

アプリケーションのインストール、 データの配布、更新をサポートする グリッドポータル構築ツールキット(PCT4G)の開発

白砂 哲^{*1}

鈴村 豊太郎^{*1}

中田 秀基^{*2*1}

松岡 聡^{*1*3}

^{*1} 東京工業大学

^{*2} 産業技術総合研究所

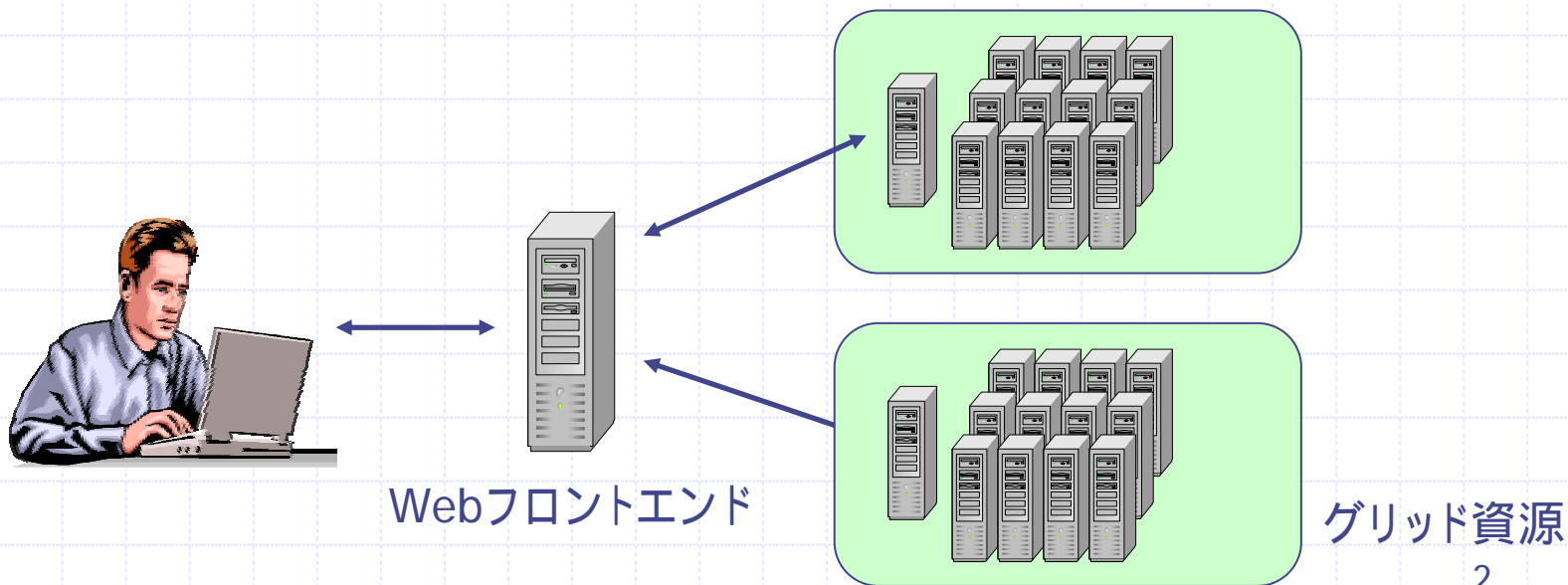
^{*3} 科学技術振興事業団

背景

– グリッドポータル – (1)

◆ グリッドポータルとは？

- 専門の知識なしにグリッド上の資源の利用を可能にするシステム
- ユーザはWebインタフェースなどを通してグリッド上のアプリケーションを利用



背景

– グリッドポータル – (2)

- ◆ さまざまなアプリケーション独自のポータルが存在
 - NCBI BLAST、GridBLAST etc.

→ 実装は煩雑

- ◆ いくつかのポータルツールキット
 - 認証やアプリケーション起動などの共通部分を提供
 - GridPort (NPACI)
 - Grid Portal Development Kit (NLANR)
 - XCAT Science Portal (Indiana Univ.)

→ アプリケーションのインストールやWebインタフェース作成はポータル構築者が別個におこなわなければならない

目的

- ◆ グリッドポータルを簡便に構築するためのツールキットPCT4Gを作成
 - ポータル構築の際の管理者の負担を軽減
 - ◆ アプリケーションのインストール、データ管理、Webインタフェース生成を自動化
 - ◆ 既存のアプリケーションの変更は不要
 - ◆ 複雑な機能や拡張性はある程度犠牲
- ◆ 実際のアプリケーション(BLAST)用のグリッドポータルの作成

発表のアウトライン

◆ PCT4Gが作成するグリッドポータルの概要

◆ PCT4Gの概要

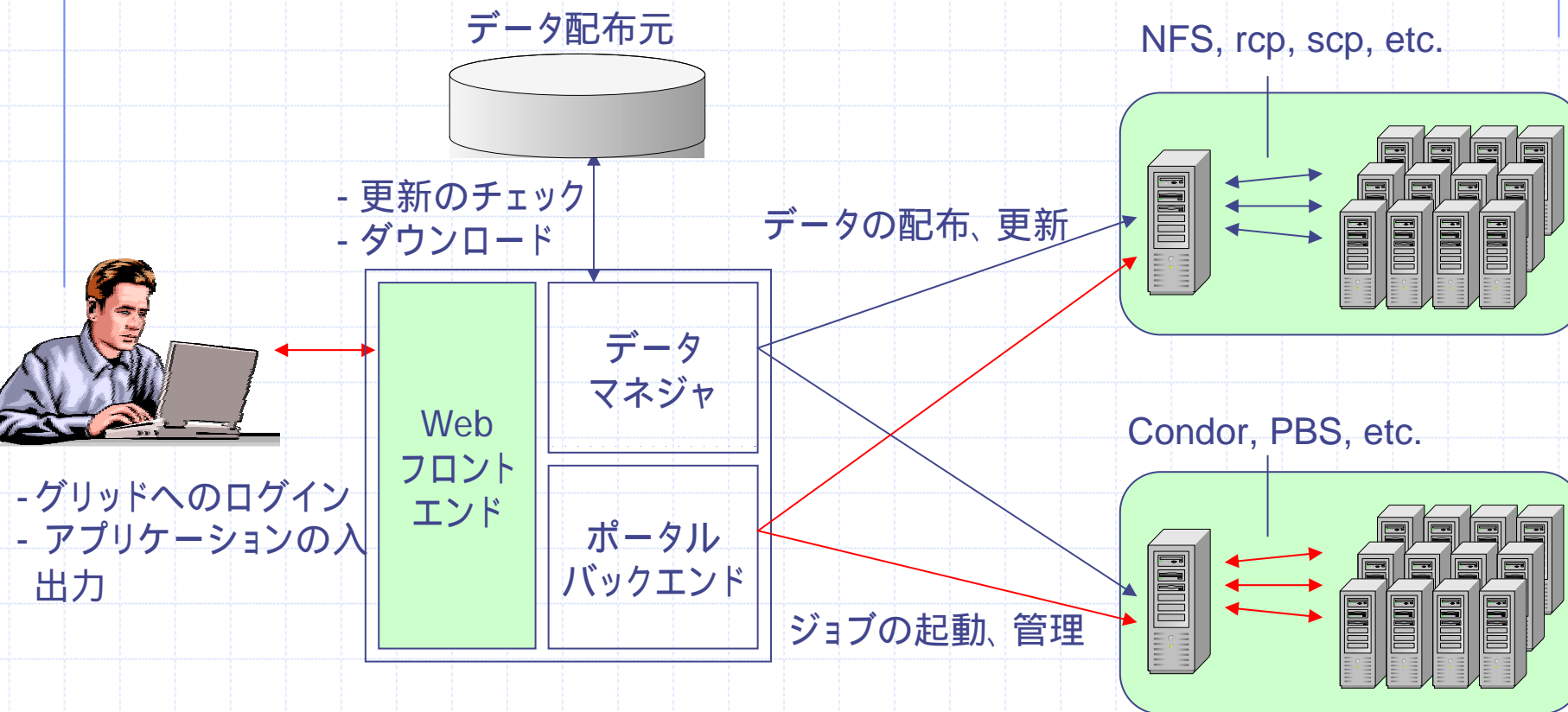
- ポータルバックエンド
- アプリケーションインストーラ
- データマネージャ
- インターフェースジェネレータ

◆ PCT4Gを用いたBLASTポータルの構築

◆ まとめ

◆ 今後の課題

PCT4Gが作成する グリッドポータル概要(1)



PCT4Gが作成する グリッドポータル概要(2)

◆ PCT4Gを用い、以下の機能を持つグリッドポータルを作成

■ Webフロントエンド

- ◆ ユーザのログイン
- ◆ アプリケーションへの入出力
- ◆ Servlet, JSPを用いて実装

■ ポータルバックエンド

- ◆ グリッド資源の管理
- ◆ ジョブの起動、管理
- ◆ Java CoGキットを用いて実装

■ データマネージメント

- ◆ アプリケーションデータの配布、アップデート

発表のアウトライン

◆ PCT4Gが作成するグリッドポータル概要

◆ PCT4Gの概要

- ポータルバックエンド
- アプリケーションインストーラ
- データマネージャ
- インターフェースジェネレータ

◆ PCT4Gを用いたBLASTポータルの構築

◆ まとめ

◆ 今後の課題

PCT4Gの概要

◆ PCT4Gの主なコンポーネント

■ ポータルバックエンド

- ◆ 作成するグリッドポータルに組み込まれる

■ アプリケーションインストーラ

- ◆ グリッド資源への既存のアプリケーションの自動インストール

■ データマネージャ

- ◆ アプリケーションデータの配布、定期的なアップデート

■ インタフェースジェネレータ

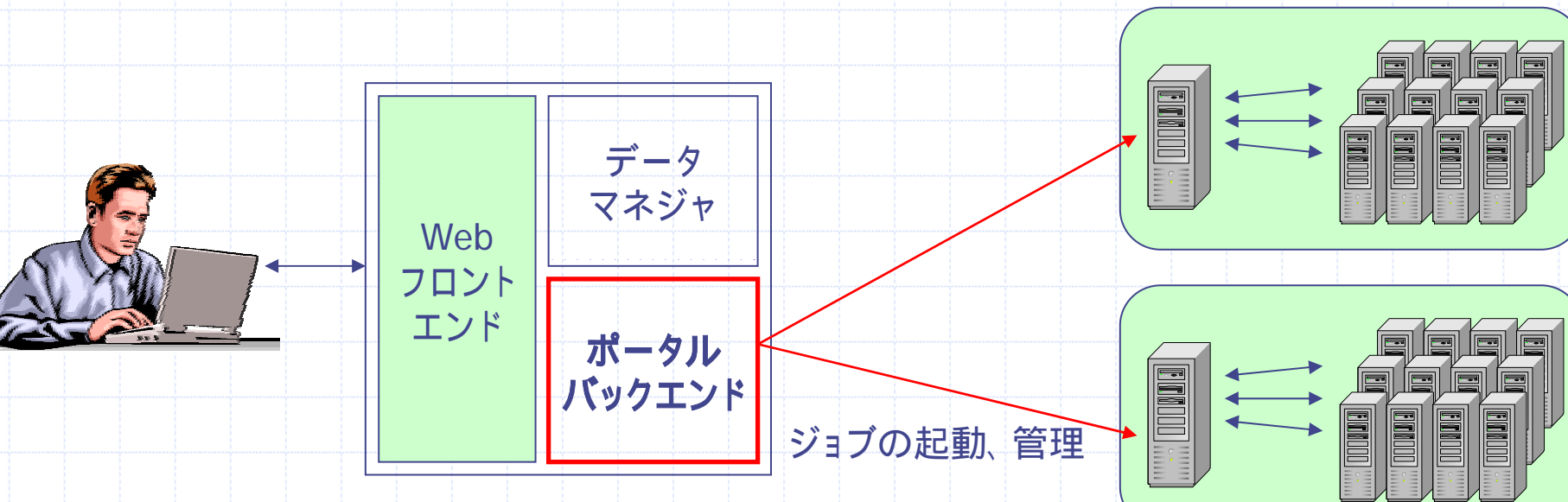
- ◆ Webインタフェースの生成

◆ 使用するグリッド資源上にGlobusツールキットがインストールされていることのみを前提

- PCT4Gをグリッド資源上にインストールする必要なし

ポータルバックエンド(1)

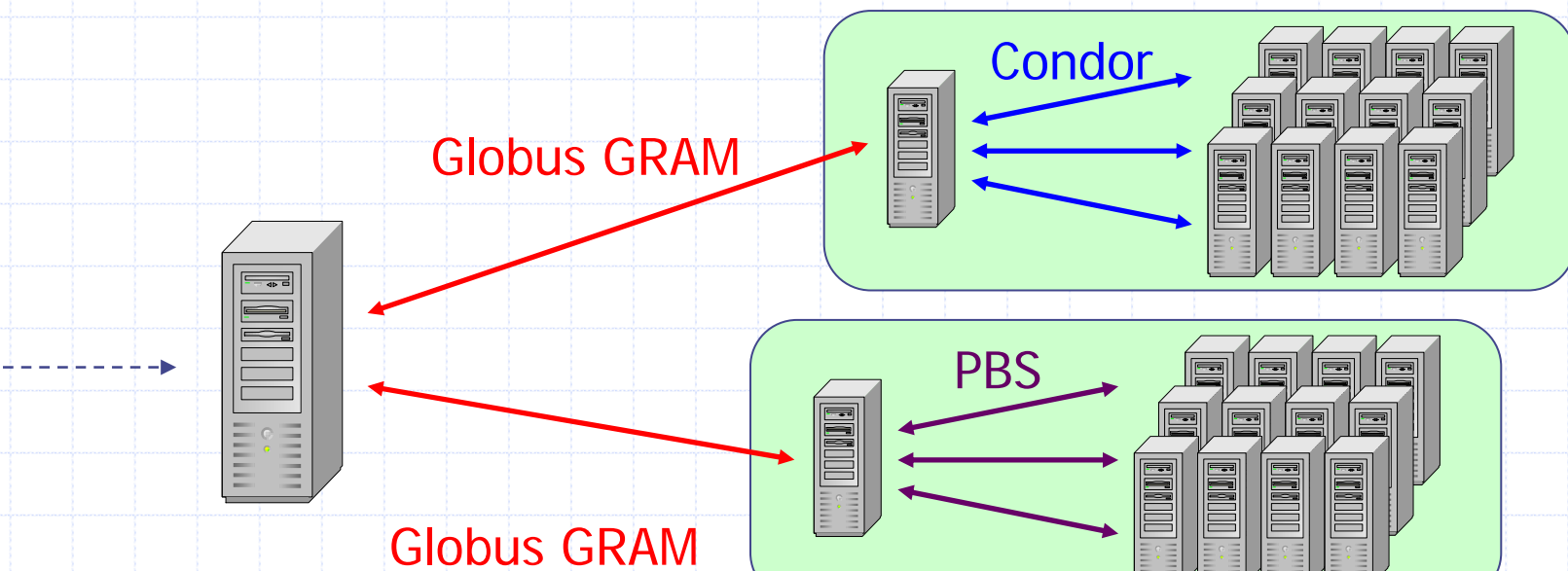
- ◆ バックエンドであるグリッド資源の管理を行う
 - ジョブ起動、管理
 - ユーザ管理、認証
- ◆ 作成されるグリッドポータルに組み込まれる
- ◆ Globus Java CoGを用いて実装



ポータルバックエンド(2)

◆ アプリケーションの起動

- 適切なグリッドリソースを選択
- GRAMを用いてアプリケーションを起動
- サイト内部では、サイト内部のジョブスケジューラを利用する
 - ◆ e.g. Condor, PBS, etc.



ポータルバックエンド(3)

◆ ユーザ認証

- シングルサインオンと呼ばれる認証方式を用いる
 - ◆ 一度のログインですべてのグリッド資源を利用可能
- ユーザは、Webインタフェースよりユーザ名とパスワードを入力
- MyProxy^{*1)}サーバのレポジトリより、ユーザ代理証明書を得
- その後のすべてのグリッド資源へのアクセスは、その代理証明書を用いて行う

*1) MyProxy[NLANR]

- ◆ 証明書のDelegationを利用した安全な証明書保持機構
- ◆ 秘密鍵をネットワーク越しに転送することなくアイデンティティを委譲

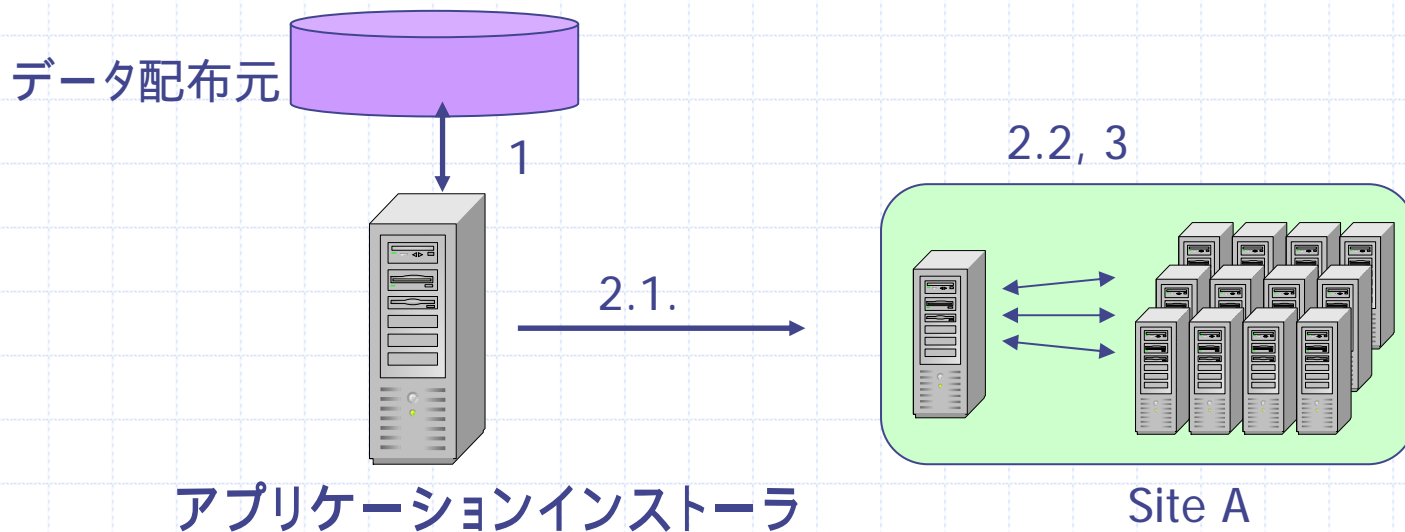
アプリケーションインストーラ(1)

- ◆ 既存のコマンドラインアプリケーションをグリッド上の各ノードにインストール
 - 設定ファイルによりインストール作業を自動化
 - ◆ アーカイブのダウンロード、各ノードへの配布
 - ◆ インストールスクリプトによるインストール
- ◆ 既存のアプリケーションへの変更は必要なし
 - アプリケーションのグリッド化、並列化などのコストを削減
 - 個々のパフォーマンス向上ではなく、複数の資源の利用による全体的なスループットの向上を目指す

アプリケーションインストーラ (2)

◆ アプリケーションのインストール手順

1. **アプリケーションアーカイブ**のダウンロード(GSI-FTP, FTP, HTTP)
2. グリッド資源上への配布
 - 2.1. 各サイトへの配布(GridFTP)
 - 2.2. サイト内の各ホストへ配布(ローカルプロトコル)
3. **インストールスクリプト**の実行



アプリケーションインストーラ(3)

◆ アプリケーションインストール設定ファイル

- XMLベースの設定ファイル
- ポータル構築者が記述
- アプリケーションインストール設定内容
 - ◆ アプリケーションアーカイブのURL (FTP, HTTP, etc.)
 - ◆ インストールスクリプト
 - ◆ 配布先の情報
 - 代表ノードのホスト名
 - ローカルファイル転送プロトコル(NFS, rcp, scp)
 - サイト内のホスト名のリスト(rcp, scpの場合)

データマネージャ(1)

- ◆ アプリケーションが用いるデータをグリッド資源上に配布、定期的に更新する
 - 設定ファイルによりインストール作業を自動化
 - ◆ データアーカイブのダウンロード、各ノードへの配布
 - ◆ フォーマットスクリプトにより、データの展開、変換(必要時)
- ◆ バイオインフォマティクス分野のアプリケーションなどに必要な機能
 - タンパク質、DNAなどのデータ
 - データは数十GBで毎日更新

データマネージャ (2)

◆ アプリケーションが利用するデータの配布、更新手順

0. データの更新をチェック

1. データアーカイブのダウンロード(GSI-FTP, FTP, HTTP)

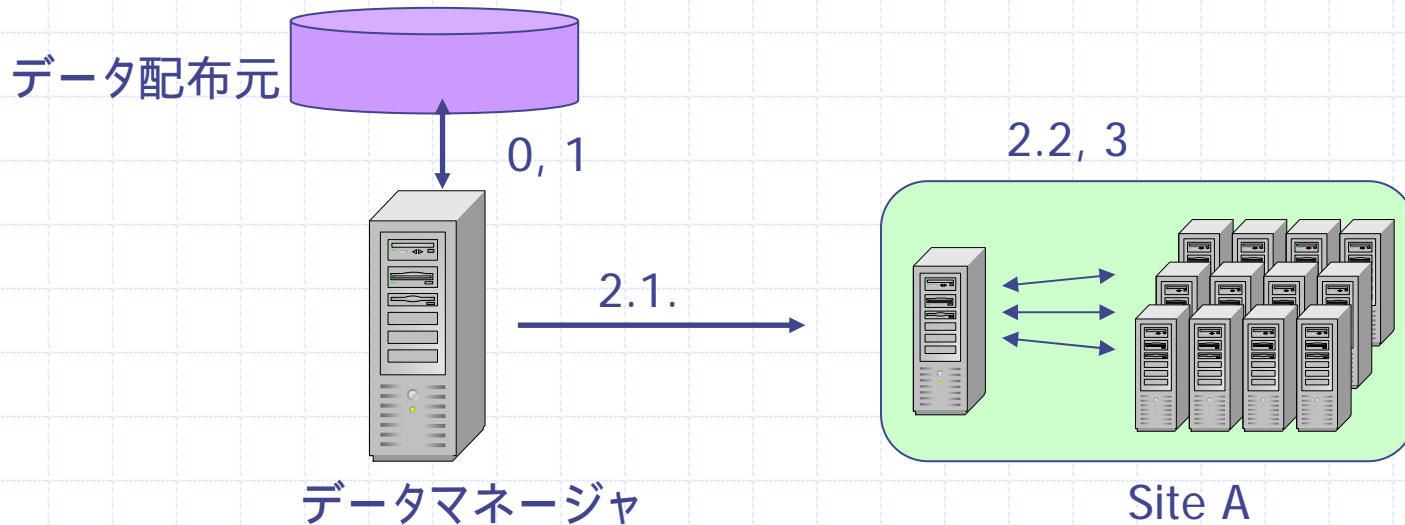
2. グリッド資源上への配布

2.1. 各サイトへの配布(GridFTP)

2.2. サイト内の各ホストへ配布(ローカルプロトコル)

3. フォーマットスクリプトの実行(Optional)

4. 0-3を定期的に繰り返す



データマネージャ(3)

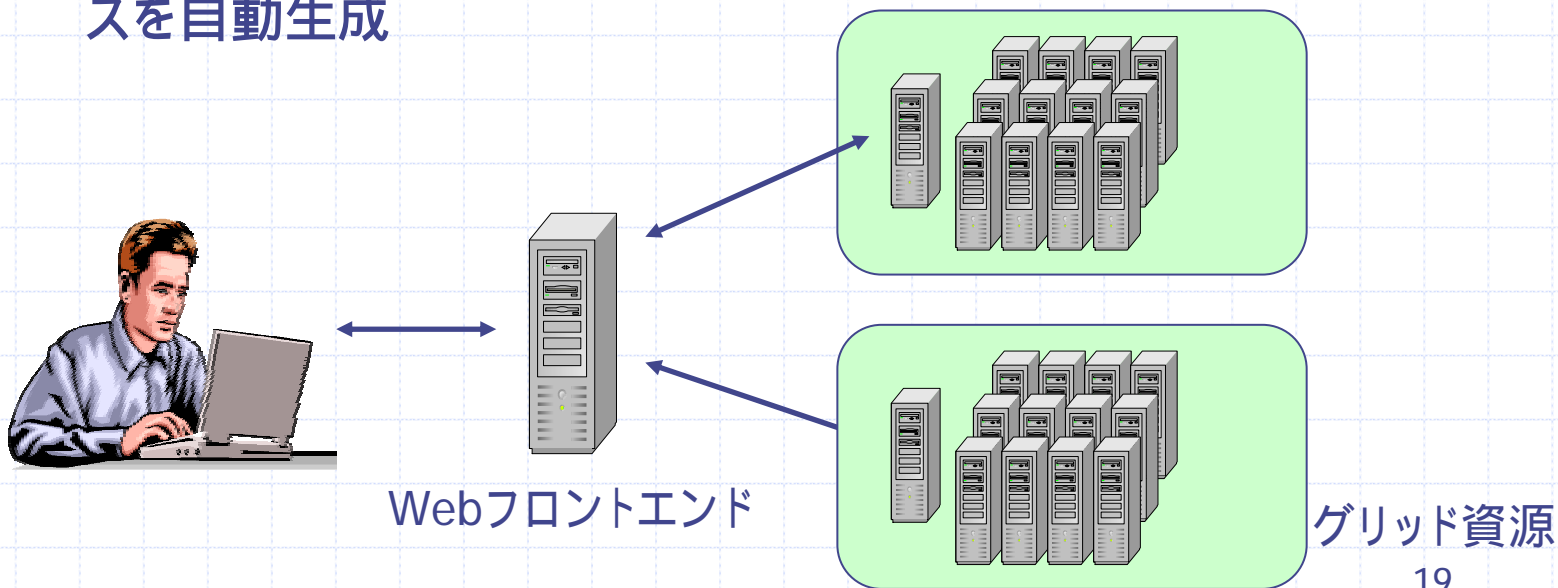
◆ データマネージメント設定ファイル

- XMLベースの設定ファイル
- ポータル構築者が記述
- 設定内容
 - ◆ 更新頻度
 - ◆ アプリケーションデータのURL (FTP, HTTP, etc.)
 - ◆ フォーマットスクリプト (オプション)
 - ◆ 配布先の情報
 - 代表ノードのホスト名
 - ローカルのファイル転送プロトコル(NFS, rcp, scp)
 - サイト内のホスト名のリスト(rcp, scpの場合)

インタフェースジェネレータ

◆ Webインタフェースの自動生成

- Gridspeedプロジェクト[Titech]で開発されたコンポーネントを利用
- “Grid Application IDL” にアプリケーションの情報を記述
 - ◆ プログラム名、引数情報
- インタフェースジェネレータがJSPでかかれたWebインタフェースを自動生成



発表のアウトライン

◆ PCT4Gが作成するグリッドポータル概要

◆ PCT4Gの概要

- ポータルバックエンド
- アプリケーションインストーラ
- データマネージャ
- インターフェースジェネレータ

◆ PCT4Gを用いたBLASTポータルの構築

◆ まとめ

◆ 今後の課題

BLASTポータル構築(1)

◆ PCT4Gを用い、BLASTポータルを構築

- BLAST (Basic Local Alignment Search Tool)
 - ◆ たんぱく質、DNAのホモロジ検索
 - ◆ 1日に1回更新されている数GBのデータを用いる
- グリッド資源としては東京工業大学のTitech Gridを利用

◆ 構築手順

- アプリケーションインストールの設定
 - ◆ NCBIのFTPサイトよりBLASTアプリケーションをダウンロード
 - ◆ Titech Grid上にインストール
- データマネジメントの設定
 - ◆ NCBIのFTPサイトよりタンパク質、DNAのデータをダウンロード
 - ◆ Titech Grid上の配布、フォーマット
 - ◆ 24時間ごとに更新をチェック
- Webインタフェースの生成

BLASTポータルの構築(2)

◆ アプリケーションインストール設定ファイル

```
<?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>
```

```
# /bin/sh
tar -zxf newdate/blast.linux.tar.Z
```

BLASTポータルの構築(3)

◆ データマネージメント設定ファイル

```
<?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>
  <destinations>
    <site>
      <type>NFS</type>
      <host>tgn003001.g.gsic.titech.ac.jp</host>
      <dir>/usr/local/blast</dir>
    </site>
  </destinations>
</date>
```

```
#!/bin/sh
prefix=`echo $file | sed -e "s/¥.Z//"`
uncompress -c $file | ./formatdb -i stdin -o T -n ${prefix}
```

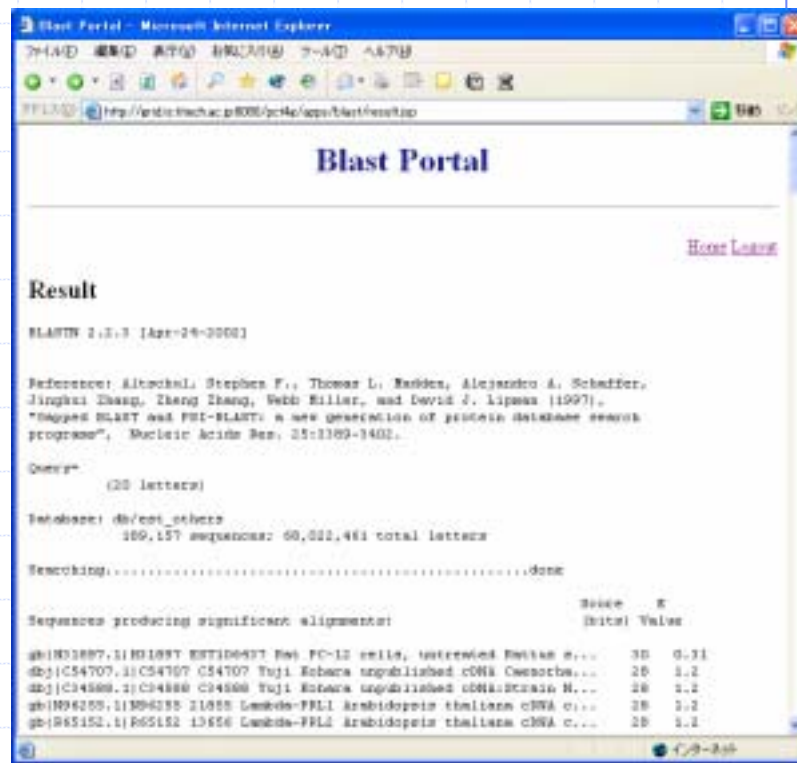
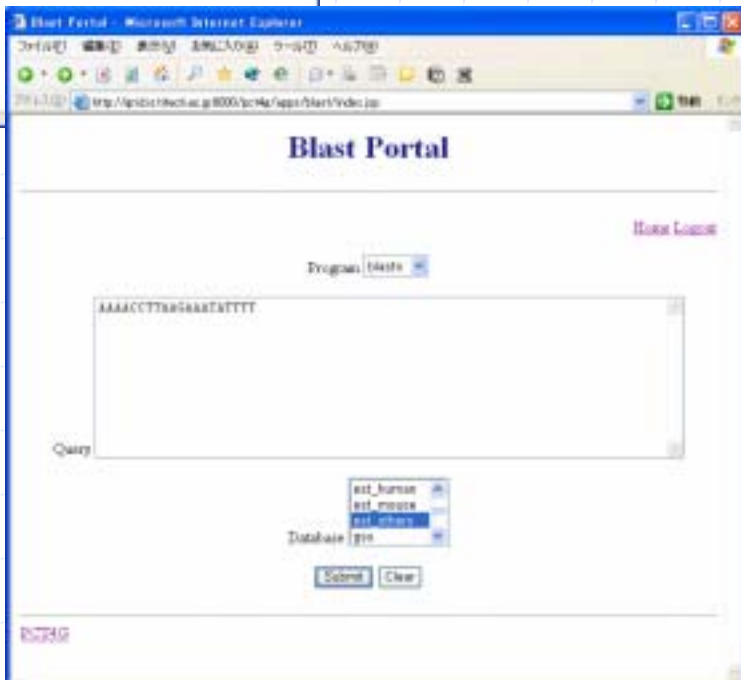
BLASTポータル構築(4)

◆ ユーザインタフェースの作成

結果表示

ログインページ

入力ページ



BLASTポータル構築(5)

◆ 評価

- 既存のツールキットを用いた方法に比べて、ポータル構築の負荷がかなり軽減された
- BLASTを利用するための最小限の機能は提供された
- BLASTに特化されたポータルと比べて、BLAST特有の機能のサポートが少ない
- BLASTに特化されたポータルと比較して、単体の性能が低い
 - ◆ BLASTに特化されたポータルでは、高性能の計算機をバックエンドに用いているため

発表のアウトライン

◆ PCT4Gが作成するグリッドポータルの概要

◆ PCT4Gの概要

- ポータルバックエンド
- アプリケーションインストーラ
- データマネージャ
- インターフェースジェネレータ

◆ PCT4Gを用いたBLASTポータルの構築

◆ まとめ

◆ 今後の課題

まとめ

◆ グリッドポータル構築ツールキットPCT4Gを作成

- アプリケーションインストール
- データ管理
- Webインタフェース生成

◆ ポータル構築ポータルの作成

- 一般ユーザが即席にグリッドポータルを構築することが可能

◆ PCT4Gを用いIBLASTポータルを構築

- PCT4Gによりグリッドポータル構築の負荷が軽減

今後の課題

◆ さまざまな機能の追加

- 一般性を保ちながら、個々のアプリケーションの要望に答える

◆ より使いやすいWebインタフェースの作成

◆ グリッド特有の機能の追加、強化

- ジョブ管理
- スケジューリング
- フォールトトレランス
- etc.

◆ 一般ユーザによるポータル構築

- ポータル構築ポータル

ポータル構築ポータル(1)

- ◆ グリッドポータルを構築するためのポータルサイト
 - Webインタフェースを通じてのポータル構築が可能
 - 一般ユーザが即席にポータルを構築
 - e.g. 自分専用アプリケーションのポータル
- 認証にはMyProxyを用いた代理証明書を利用
 - ◆ アプリケーションはユーザの権限のあるディレクトリ(ホームディレクトリ)にインストールされる

ポータル構築ポータル(2)

◆ ポータルの構築

- ポータル構築ポータルにログイン
- ポータル構築に必要な情報の入力やファイルのアップロード
 - ◆ アプリケーションインストーラ
 - ◆ データ管理

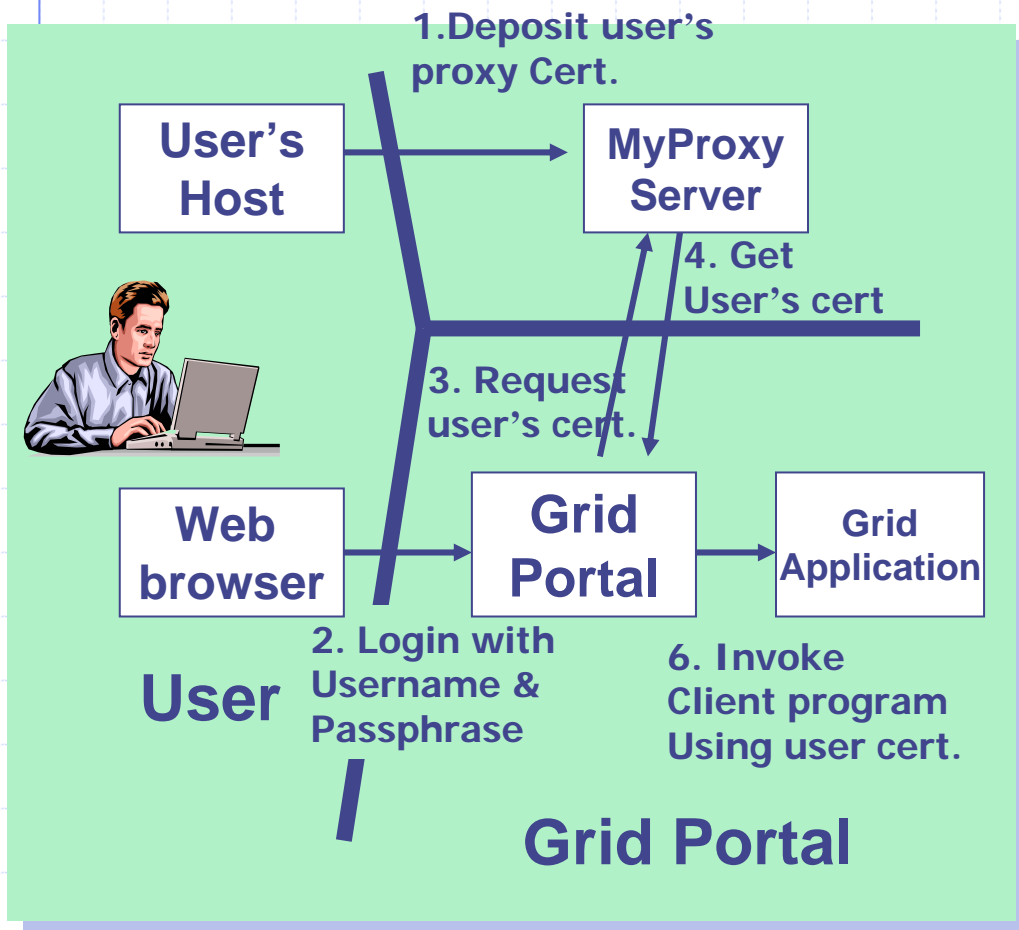
➔ 即座に利用可能



謝辞

- ◆ 本システムの開発は情報処理振興事業協会(IPA) の平成14年度未踏ソフトウェア創造事業の一環として行った

MyProxy の概要



1. ユーザのプロキシ証明書を預ける。この際にパスフレーズを指定
2. Webブラウザからユーザ名とパスフレーズを入力
3. ポータルはMyProxyサーバにアクセス
4. ユーザのプロキシ証明書を取得
5. グリッドアプリケーションを起動