Grid勉強会

11/05/10

4.2.4項～4.3.3項

高瀬　亘

GRAM概要

- Globus Resource Allocation Manager
- リソースとジョブの管理を行う
- リモート・ジョブの投入とリモート・ジョブに対するコントロール機能を提供
- 上位のサービスと下位のサービスを結び付けるのに必要とされるインタフェースやAPIやプロトコルの数を減らすことができる

スケジューラ

- ローカルスケジューラ
  - 配下にある複数マシンのそれぞれのCPUやメモリの使用率を調べて、クライアントから投入された処理に最適なノードを探して割り当てる
- メタスケジューラ
  - ローカルジューラのスケジューラ。複数のローカルスケジューラを束ねて管理する
  - 処理に対して、最適なローカルスケジューラを探して割り当てる

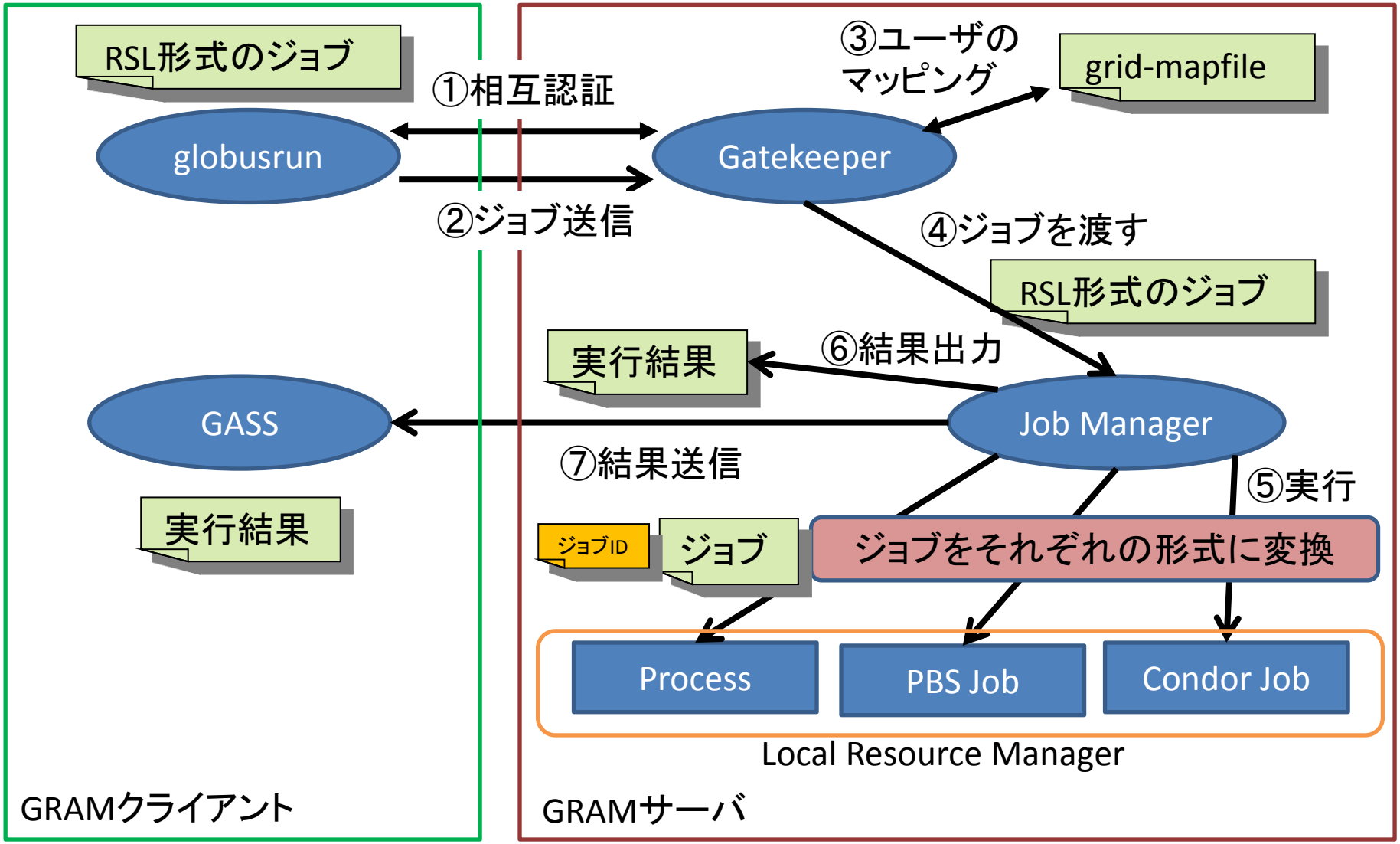
GRAMによるジョブの実行の仕組み

- 同期処理
  - リモートの実行ホストが処理が終わるまでコネクションを持続
- 非同期処理
  - リモートの実行ホストの処理をリクエストしたらコネクションを切断
  - 通常は同期処理よりも非同期処理を良く使う

同期処理の流れ

1. GSIにもとづいた相互認証を行う
2. 処理内容をクライアントからサーバへ送る
   - Resource Specification Language（RSL）言語で記述（Key-Value Pair）
   - HTTPベースのRPCを使用
3. grid-mapfileを使用してユーザのDNとサーバ上のローカルユーザをマッピング
4. マッピングしたローカルユーザで要求されたJob Managerを起動してジョブを渡す
   - ユーザが起動したプロセスとなる
5. Job Managerは要求されたジョブを実行
   - RSLファイルをそれぞれのLocal Resource Manager（LRM）用の形式に変換して処理
     - サーバ上で実行（fork）するか、他のスケジューラに引き渡す
6. 実行結果がGRAMサーバ上に仮ファイル出力される
7. Job ManagerはGASS（Global Access to Secondary Storage）のAPIを使用してGASSサーバ経由で処理結果をクライアントに戻す
   - GRAMサーバ上の仮ファイルは削除される
   - GASSサーバはGRAMクライアント・サーバ間のファイルI/Oを実現

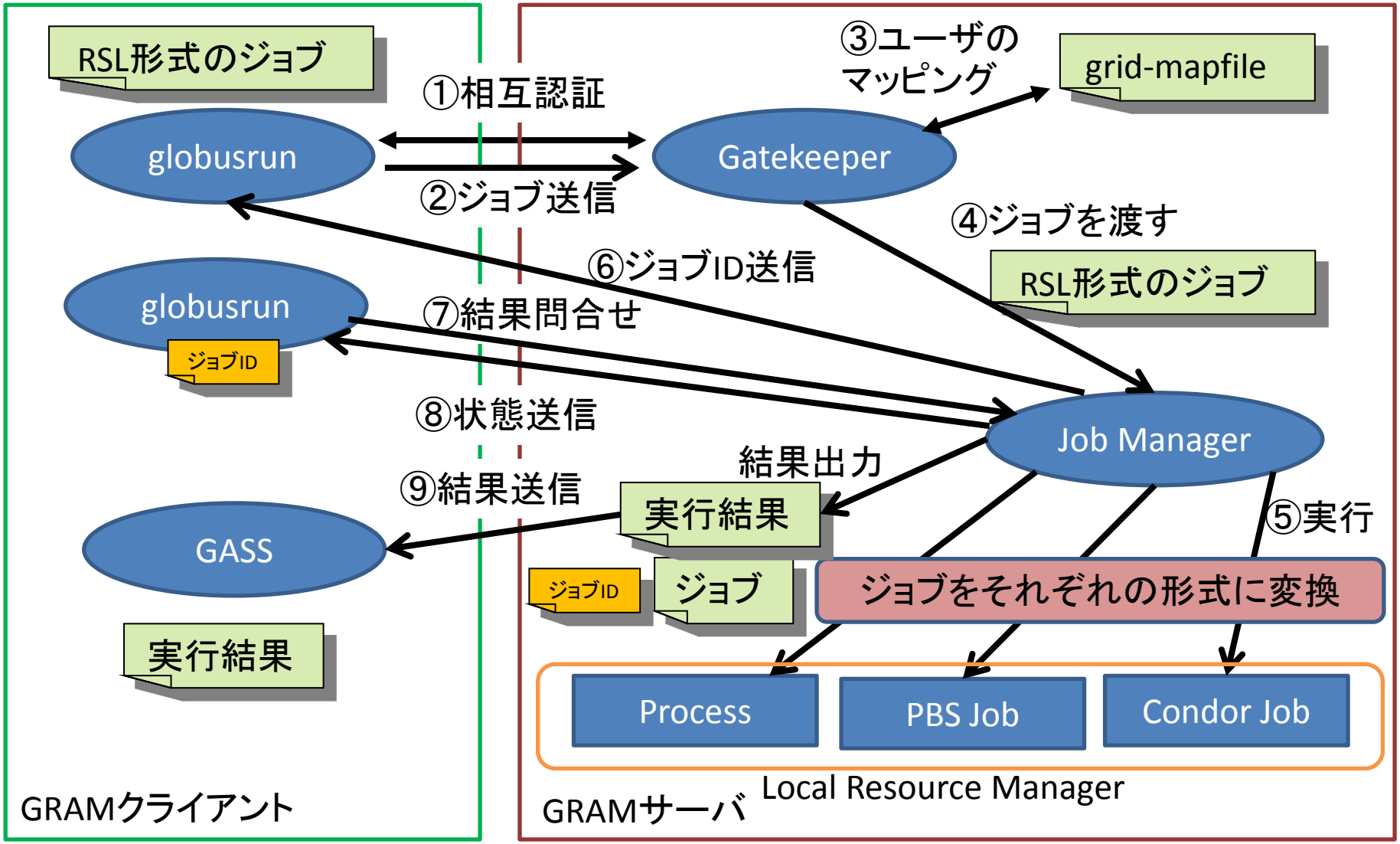
同期処理の流れ


RSL形式のジョブ
①相互認証
globusrun
Gatekeeper
③ユーザの
マッピング
grid-mapfile
②ジョブ送信
④ジョブを渡す
RSL形式のジョブ
実行結果
⑥結果出力
GASS
Job Manager
⑦結果送信
⑤実行
実行結果
ジョブID
ジョブ
ジョブをそれぞれの形式に変換
Process
PBS Job
Condor Job
Local Resource Manager
GRAMクライアント
GRAMサーバ

非同期処理の流れ

1. GSIにもとづいた相互認証を行う
2. 処理内容をクライアントからサーバへ送る
3. grid-mapfileを使用してユーザのDNとサーバ上のローカルユーザをマッピング
4. マッピングしたローカルユーザで要求されたJob Managerを起動してジョブを渡す
5. Job Managerは要求されたジョブを実行
6. 投入したジョブのジョブIDをURLでクライアントへ返してコネクションを切断
7. クライアントはジョブIDを利用してJob Managerへ処理結果を確認
   - 処理結果を問い合わせるとクライアント上でGASSサーバが起動
8. サーバはリクエストを受け取ると該当するジョブの状態を返す
9. クライアントはジョブの状態を受け取り、ジョブが終了していたら処理結果を得る

非同期処理の流れ


RSL形式のジョブ
①相互認証
globusrun
Gatekeeper
③ユーザの
マッピング
grid-mapfile
②ジョブ送信
④ジョブを渡す
⑥ジョブID送信
RSL形式のジョブ
globusrun
⑦結果問合せ
ジョブID
⑧状態送信
Job Manager
結果出力
⑨結果送信
実行結果
⑤実行
GASS
ジョブID
ジョブ
ジョブをそれぞれの形式に変換
実行結果
Process
PBS Job
Condor Job
Local Resource Manager
GRAMクライアント
GRAMサーバ

MDS概要

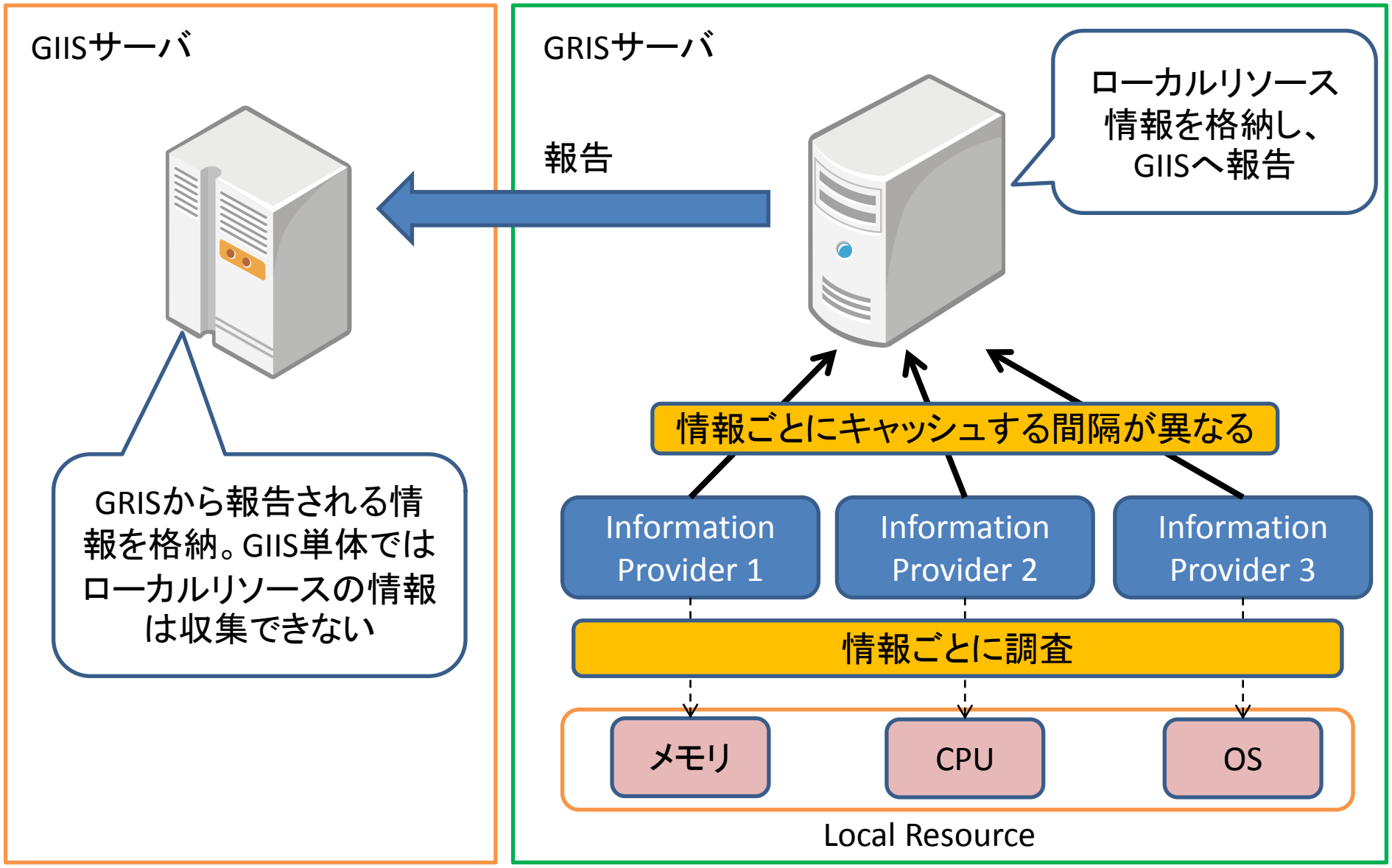
- Monitoring and Discovery Service
- リソース情報の収集を行う
- LDAPによって実装
  - NAREGIはPostgresを使用
- 標準的なインタフェースとグリッド環境で使用されるリソース情報の定義を提供
- リモートシステムのリソース情報の検索に必要だった工程の削減が実現
- ジョブの要件に合った割り振りホストを探しだす
- 上位のサービスと下位のサービスを結び付けるのに必要とされるインタフェースやAPIやプロトコルの数を減らすことができる

MDS概要

- MDSコンポーネントは以下の二つから成る

- Grid Resource Information Service（GRIS）
  - ローカルリソース情報のみを扱う
  - Information Providerの機能でリソース情報を調べキャッシュし、GRIS本体に報告
    - Information ProviderはOS、CPU、メモリ等の情報ごとに存在
    - 結果をGIISに登録することも可能
- Grid Index Information Service（GIIS）
  - GRISまたはGIISから報告されたリソース情報を管理する
  - リソース情報の検索はできない
  - グリッド環境内のマシンのリソース情報を分散化して管理
    - 対障害性の向上、スケーラビリティの確保が可能

MDS概要


GIISサーバ
GRISサーバ
報告
ローカルリソース
情報を格納し、
GIISへ報告
GRISから報告される情
報を格納。GIIS単体では
ローカルリソースの情報
は収集できない
情報ごとにキャッシュする間隔が異なる
Information
Provider 1
Information
Provider 2
Information
Provider 3
情報ごとに調査
メモリ
CPU
OS
Local Resource

MDSで検索することのできるリソース

- 静的なリソース情報
  - プラットフォームのタイプとアーキテクチャ
  - OS名とバージョン
  - ネットワークインタフェース情報（デバイス名、IPアドレス等）
- 動的なリソース情報
  - CPU情報（タイプ、バージョン、スピード、キャッシュ、使用率、数等）
  - メモリ情報（空き容量等）
  - ファイルシステム情報（空き容量、ファイルシステム名等）

MDSによるリソース情報の格納と取得の仕組み

- MDSクライアントは3つの方法でリソース情報を検索できる

- 最上位のGIISサーバへの検索
  - 全てのGRISサーバのリソース情報が得られる
- 中間のGIISサーバへの検索
  - 対象のGIISサーバの管理するGRISサーバの情報が得られる
  - 新しい情報が得られる
- GRISサーバへの検索
  - 新しい情報が得られる

MDSクライアントによる検索

- LDAPの検索の仕組みを利用している
  - Directory Information Tree（DIT）でリソース情報を格納
  - hn=hostnameで該当ホストのリソース情報を持っており（親）、dev group=XXX（CPUs, Memory, disk等）（子）がグルーピングされたリソースに対して個別に調査を行う
  - クライアントが必要条件を入力して検索をかけると、該当するhn=hostnameが出力される

GridFTP概要

- データ管理を行う
- gsiftp://ホスト名/ファイルのフルパス
- グリッド環境にセキュアなデータ転送メカニズムを提供
- GASSよりも高度なデータ転送が可能
  - GASSは1ポートだけで転送をするが、GridFTPは制御ポートに加えてデータ転送用の複数のポートを使用
  - Globus Toolkit2におけるデータ転送で通常使用
- オープンソースのソフトウェアWu-ftpサーバ拡張して作られたもの

GridFTP概要

- Wu-ftpサーバの機能に加えて以下の機能がある
  - GSIのサポート
  - サードパーティ転送
  - パラレルデータ転送
  - ストライピング転送
  - パーシャルファイル転送
  - リライアブルファイル転送

GridFTPによるファイル転送の仕組み

- サードパーティファイル転送の流れ
  - GridFTPサーバA内のファイルをGridFTPサーバBへ直接転送する
    - 通常のFTPでは、サーバAから転送させたいファイルを一旦クライアント上に取得し、そのファイルをサーバBへ転送する
    - クライアントはファイルを取り扱わないので余計なポートを開けなくてよい

1. クライアント⇔GridFTPサーバA間、クライアント⇔GridFTPサーバB間でGSIによる相互認証
   - クライアントはプロキシ証明書を、サーバはホスト証明書を使用
2. 認証に成功するとコネクションが確立し、サーバA⇔サーバB間で直接ファイル転送が行える

Replica Services概要

- Globus Toolkit2におけるデータの仮想化、分散化などのデータ・グリッド機能の一部を担う
- クライアントからのファイルサーバ上のファイルへのアクセスはReplica Servicesサーバを介す
  - クライアントは使用したいファイルが、予めどこのファイルサーバ上にあるのかを気にしなくてよい
- GridFTP＋LDAPを利用
- 論理的なデータと実際のデータとのマッピング機能を提供し、クライアントの求めるファイルがどこのホストにあるかを調べる事ができる

Globus Toolkit2利用の利点

- 認証が一度で良い
  - 利用しない場合、リモートホストへアクセスするたびに認証が必要
- 要件に合ったジョブ実行ノードをGIISサーバを使用して検索できる
  - 利用しない場合、一つ一つのノードを地道に調べていかないといけない

Globus Toolkit2の課題

- 未提供の機能は各自用意する必要がある
- 使用するプロトコルが多い
- ファイアウォールにたくさんの穴を開けないといけない

- プロトコルの問題に対応したGlobus Toolkit3
  - プロトコルの1本化
    - HTTPを使用したSOAPプロトコル
    - SOAP:XMLベースでRPCを行う
  - 現状は、使用できるポートが1つしかないため複数のサービスが集中してしまう等の問題があったため、プロトコルの1本化には向かっていない

次回

- 11/05/12 15:00
- 5.1～5.2