

Grid 上でアプリケーションの簡便な利用を可能にする Grid ポータルシステムの構築

Development of a Grid Portal Construction Toolkit

白砂 哲¹⁾

Satoshi Shirasuna

1) 東京工業大学 (〒152-8552 東京都目黒区大岡山 2-12-1 西 7 号館 1F/2F E-mail: sirasuna@is.titech.ac.jp)

ABSTRACT.

As Grid technologies get practical, a number of Grid Portals have been constructed and used in various fields to offer user friendly interfaces for Grid resources. Along with that, several toolkits to generate Grid portals have been developed in order to reduce the burden of portal developers. However, even with the aid of those toolkits, portal developers still have to install target applications on each node. In addition to that, it is necessary to keep application data up to date for some applications, especially applications in bioinformatics field. In order to automate these tasks, we are implementing a toolkit, PCT4G, which automatic application installation and data management. Also, users can construct Grid Portals of their own applications on the fly through Web interfaces of PCT4G.

1 背景

高速なネットワークの普及により、広域に分散した各種資源を集約的に利用して大規模計算を行う機構グリッドが注目を集めている。グリッドは一般に多様な管理組織に属した多数の計算資源によって構成されているため、使用には煩雑さが伴い、専門の知識を有しない一般の研究者が直接使用することは難しい。このため、グリッドの複雑さを隠蔽し豊富な計算力を簡便に利用するためのシステム、グリッドポータルが注目されている。グリッドポータルはさまざまな分野で実用されており、また、GridPort Toolkit[6] や Grid Portal Development Kit (GPDKit)[1] に代表されるグリッドポータルを構築を支援するツールキットもいくつか開発されている。

グリッドポータルとは、特別なソフトウェアサポートを持たないクライアントから、グリッド上のアプリケーションの使用を可能にするシステムである。特別なソフトウェアを使用しないという要請から、インタフェースには Web を用い、一般的な Web ブラウザからの使用を可能にする。ユーザは、Web ブラウザでポータルにログインし、利用するアプリケーションを選択し、引数やデータなどを入力する。グリッドポータルには、通常の Web アプリケーションの機能に加えて、グリッド特有の機能を実装しなければならない。

しかし、既存のツールキットは、グリッドの複雑さを一般ユーザから隠蔽しているが、グリッドポータル構築者への支援は十分ではない。例えば、アプリケーションのグリッド上の各計算機へのインストール、設定はポータル構築者が個別に行う必要がある。さらに、BLAST に代表されるバイオインフォマティクス分野のアプリケーションの多くは、最新に保たれたデータを必要とすることが多い。そのため、アプリケーションで利用するデータの配布や更新のコストもポータル構築者の大きな負担となっている。

2 目的

本プロジェクトの目的は、潜在的に多くの計算力を有するグリッド資源を有効に活用するための支援ソフトウェアを開発することである。そこで、グリッド資源の簡便な利用手段を提供するグリッドポータルに注目し、グリッドポータルを構築、保守する際のコストの削減をするソフトウェアを開発した。開発したソフトウェアであるポータル構築ツールキット”Portal Construction Toolkit for the Grid (PCT4G)”は、アプリケーションのグリッド資源上へのインストール、アプリケーションに必要なデータの配布、更新をサポートする。また、ポータル作成ポータルと呼ぶ Web インタフェースをグリッド資源上に提供することで、一般ユーザが自分専用のポータルを即席に作成できるようにする。

3 PCT4G の概要

”Portal Construction Toolkit for the Grid (PCT4G)”はグリッド上でアプリケーションを簡便に利用する枠組であるグリッドポータルを構築するためのツールキットである。PCT4G では、アプリケーションのインストールの自動化を行い、ポータル構築に伴う多数のホストへのアプリケーションの負荷を軽減する。ポータル構築者は設定ファイルを記述するだけで、グリッド上の利用する資源にアプリケーションをインストールすることができる。また、バイオインフォマティクス分野で特徴的な、頻繁に更新される多量のデータを利用するアプリケーションに対応するため、データマネージメント機能を有する。データマネージメント機能とは、アプリケーションに必要なデータの自動ダウンロード、グリッド資源上への配布を行うことである。データの初期配布後は、定期的にデータの更新をチェックし、更新があった場合には、そのデータをダウンロードし、グリッド資源上のデータを更新する。バックエンドでは、グリッド特有の機能のサポートを行うため、ポータル構築者はグリッド技術を理解する必要なくグリッドポータルが構築できる。

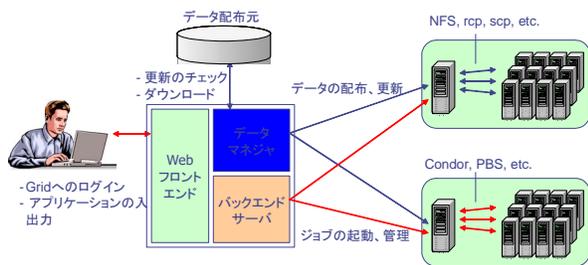


図 1: ポータル概要図

4 PCT4G の設計

図 1 は、PCT4G で作成するグリッドポータルの概要図である。作成するポータルは標準的な 3 層構造のシステムで、第 1 層はクライアントである Web ブラウザ、第 2 層はポータルシステム部、第 3 層が実際の計算を行う計算資源である。グリッドポータルの中核となる第 2 層は、ユーザインタフェースである Web フロントエンド、グリッド資源の処理を行うポータルバックエンド、アプリケーションが利用するデータの管理を行うデータマネージャからなる。以下に詳細を述べる。

Web フロントエンド Web フロントエンドは、ユーザとのインタフェースである。Web を用いることにより、ユーザは一般的な Web ブラウザでのグリッド資源乗りようが可能となる。Web フロントエンドでは、ユーザのグリッドへのログインやアプリケーションとの入出力を扱う。

ユーザのログインには、シングルサインオンと呼ばれる方式を用いる。シングルサインオンとは、一度の認証でグリッド上の複数の資源の使用権を確立する方式である。これにより、ユーザは使用する資源ごとに認証を行う必要がなくなる。

アプリケーションとの入出力では、ユーザはアプリケーションプログラムの選択、各種パラメータの設定などを行い、アプリケーション実行の指示を出す。アプリケーションの実行終了後は、ユーザは結果を Web 上から受け取る。

ポータルバックエンド ポータルバックエンドでは、Web インタフェースより入力された情報に基づき、グリッド上の資源に計算を投入する。その際に、認証、データ交換、ジョブ起動、資源管理などを行う。グリッドの複雑さはバックエンドにより隠蔽される。

データマネージャ データマネージャでは、データを扱うアプリケーションをサポートするために、データのダウンロード、グリッド上の各計算機への配布を行う。また、頻りに更新されるデータを必要とするアプリケーションをサポートするため、データの自動更新も行う。データマネージャは、定期的に配布元のデータの更新をチェックし、データが更新された場合には、データのダウンロード、各計算機への配布を行う。

上述のグリッドポータルを構築するため、PCT4G は以下に述べるいくつかのコンポーネント、ツールを提供している。

アプリケーションインストーラ アプリケーションインストーラは、アプリケーションのインストールを行うコマンドツールである。ポータル構築者が記述するアプリケーションインストール設定ファイルに従いグリッドの各計算機上にアプリケーションをインストールする。

データマネージャ データマネージャは、アプリケーションに必要なデータの配布、更新を行うデーモンである。作成されたポータル上で動作する。

ポータルバックエンド ポータルバックエンドは、作成されるポータルの内部に組み込まれ、グリッド資源の管理、計算の投入を行う。

グリッド作成ポータル グリッド作成ポータルは、一般ユーザがポータルを即席に構築するための Web インタフェースである。6 節で詳しく述べる。

5 PCT4G の実装

PCT4G の通信プロトコルについて述べる。まず、ポータルとグリッド資源間の通信には Globus プロトコルを利用する。Globus プロトコルは、アルゴンヌ研究所が中心となり開発されている Globus ツールキット [3] により採用されているプロトコルである。Globus ツールキットは、グリッド環境で必要となるプロトコルと API を実装したものであり、グリッド上でデファクトスタンダードの地位を確立しつつある。そのため、Globus プロトコルを採用することで、グリッド上の多くの資源の利用が可能になる。なお、PCT4G は、互換性、ライブラリの豊富さなどの理由から Java で実装したため、Globus の Java 用のライブラリである Java CoG Kit[7] を利用している。

前述のように、ポータルとグリッド資源間の通信には Globus プロトコルを用いるが、各グリッド資源 (サイト) 内の通信は、各サイトに適したプロトコル (ローカルプロトコル) を用いる。それは、クラスタなどをグリッド資源の場合、クラスタの代表ノードのみに Globus ツールキットをインストールし、内部では PBS や Condor[5] などのローカルプロトコルを用いる運用方法が一般的であるからである。

以下に各コンポーネントの詳細を述べる。

5.1 アプリケーションインストーラ

アプリケーションインストーラ アプリケーションインストーラは、ユーザが指定したアプリケーションをグリッド資源上にインストールする。ユーザは、アプリケーションのインストールに必要な情報を XML を用いた設定ファイル (アプリケーションインストール設定ファイル) に記述する。設定する内容を以下に述べる。

アプリケーションアーカイブ アプリケーションアーカイブの場所を、GlobusURL を用いて指定する。GlobusURL は、通常の URL の記述を拡張したもので、通常のローカルファイル (file:///...)、HTTP(http://...), FTP(ftp://...) の他に、Globus 標準の GSI 認証を用いたファイル転送方式である GridFTP(gsiftp://...) を指定できる。

インストールスクリプト インストールスクリプトは、アプリケーションのインストール、設定を行うスクリプトである。この場所には、インストールスクリプトのファイル名を指定する。

インストールする各サイトの情報 アプリケーションをインストールする各サイトの情報を記述する。各サイトの情報には、そのサイトが用いるファイル転送プロトコルの種類 (NFS、rcp、scp 等)、ホスト名、インストールディレクトリを指定する。

この設定に基づき、アプリケーションインストーラは、まずアプリケーションアーカイブを指定された URL からダウンロードし、グリッド標準のファイル転送方式である GridFTP を用いグリッド上の各サイトに配布する。サイト内部では、サイト毎に、NFS、rcp、ssh などのローカルなファイル転送プロトコルを用いて各計算ノードにアーカイブを配布する。アーカイブの配布後は、インストール

スクリプトを Globus の GRAM プロトコルを用いて実行し、アプリケーションのインストールは完了する。

5.2 データマネージャ

データマネージャは、アプリケーションが用いるデータの配布、更新を行う。アプリケーションインストーラと同様に、ユーザは、データの配布、更新に必要な情報を XML を用いた設定ファイル (データマネージメント設定ファイル) に記述する。設定する内容を以下に述べる。

更新頻度 更新頻度を時間単位で記述する。データマネージャは指定された更新頻度でデータのアップデートをチェックする。

データファイル 配布、アップデートをするデータファイルの場所を、GlobusURL を用いて記述する。

フォーマットスクリプト フォーマットスクリプトは、データを配布後に実行するスクリプトのファイル名を指定する。これは、ダウンロード後に解凍、変換などの処理が必要なデータファイルに対応するためである。

データ配布サイト データ配布サイトには、データを配布する各サイトの情報を記述する。各サイトの情報には、そのサイトが用いるローカルプロトコルの種類、ホスト名、インストールディレクトリを記述する。

初回のデータ配布はアプリケーションインストールと同様に行われる。その後は、指定された間隔毎にデータのアーカイブの更新をチェックし、更新があった場合には、再びダウンロード、配布を行う。

5.3 ポータルバックエンド

ポータルバックエンドでは、Web インタフェースからの入力を元にグリッド資源に対してジョブを投入する。グリッド上の資源を利用するための認証には、MyProxy[4] サーバから取得した代理証明書を用いる。グリッド資源が複数登録されている場合には、まず、スケジューラにジョブを投入する資源を問い合わせ、適した資源に対してジョブの投入を行う。実際のジョブの投入には、Globus の GRAM を用いる。GRAM でジョブを起動する際に、ホスト名などの他に、ローカルスケジューラの種類を指定するため、サイト内では指定されたスケジューラが用いられてジョブがスケジューリングされる。

グリッド上で行われる計算の多くは長時間かかるため、バックエンドサーバはジョブの終了を待つことなく、Web フロントエンドにジョブのオブジェクトを渡す。実際にジョブが終了した際、またはエラーやタイムアウトが起こった際には、それらの情報を該当するジョブのオブジェクトに格納する。これらのジョブオブジェクトは、Web インタフェースのジョブ管理ページより参照され、表示される。

6 ポータル作成ポータル

今まで述べて来た方法は、システム管理者がグリッドポータルを簡単に構築するものである。しかし、この方法では、自分専用のアプリケーションをグリッド上で即座に利用したいという一般ユーザの要望に答えることができない。これらのユーザは、自分で書いた解析プログラムなどをグリッド上で実行しているユーザであり、現状では、グリッド上の利用するすべての計算機へのアプリケーションのインストール、必要なデータのコピーなどを自ら行っている。

そこで、PCT4G では、ポータル作成ポータルと呼ばれる Web インタフェースを提供し、一般ユーザが個人用グリッドポータルを即席に作成することを可能にする。これにより、グリッドの詳細な知識がないユーザもグリッドの恩恵を得ることが容易になる。ポータル作成ポータルとは、一般ユーザが個人用のグリッドポータルを即席に

作成するための Web インタフェースである。一般ユーザは、Web インタフェースを用いてグリッド上にログインし、ポータル構築に必要な情報を入力し、必要なファイルをアップロードすることで、ポータルを構築できる。そのため、ユーザは自前のアプリケーションをポータルを通して作成し、自ら使用できる。

認証には、MyProxy サーバから取得される代理証明書を用いる。そのため、アプリケーションのインストールはそのユーザが書き込み権限のあるディレクトリ (多くの場合はホームディレクトリ) に行なわれる。

7 実アプリケーションへの適用

PCT4G を用いて、実際に BLAST アプリケーション用のグリッドポータルを構築した。BLAST(Basic Local Alignment Search Tool) はタンパク質、DNA のホモロジー検索を行うアプリケーションで、バイオインフォマティク分野で広く用いられているものである。いくつかの実装が存在するが、今回はアメリカの NCBI(National Center for Biotechnology Information)[2] が中心となって開発、公開している NCBI BLAST を用いたグリッドポータルを作成する。

BLAST は、検索を行うデータベースとして約 30GB のデータが必要であり、そのデータは毎日アップデートされている。そのため、既存のポータル構築ツールキットでは、BLAST ポータルを構築するのは難しい。しかし、PCT4G では、データの配布、更新をサポートしているため、BLAST ポータルの構築は、容易である。

7.1 設定ファイルを用いる方法

まず、システム管理者向けの設定ファイルを用いた方法を説明する。この方法では、すべてのグリッドユーザが利用できるポータルを作成する。

7.1.1 アプリケーションのインストール

まずは、アプリケーションインストーラを用いて BLAST アプリケーションのインストールをする。BLAST アプリケーションのアーカイブは NCBI の FTP サイトに公開されているものを用い、インストール先には東京工業大学のグリッド資源 "Titech Grid" を利用する。

図 2 はアプリケーションのインストールのために記述したアプリケーション設定ファイルである。この設定ファイルでは、NCBI 公開の FTP サーバ (ftp.ncbi.nih.gov) より BLAST アプリケーションのアーカイブファイル (blast.linux.tar.Z) をダウンロードし、Titech Grid 上のホストにインストールする。Titech Grid の各ノードは、NFS によってファイルシステムを共有しているため、代表ノード (tgn003001.g.gsic.titech.ac.jp) へのみインストールを行う。インストールディレクトリは、"/usr/local/blast" である。アプリケーションを実際にインストールするためのスクリプトには、"blast-install-script.sh" を用いる。このスクリプトは、簡単なシェルスクリプトであり、ファイルの展開方法を指示しているのみである。

7.1.2 アプリケーションデータの配布、更新

次は、アプリケーションが利用するデータの配布と、定期的な更新の設定である。

BLAST では、遺伝子配列やたんぱく質の検索を行うためのデータが必要である。また、それらのデータファイルは、毎日アップデートされている。そこで、NCBI の FTP サイトにある BLAST 用のデータファイルを Titech Grid 上に配布し、その後は、更新を毎日 1 回チェックし、再配布を行う設定をする。

図 3 が記述したデータマネージメント設定ファイルである。この設定では、NCBI 公開の FTP サイトに 24 時間に 1 回更新を確認し、更新があった場合には、そのデー

```

<?xml version="1.0" ?>
<data>
  <period>0</period>
  <data_files>
    <globusurl>
      ftp://ftp.ncbi.nih.gov/blast/
        executables/blast.linux.tar.Z
    </globusurl>
  </data_files>
  <scripts>
    <file>
      /home/sirasuna/blast/
        blast-install-script.sh
    </file>
  </scripts>
  <destinations>
    <site>
      <type>NFS</type>
      <host>
        tgn003001.g.gsic.titech.ac.jp
      </host>
      <dir>/usr/local/blast</dir>
    </site>
  </destinations>
</data>

```

図 2: アプリケーションインストール設定ファイル

タファイルをダウンロードし、Titech Grid 上のホストに配布する。BLAST で用いるデータファイルはフォーマットする必要があるため、そのフォーマットのコマンドを記述した”blast-update-script.sh”が指定されている。このシェルスクリプトは、更新があるたびに Titech Grid 上のホストで実行される。

7.2 ポータル作成ポータルを用いる方法

次にポータル作成ポータルを用いた方法を説明する。

まず、BLAST アプリケーションのインストールを行う。

図 4 がアプリケーションインストールを行う Web ページである。ここでインストールに必要な情報を入力する。入力する情報は、設定ファイルを用いた場合と同じである。ただし、一般ユーザ権限でインストールを行うため、アプリケーションはユーザのホームディレクトリにインストールする。

続いて、アプリケーションが利用するデータの配布、自動更新の設定を行う。図 5 がデータ管理を行う Web ページである。こちらも入力する内容は、設定ファイルを用いた場合と同様である。

7.3 BLAST ポータルの利用

ここまで、設定ファイルを用いる方法、ポータル作成ポータルを用いる方法の両方で BLAST ポータルを構築する方法を説明してきた。ここで、実際に構築した BLAST ポータルを利用方法を述べる。

ポータルを利用する際、ユーザはまずポータルにログインをする必要がある。図 6 がログインページであり、ユーザはユーザ名とパスワードを入力する。入力したユーザ名とパスワードは、MyProxy サーバに渡され、そのユーザに適した代理証明書がポータルに渡される。実際のグリッド資源の認証はこの代理証明書を用いて行われる。

ログイン後、ユーザが利用するアプリケーションを選択すると、アプリケーションの入力画面 (図 7) が表示される。ここでユーザはアプリケーションのオプションを選択し、データを入力する。入力後、“Submit” ボタンを押す

```

<?xml version="1.0" ?>
<data>
  <period>24</period>
  <data_files>
    <globusurl>
      ftp://ftp.ncbi.nih.gov/
        blast/db/alu.a.Z
    </globusurl>
    <globusurl>
      ftp://ftp.ncbi.nih.gov/
        blast/db/alu.n.Z
    </globusurl>
    . . . 略 . . .
  </data_files>
  <scripts>
    <file>
      /home/sirasuna/blast/
        blast-update-script.sh
    </file>
  </scripts>
  <distinations>
    <site>
      <type>NFS</type>
      <host>
        tgn003001.g.gsic.titech.ac.jp
      </host>
      <dir>/usr/local/blast</dir>
    </site>
  </destinations>
</data>

```

図 3: データマネージメント設定ファイル

と、グリッドにジョブが投入され、実際の計算が行われる。計算終了後、ユーザは結果を Web 上で受け取れる (図 8)。

8 まとめと今後の課題

グリッドポータルを構築するためのツールキット (PCT4G) の開発を行った。PCT4G では、アプリケーションのインストール、データの配布、更新をサポートし、ポータル構築のコストを削減する。また、ポータル作成ポータルを用いることで、一般ユーザが即席に自分専用のポータルを作成することが可能である。我々は実際に PCT4G を用いて BLAST ポータルを構築し、そのポータルが実際に動作することを確認した。今後の課題は、PCT4G を多くのユーザに利用してもらい、その有用性を検証することである。

9 参加企業及び機関

なし

参考文献

- [1] DOE Science Grid research and development. Grid portal development kit (gpdk). <http://doesciencegrid.org/projects/GPDK/>.
- [2] National Center for Biotechnology Information (NCBI). <http://www.ncbi.nlm.nih.gov/>.
- [3] Ian Foster and Carl Kesselman. Globus: A Meta-computing Infrastructure Toolkit. *International Journal of Supercomputer Applications*, 1997.

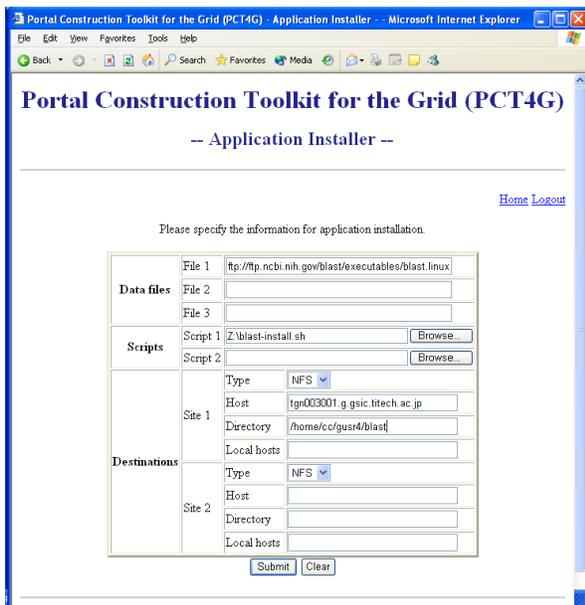


図 4: アプリケーションインストール画面

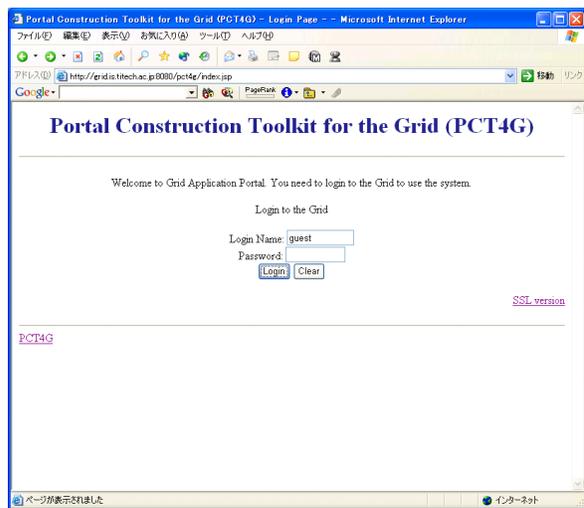


図 6: ログイン画面

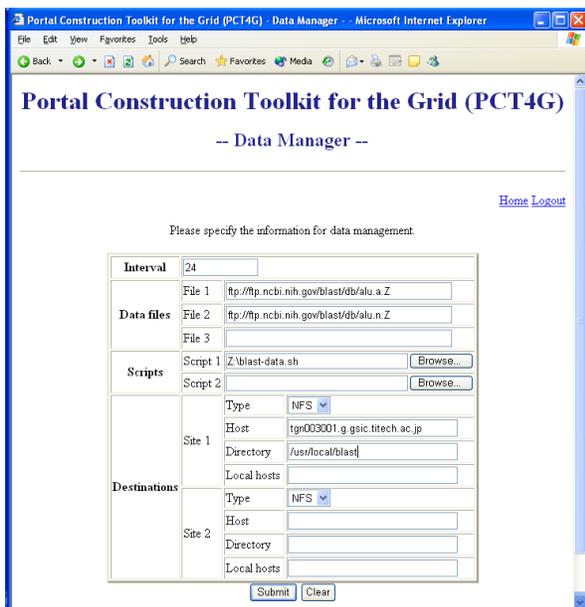


図 5: データマネージメント画面

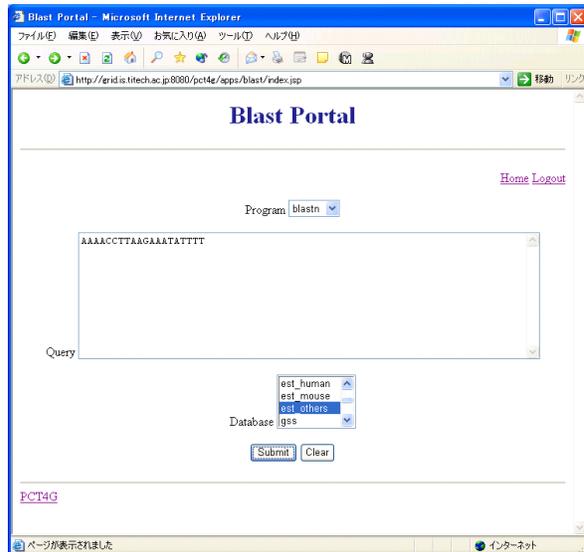


図 7: 入力画面

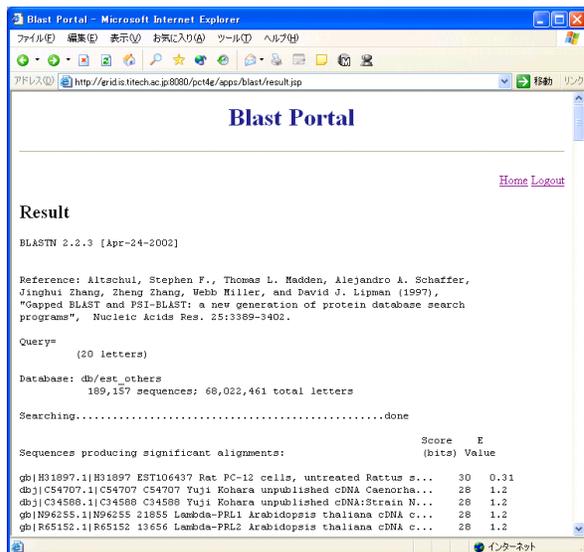


図 8: 出力画面

- [4] J. Novotny, S. Tuecke, and V. Welch. An online credential repository for the grid: Myproxy. In Proceedings of the 10th International Symposium on High Performance Distributed Computing (HPDC-10), August 2001.
- [5] R. Raman, M. Livny, and M. Solomon. Match-making: Distributed Resource Management for High Throughput Computing. In Proceedings of the 7th International Symposium on High Performance Distributed Computing (HPDC-7), 1998.
- [6] M. Thomas, S. Mock, J. Boisseau, M. Dahan, K. Mueller, and D. Sutton. The gridport toolkit architecture for building grid portals. In Proceedings of the 10th International Symposium on High Performance Distributed Computing (HPDC-10),

August 2001.

- [7] Gregor von Laszewski, Ian Foster, Jarek Gawor, and Peter Lane. A Java Commodity Grid Kit. Concurrency and Computation: Practice and Experience, Vol. 13, No. 8-9, pp. 643-662, 2001.