2005年前期　大学院科目

広域分散アプリケーション特論

月曜日　3時限

場所：情報基盤センター3F 多目的講義室

担当　青柳　睦

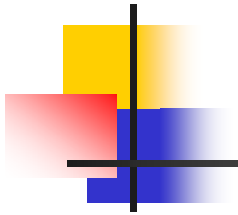
aoyagi@cc.kyushu-u.ac.jp

6月13日（月）

講義の内容，成績評価方針(server-500.cc.kyu・・)

サイエンスGrid　NAREGI

- NAREGIミドルウェアの概要
- 連成計算の概要



講義の内容

- グリッドの概要
  - Gridコンピューティングとは
  - サイエンス分野での利用
  - ビジネス分野での利用
- 計算科学の概論
  - 主要なシミュレーション手法
- サイエンスGrid　NAREGI
  - Globus, Unicoreの現状
  - NAREGIミドルウエア概要
  - 連成計算とその類型化
- Globus Tool Kit version 4

GT4の動向に依存・・

NAREGIミドルウエアのデモに変更するかもしれない

講義資料はWebで公開
server-500.cc.kyushu-u.ac.jp

NAREGIプロジェクトの概要

基本層（WP1　東工大　松岡教授）

- スーパー・スケジューラ
- 分散情報サービス
- グリッドVM

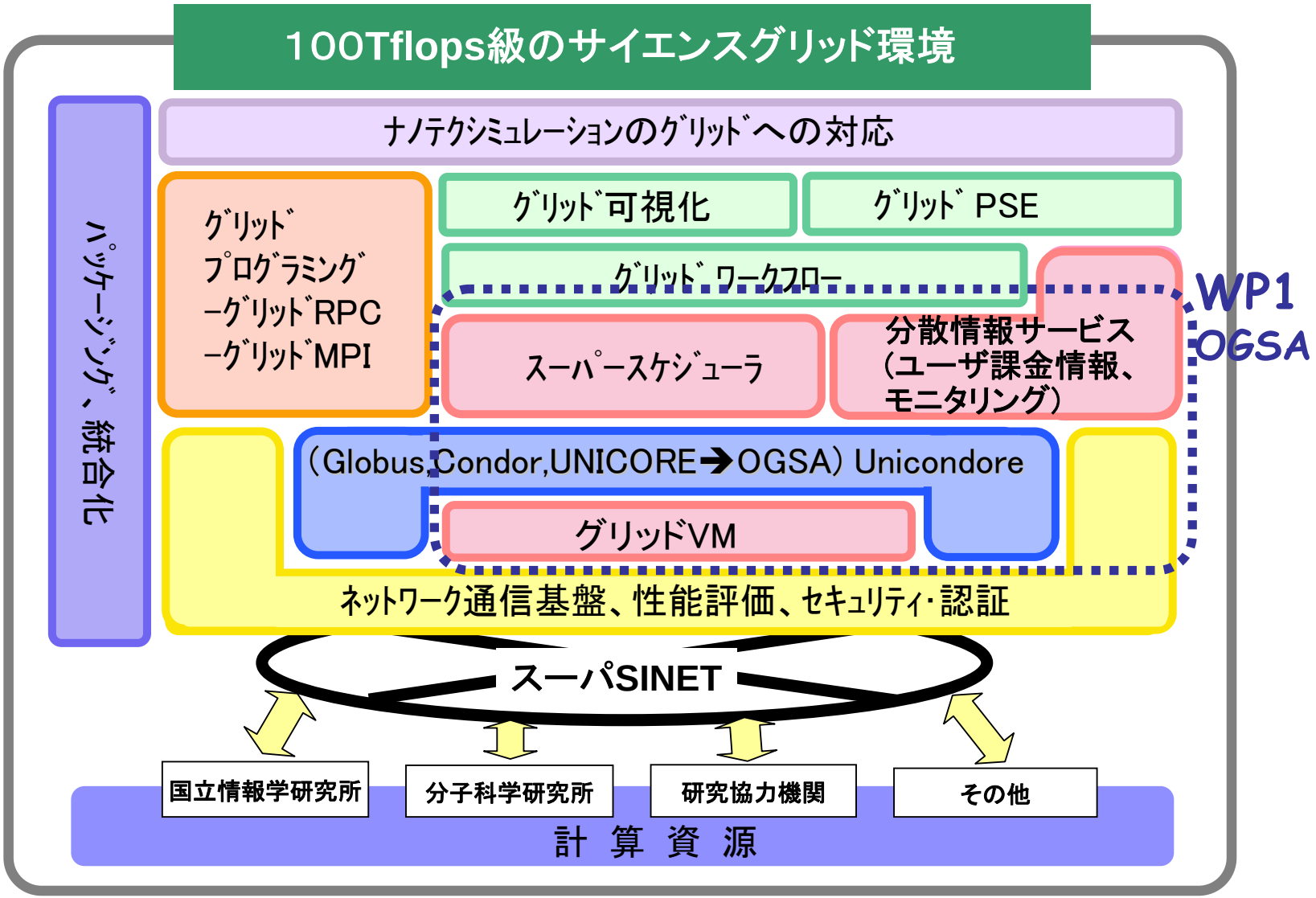
上位層
WP3　国情研　宇佐見教授
- PSE
- ワークフローツール
- グリッド可視化

Programing層
WP2　産総研　関口博士
- GridMPI
- GridRPC

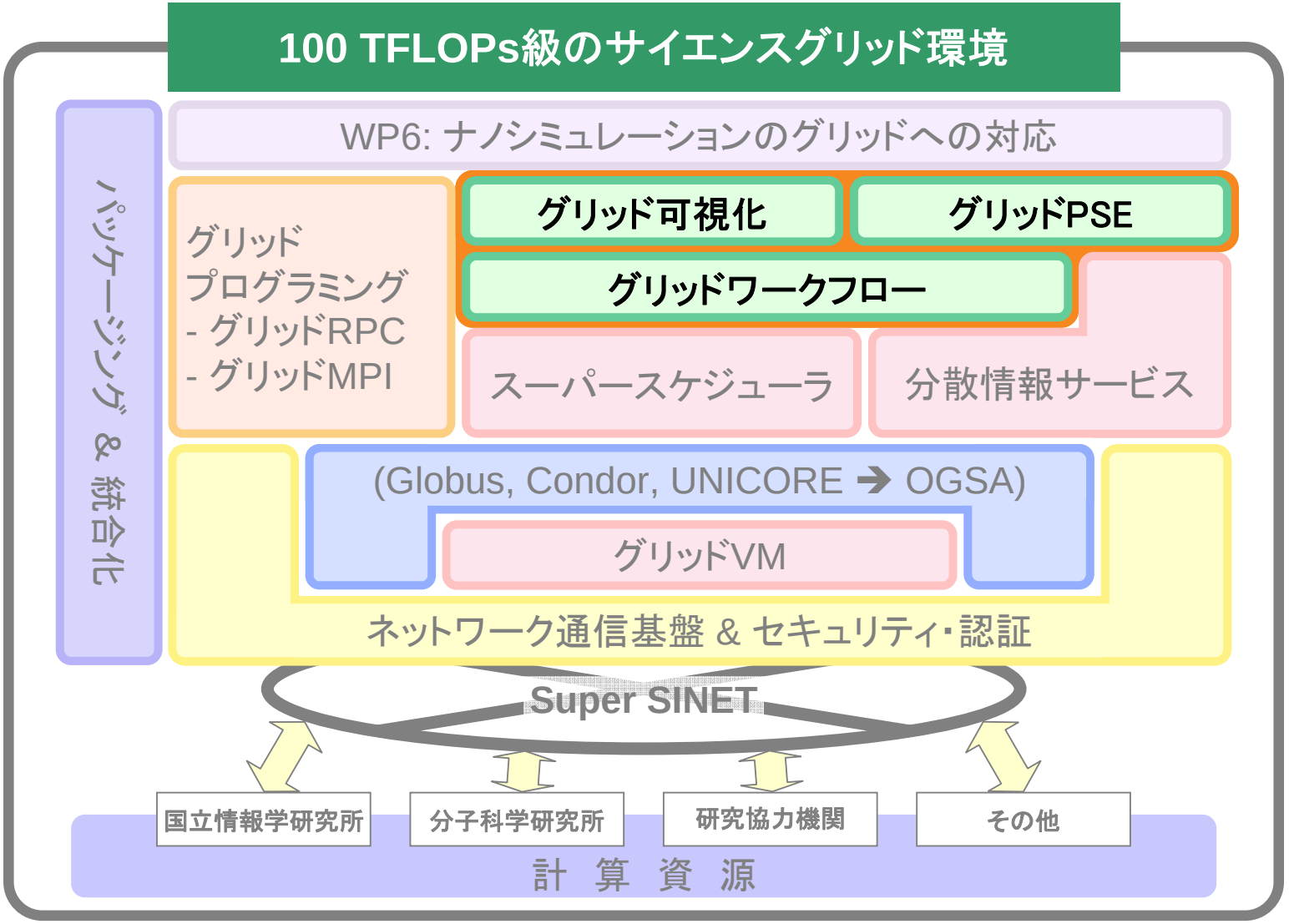
ネットワーク＆認証層
WP2　大阪大学　下條教授
- PKI
- 認証連携 VO対応

Application層
WP6　九州大学　青柳
- Mediator
- 連成計算実証試験

NAREGIソフトウェア階層とWP1 (先週の話題)

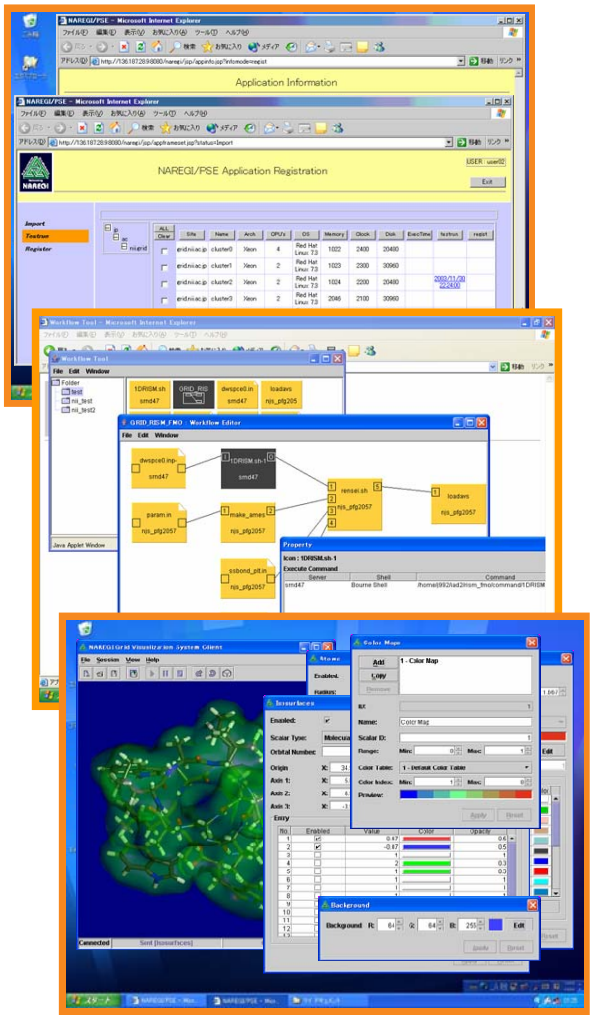


NAREGIにおける上位層

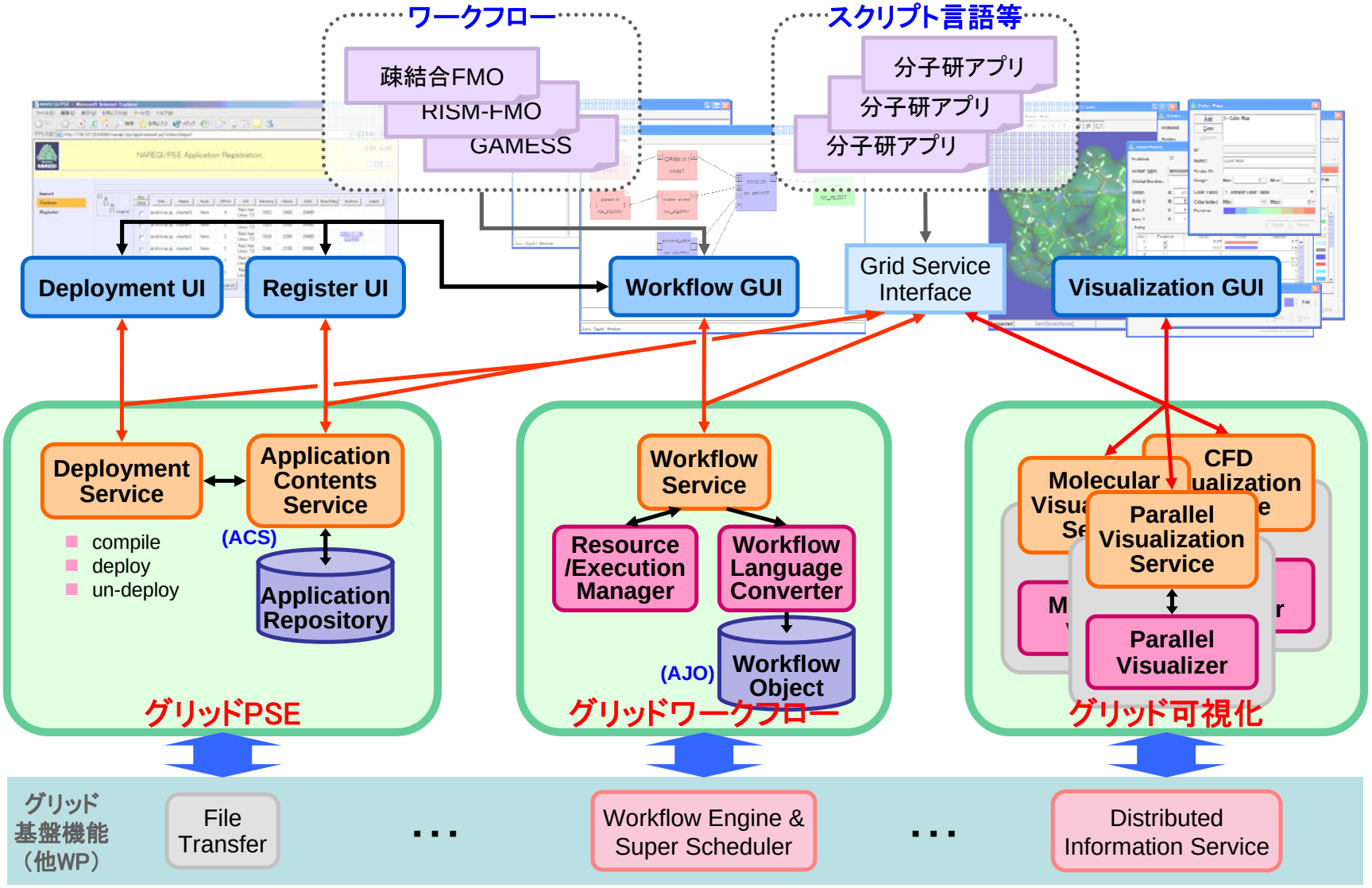


機能の概要

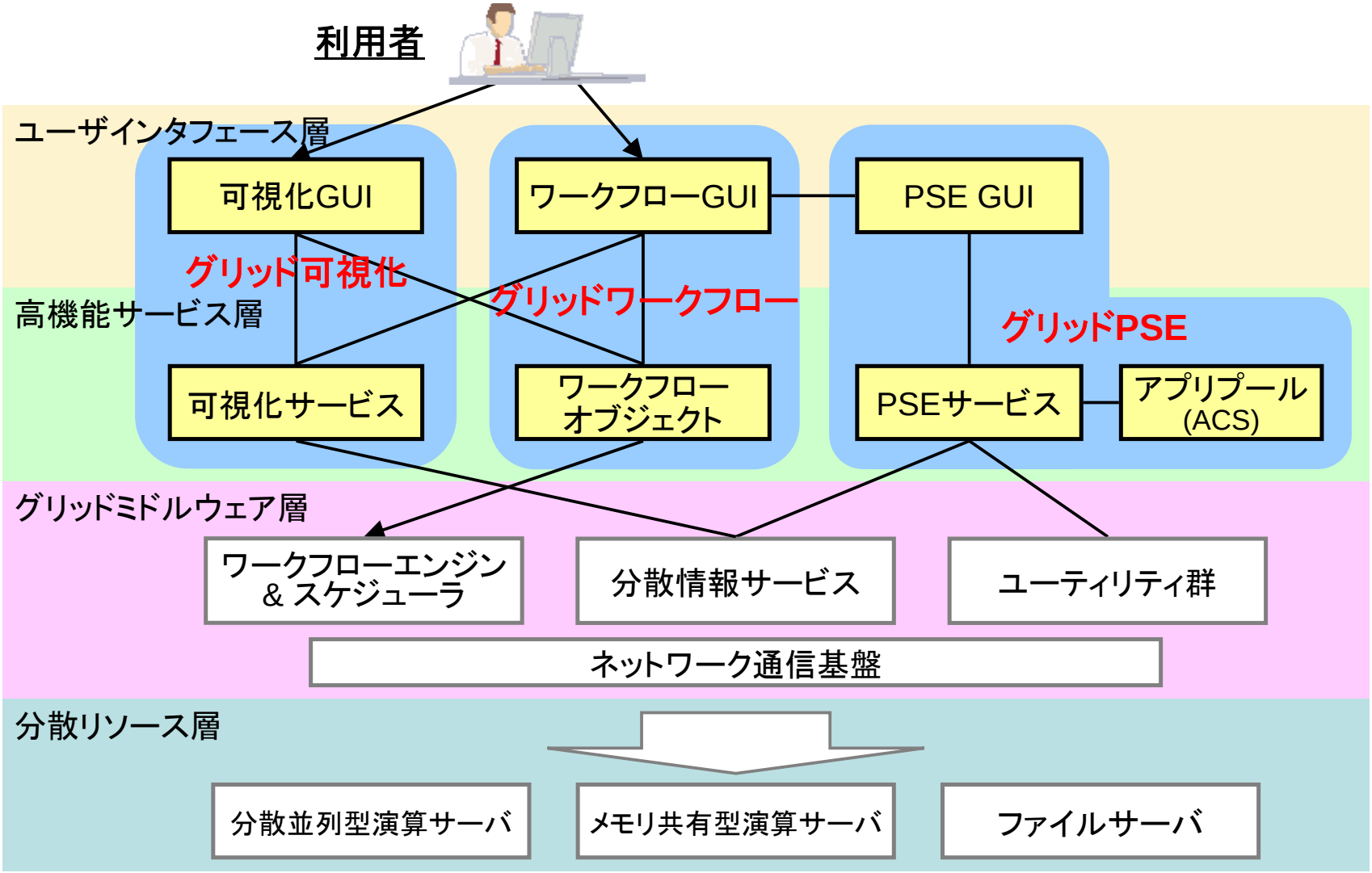
- **グリッドPSE**
  - グリッド環境へのアプリ配置
  - ユーザ要求/アプリ情報による実行支援
- **グリッドワークフロー**
  - 特定のグリッドミドルウェアに依存しないワークフロー言語
  - タスクフロー表現によるGUI
- **グリッド可視化**
  - グリッド環境上に散在した大規模データのリモート可視化
  - 汎用的な可視化グリッドサービス



システムの概要


ワークフロー
疎結合FMO
RISM-FMO
GAMESS
スクリプト言語等
分子研アプリ
分子研アプリ
分子研アプリ
Deployment UI
Register UI
Workflow GUI
Grid Service Interface
Visualization GUI
Deployment Service
Application Contents Service
(ACS)
compile
deploy
un-deploy
Application Repository
グリッドPSE
Workflow Service
Resource /Execution Manager
Workflow Language Converter
(AJO)
Workflow Object
グリッドワークフロー
Molecular
CFD
Parallel Visualization Service
Parallel Visualizer
グリッド可視化
グリッド基盤機能（他WP）
File Transfer
...
Workflow Engine & Super Scheduler
...
Distributed Information Service

相互関連図


利用者
ユーザインタフェース層
可視化GUI
ワークフローGUI
PSE GUI
グリッド可視化
グリッドワークフロー
グリッドPSE
高機能サービス層
可視化サービス
ワークフロー
オブジェクト
PSEサービス
アプリプール
(ACS)
グリッドミドルウェア層
ワークフローエンジン
& スケジューラ
分散情報サービス
ユーティリティ群
ネットワーク通信基盤
分散リソース層
分散並列型演算サーバ
メモリ共有型演算サーバ
ファイルサーバ

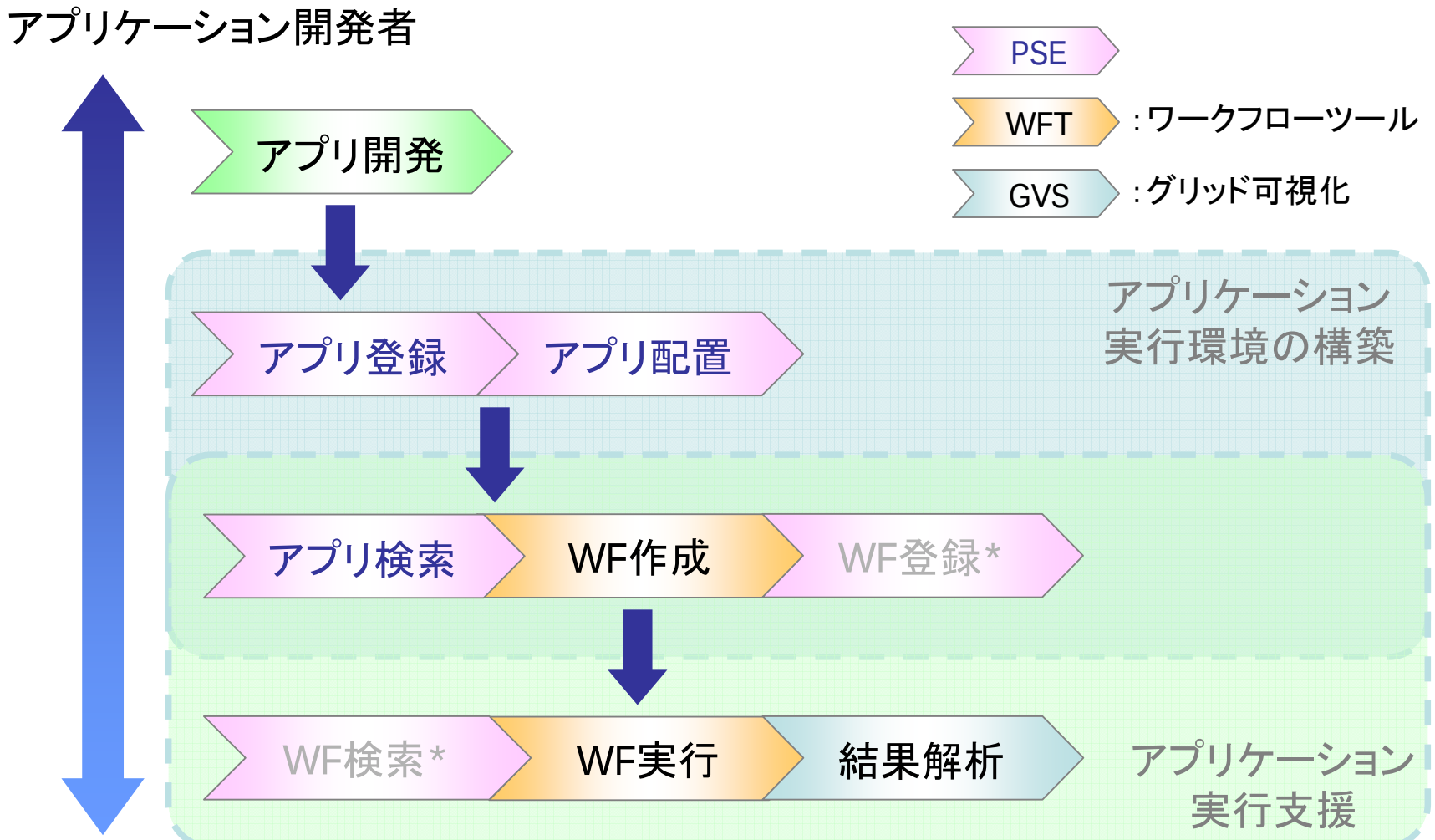
グリッドPSE：
NAREGI-PSEの背景

- アプリケーションユーザの視点：グリッド時代の研究活動
  - グリッドの知識を必要とせずともシミュレーション。
  - 各所にあるアプリケーションを容易に利用したい。
  - アプリケーションを容易に提供したい。
  - アプリケーション・シミュレーション過程と結果の再利用を行いたい。
- グリッドによるアプリケーションユーザのパラダイムシフトを支援
  - グリッドを意識することなく、グリッドを利用した研究活動ができる環境を、研究者に提供することを目指す
  - アプリケーション開発者
    - 開発したアプリケーションを、容易にグリッドへ配置・登録
  - アプリケーション利用者
    - グリッドを意識することなく、アプリケーションを実行

グリッドPSE：
NAREGI-PSEの機能

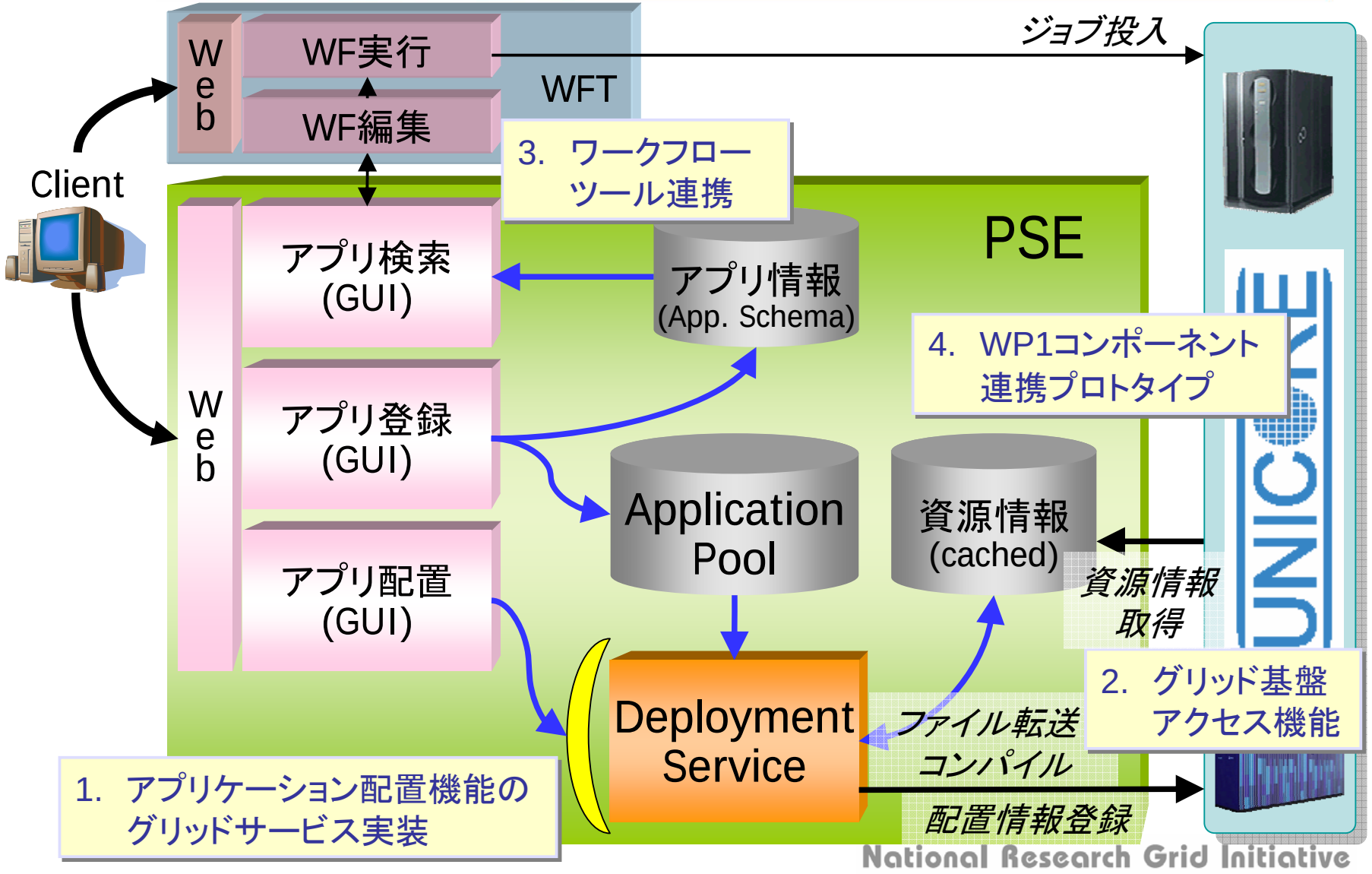
- グリッドにおけるアプリケーション実行環境の構築
  - アプリケーション配置先資源の選択を支援
  - ソースファイル：転送～コンパイル～テスト実行→登録
  - ロードモジュール：転送～テスト実行→登録
- アプリケーションの再利用支援
  - 登録アプリケーションの公開範囲を設定
  - アプリケーション情報の一つとして資源要件を管理
- NAREGIアプリケーションスキーマ
  - 「アプリケーション」をグリッドの資源として管理
- 利用者インタフェース
  - アプリケーション登録・検索・配置のGUI
  - グリッドサービス・インタフェース（Deployment）

グリッドPSE：
NAREGI-PSE利用シナリオ

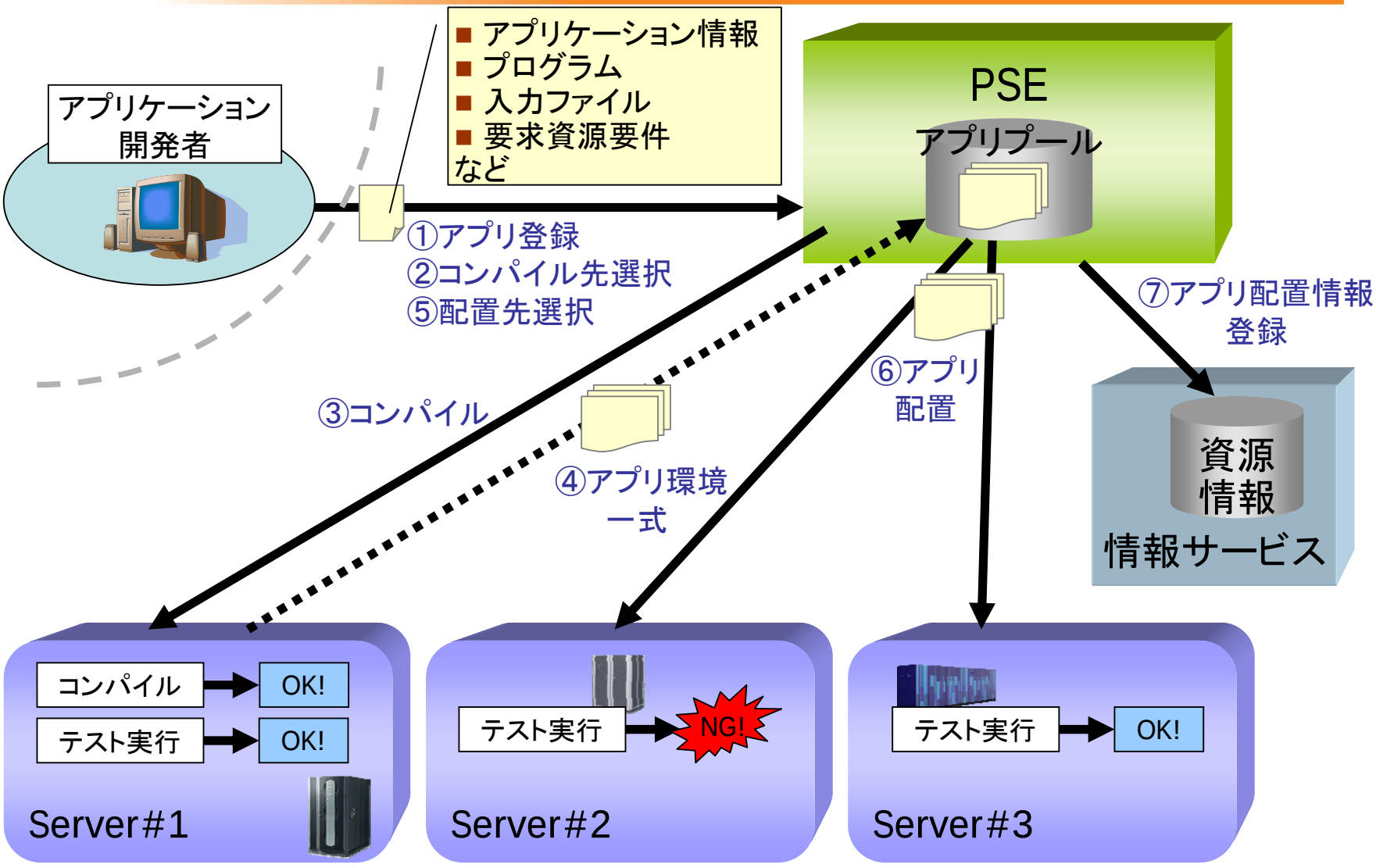


WF：ワークフロー　*：予定

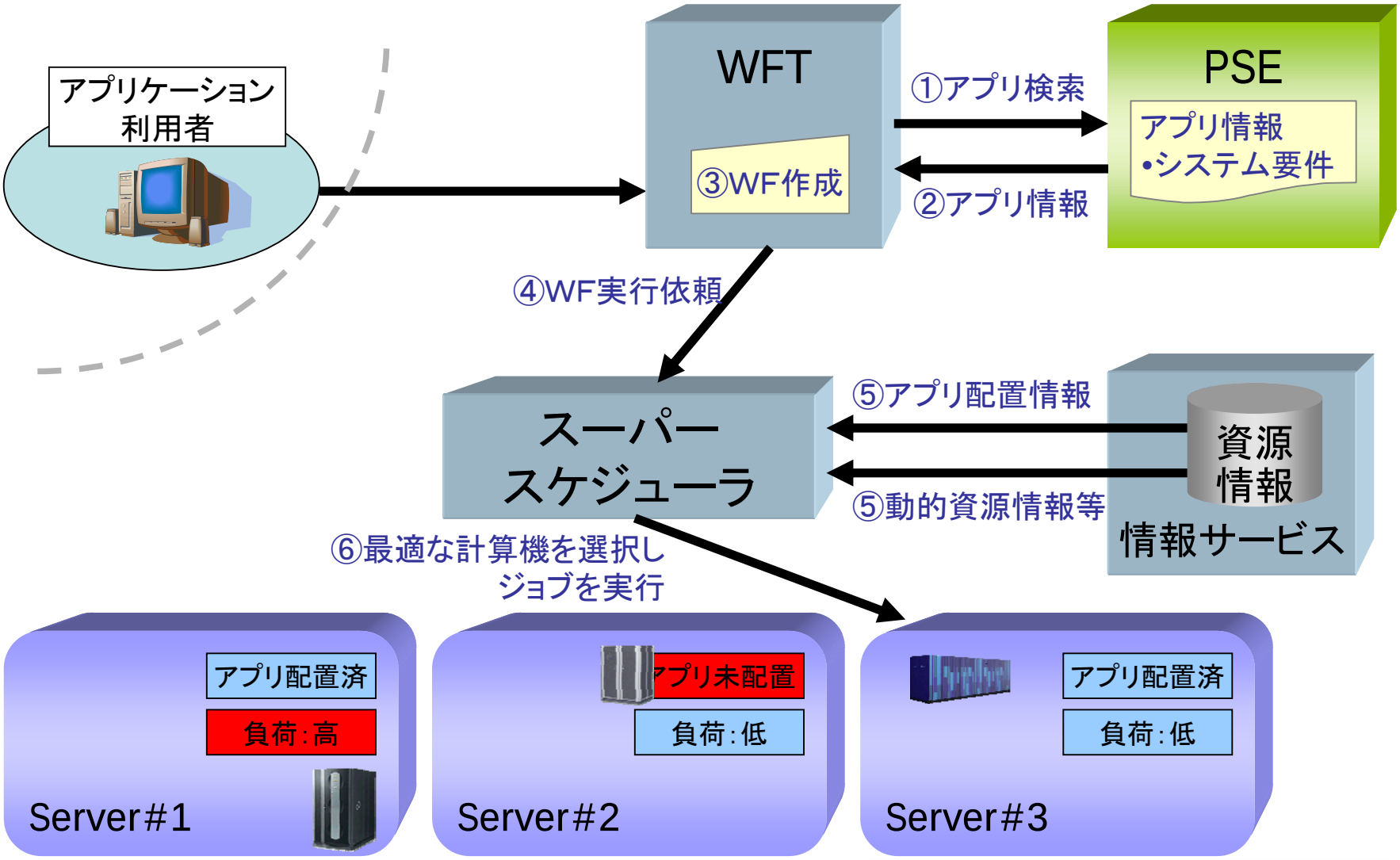
グリッドPSE：システム概要



グリッドPSE：
アプリケーション登録／配置・利用例



グリッドPSE：
アプリケーション実行・利用例

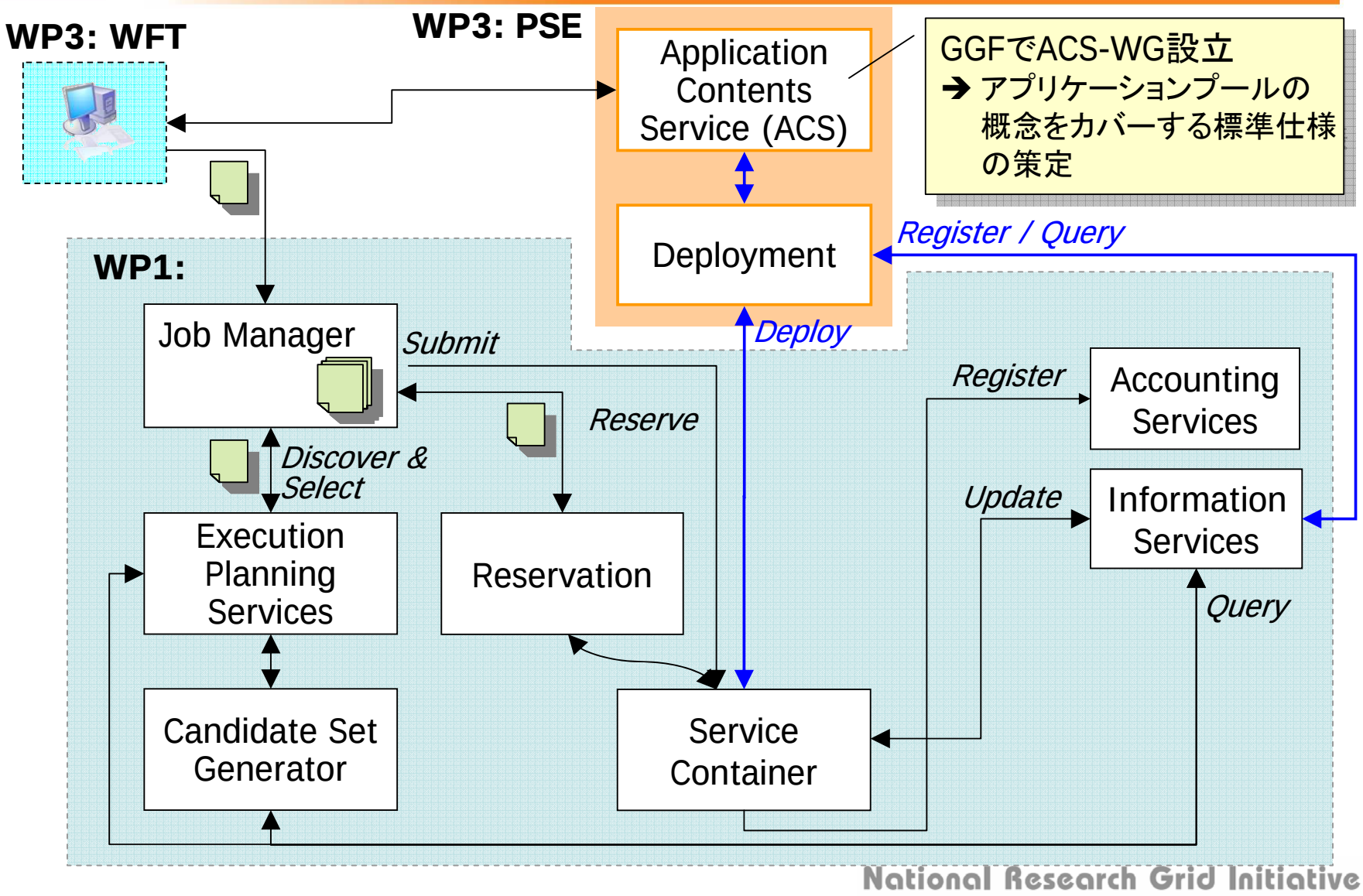


グリッドPSE：まとめ

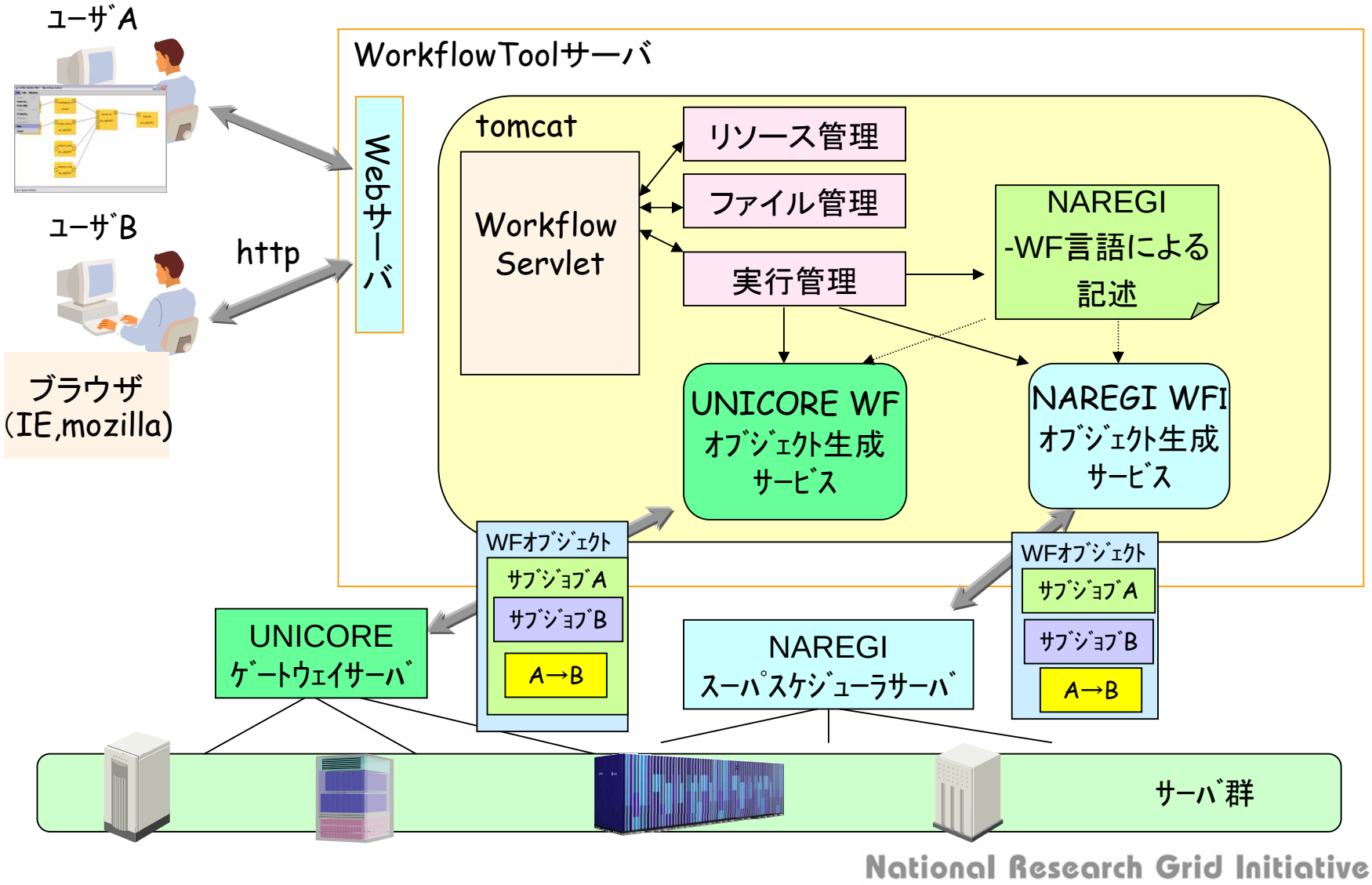
- NAREGI-PSEとは
  - 利用者がグリッドを意識せず利用できる環境を提供
  - アプリケーション開発者の支援
    - 開発したアプリケーションを容易にグリッドへ登録・配置
    - アプリケーションの再利用
  - アプリケーション利用者の支援
    - 物理的な計算機を意識せず、アプリケーションを利用

- 今後の主な開発予定
  - ワークフローツールと連携したワークフロー登録・検索機能
  - コンポーネントのWSRF対応
  - OGSAサービスとしてのアプリケーション管理
    - アプリケーションプールからApplication Contents Service (ACS)へ

グリッドPSE：
OGSAとNAREGI-PSE



グリッドワークフロー：
ワークフローツール



（補足の寄り道） Tomcat

- Apache Jakartaプロジェクトのサブプロジェクトとして開発されているオープンソースのソフトウェア.
- Javaサーブレット・JSP(Java Server Pages)を処理するアプリケーションサーバの事実上の標準.

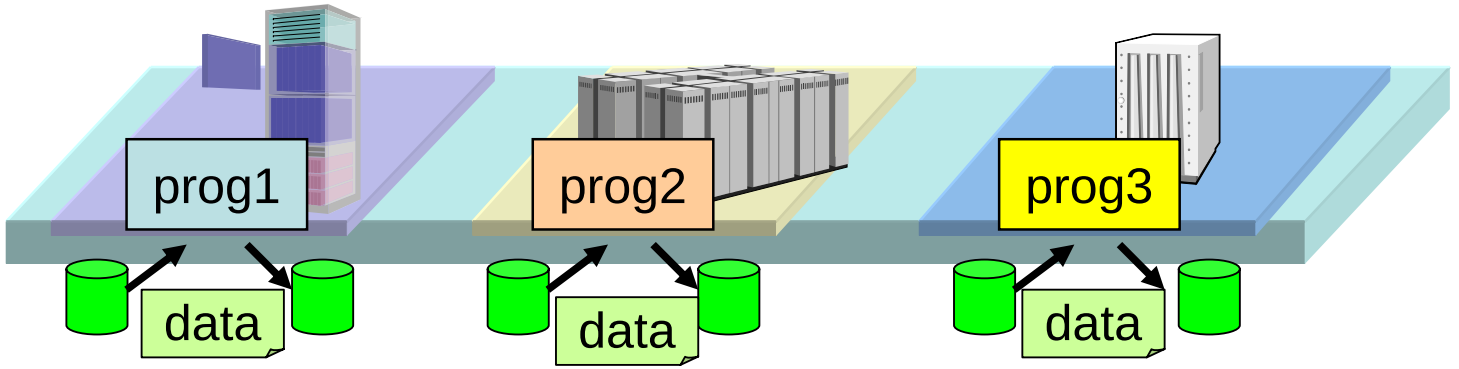
  ﾏｲｸﾛｿﾌﾄASP(Active Server Pages)はIIS付加機能だが，JSPは幅広いﾌﾟﾗｯﾄﾎｰﾑで動作
- Script言語はJavaそのものを利用.

http://jakarta.apache.org/tomcat/

グリッドワークフロー：
グリッドアプリケーションの開発工程

Step1

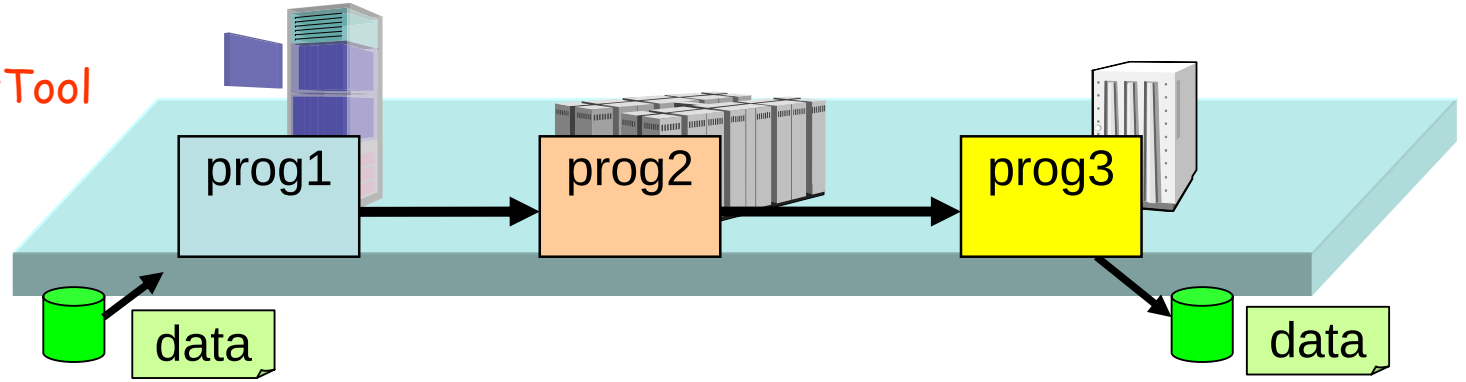
- 各コンピュータでのﾌﾟﾛｸﾞﾗﾑ開発
  （逐次ﾌﾟﾛｸﾞﾗﾑ、並列ﾌﾟﾛｸﾞﾗﾑ）



Step2

- 各ﾌﾟﾛｸﾞﾗﾑに対する計算機資源の指定（UNICORE)
- 各ﾌﾟﾛｸﾞﾗﾑの実行に必要な要件の指定（NAREGI-SS)
- 各ﾌﾟﾛｸﾞﾗﾑの依存関係の指定

WorkflowTool



グリッドワークフロー：
ワークフローツールによる開発手順（1/2）

1）WFコンポーネントとなる
プログラムアイコンの登録

UNICORE

計算機資源の指定（実行計算機、
ﾌﾟﾛｸﾞﾗﾑﾊﾟｽ　など）

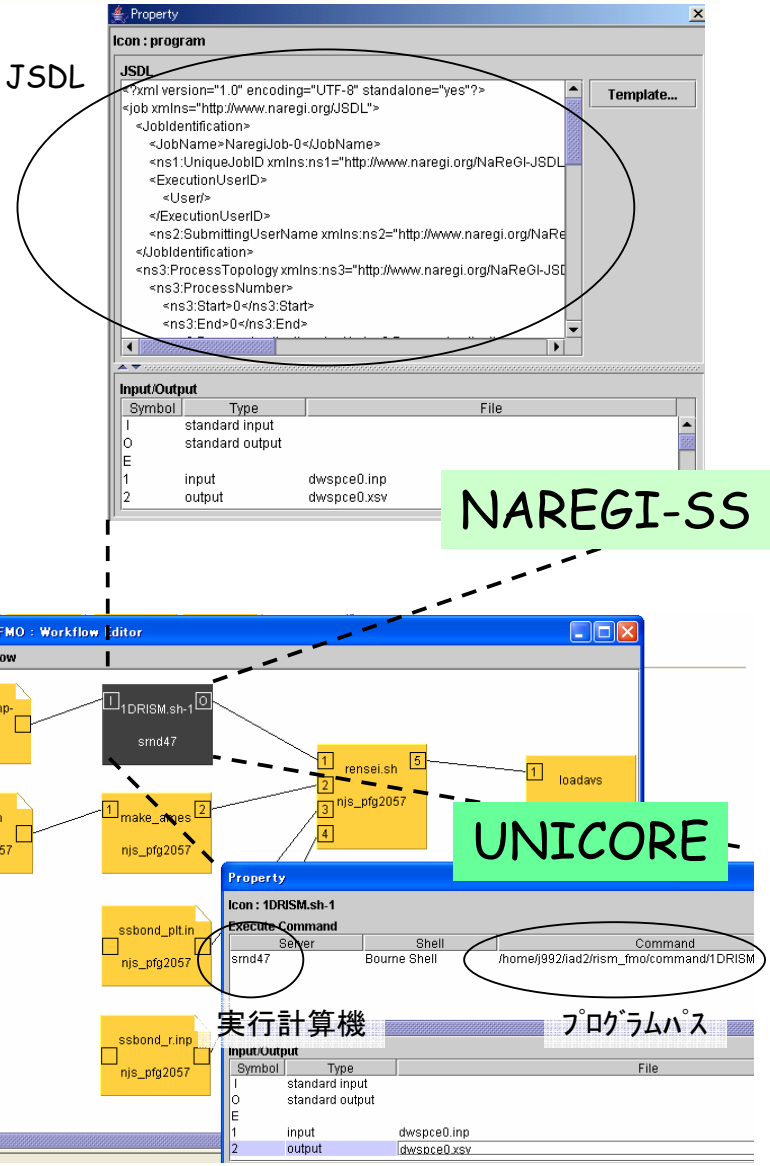
NAREGI-SS

ﾌﾟﾛｸﾞﾗﾑ実行の要件をJSDLで指定
（実行計算機など指定不要）

その他：ﾃﾞｰﾀｱｲｺﾝ、ﾜｰｸﾌﾛｰｱｲｺﾝの
登録

2）コンポーネント間の接続

制御依存関係の指定
IO（データ）依存関係の指定



NAREGI JSDL

- JSDLによる資源予約およびジョブ投入
- NAREGI向けに拡張して使用

NAREGI JSDL（一部）

属性	説明
/naregi-jsdl:SubJobID	サブジョブID
/naregi-jsdl:CPUCount	ジョブに必要な総CPU数
/naregi-jsdl:TasksPerHost	ホスト当たりのタスク数
/naregi-jsdl:TotalTasks	MPIタスク数
/naregi-jsdl:NumberOfNodes	必要ホスト数
/naregi-jdsl:CheckpointablePeriod	チェックポイント採取間隔
/jsdl:PhysicalMemory	ジョブの各プロセスが使用可能な物理メモリ
/jsdl:ProcessVirtualMemoryLimit	ジョブの各プロセスが使用可能な仮想メモリ
/naregi-jsdl:JobSatrtTrigger	予約開始時刻
/jsdl:WallTimeLimit	ジョブ実行時間
/jsdl:CPUTimeLimit	CPU時間
/jsdl:FileSizeLimit	最大ファイルサイズ
/jsdl:Queue	キュー名

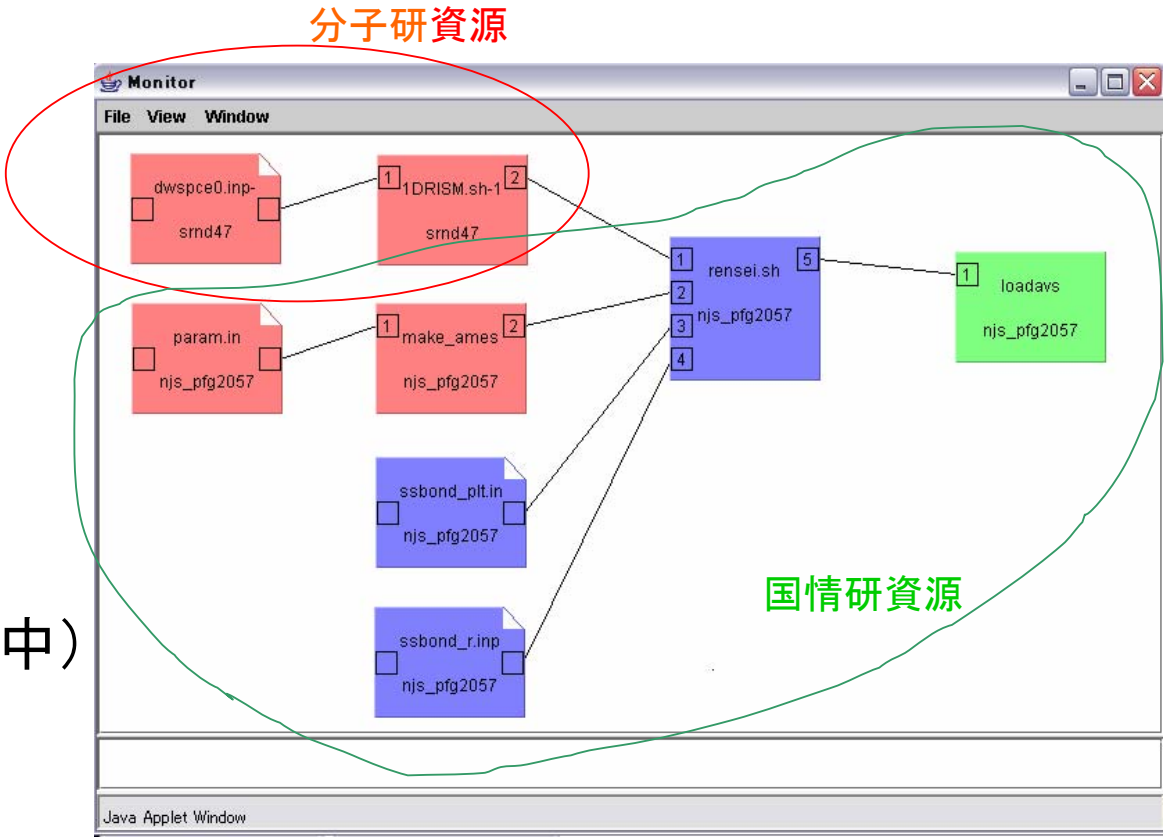
並列処理向け拡張

コアロケーションの時間指定

グリッドワークフロー：
ワークフローツールによる開発手順 (2/2)

3)計算の実行と実行状況の監視

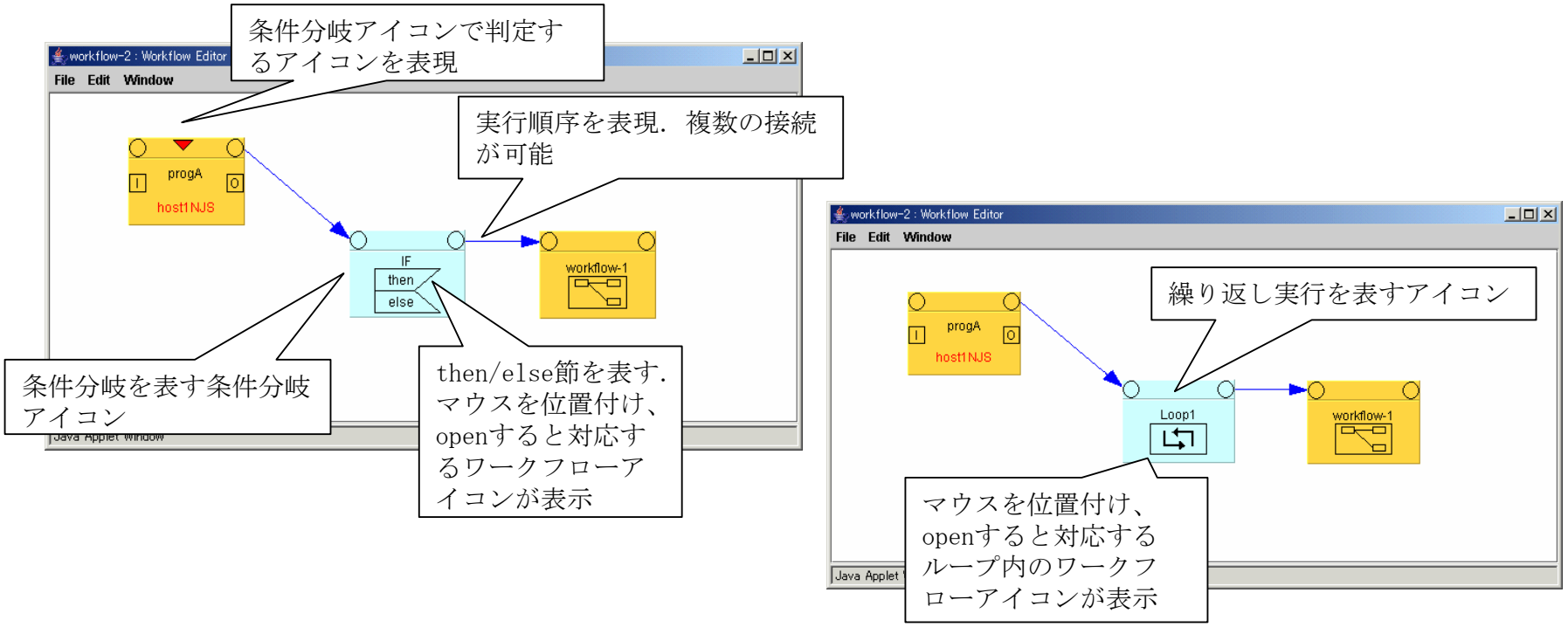
各コンポーネントの実行と
実行状態の監視



ピンク: 正常終了
ブルー: 計算中
グリーン: 待ち（準備中）
レッド: 異常終了

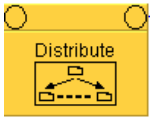
グリッドワークフロー：詳細機能の例（1/2）

- 制御構造をもつワークフロー構造
  （条件、ループ等）



グリッドワークフロー：詳細機能の例 (2/2)

- コレクティブデータ転送

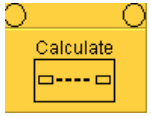


データ分散
(1計算機
→N計算機)

データ収集
(N計算機
→1計算機)

データ転送
(N計算機
→N計算機)

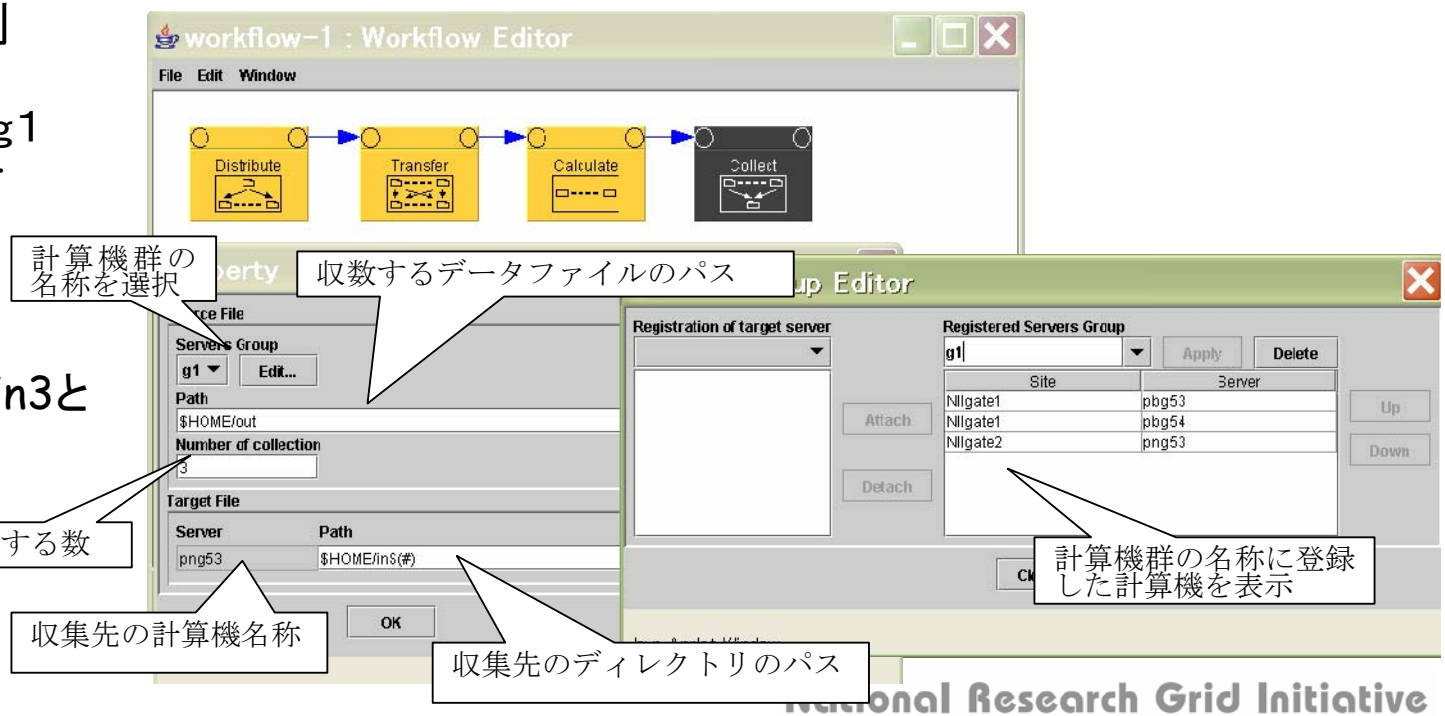
- コレクティブ計算



データ収集の例

サーバグループg1
の各HOME/out
ファイルを

Png53サーバの
HOME/in1,in2,in3と
してコピー収集



グリッドワークフロー：まとめ

UNICOREｻｰﾊﾞやNAREGI-SSｻｰﾊﾞ上での、ｸﾞﾘｯﾄﾞｱﾌﾟﾘｹｰｼｮﾝの開発と実行のために、以下の機能・特長をもつGUIﾍﾞｰｽのﾜｰｸﾌﾛｰﾂｰﾙを開発。

- ﾜｰｸﾌﾛｰ（異なるｻｲﾄ間をまたがる複数ｼﾞｮﾌﾞ、ﾃﾞｰﾀの連携）の作成
- ﾜｰｸﾌﾛｰとして、入出力の順序関係に基づくﾌﾟﾛｸﾞﾗﾑの依存関係だけでなく、ﾌﾟﾛｸﾞﾗﾑ条件ｺｰﾄﾞ等に基づくﾙｰﾌﾟや条件分岐等の制御ﾌﾛｰを記述可能
- ﾊﾟﾗﾒｰﾀｽﾀﾃﾞｨ等に有効なｺﾚｸﾃｨﾌﾞ計算、ｺﾚｸﾃｨﾌﾞﾃﾞｰﾀ転送機能を有するｱｲｺﾝ定義
- ﾜｰｸﾌﾛｰの実行、実行状況の監視

グリッド可視化：はじめに

背景

- 計算機シミュレーションにおける可視化の重要性
  - 計算結果は人間による解釈が事実上不能な膨大な数値の羅列。
  - シミュレーションは計算結果が人間に解釈されてはじめてその目的が達成される。

  可視化により初めて研究者は目に見える形で計算結果を観察、分析できる。
- グリッド環境上のデータを扱う難しさ
  - グリッド環境では計算結果データが遠隔の複数の計算サーバ上に分散。
  - 計算結果のデータ量が巨大であることが多々。

  この条件下、従来の可視化ソフトでは簡単かつ効率よく可視化することが困難。

本システムの目的

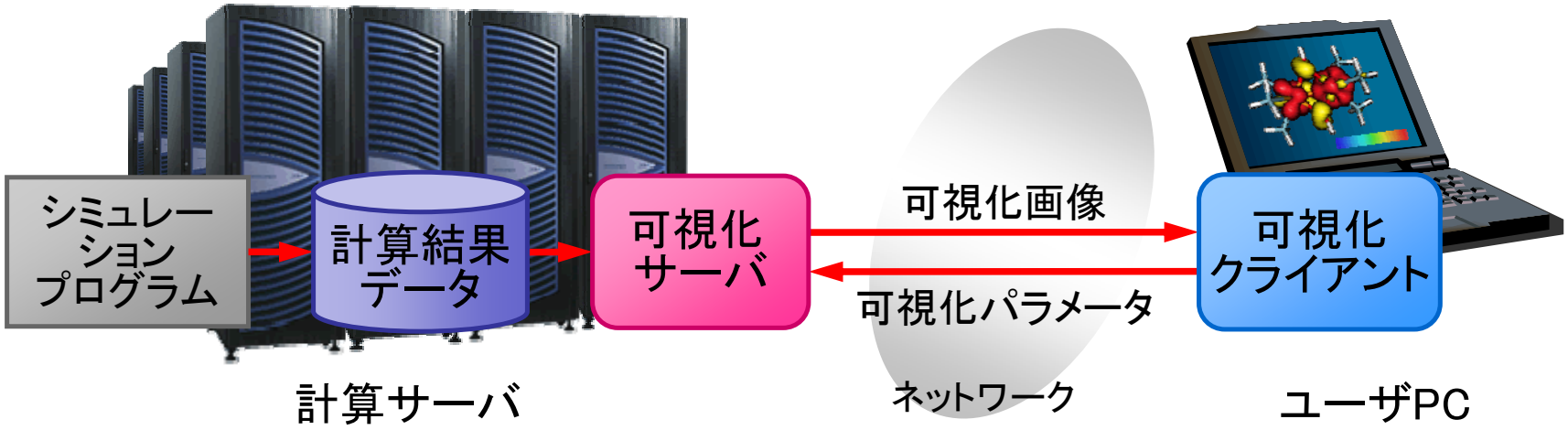
- 分散配置された大規模データの可視化

  グリッド環境上に分散されて保存された巨大な計算結果データを、
  複雑な操作や煩わしい待ち時間なしに、
  研究者のPCより遠隔から可視化できるシステムを提供。

➔グリッドの実用性と使い勝手の向上に貢献

グリッド可視化：
画像ベースのリモート可視化 (1/2)

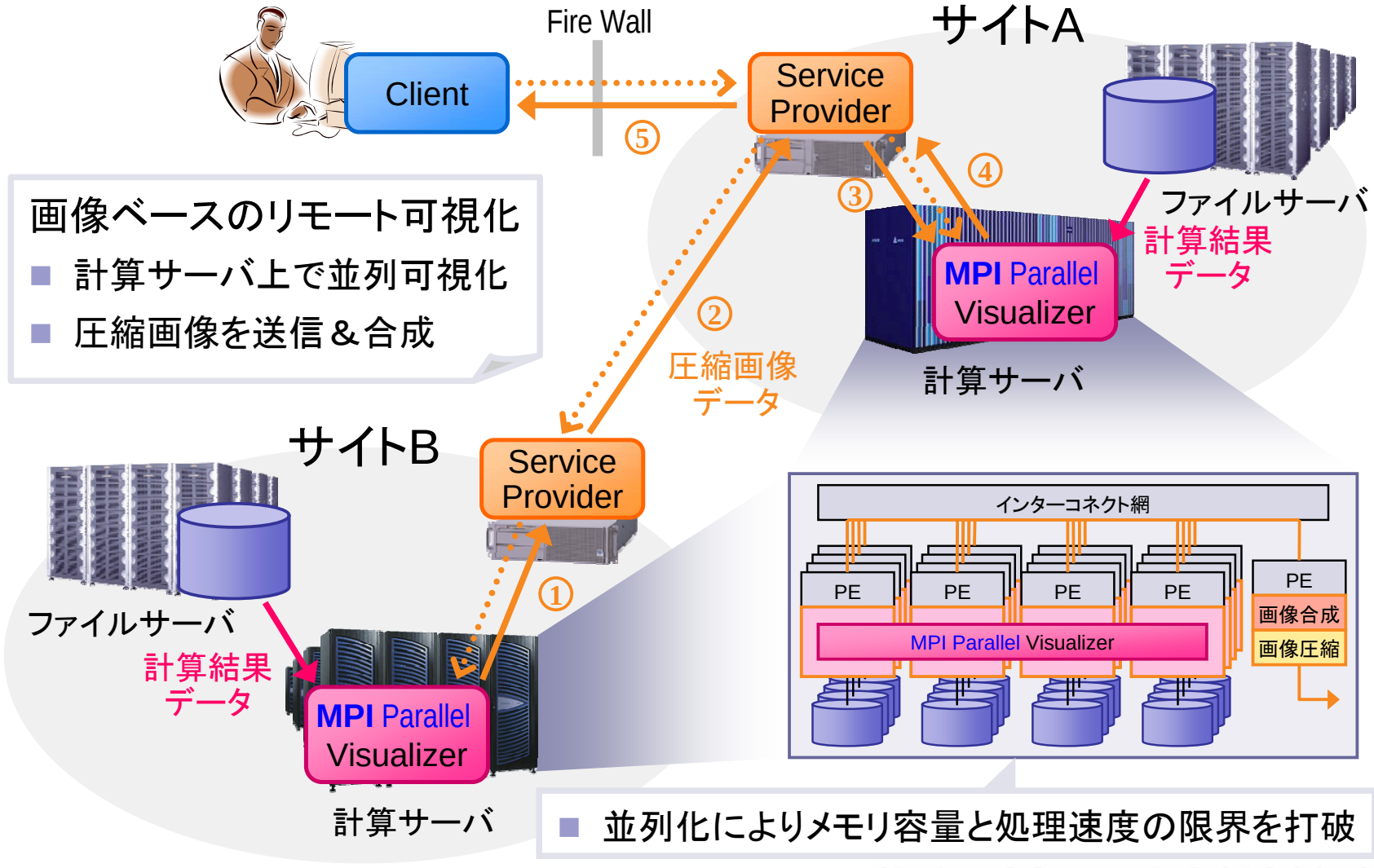
- 従来の可視化との違い
  - 計算結果データをユーザのPC上ではなく、リモートの計算サーバ上で可視化。
  - ユーザはPC上で起動したクライアントGUIによって、計算サーバ上の可視化を制御でき、そこで生成された画像を見ることができる。



グリッド可視化：
画像ベースのリモート可視化 (2/2)

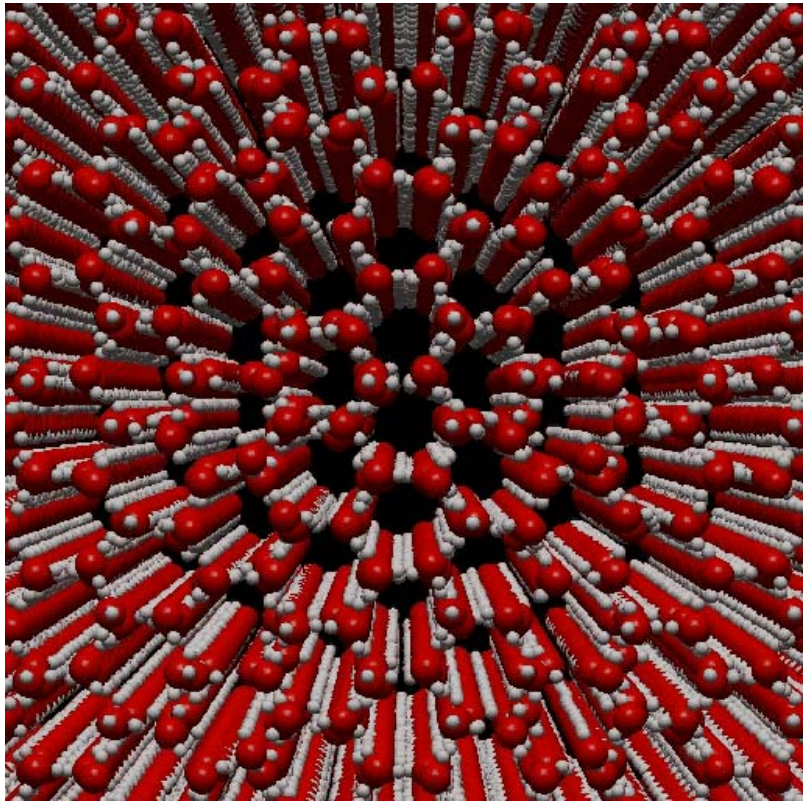
- 利点
  - 高速CPU、大容量メモリ、大容量HDDを備えたハイエンドPCは不要。
  - 大規模な計算結果データをPCにダウンロード不要。
  - マルチサイト連成/並列計算の場合でも大規模な計算結果データのサイト間転送は不要。
  - マルチサイト連成/並列計算の場合でも大規模な計算結果データの合成処理は不要。
  - ネットワーク性能や処理性能や容量の制限のために、わざわざ低解像度データにリダクション不要。
  - ネットワーク負荷が軽減。
  - どこからでもユーザのPCから可視化可能。

グリッド可視化：
大規模分散リモート可視化

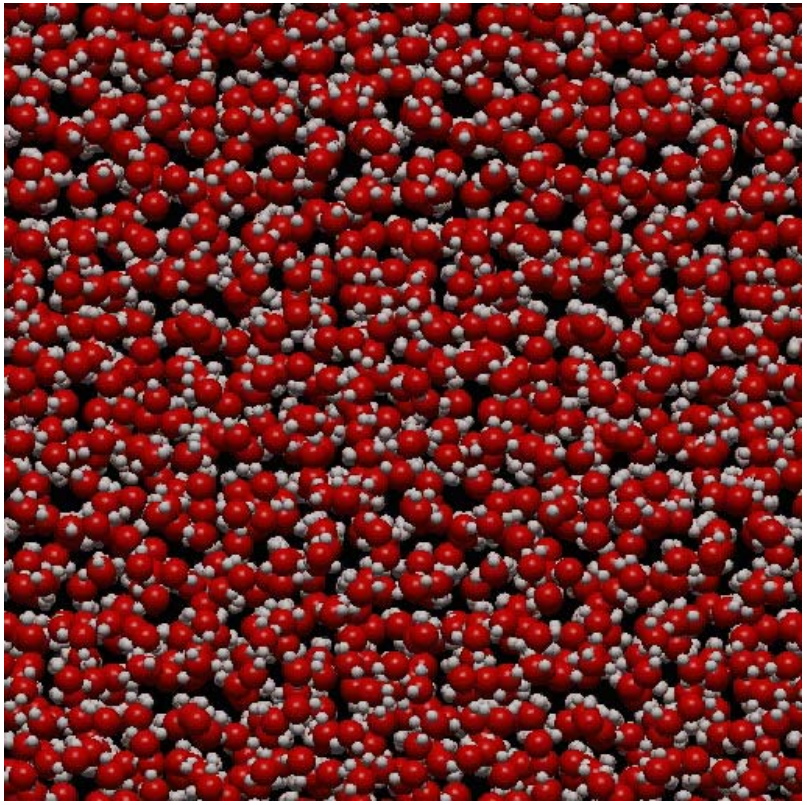


グリッド可視化：
大規模可視化の例

- 100万分子（300万原子）の運動の動画表示



氷

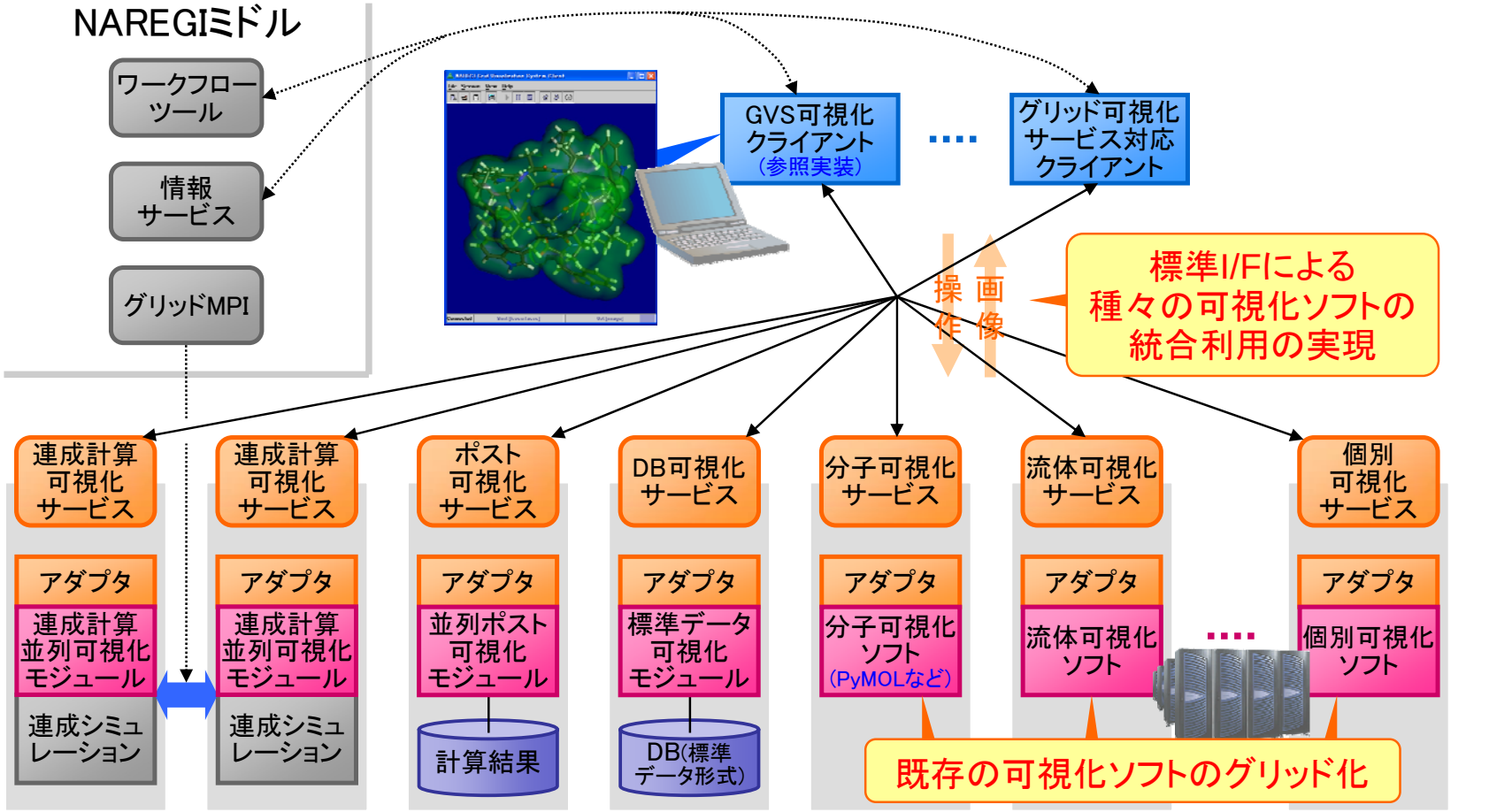


水

データ提供：分子科学研究所

グリッド可視化：
可視化グリッドサービスフレームワーク

- WSRFベースのリモート可視化統合環境への発展



グリッド可視化：まとめ

本システムの特長

- **散在するデータの可視化を手軽に**
  遠隔の計算機上で分散して計算/保存された連成計算などの大規模データを、可視化サービス連携により統合的に可視化できる。
- **ネットワークの負荷を軽減**
  遠隔の計算機上で可視化処理を行うため、大規模データの移動が不要になり、ネットワーク条件の悪い環境からもデータ規模を気にせず可視化できる。
- **大規模データを高速に可視化**
  並列可視化モジュールを使用の場合、可視化処理は並列に実行され、数百万原子規模の分子データも1秒程度で可視化できる。
- **様々な形態の可視化を簡便に**
  WSRFによる可視化標準I/Fにより、1つのクライアントGUIで様々な形態の可視化を統合的に行える。
- **種々の可視化ソフトをグリッド化**
  アダプタを用意することで、既存の種々の可視化ソフトをこの統合環境の枠組みに組み込める。

NAREGIプロジェクトの概要

基本層（WP1　東工大　松岡教授）

- スーパー・スケジューラ
- 分散情報サービス
- グリッドVM

上位層

WP3　国情研　宇佐見教授

- PSE
- ワークフローツール
- グリッド可視化

Programing層

WP2　産総研　関口博士

- GridMPI
- GridRPC

ネットワーク＆認証層

WP2　大阪大学　下條教授

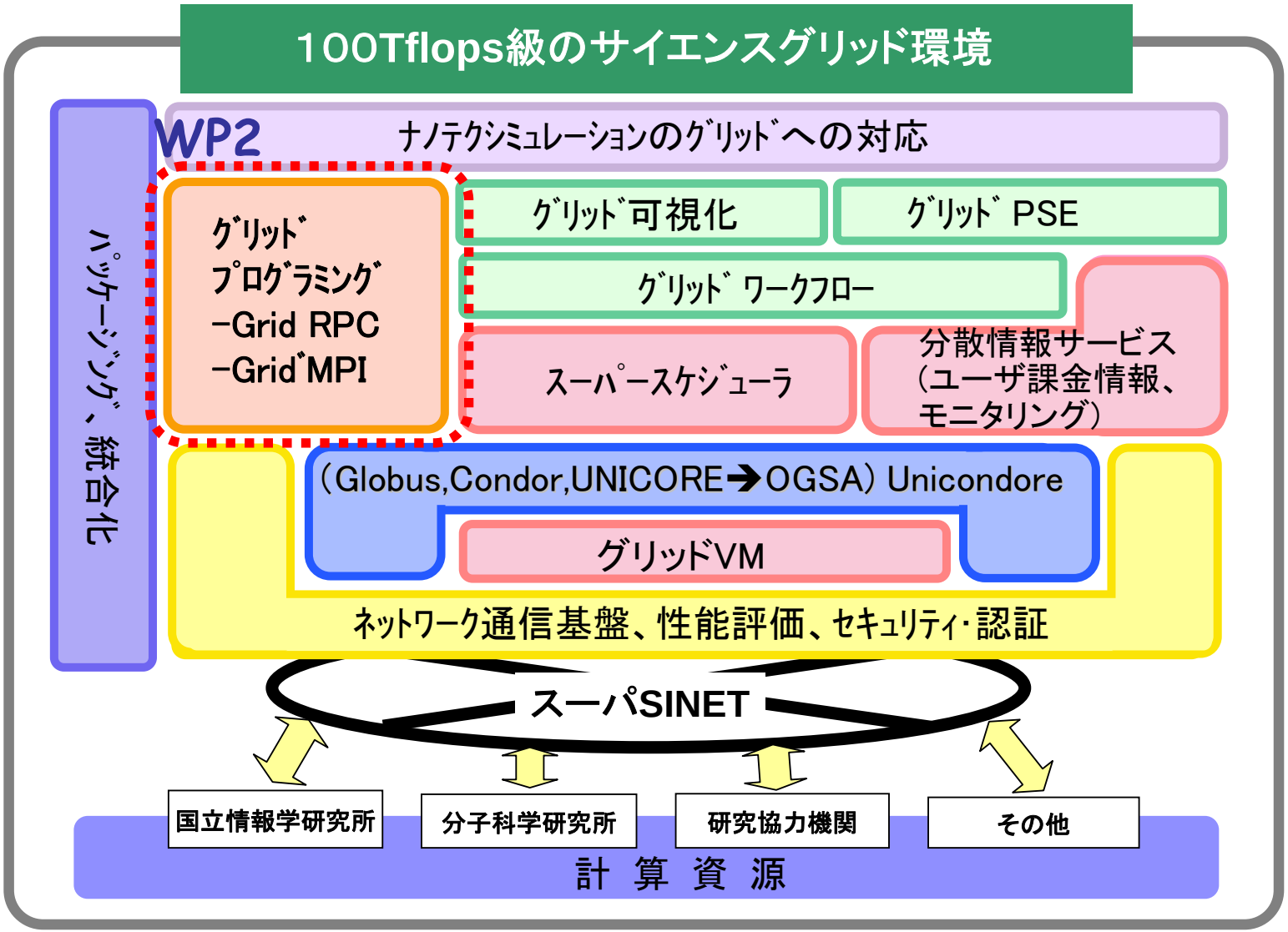
- PKI
- 認証連携 VO対応

Application層

WP6　九州大学　青柳

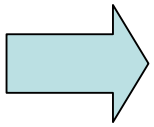
- Mediator
- 連成計算実証試験

NAREGIソフトウェア階層とWP2



NAREGIグリッドプログラミング環境（WP2）

- 目的
  - （資源）遠隔地に設置され、高速なネットワークで接続された複数のコンピュータ（クラスタやSMP機）
  - （環境）標準的なグリッド機能(セキュリティ、情報サービス、実行環境） e.g. NAREGI, GT3.0
  - VO/VCにおけるプログラム/ソフトウエア開発環境を提供
- 設計指針
  - 従来の並列プログラミングとの親和性がよいこと
    - 大規模計算ユーザのプログラム移行性
  - グリッド環境をユーザが意識しないこと
    - 独特のコマンド等を覚える必要がない
  - 小規模から大規模まで安定して拡張できること
    - 高いスケーラビリティ（100TFlops規模のグリッドでも動作する）



Grid RPC / Grid MPI を採用

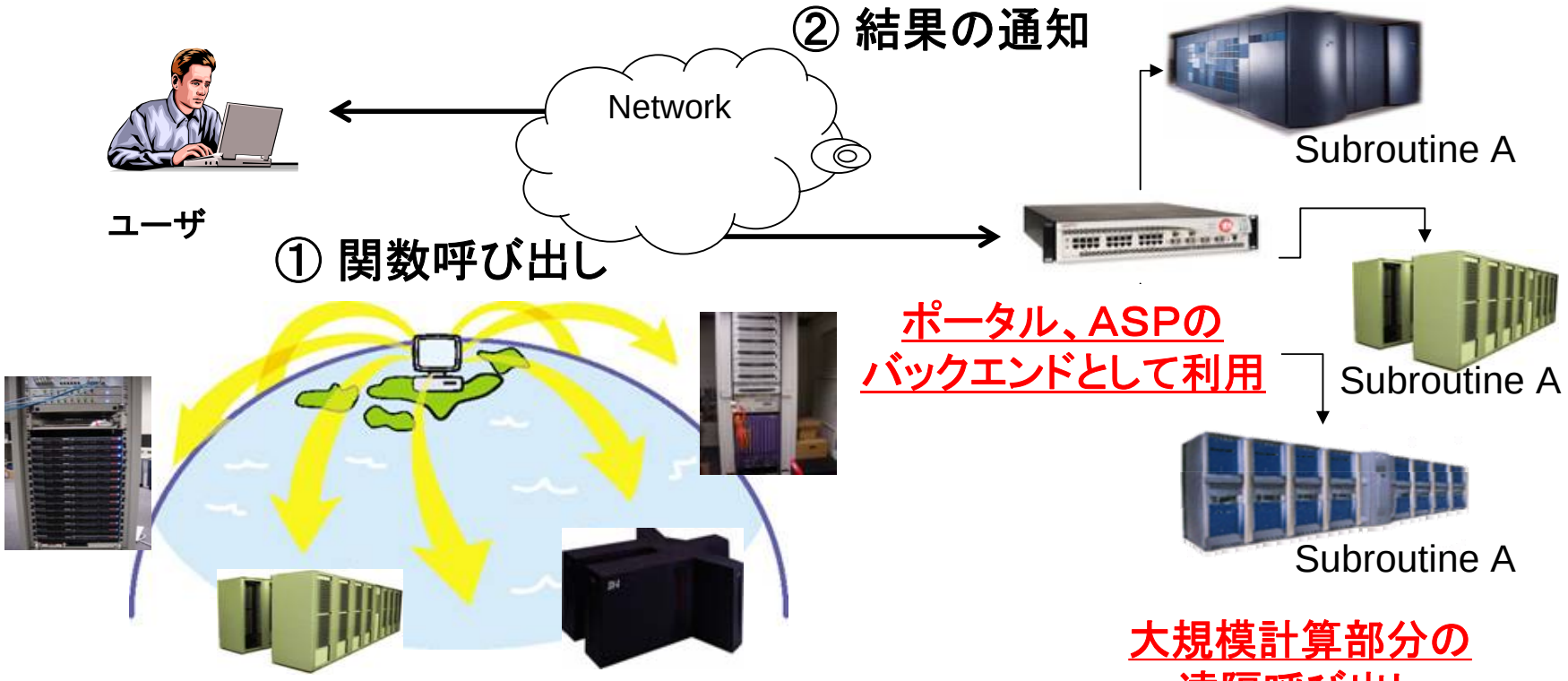
Grid MPI と Grid RPC

	Grid MPI	Grid RPC
特徴	並列プログラムの構造を理解する必要があり、若干困難であるが、記述能力は高い。並列性が複雑に絡み合う場合に有効。	特定の計算部分を遠隔実行するだけなので、VO/VCにおけるプログラムが容易。比較的単純な並列性を記述する場合に有効。
	MPI プログラムがそのまま動作可能	RPC呼出手続きの記述を若干追加
GGF 関連 WG	IMPI/OpenMPI	Grid RPC WG
主な実装	MPICH-G/2, PACX-MPI, STAMPI, Grid MPI, OpenMPI	Ninf, NetSolve

Grid RPC

- 認定ソフトウエア naregi-wp2-rpc-041221
  - –Ninf-G version 2.3.0 のリリース
- 大規模実証実験(Ninf-G2)
  - –アプリケーションの長時間実行による評価
  - –アプリケーションの実行による実用性評価
- Ninf-G versin 3.0.0aの開発

GridRPC 機能・動作概要



グリッド上の複数の高性能計算機を利用した大規模計算

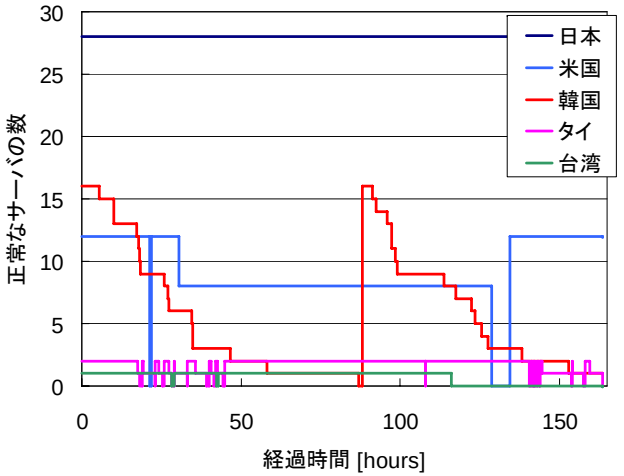
Ninf として 1994から開発を開始 → 成果を NAREGI に組み込む

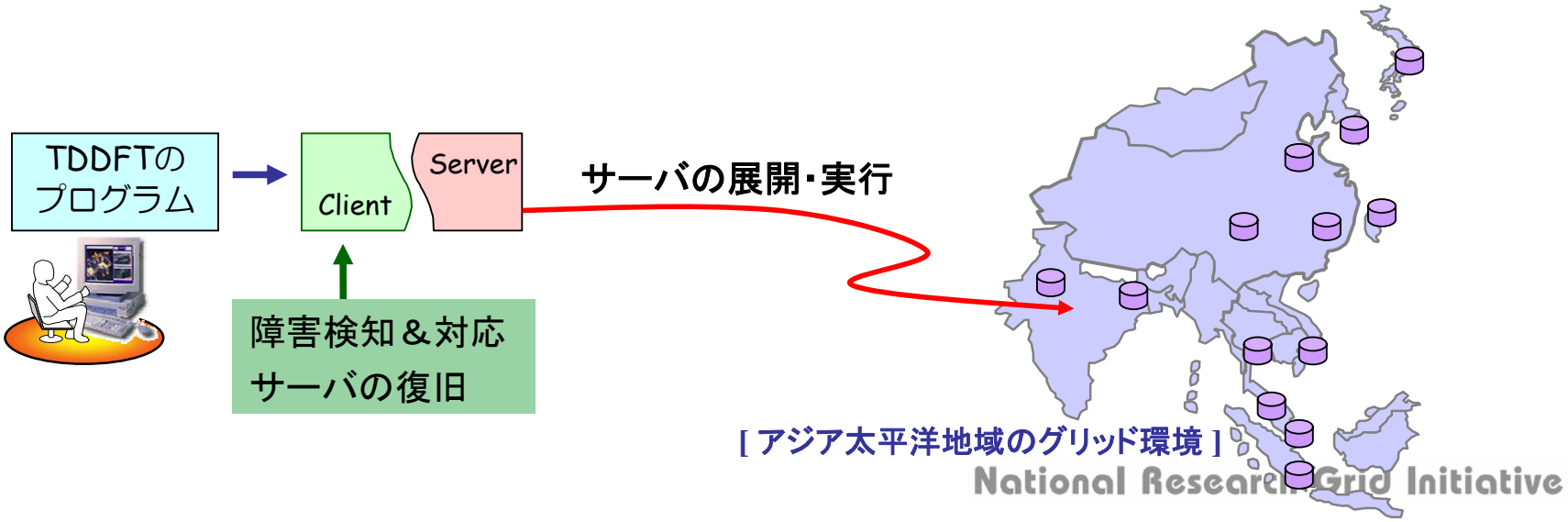
GridRPCの開発計画（前期）

	H15	H16	H17
計画	Ninf-G Version 2 (Ninf-G2)を開発。各課題に対応したGridRPC APIの実装を行なう。 また、GGFにおいてGridRPC APIの標準化を進める。	Ninf-G2の評価および改良。 GT3を用いたNinf-G Version 3 (Ninf-G3) のプロトタイプ開発。性能・機能の検証。	Ninf-G3正式版開発。
成果物	Ninf-G Version 2.0.0a を SC2003において配布。 Ninf-G Version 2.0.0 を年度末に配布。	Ninf-G Version 2.1 を SC2004において配布。 Ninf-G3 プロトタイプ	Ninf-G Version 3.0 を年度末に配布。 GT4, Univa Globus等に対応したNinf-G Version 4 の開発
標準化	GGF-7にてGridRPC WGの立ち上げ。	GGF-12にて標準APIの参照実装を提供。 GridRPC WG。	

アプリケーションの長時間実行による評価 ― 国際グリッドテストベッド上でのTDDFT ―

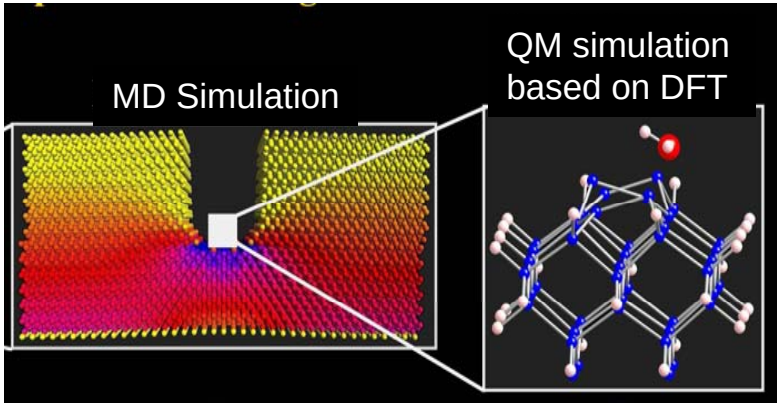
- 目的
  - Ninf-G2の品質の検証
  - Ninf-G2のエラー検知機能の検証
  - 耐障害機能の実装技術開発
- 実験
  - 量子化学計算（TDDFT）をNinf-G2を用いて実装
  - アジア・太平洋地域の国際的な試験環境上で長時間実行
    - 8カ国、10機関により提供される合計210CPUのテストベッド

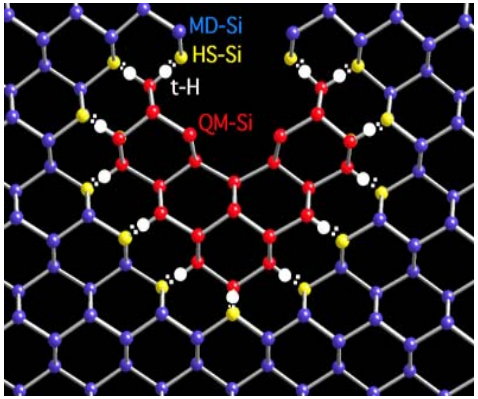




アプリケーションの実装による実用性評価 ― Hybrid QM/MD simulationの実装 ―

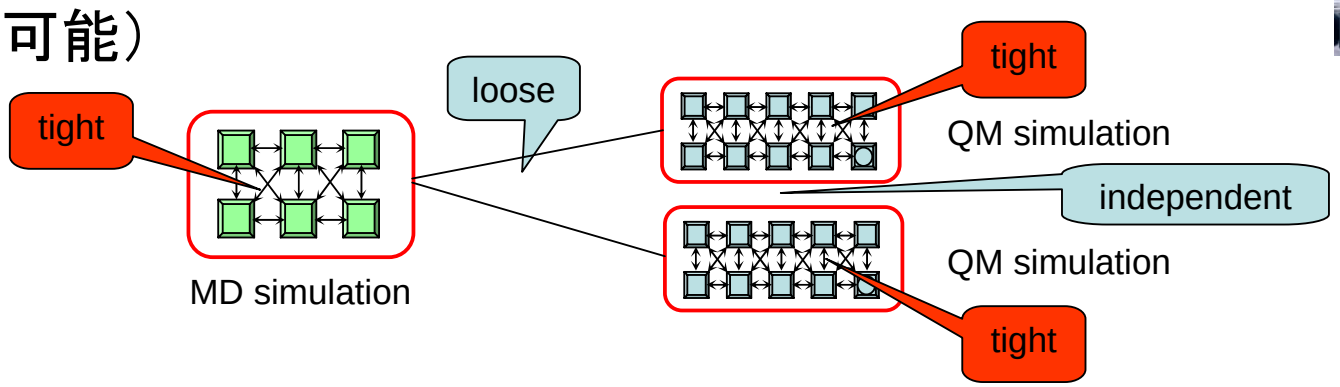
- 大規模なAtomistic simulationの高精度実行を可能にするため、MD SimulationとQM simulationを連携
  - MD simulation
    - 全領域の原子の振舞いを計算
    - 経験的原子間ポテンシャルを用いた古典MDシミュレーション
  - QM simulation
    - 興味のある領域のみを対象に実行，MDの結果を修正
    - 密度汎関数 (DFT)に基づくQMシミュレーション



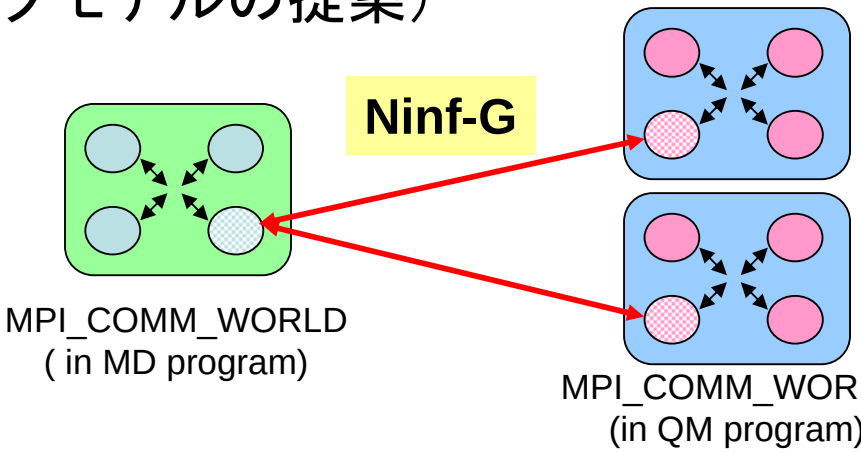


アプリケーションの実装による実用性評価 ― Hybrid QM/MD simulationの実装―（続き）

- アプリケーションの特徴
  - 複数のQM領域を定義可能（個々のQM領域は独立に計算可能）

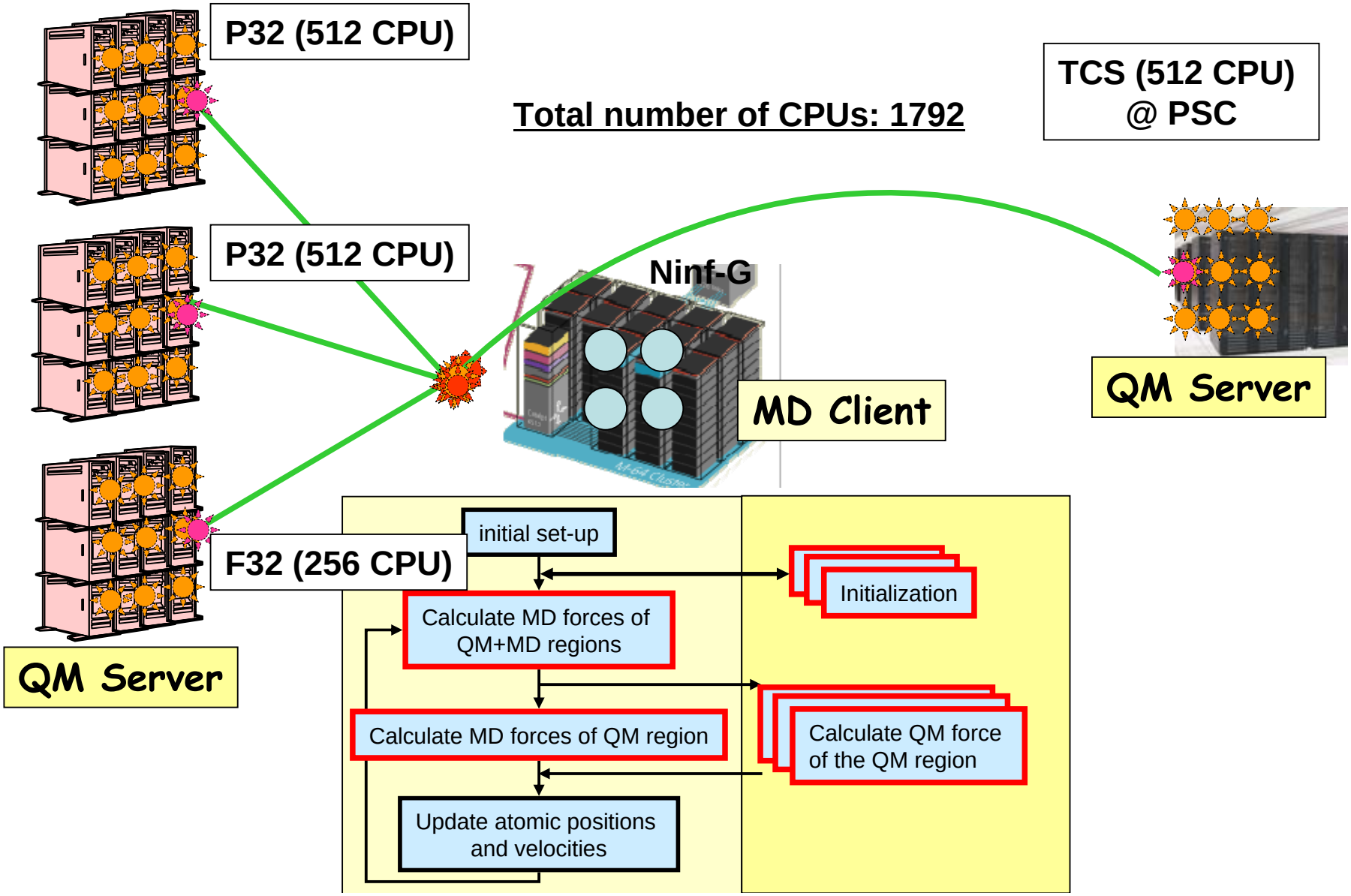


- Ninf-GとLocal MPIを組み合わせて実装（新たなプログラミングモデルの提案）



- 2プログラム(MD + QM)
- シミュレーション内通信：Global communicatorを用いたMPI通信
- シミュレーション間通信：Ninf-Gを利用

実験におけるアプリケーションの動作



専用計算機対応GamessのRPC化

Source: 青柳研 M1 岩切進悟君(卒研の図から引用)

クライアント側

RPC化

まとめた関数を
クライアントプログラムから
遠隔で呼び出す。

計算開始

入力

密度、初期フォック行列設定

クライアントプログラム

（RPC CALL）

まとめた関数

チップの初期化

電子反発積分計算
Fock行列の計算

対角化

収束判定

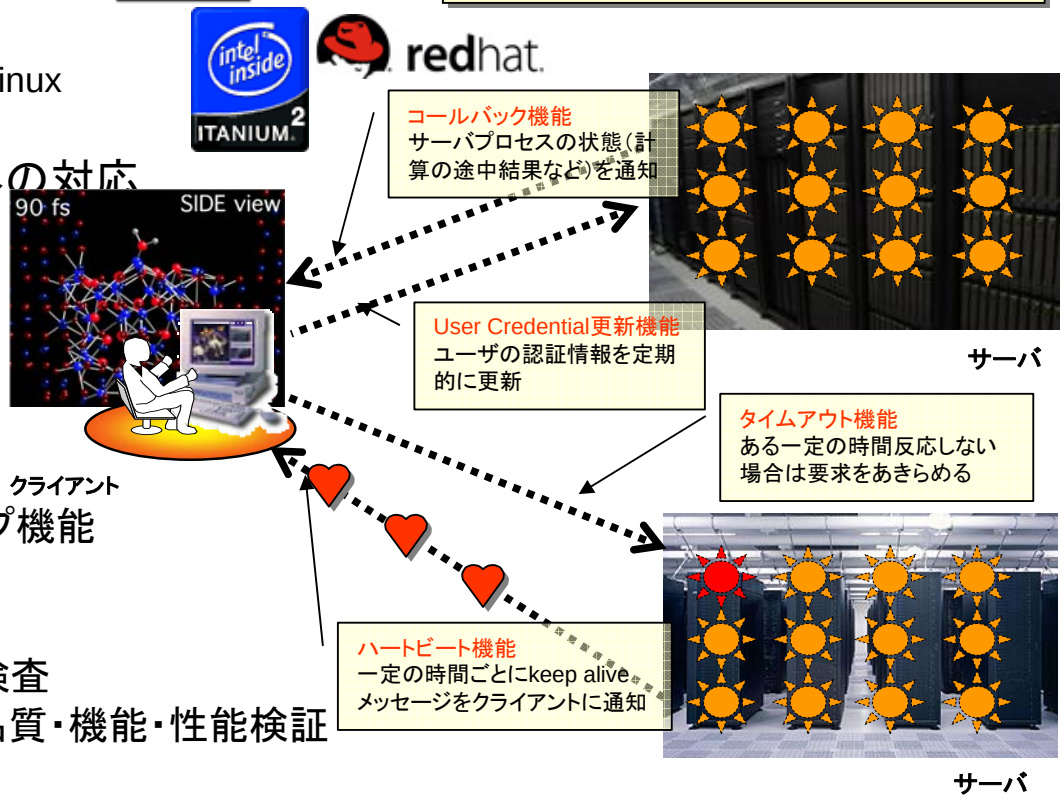
サーバー側

(専用ハードウエア)

GridRPC グリッドプログラミング環境【まとめ】

- 高性能システムへの対応
  - 64ビットアーキテクチャのサポート
    - AMD Opteron / SuSE Linux
    - Intel IA64 / RedHat Enterprise Linux
  - MPIプログラムの起動方法変更
- 長時間実行、信頼度の低い環境への対応
  - ハートビート機能
  - クライアントコールバック機能
  - タイムアウト機能
  - User credential更新機能
- 性能改善
  - 送信データの圧縮機能
  - データ送信と計算のオーバーラップ機能
  - 実装上の最適化
- 頑健化
  - 耐久テストプログラムによる品質検査
  - 大規模アプリケーションを用いた品質・機能・性能検証

2003年11月～2004年9月
Ninf-G Version 2 の
ダウンロード数： 498（5大陸、17カ国）



Ninf-G2を用いた応用研究が国内シンポジウムで最優秀論文賞を受賞
網崎　孝志他「分子動力学専用計算クラスタの開発とそれを利用した計算資源提供型グリッドの試み」
ハイパフォーマンスコンピューティングと計算科学シンポジウム　2004

Grid MPI

認定ソフトウエア naregi-wp2-mpi-041106

MPI-IO、リモート書きこみ、動的プロセス生成等の機能をMPI-2.0 標準仕様に準拠させる改良を実施。

MPI-IO、リモート書込み、動的プロセス生成等の機能などをMPI-2.0 標準仕様に準拠。テストスイート(MPI Validation Suite)にて動作を確認。

Grid MPIによりTCP/IPの通信トラフィックが極端に性能低下する現象を明らかにしシステム改良を実施。

Grid MPI 開発計画(前期)

	H15	H16	H17
計画	(開発課題) •MPI-1機能&IMPI •TCP/IP輻輳制御アルゴリズムのプロトタイプ実装 (研究課題) •トポロジを考慮した通信機構の設計とプロトタイプ作成 •チェックポイント機能の設計	(開発課題) •MPI-2機能への拡張と品質向上 •ベンダMPIとのインターフェイスの設計 (研究課題) •TCP/IP輻輳制御アルゴリズムの実装と評価 •トポロジを考慮した通信機構の評価	(開発課題) •GridMPIの全体評価および調整 •チェックポイント機能の実装 •ベンダMPIとのインターフェイスの実装 •トポロジを考慮した通信機構の実装
成果物	Grid MPI Evaluation Version SC03にて配布(MPI-1機能＋IMPI) Grid MPI Version 0.1 (MPI-1機能+IMPI+TCP/IPチューニング)	Grid MPI Version 0.2 SC04にて配布(MPI-2機構)	Grid MPI Version 1.0 SC05にて配布(チェックポイント機能付き、ベンダMPIインターフェイス込み)

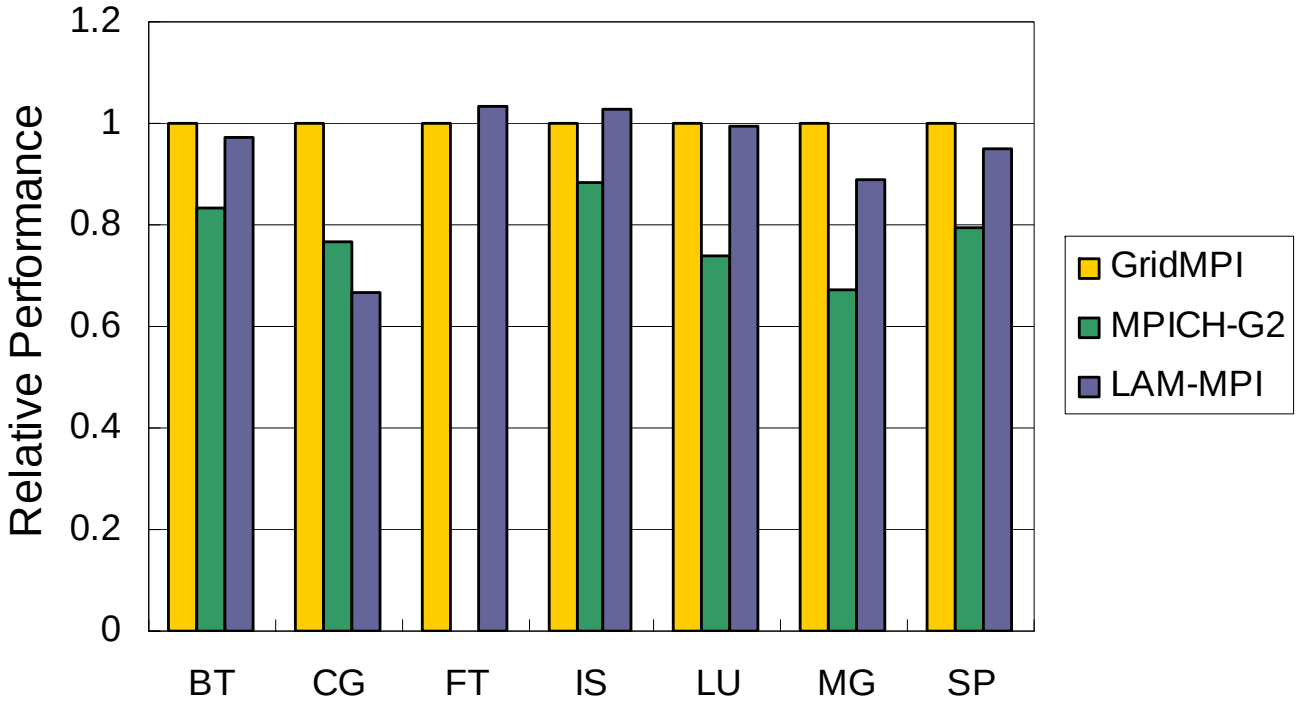
主要なフリーMPI実装との互換性比較

	ANL Test Suite	Intel Test Suite
GridMPI ver.0.2	100% Pass (Fail: 0/142)	100% Pass (Fail: 0/493)
MPICH-G2 ver.1.2.6 + Globus3.2	90.8% Pass (Fail: 13/142)	87.2% Pass (Fail: 63/493)
LAM-MPI ver.7.1.1	92.3% Pass (Fail: 11/142)	92.3% Pass (Fail: 16/493)

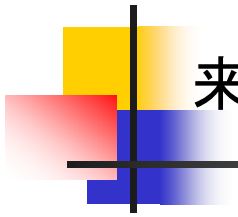
- ANL Test Suite（Argonne National Laboratoryが開発）
- Intel Test Suite（Intelが開発）

主要なフリーMPI実装との性能比較

NAS Parallel Benchmarks (NPB2.3)



– GridMPIを1とした時の相対性能を表示
  - 8ノードクラスタを2組接続
  - LAM-MPIはクラスタ構成、バイトオーダー変換なし
  - MPICH-G2ではFTは実行できず



来週（6月20日）の講義は情報基盤センター多目的講義室

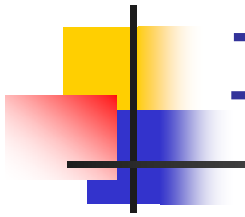
次週は

- 連成計算の類型化
- Mediator/GridMPI 連成プログラミング

（プログラミング層の続き）

を予定しています.

レポート課題2　(2005.5.23 出題)の〆切は6月6日（月）
深夜ですが，少しなら遅れて提出しても受理.



再来週(6/27)の予定

- GGF14(シカゴ)に(たぶん)出張のため代講(高見，大庭)者によるGrid計算の実演デモを予定しています.

  Unicore＋Naregi上位層

  Globus(ver.3.2)