘密鍵の転送もないセキュアな方式です [参考文献18]。一方、外部よりアクセスする場合に、第三者サーバーをどこに置くかという問題があります。なぜならDMZ上のマシンは外部から不正攻撃される可能性があり、プロキシ情報を盗まれる恐れがあるからです。インターネットなどの非セキュア・エリアに存在するユーザーが、本来ならセキュア・エリアに置くべき情報をどのように扱うのかという問題であり、ファイアウォールの通常のセキュリティ・ポリシーでは外部と内部との直接接続は許されていないため、何らかの解決が必要となります。

- データベースはイントラネットに存在し、必要に応じてGridFTPプロトコルにてデータ転送を行うことができます。またユーザーに対するデータの表示やダウンロード・サービスはSSL上にて実施すればインターネット部分におけるセキュリティは確保できます。しかしイントラネット部分においてはセキュリティを確保できません。Gaussianに関しては情報そのものが論文の「ネタ」ともなる研究者にとって貴重な財産であり、イントラネット部分においてもセキュリティの確保が課題となります。
- アプリケーション・インターフェース、可視化、データ / 課金管理、リソース管理など、2.2節および2.3節で述べてきた項目のうちGlobus Toolkitの範囲外となる部分については別途ツールを検討するか、または開発を行う必要があります。

3. Gaussianポータル・サーバーの概要

前節で述べた要件に基づき、お客様にてGaussianポータル・サーバーの設計が行われました。ここではその概要を論述するとともに、実装例を紹介します。

3.1. 設計概要

3.1.1. ユーザー管理と認証・認可

ユーザーは、Webブラウザ経由で本システムにサインオンします。従ってDMZ領域にWebアプリケーション・サーバーを用意し、httpsプロトコルを用いてユーザー認証を実施します。

Gaussianジョブを実行するバックエンド計算エンジンは複数システムを想定します。グリッド・システムですから、その実行にはプロキシ・クレデンシャルを生成する必要があります。セキュリティを確保しながらこれを生成するには、次の方法があります。

- セキュリティーを高めるためにユーザーごとにプロキシを生成させる場合には、ユーザーが独自に生成したプロキシ情報を、何らかの方法でイントラネット側のセキュア・エリアに設置するサーバーに通知する仕組みを構築します。
- 個人ごとのプロキシ生成は行わずに、サインオンを済ませたユーザーに対して、ある特定IDのプロキシをシステム内部で自動的に生成し、そのプロキシでグリッド・システムを使用します。

前者は、グリッド・システムを使用する際に各ユーザーがそれぞれ別のプロキシ・クレデンシャルを使用するため、セキュリティがより高まります。またクレデンシャルを用いて他サイトの計算エンジンを使用する場合も、クレデンシャル内に個人情報(DN: Distinguished Name)が含まれているため、他サイトにおける認可や課金計算も比較的容易です。一方、後者はDMZなどを意識する必要がなく、実装が容易な反面、クレデンシャル内の個人情報だけでは実際のユーザーを特定できないことから、認可や課金計算ができるようにユーザー名を付加して他サイトに通知する仕組みを構築しなければなりません。

グリッド・システム本来の概念からすれば前者を採用すべきですが、実装が容易ではないことと、将来的にはGlobus Toolkitの拡張が予想されることから、今回は後者を採用することにしました。

3.1.2. アプリケーション・インターフェース

本論文ではGaussianのみを想定していますが、今後の拡張を考慮して次の機能を有する画面を設計することとしました。

- 入力条件ファイルの指定およびユーザー端末からのアップロード機能

- チェックポイント・ファイルの有無の指定、および「あり」の場合のファイルの指定、アップロード機能
- 過去の計算実績の中に、同一の入力条件があるか否かを検索する機能、および検索結果を表示する機能
- 分子構造を3次元表示する機能
- ジョブの実行指定機能およびジョブ状態管理表示の機能
- ジョブの結果表示、出力ファイルとチェックポイント・ファイルのダウンロード機能

将来的にGAMESSやNWChemなどのアプリケーションを組み込む場合でも、上記のインターフェースはそのまま流用できます。ただし条件指定などのアプリケーション独自の機能の一部は、入力後にシステム内で自動的な調整が必要となります。

3.1.3. 可視化

3.1.2節で述べたように、分子構造を3次元表示する機能を構築しています。なお過去の計算実績を検索した場合には、複数の事例を同時表示して比較することも研究者には欠かせないため、可視化画面は別ウィンドウを起動して表示させることとしました。インターフェースの観点から、このように実際の利用者の立場で設計することも重要です。

3.1.4. 並列実行ライブラリー

Gaussianを複数のCPUを用いて並列実行する場合にLindaを使用しますが、残念ながらGlobus Toolkit上で動作させることはできません。Gaussianの並列実行は同一サイトにあるSMPマシンや分散クラスターなどの閉じた環境でのみ行うことができます。もっともGaussianでは通信が多数発生するため、地理的に散在しているマシン同士で並列実行することは現実的ではありません。

3.1.5. システム / リソース管理

本サーバーは同一サイト内において、RS/6000ワークステーション8台を有するシステムと、108台のxSeries™から成るLinuxクラスター・システムの二つのシステムを想定しています。また近い将来には、別サイトにおいて複数のワークステーションなどを用いた同様のシステムを構築し、グリッド・システムとして接続する予定です(図3)。

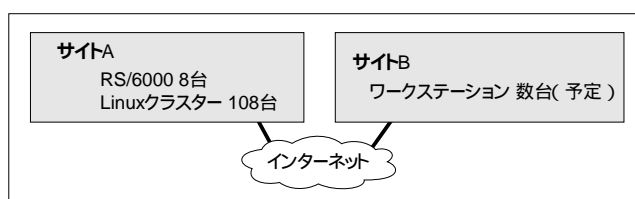


図3. Gaussianポータル・サーバーのCPUエンジン

RS/6000のPowerPC Architecture™とxSeriesではCPUの性能が異なるため、システムの負荷状況を見ながら動的にジョブ・アサインをすることは容易ではありません。一般的なCPUの性能比を基に係数を乗じて空き具合の尺度とすることが考えられますが、今回は比較的短時間に終了するジョブと長時間ジョブとでクラスを分け、投入するシステムをクラスで類別することにしました。

またRS/6000システムではジョブ投入制御にAIX LoadLevelerを、xSeriesではPBSを使用する予定です。

3.1.6. データ / 課金管理

バックエンドのCPUエンジンを統括するGlobusサーバーとWebアプリケーション・サーバーとの間はGridFTPを用いることでセキュアなデータ転送を行えます(図2)。一方、データベースに含まれている各種情報やイントラネット部分については2.4節にて述べたように機密性の問題が残りますが、以下の理由から今回は特に考慮を行わず、今後の課題としました。

- 当該部分の機密性を高めるには、データベース内の情報を含めて暗号化を実施し、暗号化された状態でデータを送受信する必要があり、開発工数が膨大になること。また技術的

に比較的完成されたものであり、必要に応じてアドオンが可能であること。

- イン트라ネットにおいては既に多数のGaussianジョブが実行され、計算結果に関するデータも平文の状態ネットワーク上を流れていること。

セキュリティーを含むデータ管理はグリッド・システムにおいて重要な項目の一つとなりますが、既に運用されているシステムとのバランスを考慮し、コスト・パフォーマンスの良いシステム構築を心掛けるべきです。

データベースに蓄積する計算結果ファイルについては、アクセスおよび公開に関するランク付けを実施し、インターネットなどの外部公開できるものとできないものを区別できるようにしました。

課金管理に関しては、現在は同一サイトのみであることから、ジョブの実行CPU時間のみを対象とし、実行ユーザーごとに当該情報を取得し、課金を含むユーザー管理データベース上に蓄積することとしました。また実際にジョブの実行を行わずに、データベースの過去実績ファイルの検索結果をもって計算結果を得た場合にも相当する課金を行う仕組みとしています。

3.1.7. その他

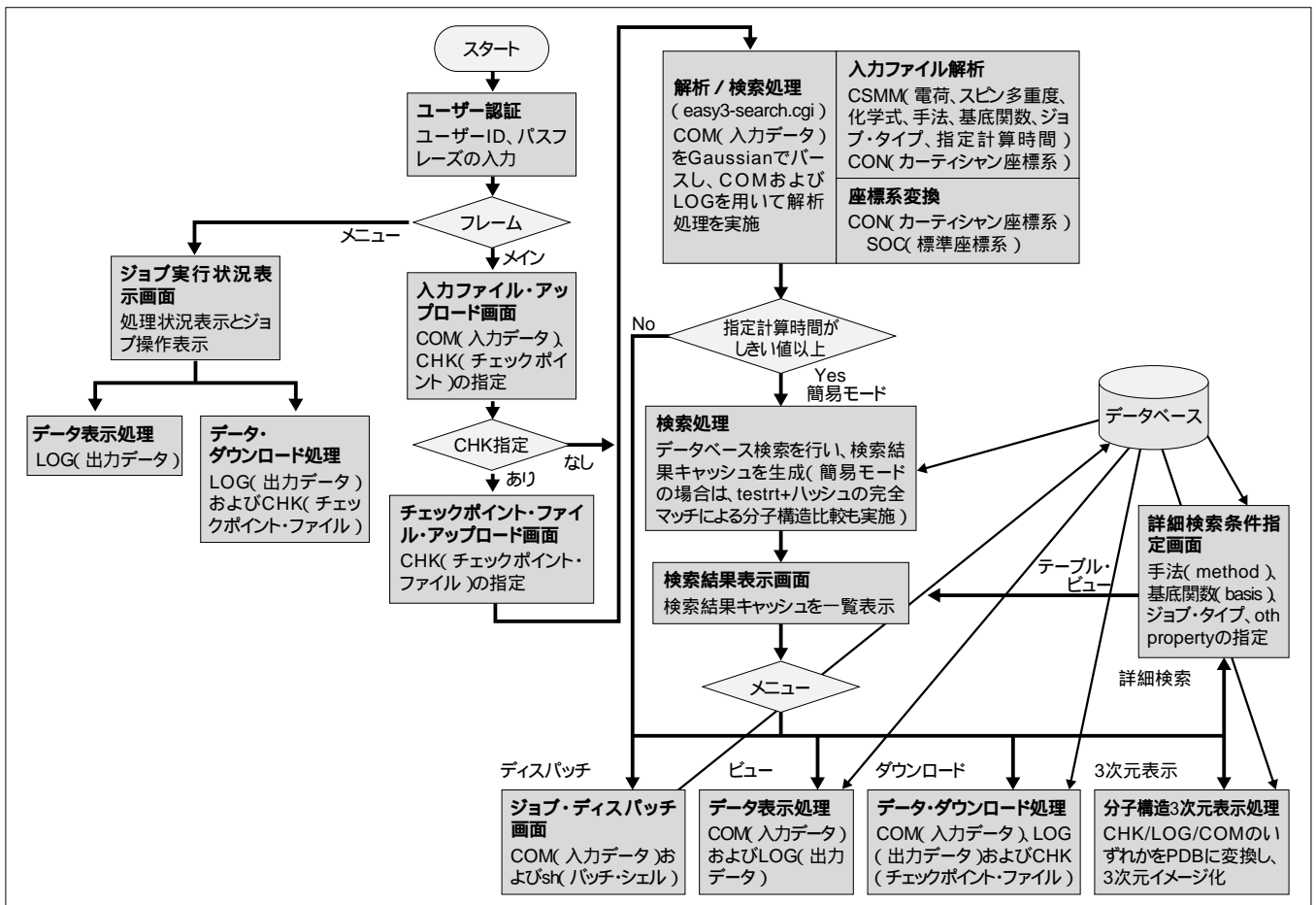


図4. Gaussianポータル・サーバーの処理フロー (出典: 参考文献6)

チェックポイント・ファイルを扱うシステムに特有の問題として、データ・ファイルのフォーマットに注意を払う必要があります。Intel系のリトル・インディアン方式とPowerPCなどのビッグ・インディアン方式の違いをシステム内で吸収させる必要があります。一般に、チェックポイント・ファイルのレコード単位のフォーマットが明確に定義されていれば変換は可能ですが、現実にはそれほど容易ではありません。さらに同一ファイル内に整数型・実数型・倍精度などの複数の型が混在する場合は変換が不可能です。幸いGaussianにおいては、バイナリー・ファイルの精度を落とすことなくテキスト・ファイルに変換するchkmoveツールが用意されているため、データベース上にはテキスト・ファイルの状態で保持させることとしました。

3.2. 設計例

前節で述べた設計概要を基にまとめた処理フローを図4に示します。

データベース検索に関しては、第1段階として入力ファイルで指定された情報をハッシュ化し、完全マッチにより検索を実施しています(図の中央、「検索処理」の部分)。しかし実際には、わずかに入力条件の異なるものや、計算手法の種類や座標系の記述方法の違いなどがあり、完全マッチで検索できないケー

スが少なからず存在します。このためGaussianポータルにおいては、以下の項目を利用した詳細検索を可能としています(図の右下方、「詳細検索条件指定画面」の部分)。

- 計算手法
- 基底関数の種類
- ジョブ・タイプ
- そのほかの特性

また図5に検索処理に関する詳細フローを示します。

3.3. 実装例

前節で述べたポータル・サーバーを、以下の環境を用いて構築しており、ほぼ開発が終了し、テスト段階に入っています。

- Webアプリケーション・サーバー
 - ハードウェア
 - xSeries
 - ソフトウェア
 - RedHat Linux[参考文献19]
 - Apache[参考文献20]
 - Globus Toolkit
 - OpenSSL[参考文献21]
 - Java™2, Perl, C

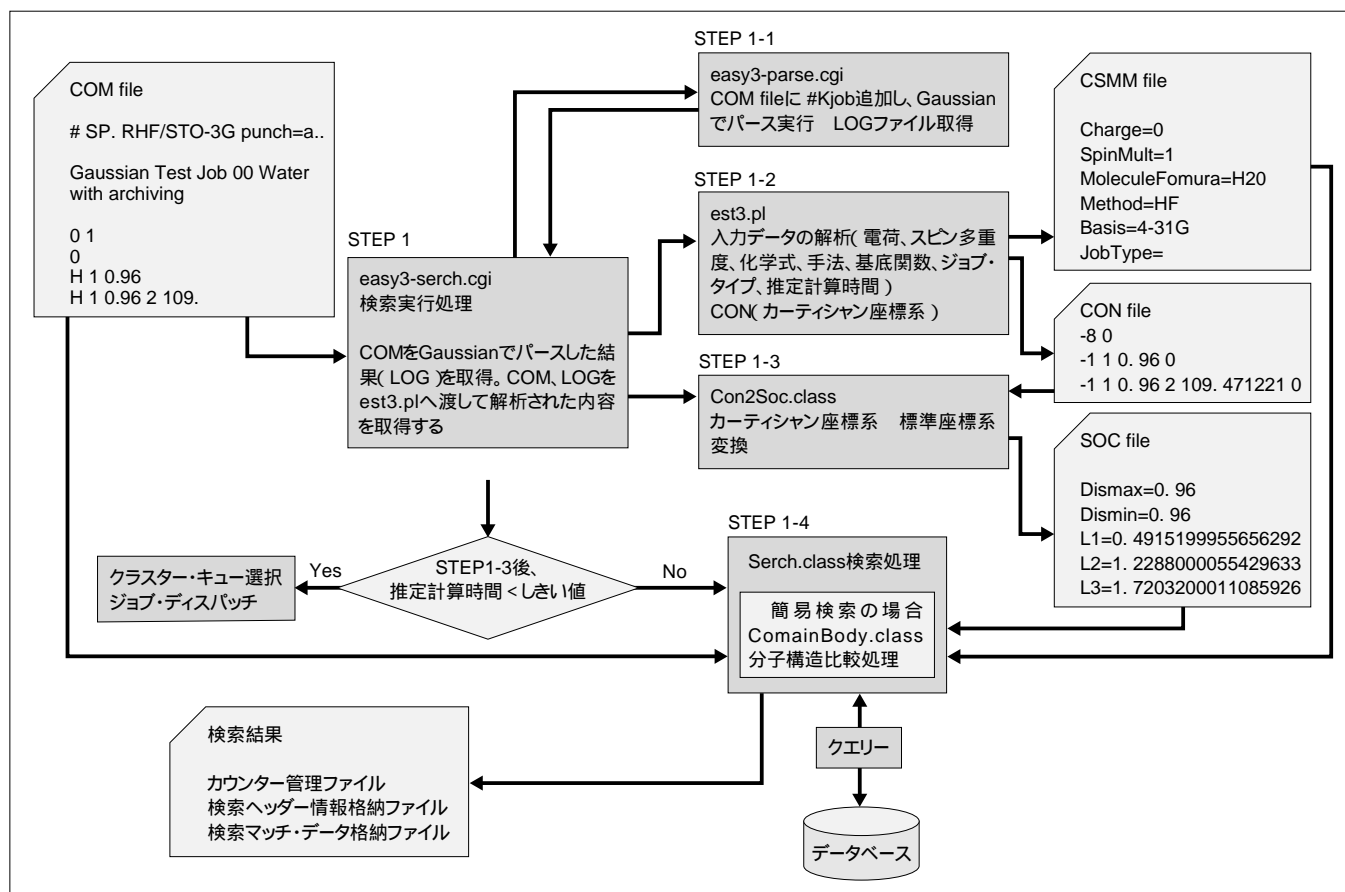


図5. 検索処理 詳細フロー

No	Job ID	Methods	Goals	Notes	Resources
1	422E-15	GCFM	3-210	MP TEST 3-210 GCFM Gaussian Test Job 03 / STEEPEST DESCENT SCF 3-210/0-210 ETHYLENE PBCENTERV Oracle / nasa000.com	Input: View / Download Output: View / Download CHECK: <input type="checkbox"/> 3D View: <input type="checkbox"/>
2	422E-08	GCFM	3-210	MP TEST 3-210 GCFM Gaussian Test Job 03 Carnation-4 alpha / STEEPEST DESCENT SCF 3-210/0-210 ETHYLENE PBCENTERV Oracle / nasa000.com	Input: View / Download Output: Nothing CHECK: <input type="checkbox"/> 3D View: <input type="checkbox"/>
3	422E-08	GCFM	3-210	MP TEST 3-210 GCFM Gaussian Test Job 03 Carnation-3 alpha / STEEPEST DESCENT SCF 3-210/0-210 ETHYLENE PBCENTERV Oracle / nasa000.com	Input: View / Download Output: Nothing CHECK: <input type="checkbox"/> 3D View: <input type="checkbox"/>
4	422E-15	GCFM	3-210	MP TEST 3-210 GCFM Gaussian Test Job 03 / STEEPEST DESCENT SCF 3-210/0-210 ETHYLENE PBCENTERV Oracle / nasa000.com	Input: View / Download Output: Nothing CHECK: <input type="checkbox"/> 3D View: <input type="checkbox"/>

図6. 検索結果表示例 (出典: 参考文献 6)

- Globusサーバー
 - ハードウェア
 - xSeries
 - ソフトウェア
 - RedHat Linux
 - Apache
 - Globus Toolkit
 - OpenSSL
 - Java2、Perl、C
- データベース・サーバー
 - ハードウェア
 - Compaq Alphaサーバー
 - ソフトウェア
 - Tru64 UNIX®
 - Oracle
- Gaussian CPUエンジン1
 - ハードウェア
 - xSeries 108台
 - ソフトウェア
 - RedHat Linux
 - Globus Toolkit
 - OpenPBS
 - Gaussian98
- Gaussian CPUエンジン2
 - ハードウェア
 - RS/6000 8台
 - ソフトウェア
 - AIX5L
 - Globus Toolkit
 - AIX LoadLeveler

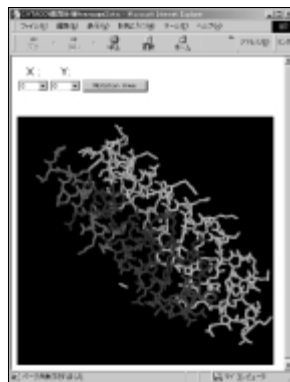


図7. 分子構造3次元表示例 (出典: 参考文献 6)

Gaussian98

図6に検索結果の表示例を、図7に3次元表示例を示します。3次元表示では、X軸およびY軸方向に回転して再表示できるように回転度数を入力できるようになっています。

4. 今後の展望

量子化学におけるGaussian計算は非常に需要が高く、フレンドリーなポータル・システムを構築することは研究の生産性向上に大きく寄与します。特に本論文で述べたグリッド・システムは次の特長を有します。

- Gaussianの入力条件ファイルを基に、過去の計算実績データベースを検索し、同一条件のものがあれば計算を実際に行わずに計算結果を即時表示できます。
- Globus Toolkitを用いたグリッド・インフラストラクチャー上に構築しています。またSSLを用いて、外部からもセキュアに使用できます。
- 現在は1サイトのみでの運用ですが、他サイトへの拡張が容易にできる設計となっています。
- 異質アーキテクチャーである二つの計算エンジンを有し、予想計算実行時間などの情報を基にシステムが自動的にジョブの振り分けを行っています。

現段階では一つのサイトのみでの運用ですから、グリッド・システムというよりは単なるポータル・システムの観がありますが、基本的なグリッドの機能は有しています。2003年の早い時期にはサイトがもう一つ追加される予定もあるとのことであり、その時点における課題と今後の展望を次のように考えています。

- サイト間での課金関連を含む運用方法の確立
 - サイト間での課金に関してはある意味「決め事」であり、今回もCPU実行時間をベースに課金する予定です。しかしこれまでは述べてきませんでした。ベンダー・アプリケーション・ソフトウェアによってはライセンス上の問題が発生する可能性があります。すなわちあるサイトで有効なライセンスを有している場合に、他サイトのユーザーが無条件にそのソフトウェアを使用できるのかという問題です*5。これはグリッド・システムにおいて今後議論され、また解決されるべき問題の一つです。
- セキュリティ
 - ユーザーのサインオンに関しては、現在は最低限の認証を実施しています。特にジョブの実行時間が長期間にわたる場合、途中でクレデンシャルの有効期間が超過したり、いったんサイン

オフし、後でサインオンし直すといったケースも考えられます。またキオスク端末や外出先などからジョブの実行状況を確認したくなることもあるでしょう。プロキシを作成するための秘密鍵はユーザー自身の責任において管理することが最も好ましいのですが、多種多様な使用状況を想定すると、個人所有のPCなどに秘密鍵を保持することは可能でも、他人のPCを借用する場合やキオスク端末ではプロキシの作成ができなくなります。この場合には、使用可能なアプリケーションやコマンドを限定するなどの運用が必要になるでしょう。

- アプリケーション・ソフトウェアの^{いんべい}隠蔽

一般に、ユーザーは「何を行いたいか」を指定するべきであり、「何を用いてどのように行うか」はシステムが自動的に判断することがアプリケーション・ポータル・サーバーの究極の目標と考えられます。量子化学においてもGaussian、GAMESS、NWChemのどれを使用するかをユーザーが明示しなくても、システム内部で自動的に判断する必要があります。ソフトウェアにより機能の差や得手 / 不得手があるので、Gaussianを中心に置き、ある特定作業を行いたいときはポータル画面に選択ボタンを設けてスイッチするといった運用が有効です。

- 種類の異なるアプリケーションへの対応

さまざまなアプリケーションに対応可能な汎用的なグリッド・システムを簡単に構築できるミドルウェアを整備することは、今後の研究開発にとって非常に有用です。上記の量子化学アプリケーションに加えて、今後は流体 / 構造力学などの科学技術計算にも対応できるように拡張することになるでしょう。現在ではWeb画面の遷移を含めてアプリケーションの処理フローは、アプリケーションごとに作成しているのが現実です。もしこれが汎用的に作成できるようになれば、アプリケーションのアドオンを容易に行うことができ、グリッド技術の発展に寄与するだけでなく計算化学者にとっても朗報となります。

*5: 本論文が対象としているGaussianについては両サイトとも有効なライセンスを有しており、一部の運用上の制限を除いて問題はありません。

5. おわりに

本論文では、グリッド技術を用いてユーザーに簡便なWebインターフェースを適用し、Gaussianアプリケーションを簡単に実行できるポータル・サーバーについて論じました。また実際の設計例と実装例を紹介しました。本システムの開発はほぼ終了し、今後は運用を通じて改良を重ねていく予定です。

グリッドは最近になって急速に世の中に広まり、各メーカー / 研究者がさまざまな研究・開発を行っています。応用範囲が広

いだけに、科学技術計算を対象としたグリッド・システムは、グリッド全体から見たときにそれほど大きな比重を占めないかもしれませんが、科学の発展にとって大きな意義があり、インターネットを介したセキュリティーやユーザー親和性など、今後解決しなければならない項目も多数存在します。

本システムを複数のサイトへ展開し、またバックエンドにて処理を行うアプリケーションの種類や数を増やすことは、今後のこの分野における技術革新に大きな貢献となります。

(ページ数および表記上の観点から、著者の了解を得て編集部にて手を入れてあります)

[参考文献]

- [1] Ian Foster, Carl Kesselman. The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufmann, 1998.
- [2] Gaussian Inc.: <http://www.gaussian.com/>
- [3] 関口 智嗣「TACC Quantum Chemistry GRID」『日本化学会情報化学部会誌』Vol. 20, 2002年
- [4] 西川 武志・長嶋 雲兵・関口 智嗣「Quantum Chemistry GRID/Gaussian Portalの構築」『情報処理学会HPC研究会第90回報告』, 2002年5月
- [5] 産業技術総合研究所 グリッド研究センター:
<http://unit.aist.go.jp/grid/QCgrid/>
- [6] 西川 武志・長嶋雲兵・関口智嗣「Quantum Chemistry GRID/Gaussian Portal Phase2」『情報処理学会HPC研究会第92回報告』, 2002年10月
- [7] 西川 武志「TACC Quantum Chemistry GRID」『計算工学』Vol.7 No.4, 2002年
- [8] 西川 武志・長嶋 雲兵・関口 智嗣「計算システムおよび方法」特願2002-102679, 2002年4月
- [9] GLOBUS project: <http://www.globus.org/>
- [10] Ames Laboratory/Iowa State University:
<http://www.msg.ameslab.gov/GAMESS/>
- [11] Environmental Molecular Sciences Laboratory at Pacific Northwest National Laboratory: <http://www.emsl.pnl.gov:2080/docs/nwchem/>
- [12] National Tele-immersion Initiative:
<http://www.advanced.org/teleimmersion.html>
- [13] The MPI (Message Passing Interface) Forum:
<http://www.mpi-forum.org/>
- [14] Mathematics and Computer Science Division at Argonne National Laboratory: <http://www.hpcclab.niu.edu/mpl/>
- [15] IBM Corporation. IBM LoadLeveler for AIX 5L:
Using and Administering V3.1. SA22-7881-00
- [16] Platform Computing Inc.: <http://www.platform.com/products/wm/LSF/>
- [17] Veridian Systems: <http://www.openpbs.org/>
- [18] 中田 秀基・斎藤 真幸・鈴村 豊太郎・田中 良夫・松岡 聡・関口 智嗣「Gridポータル構築ツールキットNinf-Portal」『情報処理学会JSP2002』, 2002年5月
- [19] Red Hat Inc.: <http://www.redhat.com/>
- [20] The Apache Software Foundation: <http://www.apache.org/>
- [21] The OpenSSL Project: <http://www.openssl.org/>