

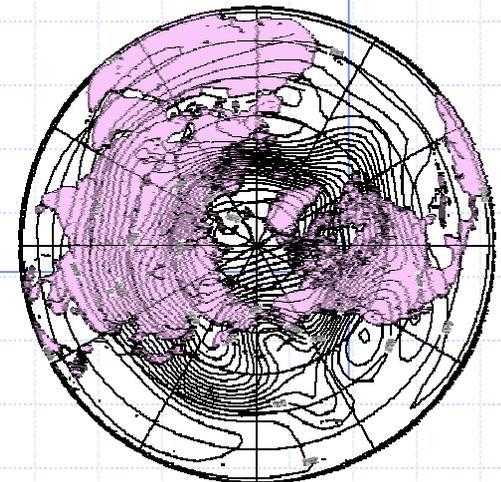
気象予報グリッドポータルの開発

首藤一幸 武宮博 平野基孝
田中良夫 関口智嗣

産業技術総合研究所 グリッド研究センター
(株)SRA

Barotropic Height

199901300800Z H1-0000

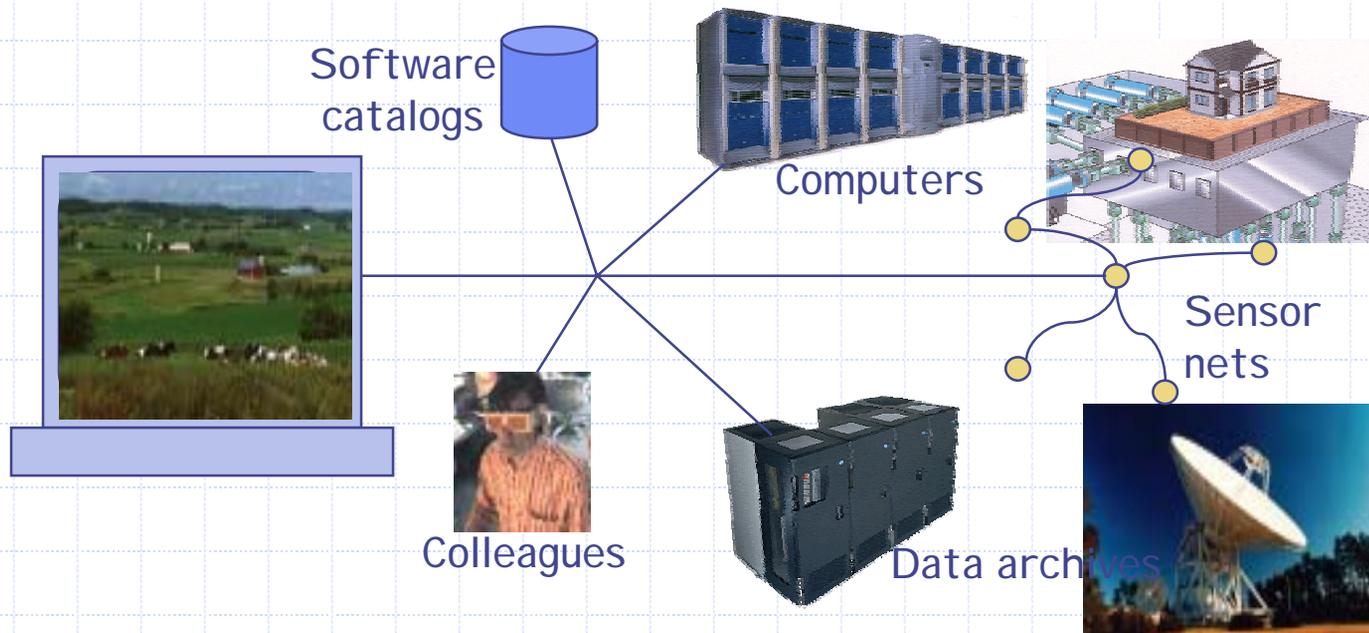


Grid という コンセプト

ドキュメントのウェブから 計算・デバイス・ソフトウェアのウェブへ

◆ ネットワーク上の様々な計算・情報資源を、
どこからでも水道や電力のように容易にセ
キュアに利用できるようにする利用技術

- ◆ 高性能計算機 デバイス・センサー
ソフトウェア・ライブラリ データベース 人 ...



グリッド ミドルウェア

◆ アプリケーション開発者に便宜を提供

■ 下位ミドルウェア

- ◆ セキュアな遠隔プログラム起動や通信
Globus Toolkit

Grid アプリケーション

MPICH-G2

Ninf-G

■ 上位ミドルウェア

- ◆ 容易なプログラミングを支援
MPICH-G2 (メッセージパッシング), Ninf-G (RPC)

Globus Toolkit

◆ アプリケーション利用者に直接提供

■ グリッドポータル

- ◆ ウェブブラウザからの容易なグリッドアプリケーションの利用
(ポータル: Yahoo! などウェブへの入り口がこう呼ばれた)
- ◆ 構築キット: GSDK, HotPage など

気象予報グリッドポータルの開発

1. 逐次プログラムのグリッド化 (gridification)

- 気象予報プログラムを、Ninf-G を用いてグリッドアプリケーションに仕立てた。

2. グリッドアプリのポータル化 (portalization)

- ポータル構築キット GridLib に載せ、ポータルに仕立てた。
- アプリケーション側の変更は皆無。
 - ◆ アプリ記述ファイルを作成し、GridLibに登録した。

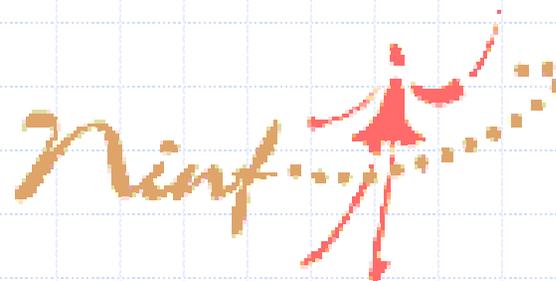
Ninf-G, GridLib : 産総研グリッド研究センターで開発しているグリッドミドルウェア

気象予報グリッドポータル のアーキテクチャ



- ◆ 気象予報アプリは、**Ninf-G** を用いた (非同期) 遠隔呼び出しで、分散処理を行う。
- ◆ ポータル構築キット **GridLib** は、ウェブブラウザからの利用に必要な機能を提供する。

Ninf-G



- ◆ グリッド上で動作するアプリの**容易なプログラミング**を可能とする**(上位)ミドルウェア**
 - 遠隔呼び出し(RPC)という使いやすいプログラミングモデル
 - GGF (Global Grid Forum) APM (Advanced Prog. Model) で作成された**GridRPC 仕様**に準拠
 - GT2 (**Globus Toolkit 2.x**)の**プログラム起動機能**(GRAM)と**通信機能**(Globus IO)を利用
- ◆ アプリの新規開発だけでなく、既存アプリのグリッド化も容易にする。



GridLib - ポータル構築キット

- ◆ ASP (Application Service Provider) としての仮想スーパーコンピュータセンターの構築を支援
- ◆ 既存のアプリケーションをわずかな手間でポータル化
 - ウェブブラウザからの入力パラメータをコマンド行を通じてアプリに渡すため、**アプリ側の変更は不要**
 - アプリケーション記述ファイルを用意し、GridLibに登録する
- ◆ 利用実績
 - Gaussian Portal Phase 2 (産総研, 日本IBM)
 - ◆ 産総研内でTACC(コンピュータセンター)がサービス中

容易なポータル化

- アプリケーション記述の登録
- パラメータ入力画面の自動生成

GridLib

* option	<input type="radio"/> do not hide entries starting with . <input type="radio"/> use a long listing format
screen width	this type is 'integer'. Range : 0.0 < x <= 200.0 <input type="text" value="40"/>

```
<?xml version="1.0" ...  
<application>  
  <appname>ls</appname>  
  <argspec>/bin/ls %option% %width%</argspec>  
  <arglist>  
    <args use="required">  
      <title>option</title>  
      <radio name="option">  
        <option value="-a">do not hide entries ...</option>
```

GridLib の機能

グリッドポータルに
必要な機能ひと通り

◆ サインオンの受け付け

- パスワード、ブラウザが提示する証明書に基づいて利用者を認証

◆ ジョブの実行管理

- Globus Toolkitのジョブ管理機能GRAMを利用して、投入、取り消しなどを行う。
- 実行中、完了などの状態を管理し、利用者に提示する。

◆ ジョブの入出力管理

- パラメータの入力やファイルの転送を受け付け、アプリに渡す。また、ジョブの出力(標準出力やファイル)を利用者に返す。

◆ アカウンティング

- プロセッサ、ディスクの使用量に基づく利用記録を採る。
 - ◆ コンピュータセンターには欠かせない。

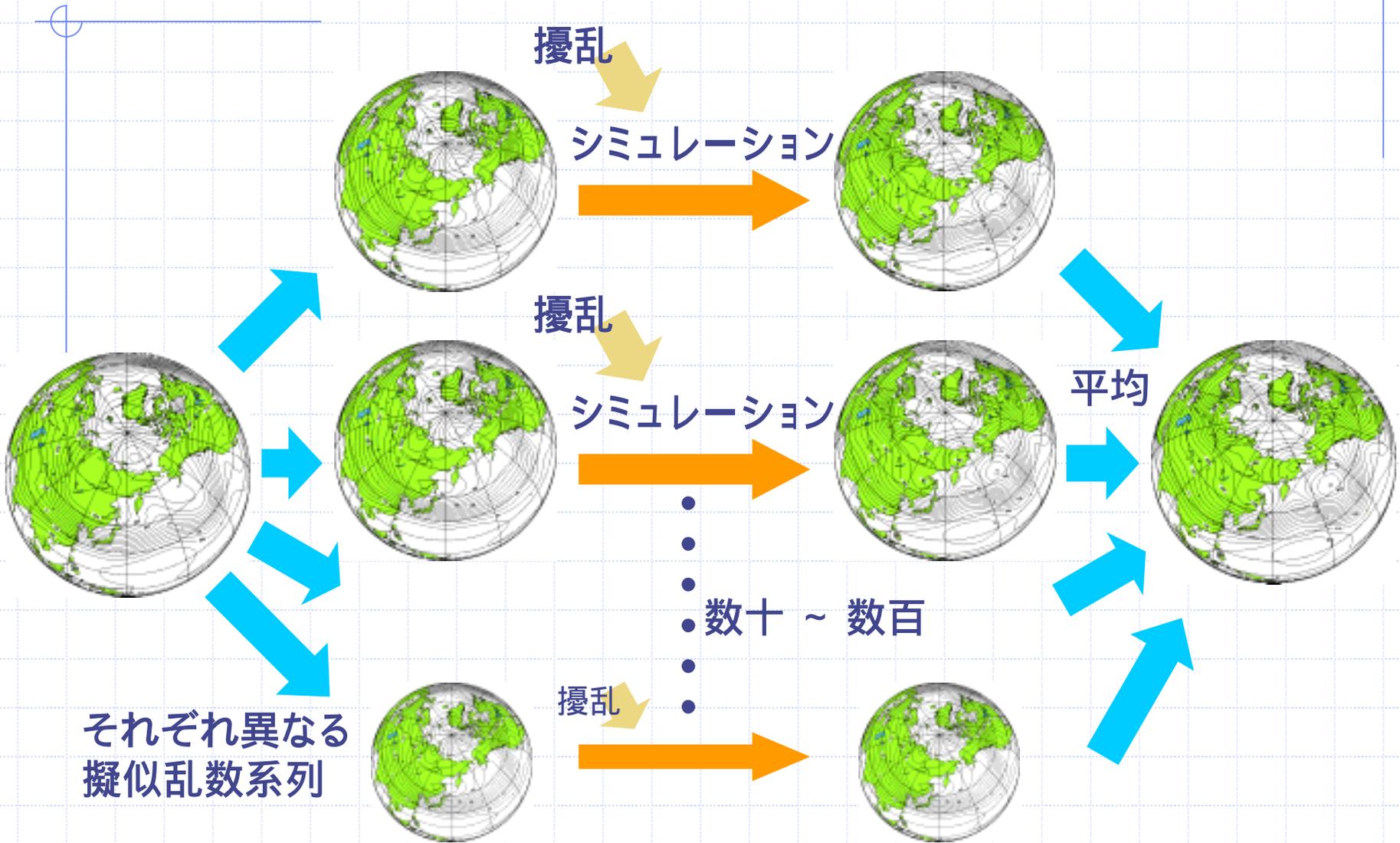
予報モデルとそのグリッド化

◆ 順圧 S-Model

- 中・長期予報のための大気大循環モデル
- 大気の鉛直平均場(順圧成分)を予測する
 - ◆ 順圧大気ではカオス性が抑制される。
予報限界を延ばしている。
- 中・長期の気象に大きな影響を与えるブロッキング高気圧を精度良く再現
- 筑波大 田中先生が開発
 - ◆ FORTRAN 77プログラム : smodel.f

アンサンブル予報

擾乱を加えつつ行った複数のシミュレーション結果の平均をとる



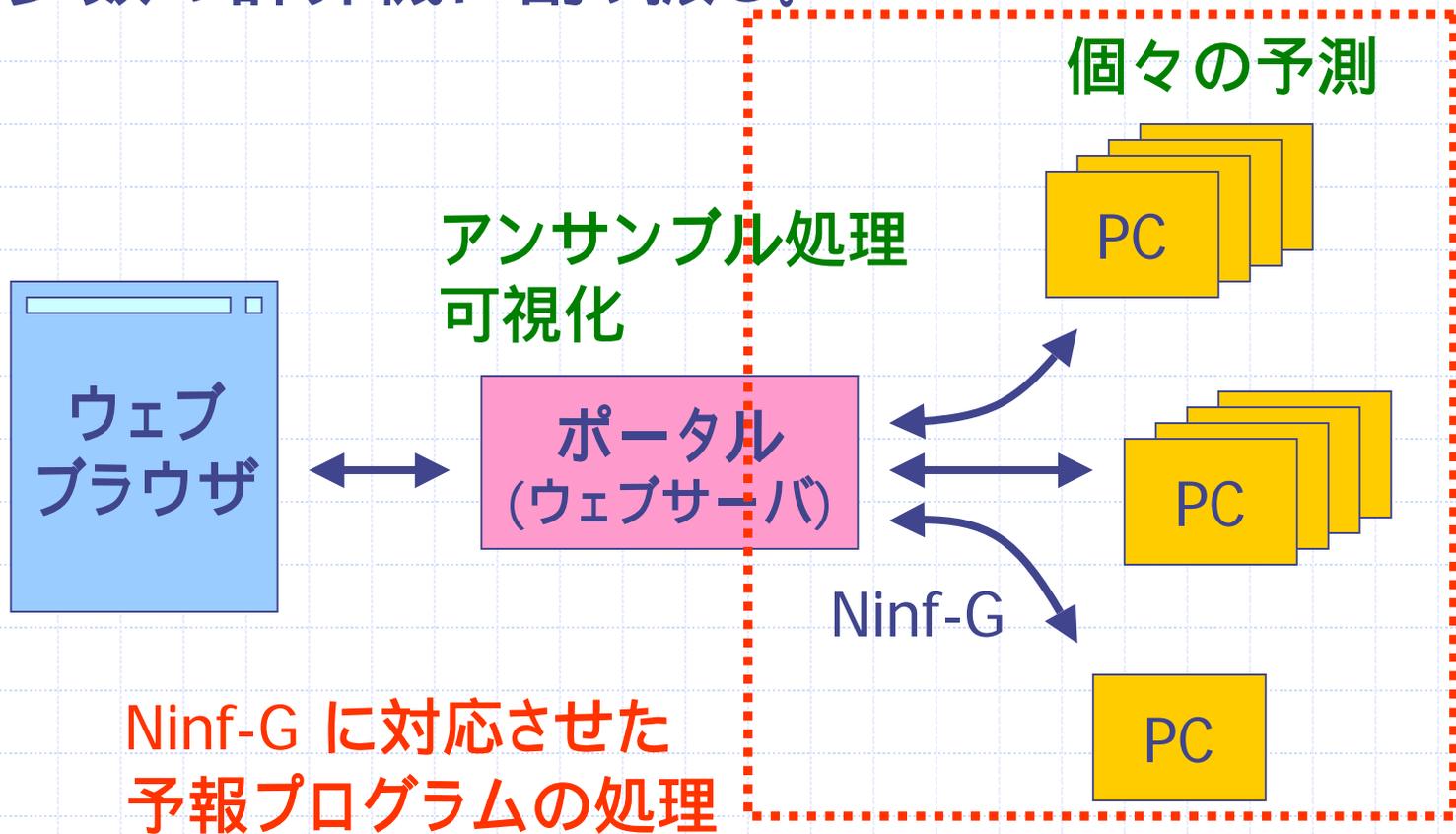
アンサンブル予報

- ◆ カオス的振る舞いによって生じる予報のずれを抑制できる。
 - 観測結果中の誤差や、計算時の丸め誤差は不可避
- ◆ 気象庁の場合、擾乱を加えた20～50の初期状態を用意している
予報限界が1日延びたと言われている。
- ◆ 各シミュレーションは完全に独立して行える。
 - 典型的なパラメータサーベイ。
 - 通信遅延への耐性が高い。

グリッド向き

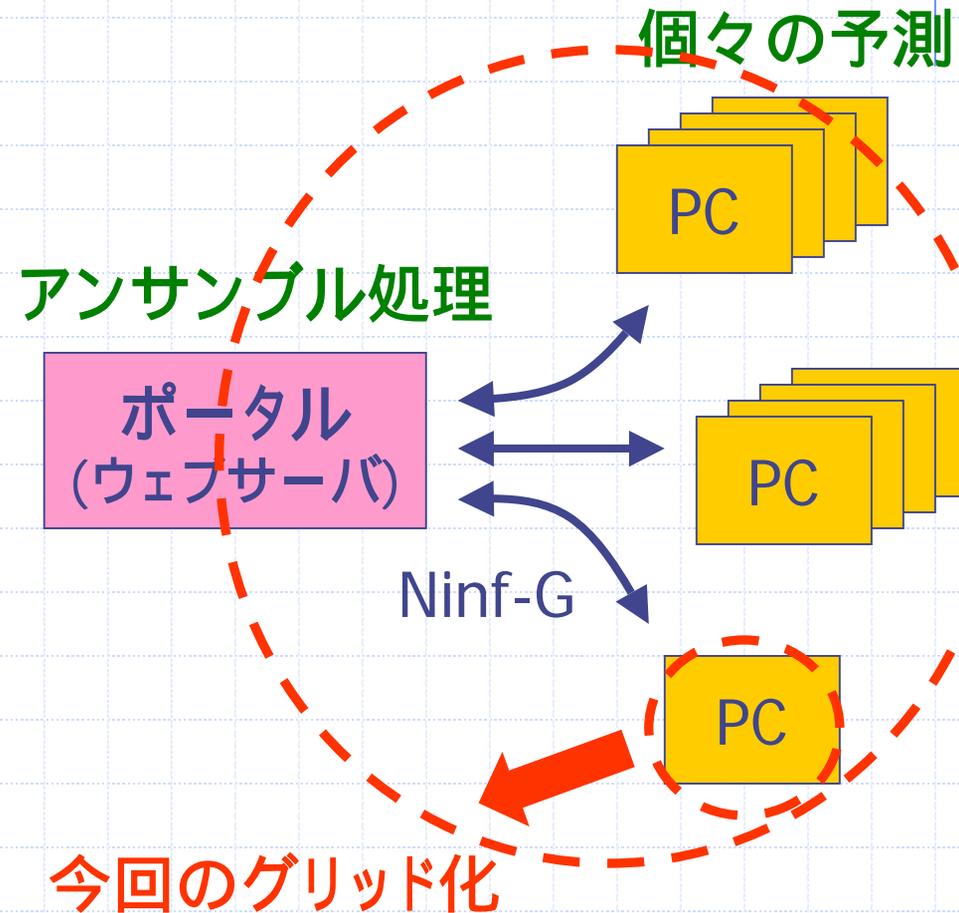
アンサンブル予報のグリッド化

- ◆ アンサンブル予報の各シミュレーションを、Ninf-G を用いた遠隔呼び出しで、多数の計算機に割り振る。



予報プログラムの Ninf-G 対応

- ◆ 逐次 FORTRAN プログラムを、Ninf-G を用いてグリッド化
 - 非同期遠隔呼び出しで並列処理
- ◆ アンサンブル予報の各シミュレーションを遠隔呼び出し
 - 順圧 S-Model で予報を行う



予報プログラムの Ninf-G 対応

1 日程度の作業で済んだ

1. caller ・ callee 間のデータ依存の除去

- Ninf-G では caller, callee間で共有できるデータは引数と返り値に限られる。
- FORTRANの COMMON変数を引数に変更した。

2. 関数呼び出しの、遠隔呼び出しへの書き換え

- GridRPC仕様の呼び出し関数 `grpc_call_async` や `grpc_wait_any` に置き換えた。
- 10 行程度の変更。

予報プログラムの Ninf-G 対応

3. スケジューリングルーチンの作成

- 現実装は、呼び出し先計算機を決定するメタサーバを利用していない。
そのため、caller側プログラムにセルフスケジューリングを実装。
- 50行程度の追加。

4. 呼び出しインタフェース定義ファイルの作成と、MDSへの登録

- インタフェース定義ファイルを Ninf-Gのns_genコマンドに与えることで、次のファイルが生成される：
 - ◆ MDSに登録する LDAP データ
 - ◆ callee側スタブプログラム `_stub_XXX_.c`

(上位)ミドルウェア Ninf-G の提供するもの

- ◆ Globus Toolkit 2.x の提供する機能はごくごく基本的なもの
 - ジョブ管理機能, API GRAM
 - 計算資源情報管理機能 MDS
 - 通信機能 Globus IO
- ◆ これらを直接用いてのアプリ開発は大変
 - 多大な労力を要する
 - bug-prone
 - デバッグも大変
- ◆ Ninf-G は、遠隔呼び出しプログラミングモデルという枠を与えることで、開発、保守、グリッド化を容易にする。

本ポータルの利用イメージ

- ◆ サインオン
- ◆ アプリケーション選択
- ◆ パラメータ入力
- ◆ ジョブ投入
- ◆ ジョブ一覧の閲覧
- ◆ 結果の取得

アプリケーション選択

アドレス(D)  https://koume.hpcc.jp/top.html 移動



AIST GridLib Virtual Computer Center

National Institute of Advanced Industrial Science and Technology

Portal User ID :
User:35

- MENU -

- Select Application
- Job Status
- Accounting Info
- Personal Info
- Sign off

Virtual Computer Center

Application Category

- All
- utility
- archiver
- Weather Forecast**
- phoenics

Search Application

Application List

Application Name	Category	Status	Application supplier
weather simulation(Ver. 1)	Weather Forecast	operating	AIST
weather simulation(Ver. 2)	Weather Forecast	operating	AIST
weather simulation(Ver. 3)	Weather Forecast	operating	AIST

   インターネット

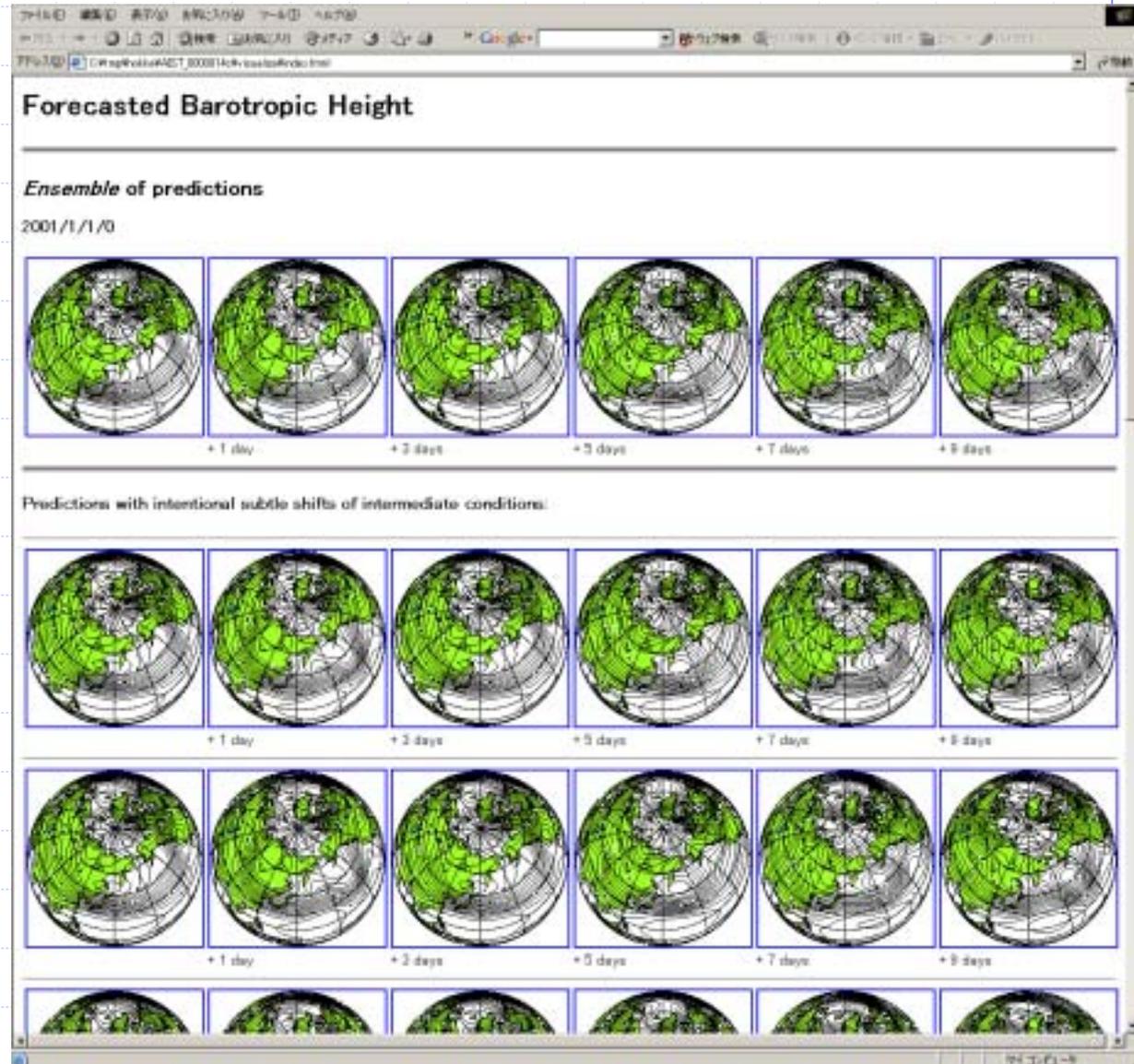
パラメータ入力

* start date	this type is 'string'. <input type="text" value="2001/01/01"/>
* start time	<input type="text" value="0"/> ▼
* end date	this type is 'string'. <input type="text" value="2001/01/11"/>
* end time	<input type="text" value="0"/> ▼
* number of simulation	this type is 'integer'. Range : $x \leq 100.0$ <input type="text" value="10"/>
* number of hosts: AIST(japan)	this type is 'integer'. Range : none <input type="text" value="1"/>
* number of hosts: KISTI(Korea)	this type is 'integer'. Range : none <input type="text" value="0"/>
* number of hosts: KU(Thai)	this type is 'integer'. Range : none <input type="text" value="0"/>

登録したアプリケーション記述ファイルから、パラメータ入力ウェブページが自動的に生成される。

結果の可視化

Ruby スクリプトから
可視化ツール GMT
を呼び出している。



広域テストベッドでの実験

PCクラスタ

産総研,
Japan

PCクラスタ

KISTI,
Korea

PCクラスタ

Kasetsart U.,
Thailand

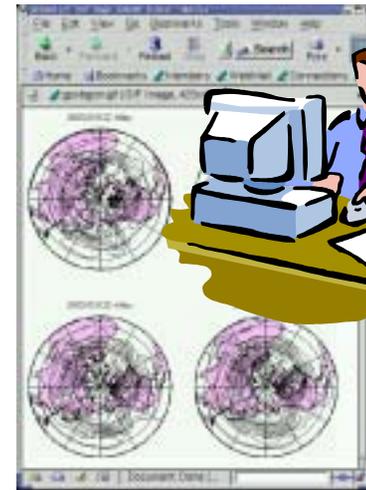


Asia-Pacific Grid

テストベッド

気象予報
ポータル

産総研,
Japan



SC2002会場,
Baltimore

広域テストベッドでの実験

◆ ApGrid テストベッドの一部

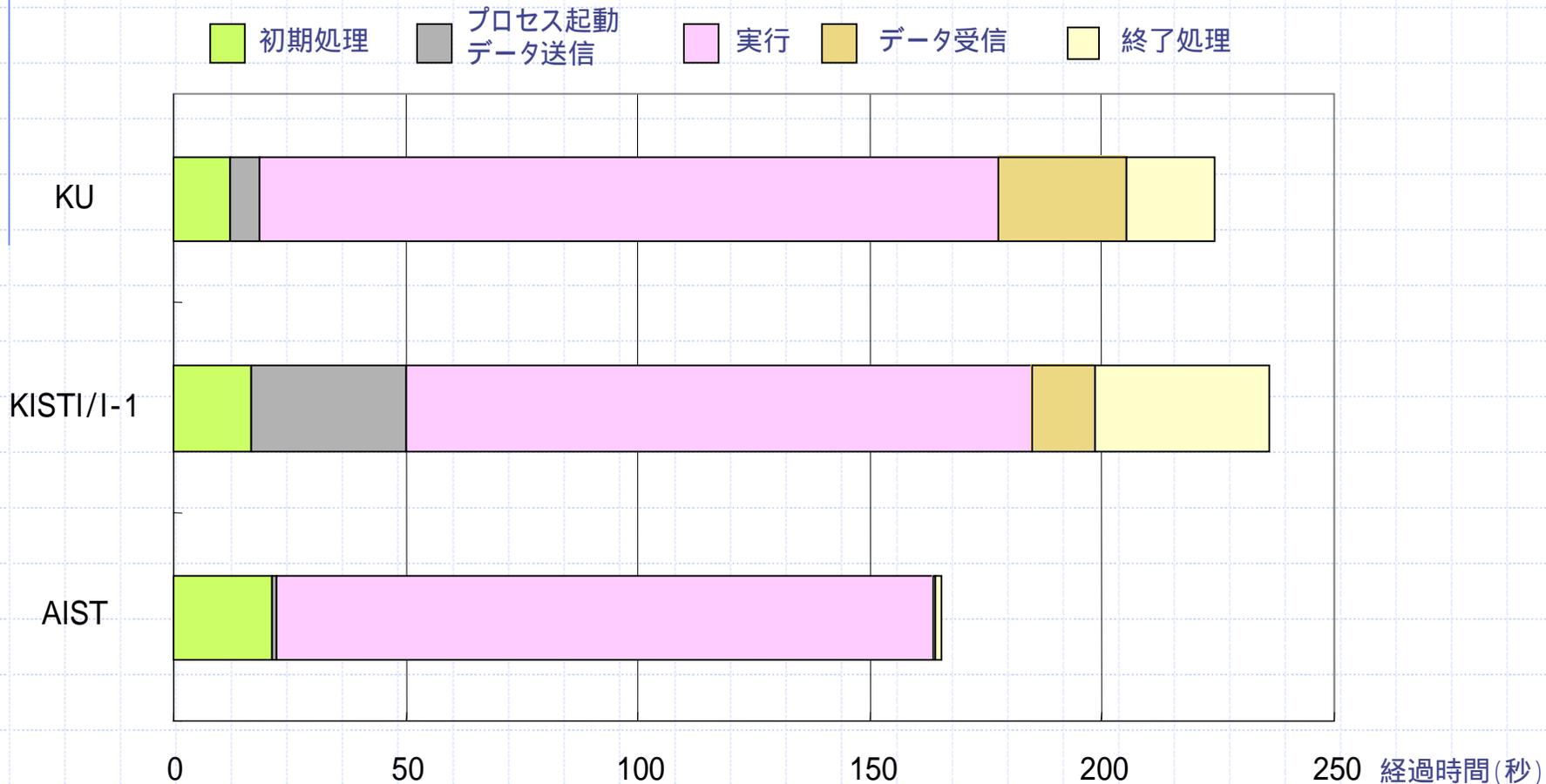
- www.apgrid.org
- 産総研 Linux クラスタ
- 韓国 KISTI Linux クラスタ
- タイ Kasetsart 大 Linux クラスタ

◆ 合計6台、158プロセッサ

◆ SC2002 (11月) にてデモ。

性能の予備評価

- ◆ 100日間のシミュレーションを 1台に対して呼び出し
- ◆ シミュレーション自体に要した時間はいずれも150秒程度
- ◆ 起動その他のオーバーヘッドが、その15%(LAN)から 30% or 40%(WAN)



性能の予備評価

◆ Ninf-G / Globus Toolkit を用いることによる 様々なオーバーヘッド

■ 初期処理

- ◆ MDS へのアクセスに 10 ~ 20秒

■ ジョブ起動

- ◆ KISTI ではジョブ管理システム PBS を用いてバッチ運用を行っている。GRAM での起動要求から実際の起動まで、20秒程度を要している。

■ データ転送

- ◆ KISTI, KU への呼び出しでは、データの送受信にそれぞれ10秒程度を要する。

■ 終了処理

- ◆ GRAM が callee の終了を検知するのに 10 ~ 30秒を要する。

注) GRAM: Globus Toolkit のジョブ管理機能

MDS: Globus Toolkit の計算資源情報管理機能

オーバーヘッド削減策

- ◆ MDS を利用せずに済むように、遠隔手続きの情報を caller 側にファイルとして用意しておく。
- ◆ GRAM が、起動したプロセスの終了を検知するために行うポーリングの間隔を短縮する。
 - 既定値は 30秒。

まとめ

- ◆ 既存 FOTRAN プログラムの**グリッド化とポータル化**を通して、容易さを検証した。
 - Ninf-G 対応(グリッド化)は 1日程度の作業で済んだ。
 - GridLib への搭載にあたっては、アプリ側の変更は一切不要であった。
 - ポータルの利用は、ウェブインタフェースで直観的に行えた。
- ◆ **ApGrid テストベッド**で動作させた。
 - グリッド化によるオーバーヘッドは、1回の遠隔呼び出しあたり、ときに数十秒に及んだ。
 - ◆ 改善の余地あり & 要改善
- ◆ **今後**
 - 実運用を通じて、GridLib、Ninf-G、グリッドアプリ一般、ポータル一般の問題を洗い出していく。