

Grid World 2005

## グリッドに必要なコンポーネント

合田 憲人  
東京工業大学 物理情報システム専攻



Kento Aida, Tokyo Institute of Technology

## グリッドの定義

- **定義** [<http://www.jpgrid.org/about/index.html>]
  - グリッドは広域ネットワーク上の計算、データ、実験装置、センサー、人間などの資源を仮想化・統合し、必要に応じて仮想計算機 (Virtual Computer) や仮想組織 (Virtual Organization) を動的に形成するためのインフラです。
- **グリッドのチェックリスト** [<http://www.gridtoday.com/02/0722/100136.html>]
  - 集中管理されていない分散した資源のコーディネーション
  - オープンスタンダードなプロトコルやインターフェースの利用
  - 単純には得られない質の高いサービスの提供

Kento Aida, Tokyo Institute of Technology

# グリッドを使ってできること

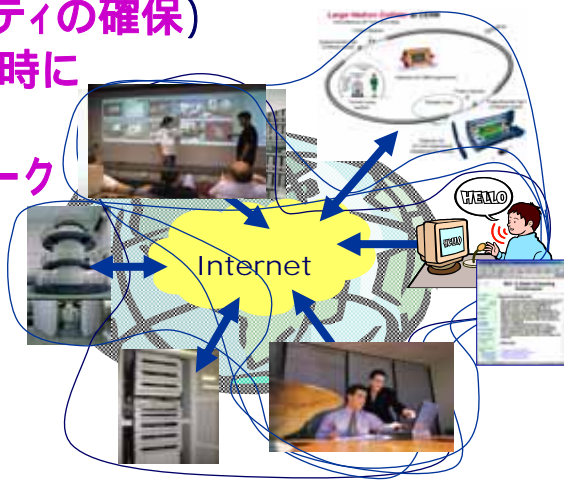
ネットワーク上の情報資源(サービス)を

- 安全に(セキュリティの確保)
- 安定して(必要な時に必要なだけ)
- 簡単に(ネットワークを意識しない)

利用する.



**仮想組織**



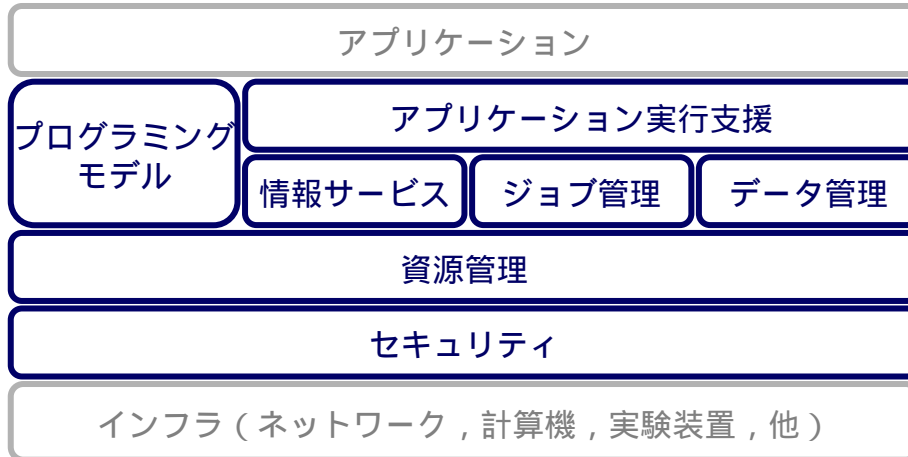
Kento Aida, Tokyo Institute of Technology

# グリッドの分類



Kento Aida, Tokyo Institute of Technology

## グリッドの要素技術

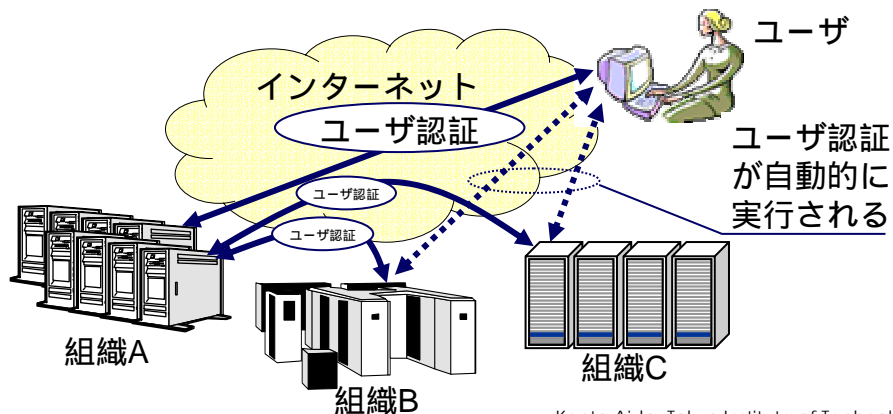


Kento Aida, Tokyo Institute of Technology

## セキュリティ(シングルサインオン)

### □シングルサインオン

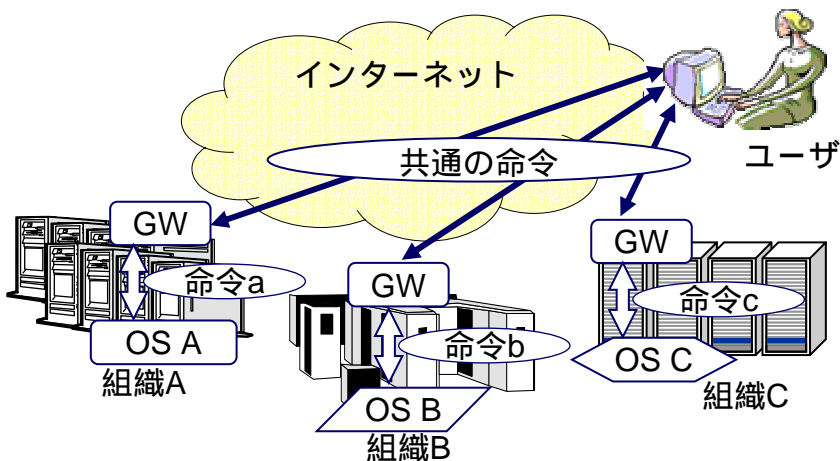
- 仮想組織上の1台のマシン上でユーザ認証
- 他のマシン上での認証は自動的に行われる。



Kento Aida, Tokyo Institute of Technology

# 資源管理

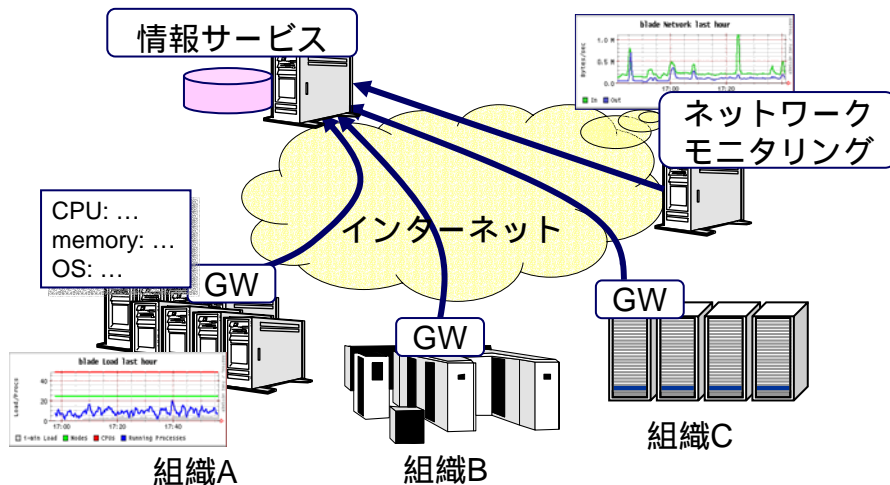
□マシンごとに異なる実行方式, 命令を吸収し, 共通のインターフェースを提供.



Kento Aida, Tokyo Institute of Technology

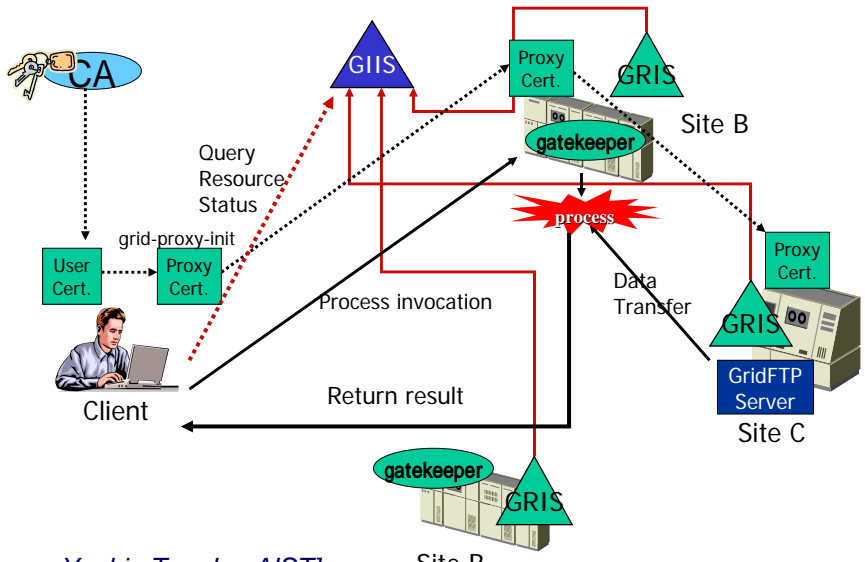
# 情報サービス

□グリッド上の資源情報の管理・提供



Kento Aida, Tokyo Institute of Technology

# Big picture of the GT2

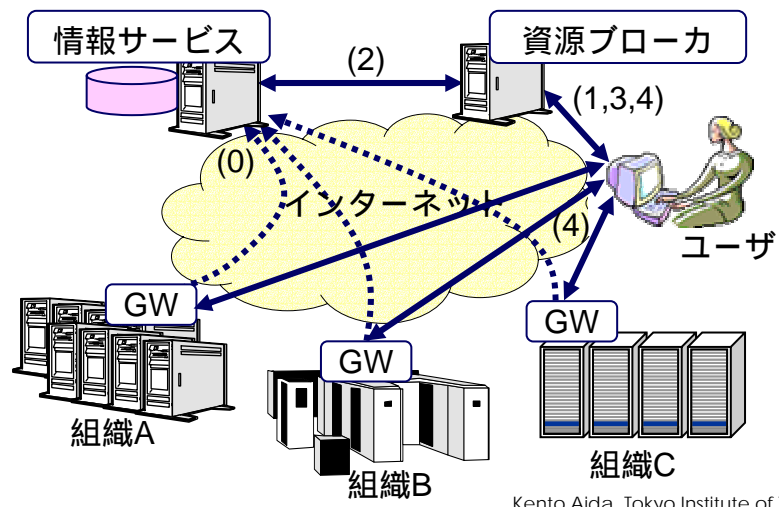


[source: Yoshio Tanaka, AIST]

Kento Aida, Tokyo Institute of Technology

# ジョブ管理

□ 資源選択, スケジューリング, ジョブの実行制御



Kento Aida, Tokyo Institute of Technology

## Condor

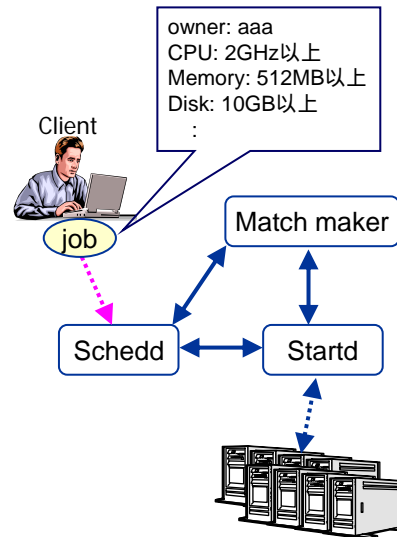
### □ High Throughput Computing

□ ClassAds mechanism  
によるジョブ要求と計算資源のマッチング

□ Check pointingによる耐故障性

### □ Globus上への実装

□ Condor-G



[ <http://www.cs.wisc.edu/condor/> ]

Kento Aida, Tokyo Institute of Technology

## Nimrod

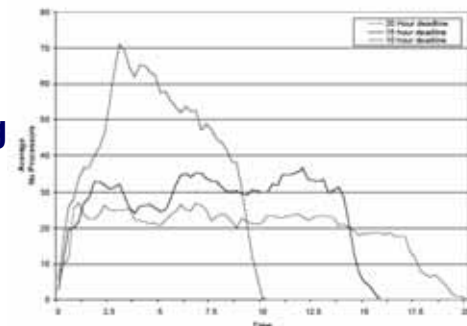
### □ パラメータサーベイ型アプリケーションを対象

□ computational  
economy

□ deadline scheduling

### □ Globus上への実装

□ Nimrod/G



[source: D. Abramson, et al., "High Performance Parametric Modeling with Nimrod/G: Killer Application for the Global Grid?," IPDPS2000, 2000 ]

[ <http://www.csse.monash.edu.au/~davida/nimrod.html/> ]

Kento Aida, Tokyo Institute of Technology

## スケジューリング / モニタリング

### □ AppLes [ <http://grail.sdsc.edu/> ]

□ スケジューリングエージェントを使ったグリッド上でのアプリケーションスケジューリング方式を提案

□ 活動はGRAIL (SDSC/UCSD) へ継承

➢ スケジューリングツール, シミュレーション, プログラミングツール, アプリケーション, ...

### □ NWS [ <http://nws.cs.ucsb.edu/> ]

□ グリッド上の計算資源, ネットワーク資源のモニタリングツール

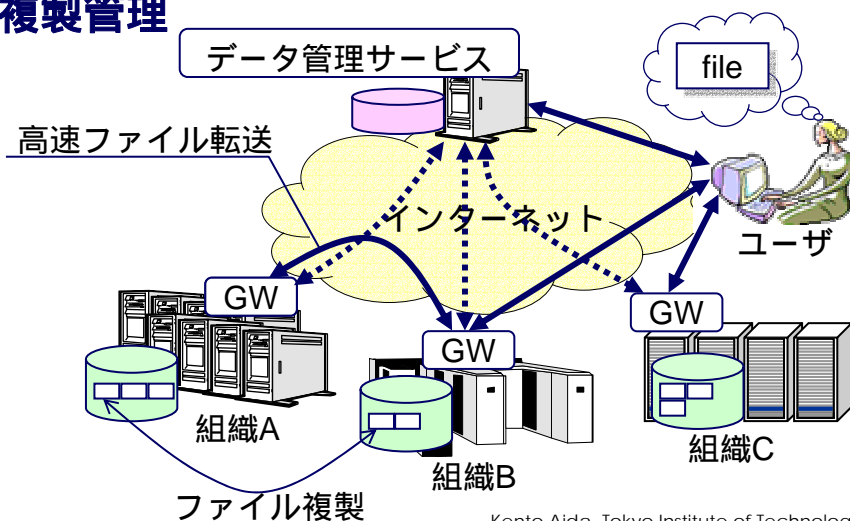
□ 統計的手法に基づく負荷予測

□ スケジューリングツールとの連携

Kento Aida, Tokyo Institute of Technology

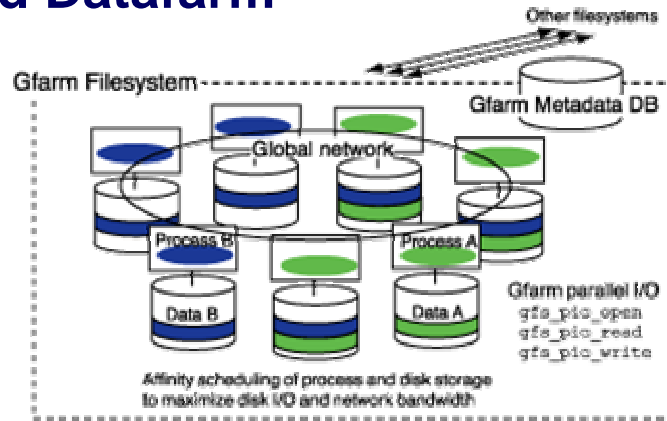
## データ管理

□ 分散ファイル管理, 高速ファイル転送, 複製管理



Kento Aida, Tokyo Institute of Technology

# Grid Datafarm

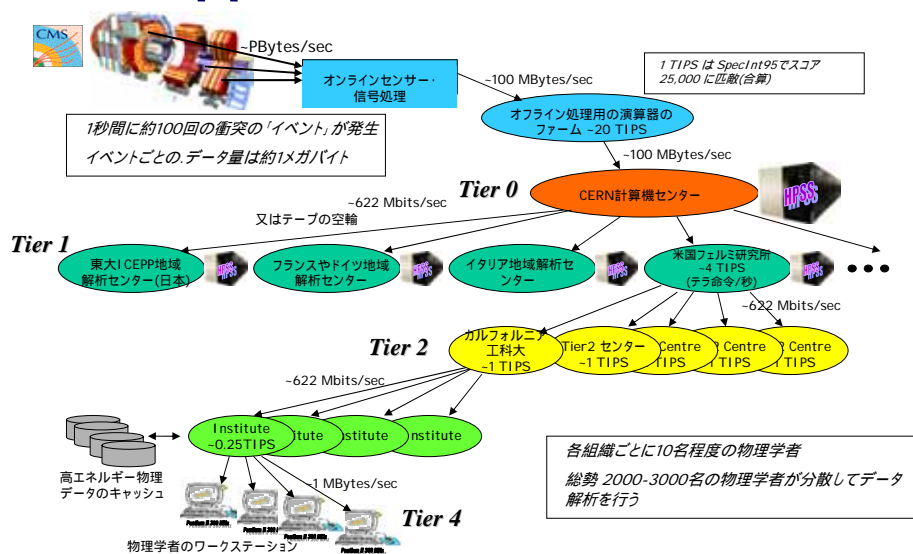


- Peta-to-Exascale Global Filesystem on unified CPU/storage cluster
- Parallel I/O and parallel processing with local I/O scalability

[source: Osamu Tatebe, AIST]

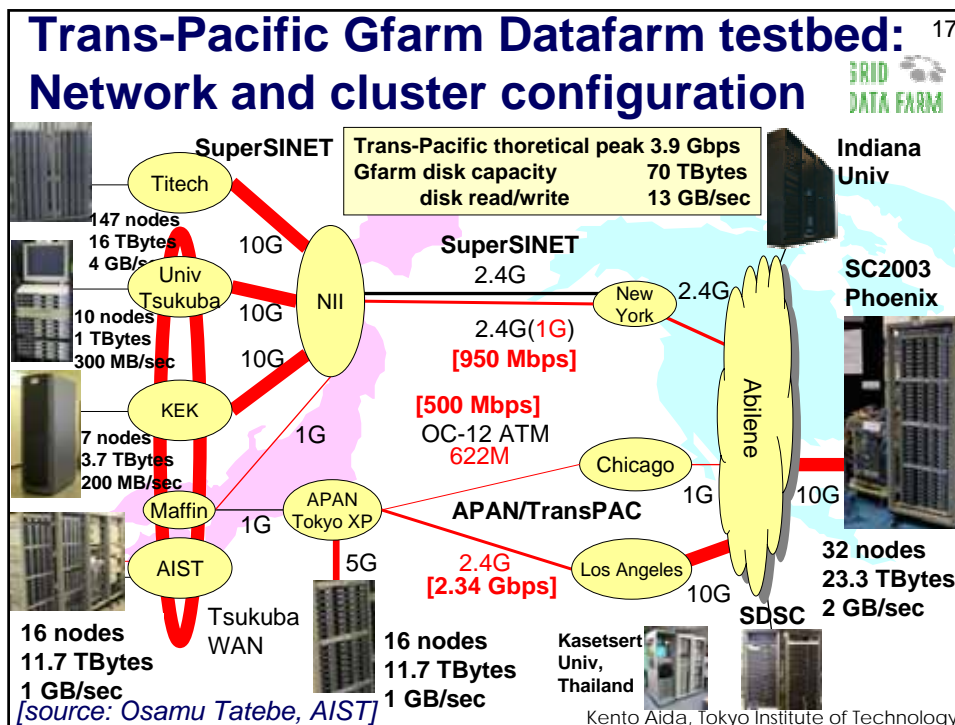
Kento Aida, Tokyo Institute of Technology

# HEP Application



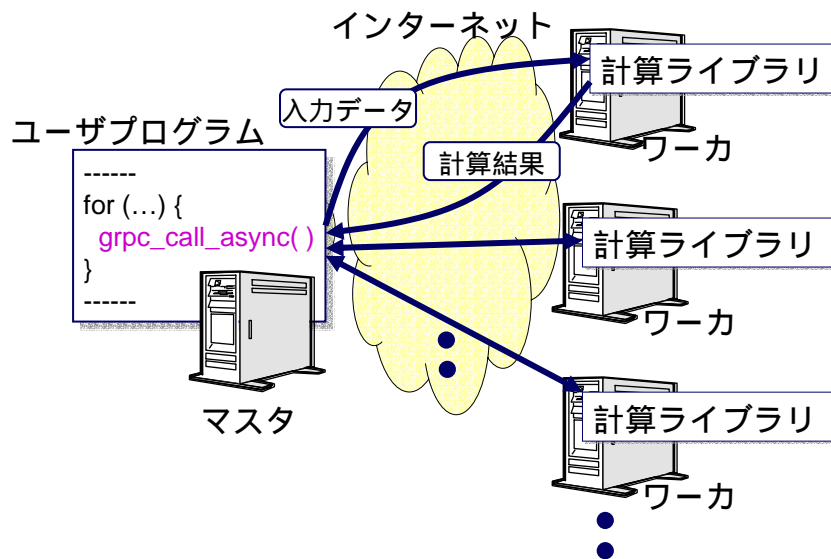
[source: Harvey Newman, CalTech]

Kento Aida, Tokyo Institute of Technology



- 18
- ## プログラミングツール
- MPI
    - MPIによるプログラミングモデル
      - MPICH-G2, GridMPI, ...
  - GridRPC
    - RPCに基づくグリッド上のプログラミングモデル
      - Ninf-G, OmniRPC, NetSolve, ...
  - Master Worker Template
    - マスタ・ワーカ型プログラムを開発するためのテンプレート
      - MW, AMWAT, ...
- Kento Aida, Tokyo Institute of Technology

## GridRPC



Kento Aida, Tokyo Institute of Technology

## GridRPC (続)

- **Ninf-G** [ <http://ninf.apgrid.org/> ]
  - GridRPCのリファレンスインプリメンテーション
  - Globus Toolkit上に構築
    - グリッド上のセキュリティに関する機能(ユーザ認証, 通信の暗号化等)はGlobusの機能を使用.

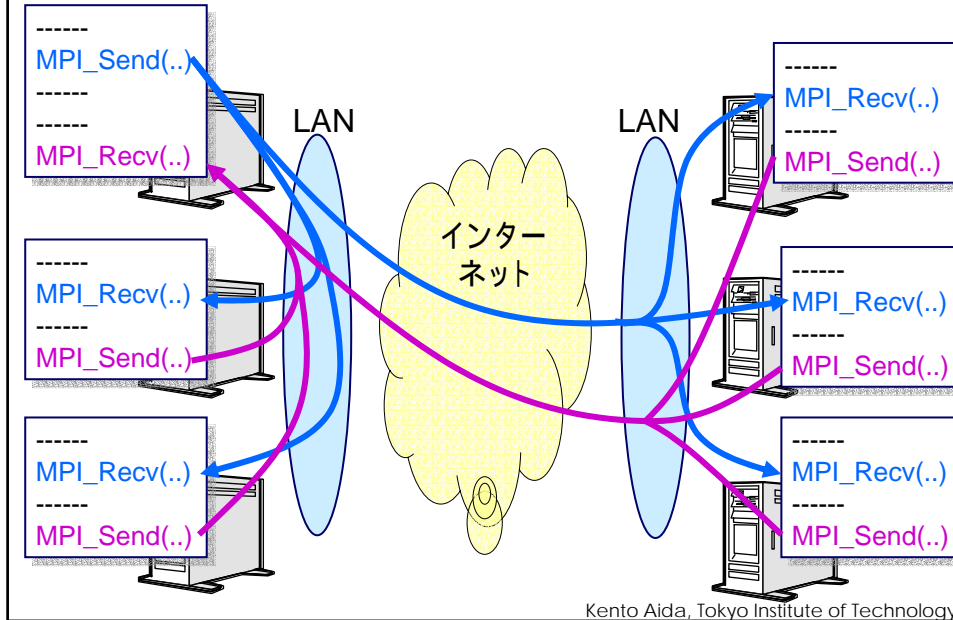
```
for (i = start; i <= end; i++) {
  SDP_search(argv[1], i, &value[i]);
}
```



```
grpc_function_handle_init(&hdl, ..., "SDP/search");
for (i = start; i <= end; i++) {
  grpc_call_async(&hdl, argv[1], i, &value[i]);
}
```

Kento Aida, Tokyo Institute of Technology

## グリッド上でのMPIプログラミング



## アプリケーション実行支援

### □ポータル

- グリッド上で提供されているアプリケーションの探索, 実行, また実行中のアプリケーションの監視や制御を行うための窓口的な機能を提供.

➢Webページ

- ワークフローとの連携

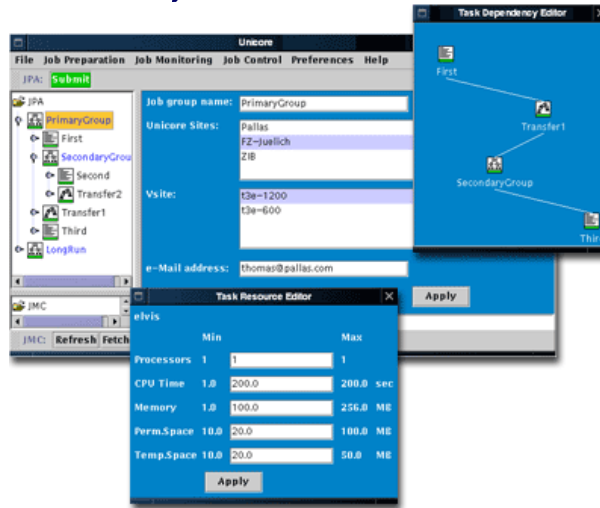
### □ワークフロー

- アプリケーション間の実行順序や入出力による制約をDAGにより表現.
- ワークフローエンジンによる複数アプリケーションの実行制御.

Kento Aida, Tokyo Institute of Technology

# アプリケーション実行支援環境の例 (UNICORE)

23



[source: <http://www.unicore.org/unicore.htm>]

Kento Aida, Tokyo Institute of Technology

# 並列処理技術とグリッド技術の比較

24

	並列処理	グリッド
インフラ	<ul style="list-style-type: none"> <li>単一組織内の計算資源</li> <li>複数の計算資源を用いて高性能計算を実行するための計算環境を提供.</li> </ul>	<ul style="list-style-type: none"> <li>複数組織に分散した情報資源</li> <li>インターネットに接続された情報資源を安全に, 安定して, 簡単に利用するための情報サービスを提供.</li> </ul>
利用目的	高性能処理の実行. ユーザの利益は, 処理時間短縮, 信頼性向上.	情報サービスの利用. ユーザの利益は, 処理時間短縮, 信頼性向上, 利便性向上, 対費用効果向上等, 多様化.
処理の内容	<ul style="list-style-type: none"> <li>計算 (並列計算)</li> <li>データ処理 (分散データベース)</li> </ul>	<ul style="list-style-type: none"> <li>計算 (グリッドコンピューティング)</li> <li>データ処理 (データグリッド)</li> <li>実験・測定</li> <li>遠隔会議 (アクセスグリッド)</li> </ul>
鍵となる技術 (取り組むべき研究課題)	<ul style="list-style-type: none"> <li>アーキテクチャ (計算機アーキテクチャ, ネットワーク)</li> <li>システムソフトウェア (OS, コンパイラ, ランタイムシステム, プログラミング環境, 実行支援環境)</li> <li>並列化技法 (スケジューリング, 最適化アルゴリズム)</li> <li>応用</li> </ul>	<ul style="list-style-type: none"> <li>ミドルウェア (セキュリティ, 資源管理, 情報サービス, ジョブ管理, データ管理, プログラミング環境, 実行支援環境)</li> <li>インフラ構築</li> <li>応用</li> </ul>

Kento Aida, Tokyo Institute of Technology

## まとめ

- グリッドは様々な要素技術(サービス)の集合
  - セキュリティ, 情報サービス, 資源管理
  - ジョブ管理, スケジューリング
  - プログラミングツール
  - アプリケーション実行支援
- 複数サービスの構成方法
  - GGFでの議論
    - OGSA, WSRF
  - 実用化にむけた研究