

Grid Computing

ービジネス分野での応用ー

Web サービスと OGSA

Web Service and Open Grid Service Architecture

(株) 富士通研究所

岸本光弘 hiro.kishimoto@jp.fujitsu.com

グリッドと Web サービス

科学技術分野において、グリッドコンピューティングは目覚ましい成功を収めてきた。Globus Toolkit や UNICORE に代表されるグリッドミドルウェアは、複数のスパコンセンタ間を接続し、認証や認可といった実用的なセキュリティ基盤を提供し、分散リソースの検索とモニタを実現している。さらに、遠隔サイトでのアプリケーションの遠隔実行や、広域分散したデータへの高速アクセス手段を提供してきた。その結果、多数のグリッドミドルウェアがオープンソースとして提供され、多くの実用的なプロジェクトが活動している。

しかし、現在の Globus Toolkit は、HTTP、LDAP、FTP など、多数のプロトコルを採用している。また、遠隔実行やセキュリティの実現にも、複数の手段を採用しており、分かりやすいとはいいがたい。さらに、科学技術分野以外のビジネス分野へ適用するためには、データベースアクセスやシステムを構成する機器の管理、ワークフロー制御の機能、システムの仮想化を提供することで、エンドユーザレベルでの性能 (QoS) を保証することが必要となる。

これらはいずれも難しい課題であるが、上記の多くは、すでにビジネス分野において、Web サービス技術として検討され技術開発が進められている。Web サービスは、ビジネスアプリケーションを統合する技術であり、インタフェース情報を記述する XML ベースの WSDL (Web Service Description Language)、通信のデータ形式を規定する HTTP ベースの SOAP (Simple Object Access Protocol) といった技術が W3C 等で標準化されている。グリッドの研究コミュニティは、共通した課題に対して異なる技術標準を開発するのではなく、

これら、Web サービスの技術を評価し、必要な拡張を施した上で採用していくべきと考える。

一方、Web サービス側から考えると、アプリケーションプログラムの統合だけでなく、異機種 (マルチベンダ) プラットフォームや、分散システムベースのシステム構築時に必要な、本質的に難しい問題を、グリッド技術を使うことで、解決できる可能性がある。すなわち、ベンダごとに個別のインタフェースを使用するのではなく、仮想化により標準化されたインタフェースを使用することで、システムを統一的にモニタおよび制御することが可能になる。

異なるアプリケーションを結合する Web サービス技術と、分散した資源を仮想化して共有する技術であるグリッドコンピューティングを融合したものを、Open Grid Service Architecture (OGSA) と呼ぶ。OGSA はグリッドコンピューティング技術を、科学技術分野だけでなくビジネス分野にも適用拡大することを目指している¹⁾。

具体的には、OGSA は、異機種混合かつ分散環境でのシステム構築の複雑さを減らし、高信頼かつスケラブルで、エンドユーザから見た性能保証が可能なシステム構築を可能にする。さらに、膨大な数のサーバを管理するデータセンタの運用管理の効率化と自動化を可能にすることを目指している (図-1 参照)。

グリッドサービス

OGSA では、Web サービスの標準技術をグリッドコンピューティングに適用する。これにより、従来のように計算資源の遠隔利用という限定された機能から、資源・情報・サービスを仮想化・均一化し、グローバルに共有可能になる。アプリケーションに加えて、サーバや



図-1 グリッド技術と Web サービス技術の統合

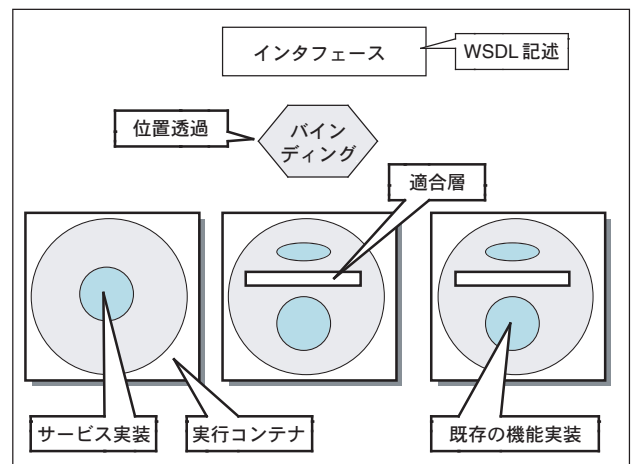


図-2 グリッドサービスのモデル³⁾

ストレージといったすべての計算資源を仮想化する新しいインタフェースを、グリッドサービスと呼ぶ。

グリッドサービスは、扱う資源の状態情報と共通操作を規定し、アプリケーションを含むヘテロな計算資源を統一的に扱うことができる。グリッドサービスでは、以下の Web サービス標準技術を採用する。

- XML (Extensible Markup Language)

XML はインターネットの標準として W3C により勧告された拡張可能なマークアップ言語で、文書管理および電子データ交換フォーマットの規定に使われる。

- WSDL (Web Services Description Language)

XML ベースの Web サービス記述言語であり、Web サービスの持つ機能 (portType) と、それを利用するためのメッセージ (operation) を記述する方法を定義する。また、通信プロトコルやエンコード方式を、機能定義と独立に指定できる。

- SOAP (Simple Object Access Protocol)

XML と HTTP 等をベースとした通信プロトコル。SOAP メッセージは、XML 文書にエンベロープと呼ぶ付加情報を付けたもので、HTTP などのプロトコルを用いて通信する。

グリッドサービスは、Web サービスの拡張スキーマとして定義される。グリッドサービスの基本仕様は、Open Grid Service Infrastructure (OGSI) として、Global Grid Forum (GGF) の OGSI-WG で策定され、初版の仕様書が公開されている²⁾。

グリッドサービスは、WSDL を採用することで、インタフェースと具体的な実装とを、完全に分離する。さらに実装との通信手段 (トランスポート) もバインディ

ングの機構により、複数の選択肢の中から最適なものを選択することができる。たとえば、リモートサーバにアクセスする場合には SOAP を、同一のサーバ内でアクセスする場合は、高速で信頼性のある通信手段を選択して、柔軟で効率的なアクセスを実現する (図-2 参照)。

Web サービスが、継続して提供されるサービスを前提にしているのに対し、グリッドサービスは、動的かつ一時的なサービスの生成消滅が頻繁に発生するという特徴を持つ。その結果、グリッドサービスはいくつかの点で Web サービスを拡張したものになっている。

すなわち、グリッドサービスインスタンスを管理するために、インスタンスへの参照 (ハンドル) と、インスタンスの集合を管理する仕組みを導入する。また各グリッドサービスインスタンスは寿命と内部状態を持ち、内部状態は XML を用いて記述される。インスタンスの状態変化は、非同期通知機構によりクライアントに通知される。さらに、これら機能を実装するために、WSDL インタフェースの継承機能が必要になった。

しかし、現時点での最新仕様である WSDL 1.1 では、グリッドサービスが必要とする portType の継承機能、および portType への情報要素の付加 (Open Content Model) ができない。この2つの言語拡張は、OGSI-WG から W3C に拡張提案されており、次版の WSDL 1.2 で採用されることが決まっている。

上記2点以外の拡張は、グリッドサービスとして共通で持つべきインタフェースを WSDL 1.1 を用いて規定することで実現している²⁾。

OGSI が規定している portType と Operation の一覧を表-1 に示す。このなかで GridService だけが必須の portType であり、それ以外の portType はオプションである²⁾。



| | portType | Operation | |
|----------|--------------------------|---|--------------------------------------|
| 基本機能 | GridService | findServiceData setServiceData requestTerminateAfter requestTerminateBefore destroy | サービスデータの参照と更新 インスタンス管理および 終了指示 |
| ネーミング | HandleResolver | findByHandle | GSHからGSRへの変換 |
| 非同期通知 | NotificationSource | subscribe | 通知の購読依頼 |
| | NotificationSubscription | | 購読情報の取得 |
| | NotificationSink | deliverNotification | 非同期通知 |
| インスタンス生成 | Factory | createService | インスタンスの生成 |
| サービスグループ | ServiceGroup | | グループの管理 |
| | ServiceGroupEntry | | 関係の定義 |
| | ServiceGroupRegistration | add remove | グループの追加と削除 |

表-1 OGSi が規定する portType の一覧表

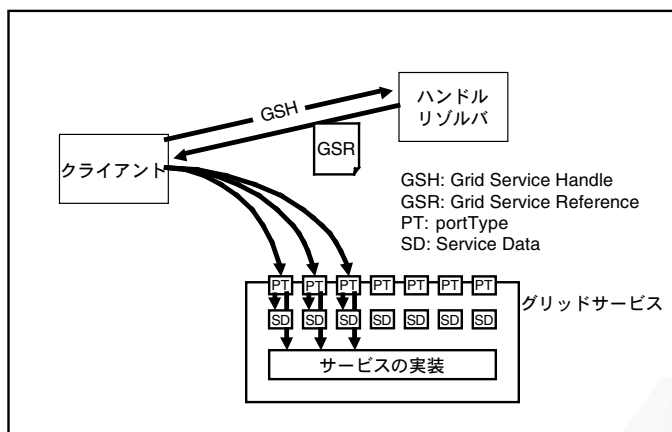


図-3 グリッドサービスのGSHとGSR

• 名前付けとバインディング

グリッドサービスは、ユニークな Grid Service Handle (GSH) を持つ。ハンドルリゾルバは、GSH から Grid Service Reference (GSR) に変換する。GSR は、WSDL 等で記述された具体的なインタフェース記述である。前述のように GSR は、複数のバインディングを含むことができ、クライアントはその中から最適なものを選択できる (図-3 参照)。

• 情報モデル (サービスデータ)

グリッドサービスの持つ情報は、XML で表したサービスデータ (SD) として提供される。サービスデータは参照と変更ができる。サービスデータはオブジェクト指向言語におけるクラス定義中のインスタンス変数に似ている。

• ライフサイクル

グリッドサービスのインスタンスは、Factory によ

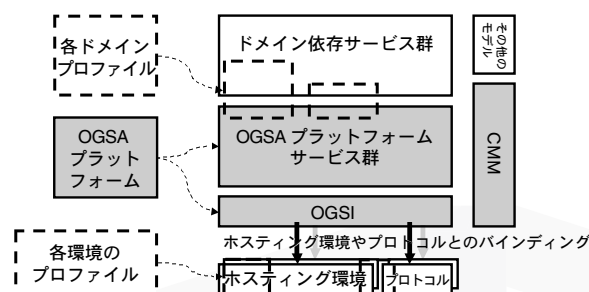


図-4 OGSA プラットフォーム³⁾

て作られる。インスタンスは明示的に破壊することもできるし、ソフトステート (リース期間) を指定することで、間接的に消滅させることもできる。

• 非同期通知

サービスデータの変化を非同期に通知するための、送付依頼と、実際にイベントを送付するインタフェースを持つ。

OGSA プラットフォーム

OGSA のアーキテクチャは、異機種プラットフォームで共通な分散リソース管理モデルを提供し、エンドユーザーに対し性能 (QoS) を保証するものである。さらに、システムの自律制御の基盤を提供し、個別の専用システムではなく、制御機能をコンポーネント化して、組合せ可能とすることを目指している。

GGF の OGSA-WG が検討している、OGSA プラットフォームの全体像を図-4 に示す。前述の OGSI は、グ

| 名前 | 機能概要 |
|------------------------------|--------------|
| Service Groups and Discovery | サービスの登録と検索機能 |
| Service Domain | ディレクトリサービス機能 |
| Security | セキュリティ機能 |
| Policy | ポリシー管理機能 |
| Data Management | ファイルやデータベース |
| Message & Queuing | メッセージ機能 |
| Event | イベント機能 |
| Distributed Logging | 分散ログ機能 |
| Metering and Accounting | 計量と課金サービス機能 |
| Administration | 管理サービス機能 |
| Transaction | トランザクション機能 |
| Grid Service Orchestration | ワークフロー機能 |

表-2 OGSA-WG で議論しているプラットフォームサービス一覧

| | | |
|---|---|--|
| システム 構成管理 - System configuration management - Resource instrumentation management | 自律制御 - Distributed Cluster - Adaptive Resource Allocation - Distributed Load Balancing | |
| | リソース管理 - Resource Reservation - Brokering - Policy Management - Deployment | ジョブ実行管理 - Task Management - Scheduling - Distributed Job Workflow |
| その他の機能 - Distributed log & trace - Distributed accounting -Reliable Messages - Distributed security - System monitoring -Service Collection - Database abstraction - User Management | | |

図-5 商用ビジネスコンピューティングに必要なプラットフォームサービス群

リッドサービスを定義し管理するものとして、この図の中では下位のモジュールとして示されている。

OGSI の上位に位置する OGSA プラットフォームサービス群は、OGSI の機構を使いながら、サービスの検索や監視、データのアクセスや統合といった、OGSI では直接は提供していない、より高度な機能を提供する。

OGSA プラットフォームの3つ目の要素が、Common Management Model (CMM) と呼ばれる共通モデルである。ハードウェアおよびソフトウェア要素を、標準操作インタフェースを持った、グリッドサービスとして表すものである。従来の、システム管理の標準である MIB (Management Information Model) や CIM (Common Information Model) は、システムリソースの静的および動的な状態情報を共通化するものであるが、CMM は、状態情報に加え、標準操作 API を規定する。それにより、クライアントは、システムリソースごとの詳細仕様を知らなくても、標準 API を使って制

御が可能になる。

また、直接 OGSA プラットフォームには含まれないが、グリッドサービスを実行するホスティング環境と、通信プロトコル、さらには、特定のアプリケーション領域 (ドメイン) 向きのサービス群の位置づけも図-4で示されている。

OGSA プラットフォームサービス

OGSA プラットフォームサービスは、既存のプラットフォームの OS が提供している機能を拡張したものである。相互運用性を保証するためには、標準のインタフェース仕様を規定する必要がある。現在、GGF の OGSA-WG において、サービスの特定と優先度付けが活発に議論されている。表-2に、文献3) で述べられているサービスの一覧を示す。

我々は、商用のビジネスコンピューティングに必要な OGSA プラットフォームサービスを整理し、表-2を支持しさらに補完することを目的として、OGSA-WG に対して積極的に提案を行っている (図-5 参照)⁴⁾。

- ジョブ実行管理機能
ジョブリクエストを受け付けて、ユーザの認証を行い、予約から実行完了までジョブの管理を行う。Globus Toolkit 2 の GRAM を拡張した機能。
- リソース管理機能
システムが持つすべてのリソースを管理し、ジョブリクエストに応じて、ブローカリングを行い、リソースの事前予約および実際の割付を行う機能。ジョブの



| 分類 | 機能 | α版での提供範囲 |
|-----------|--------------------------|---------------------------|
| OGSI 関連 | OGSI 参照実装 | |
| | OGSA セキュリティ | GSI ベースの SOAP セキュリティ |
| | Java SDK | グリッドサービス作成支援ツール |
| | 実行環境 | 4種類のホスティング環境を実現 |
| グリッドサービス | OGSA-DAI | 英国 eScience が開発した DB サービス |
| | GRAM | Java ラッパーで実現 |
| | GridFTP | GT2 互換機能 |
| | MDS index サービス | Java による実装 |
| | Reliable File Transfer | グリッドサービスのプロトタイプ実装 |
| | Replica Location Service | OGSA-Security 互換 |
| クライアントAPI | Java API | WSDL ベースの Java バインディング |
| | GRAM クライアントAPI | GT2 互換 |
| | Grid FTP クライアント | GT3 セキュリティをサポート |
| ツール | GT3 グリッドサービス | サンプル, 解説書, デモプログラム |
| | Grid FTP クライアントツール | GT3 セキュリティ互換 |

表-3 GT3 αの提供機能一覧

実行に必要なプログラムやデータのデプロイメントを行う。

- システム構成管理機能
前述のCMMに基づき、システムの構成要素をすべて一元的に管理する機能。Globus Toolkit 2のMDSに対応する機能。
- 自律制御機能
システム構成要素の故障に対応し、フェイルオーバーを行ったり、処理負荷の増大に対応し、計算資源を追加割付したりする機能。
- その他の基本機能
上述の機能を実現する基礎となる、セキュリティ機能やユーザ管理の機能、データサービス機能等。

OGSIの実装

OGSIの正式仕様の策定に先行して、複数の研究グループがOGSIの参照実装を開発している。Globusチームと欧州富士通研究所では、OGSIのドラフト仕様をもとに、参照実装をそれぞれ独立に開発済みで、オープンソースとして公開している。

■ Globus Toolkit 3

Globus チームは、OGSI 機能を実装した Globus Toolkit 3 (GT3) のα版を2003年1月に公開した。GT3は、OGSIの参照実装という側面と、従来のGT2の後継という2つの側面を持つ。そのため、GT3は、OGSI実装と、上位のGT2相当のグリッドサービス群から構成されている。GT3 αの提供機能一覧を表-3に示す⁵⁾。

OGSI実装はSOAP処理系にはapache Axisを使用し、Axisに深く依存した実装になっている。GT2のGRAMとGridFTPは、ラッピングによりグリッドサービス化して提供されている。一方、MDSは、グリッドサービスとして再実装された。セキュリティ機能は、従来のGSIを利用するトランスポート層での実装と、新規に開発した、やはりGSIベースのSOAP通信層でのWS-Security実装が提供されている。GT3の正式版は、2003年の6月に公開が予定されている。

■ UNICORE デモンストレータ

欧州富士通研究所では、EuroGridでも採用されたUNICOREのOGSI対応として、デモンストレータを開発し公開した。デモンストレータは、UNICOREの拡張用インタフェースにMind Electric社のアプリケーションサーバであるGLUEベースの、SOAPエンジン

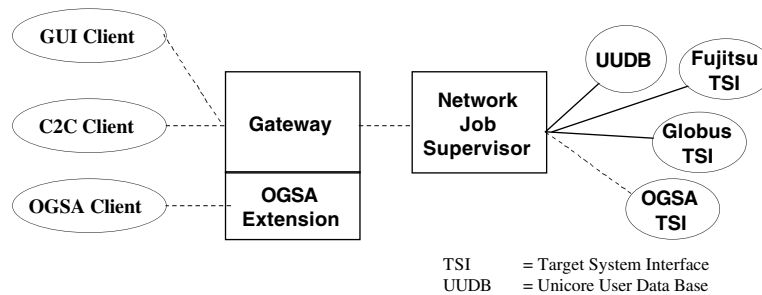


図-6 UNICORE デモンストレータの構造

とサービスコンテナを追加することで実現されている(図-6 参照)⁶⁾。デモンストレータを使えば、OGSI インタフェース経由で、実際に UNICORE サーバを利用することができる。英国の RealityGrid プロジェクトでは、UNICORE デモンストレータを使ったグリッドテストベッドを構築した。

展望

グリッド技術の標準化団体である GGF は、多数の作業部会と研究部会から構成されており、さまざまな技術領域の標準仕様策定を推進している。OGSA は今後のグリッドコンピューティングの主流になっていく重要なアーキテクチャであり、多くの部会で OGSA 対応の検討が始まった。OGSA が浸透した結果、今後は、複数の作業部会が勧告する技術仕様間の調整が大きな問題になる可能性がある。OGSA-WG が全体の調整と推進を行う役割を担っているが、各作業部会の自主性と生産性を維持しつつ、システム全体の整合性をとっていくことが、OGSA の成功の鍵となると思われる。

参考文献

- 1) Foster, I., Kesselman, C., Nick, J. and Tuecke, S.: The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration, Globus Project (2002), <http://www.globus.org/research/papers/ogsa.pdf>
- 2) Tuecke, S., Czajkowski, K., Foster, I., Frey, J., Graham, S., Kesselman, C., Maquire, T., Sandholm, T., Snelling, D. and Vanderbilt, P.: Open Grid Services Infrastructure (OGSI) Version 1.0, http://www.gridforum.org/ogsi-wg/drafts/draft-ggf-ogsi-gridservice-29_2003-04-05.pdf
- 3) Foster, I. and Gannon, D.: The Open Grid Services Architecture Platform, <http://www.ggf.org/Meetings/ggf7/drafts/draft-ggf-ogsa-platform-2.pdf>
- 4) Kishimoto, H., Savva, A., and Snelling, D.: OGSA Fundamental Services: Requirements for Commercial GRID Systems version 1.9, (Oct. 2002), http://www-unix.gridforum.org/mail_archive/ogsa-wg/pdf00002.pdf
- 5) Status and Plans for Globus Toolkit™ 3.0, <http://www.globus.org/toolkit/gt3-factsheet.html>
- 6) Snelling, D.: Readme for prototype GRID Service, Fujitsu Laboratories of Europe, 10th May 02, Version 1.0, <http://www.unicore.org/>
(平成 15 年 4 月 27 日受付)

