

資源予約と連携した階層型分散資源モニタリングシステムの設計

竹房 あつ子^{†1} 中田 秀基^{†1} 柳田 誠也,^{†1,†2}
岡崎 史裕^{†1} 工藤 知宏^{†1} 田中 良夫^{†1}

グリッドとネットワークプロビジョニング技術を用いることで、複数管理組織を跨る高品質の仮想計算基盤を動的に構築・提供することが可能になった。しかしながら、分散する多様な資源で構成される仮想計算基盤の利用者が、確保した資源群の状況を把握するのは困難である。本研究では、資源予約により複数管理組織を跨る資源群を利用するユーザに対し、利用資源のモニタリング情報を収集し、各管理者の開示ポリシーに基づき資源情報を提供する、階層型分散資源モニタリングシステムを提案する。提案システムでは、GridARS コアロケーションフレームワークを用いて資源予約との連携を実現する。本稿では、提案システムの設計について述べるとともに、本システムによる計算機およびネットワーク資源のモニタリング情報の収集・提供方針を述べる。

Design of a Hierarchical Distributed Resource Monitoring System in Cooperation with Resource Reservation

ATSUKO TAKEFUSA,^{†1} HIDEMOTO NAKADA,^{†1} SEIYA YANAGITA,^{†1,†2}
FUMIHIRO OKAZAKI,^{†1} TOMOHIRO KUDOH^{†1} and YOSHIO TANAKA^{†1}

Grid and Network provisioning technology enabled to construct high-quality virtual computing infrastructures spanning several administration organizations. However, it is still difficult for the users to monitor the usage of distributed and various resources. We propose a hierarchical distributed resource monitoring system that gathers information based on resource reservation and filters information with the policies specified by the administrators. The system co-works with GridARS co-allocation framework to retrieve resource reservation information. In this paper we introduce the design of the proposed system and show an initial adaptation of the system with computer and network resource monitoring.

1. はじめに

グリッドとネットワークプロビジョニング技術を用いることで、複数管理組織を跨る高品質の仮想計算基盤を動的に構築・提供することが可能になった。我々の開発する GridARS¹⁾ システムでは、複数資源マネージャと連携し、要求された資源性能を保証する資源群を事前予約に基づき同時確保するコアロケーションフレームワークを提供する。これにより、ユーザは WSRF に基づく標準インタフェースから時間や資源性能に関する要求を送るだけで、分散する複数組織の多様な資源を容易に確保することができる。

仮想計算基盤の構築事例として、G-lambda プロジェクト²⁾ と米国 EnLIGHTened プロジェクト³⁾ の共同実験が挙げられる。G-lambda では GridARS, EnLIGHTened では HARC⁴⁾ コアロケーションシステムを用いて、日米を跨ぐ3つのネットワークドメイン

と10サイトのクラスタ計算機を管理する複数資源マネージャと連携して、ユーザの要求する資源群を確保し、動的に構築された仮想計算基盤上で MPI で実装された並列アプリケーションの実行や HD ビデオストリーム通信に成功した⁵⁾。

しかしながら、仮想計算基盤を構成する資源は多様であり、管理者が複数存在し、かつ分散しているため、資源を確保した利用者が予約した資源群の状況を把握するのは困難である。G-lambda と EnLIGHTened の共同実験では、予約資源モニタサービスを開発して分散する資源の予約完了状態、利用可能状態、解放状態等の予約情報を可視化し、直観的に予約情報を収集・提供することができた。しかし、デモでの利用を目的としているため、認可は行わず全ドメインの全資源予約情報を開示している。また、各資源の予約状況を管理する資源マネージャ群から提供される資源予約に関する情報のみしか扱っていないため、仮想計算基盤の利用時におけるホスト間のコネクティビティの有無、帯域、遅延、CPU稼働率等の資源モニタリング情報を容易に知ることはできない。よって、詳細な資源状

^{†1} 産業技術総合研究所 National Institute of Advanced Industrial Science and Technology (AIST)

^{†2} 数理技研 SURIGIKEN Co., Ltd.

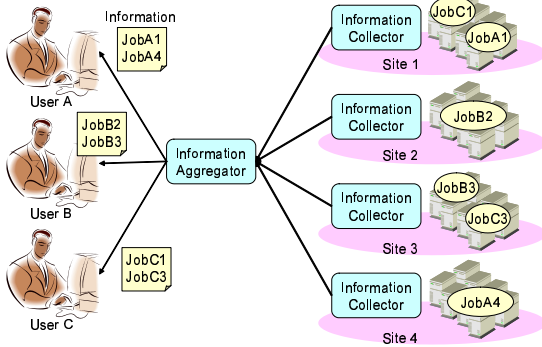


図1 情報サービスシステムの概要

況をドメインごとの開示ポリシーに従って提供するモニタリングシステムが必要である。

本研究では、資源予約管理システムと連携した階層型分散資源モニタリングシステムを設計する。我々は認可モデルとポリシー記述言語の標準である XACML (eXtensible Access Control Markup Language)⁶⁾を用い、サイト毎に情報開示ポリシーを定義して、各ユーザからの情報アクセス制御を可能にする情報サービスシステムを開発している⁷⁾。これを改良し、より大規模かつ複雑な管理ドメイン構成に対応する、階層型モニタリングシステムを設計した。資源予約との連携では、GridARSの資源管理システムから資源予約に関する情報を取得して、資源モニタリングを実施するための機構を新たに追加する。また、本モニタリングシステムで授受される資源情報には、汎用性を考慮して標準的なデータ表現である GLUE バージョン 2.0⁸⁾を拡張して用いる。提案システムにより実際に取得・提供可能な資源情報として、計算機とネットワークに関する資源モニタリング情報の収集、提供方針を述べる。

2. XACML を用いた情報サービスシステムの概要と課題

先行研究である XACML を用いた情報サービスシステムの概要を図1に示す。本情報サービスシステムは複数サイトに分散する各ユーザの情報を統合して提供する Information Aggregator (IA) と、各サイトにおいて認可された情報を提供する Information Collector (IC) からなる。IA はユーザから情報取得要求を受け取ると、関連するサイトの IC にそのユーザの権限を移譲して問い合わせる。各 IC では、XACML を用いて認可判定を行い、開示可能な情報を提供する。IA は各 IC から取得した情報を統合し、要求したユーザに送信する。

本情報サービスシステムは、IA と IC からなる2層構造であり、IA、IC はそれぞれ WSRF⁹⁾ インタフェースを提供している。しかしながら、実環境では図2の

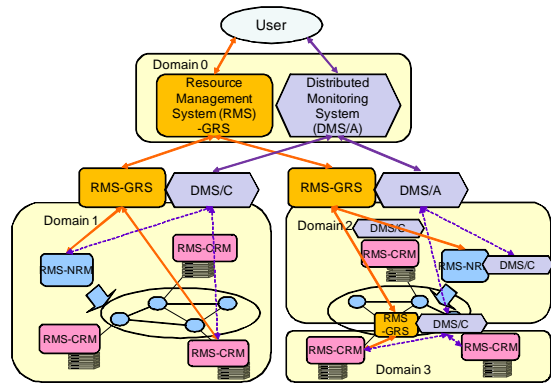


図2 階層型分散資源モニタリングシステムの概要

Domain0, Domain2, Domain3 のように管理ドメインが階層的に構成され、上位の資源管理システムが下位のドメインの存在を知らされない場合がある。また、Domain1 のように複数資源を一括して管理するドメインも存在するため、どこの層で認可判定を行うべきかは各管理ドメインごとの判断に任される。さらに、資源予約との連携も実現しておらず、予約資源のモニタリングを行うための機構も必要となる。

3. 階層型分散資源モニタリングシステムの設計

3.1 モニタリングシステム (DMS) の概要

図2に本研究で提案する階層型分散資源モニタリングシステム (DMS) の概要を示す。図中の DMS は本研究で提案するモニタリングシステムモジュール、RMS は資源管理システムモジュールを示し、RMS はグリッド資源コーディネータ/スケジューラ (GRS) と計算資源マネージャ (CRM) およびネットワーク資源マネージャ (NRM) からなる。CRM と NRM は対象資源である計算機およびネットワークの資源を管理し、提供する資源サービスを表す。Domain0-3 は資源の管理ドメイン/組織を示す。2節で述べたように、実環境では一組織が計算資源やネットワーク等の複数資源を提供する場合や、他の組織の RMS を介して間接的にユーザに資源提供を行う場合もある。

本研究では、先行研究における IA と IC の“インタフェース”と“機能”を分離し、新たに IA、IC 統一のインタフェースを策定して、図2に示すように DMS を階層的に構成できるように設計した。また、各 DMS モジュールではその管理ドメインの判断により、IA の情報集約機能、IC の XACML を用いた認可判定に基づく情報提供機能を使い分けられることができるようにした。図2において、DMS/A は IA の機能を有する DMS、DMS/C は IC の機能を有する DMS を表す。これにより、どの階層でも認可に基づく情報提供が行えるようになるとともに、上位の DMS からは

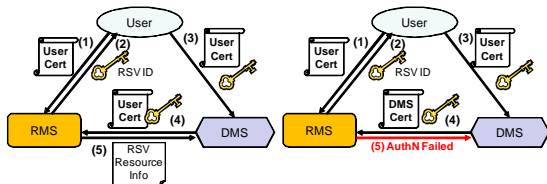


図 3 RMS と DMS 間での認証と資源情報取得. 左は権限委譲する場合. 右は権限委譲しない場合.

下位の DMS の構成によらず, 同じ手続きにより情報収集が可能になる.

資源予約との連携では, GridARS システムを用いた RMS と連携し, 資源予約時の予約 ID を鍵として GridARS から予約情報を取得し, 関連する RMS において予約された資源のモニタリングの実行を可能にする.

3.2 資源予約との連携と認証

GridARS は, WSRF に基づく事前予約インタフェースを提供するコアケーションフレームワークである GridARS-WSRF と, GridARS-WSRF を介して受け取ったユーザの時間と資源の数および性能に関する要求をもとに, 適切な資源群を探して同時予約手続きを行う GridARS-Coscheduler からなる. GridARS-WSRF は全 RMS モジュールで, GridARS-Coscheduler は GRS で利用することができる.

GridARS を用いて資源予約手続きを行う場合, 各予約要求に対してそれらを参照するためのエンドポイントリファレンス (EPR) が返される. これが予約 ID となり, GridARS のユーザは資源予約手続きを行ったり, 資源予約に関する情報を取得することができる. よって, DMS に対してモニタリング要求を送る場合, 資源予約の際に受け取った予約 ID を渡すことにより, DMS が資源予約に関する情報を閲覧できるようにする.

GridARS-WSRF は GSI (Grid Security Infrastructure) をサポートしており, ユーザ/RMS と RMS 間の資源予約手続きにおいて認証機能を利用することができる. GSI の認証では, ユーザの権限を委譲して下位への手続きを行う場合と, 権限委譲しない場合がある. 研究機関同士の相互計算機利用を目的とした従来のグリッドでは, 権限委譲してシングルサインオンによる資源利用が有効であった. 一方, 商用サービスではサービスが仲介される場合, 一般に上位のユーザの情報を下位のドメインに対して公開しないため, 権限委譲しない可能性が高い.

権限委譲しない場合, 資源予約と DMS の連携において図 3 に示すような問題が発生する. 図 3 では, 左が権限委譲する場合, 右が権限委譲しない場合を示している. 権限委譲する場合 (左) では, (1) ユーザの証明書で予約手続きをし, (2) 予約 ID を取得する. (3) ユーザの証明書で予約した資源のモニタリング要求を

予約 ID とともに送ると, (4) DMS がユーザの証明書と予約 ID を用いて, (5) 予約情報を取得することができる. 一方, 権限委譲しない場合 (右) では, (4) のフェーズでユーザ証明書を利用することができないため, (5) のフェーズで予約情報の取得に失敗してしまう.

権限委譲しない場合においても DMS が予約情報を取得できるようにするには, RMS と DMS を同じホストのサービスコンテナでサービスするか, あらかじめ RMS と DMS 間で認証に関する契約が行われていなければならない. しかしながら, RMS と DMS は密接に関係しており, 双方は同じ管理者によって運用される可能性が高い. 本研究では, 運用上の利便性を考慮して, 前者を前提として開発を進めることにした.

3.3 DMS インタフェースと情報取得プロセス

DMS では先行研究の IA, IC のインタフェースを共通化した, インタフェースを提供する. 表 1 に DMS の主なサービスオペレーションを示す. 表中のモニタリング ID は個々のモニタリング要求に対して DMS から提供されるものであり, 予約 ID は RMS から資源予約の際に発行される ID を表す. 各オペレーションの機能は図 4 において説明する.

図 4 は資源予約終了時からモニタリング情報を取得するまでのプロトコルシーケンスを示す. ここでは, 図 2 のような 4 つの階層的なドメインの構成を想定する. 図中の灰色の矢印は平行して同時に手続きを行っていることを表している.

まず, ユーザは Domain0 の RMS0 を介して Domain1 と Domain2 の資源を事前予約で確保し, ユーザは RMS0 から予約 ID RsvID0 を受け取る. その際, RMS0 から RMS1, RMS2 に対して予約手続きが行われており, RMS0 は予約 ID RsvID1, RsvID2 をそれぞれ受け取って, それらを RMS0 が保持している. RMS2 では, さらに Domain3 の RMS3 に対して予約手続きを行っており, 予約 ID RsvID3 を取得している. 予約手続きの詳細は文献¹⁾を参照されたい.

予約手続きが完了すると, ユーザは RsvID0 を用いて DMS0 に対して create でモニタリング手続きの開始要求を送り, DMS はそのモニタリング手続きのためのモニタリング ID MID0 を返す. 以降のオペレーションでは, ユーザはこのモニタリング ID を用いて手続きを行う.

次に, ユーザは DMS0 に対して configure でモニタリングで取得したい資源情報セットを設定する. 資源情報セットは取得したい資源情報をあらかじめ指定するものであり, 資源情報の詳細は 4 節で述べる. configure では, DMS0 は RsvID0 を用いて RMS0 から予約情報を取得し, Domain1 と Domain2 に関連する予約があることを知り, DMS1, DMS2 に対して同様に create と configure を行う. DMS2 でも同様に DMS3 に対して処理する.

表 1 DMS の主なサービスオペレーション

オペレーション名	機能	入力 / 出力
create	モニタリング手続き開始	予約 ID / モニタリング ID
configure	収集する情報セットを指定	モニタリング ID, 情報セット / -
start	情報収集を開始	モニタリング ID (, 時刻) / -
stop	情報収集を停止	モニタリング ID (, 時刻) / -
getInformation	指定した情報の取得	モニタリング ID (, 時間帯) / モニタリング情報

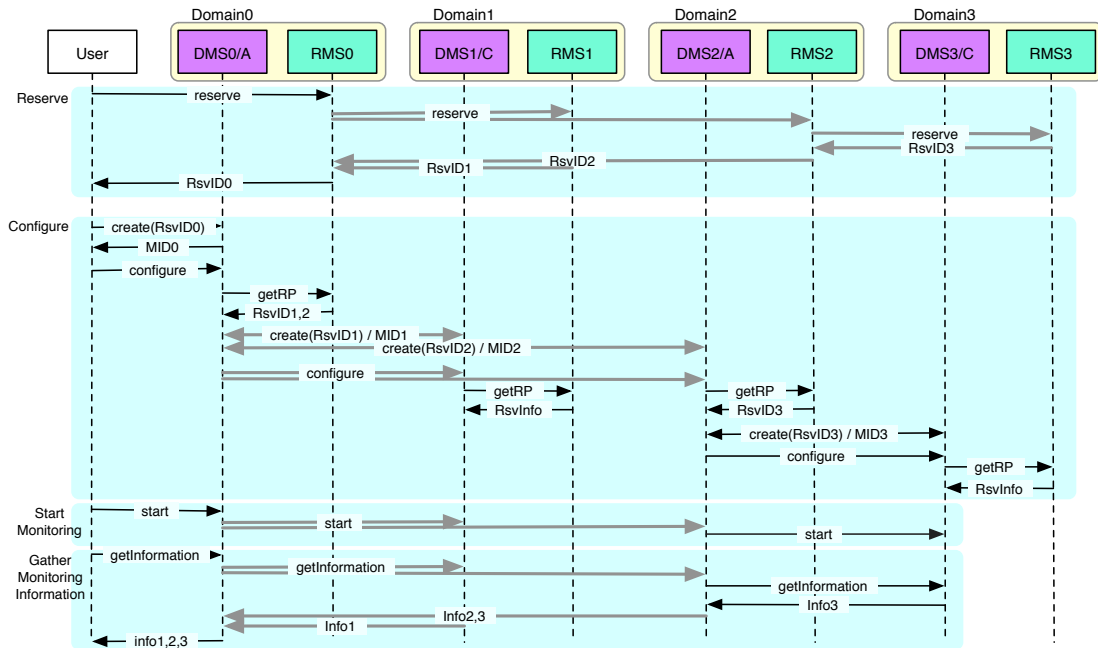


図 4 情報取得プロセス

その後、ユーザが DMS0 に対して start を実行すると、DMS0 が DMS1 と DMS2 に、DMS2 が DMS3 に start を発行し、最終的に最下位の DMS において定期的な資源情報の収集処理が開始される。この際、時刻情報を同時に送ると、指定した時刻から情報収集を開始することができる。資源予約との連携では、資源が利用可能になる前に start が実行された場合は、予約開始時刻に情報収集が開始される。同様に、stop は最下位の DMS において資源情報の収集処理を停止させるために利用する。時刻が指定された場合は、その時刻に収集処理を停止する。

最下位の DMS で収集した資源情報を取得するため、ユーザは DMS0 に対して getInformation を発行すると、DMS0 から下位の DMS に getInformation が発行されていき、最終的に関連する予約資源の資源情報 Info1,2,3 が各層において適宜認可されて取得することができる。また、時間帯を指定すると、指定した時間帯の資源情報のみを受け取ることができる。オプションとして、WS-Notification¹⁰⁾に基づく資源情報提供 (subscribe / unsubscribe) も可能にする。

3.4 DMS におけるデータ表現

DMS 間で授受される資源情報には、汎用性を考慮して標準的なデータ表現である GLUE バージョン 2.0⁸⁾ を拡張して用いる。GLUE では、主に計算資源に関する情報の XML データ表現が定義されているが、ネットワーク資源に関する情報は定義されていない。また、モニタリングで扱う時系列データに関する定義もない。よって、ネットワーク資源および時系列データに関する拡張を行って用いる。

4. DMS における資源モニタリング情報の収集

DMS における資源のモニタリングでは、先行研究の IC の機能を利用する。図 5 に IC の概要を対象資源および資源管理システム (RMS) とともに示す。IC では、図中の MA で表すモニタリングエージェント群が収集した情報を、XACML を用いた認可判定を行った後、ユーザに提供される。以下に、計算機およびネットワーク資源のモニタリングエージェントによる情報収集の設計指針について述べる。

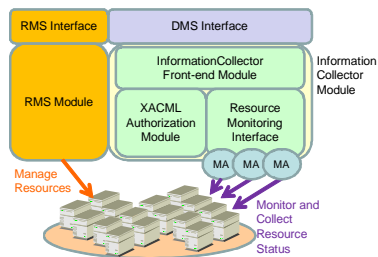


図 5 IC の概要

4.1 計算機資源の情報収集

収集される計算機資源情報は、(a) 予約自体に関する情報、(b) 当該予約に関連する資源利用に関する情報、(c) アプリケーションの実行に関する情報からなる。(a) は、当該予約の内容を表したものであり、予約 ID、予約した CPU 数やメモリ量、当該予約の開始時刻および終了時刻、予約のステータス等が含まれる。(b) は、当該予約に関連する資源利用状況の統計情報を表したものであり、使用 CPU 時間、使用メモリ量、使用ディスク量等が含まれる。ただし、(b) は予約単位に集計されたものであり、Ganglia¹¹⁾ のように各ドメインで実際に動いているプロセスの統計情報をそのまま列挙したものではない。また、SMP やマルチコアを搭載した計算機のように、1 ノードに複数のユーザのジョブが混在する場合においても、他のユーザの処理状況は公開されない。(c) は、特定のファイルに書き出されたアプリケーション固有の実行ログを示す。これにより、ユーザが計算機に直接ログインできない環境でも、DMS の枠組みで容易にアプリケーションの実行状態を取得することができる。

(a)、(b)、(c) の情報収集は次のように行われる。(a) は、CRM に対して問い合わせを行い取得する。CRM への問い合わせは、予約情報を取得する際と同様に、CRM の Web サービスインタフェースを介して行われる。(b) は、現在稼働中のプロセスについては、OS の提供するプロセス情報インタフェースを利用して取得し、終了したプロセスについては、ジョブ管理ミドルウェアのログファイルや OS の提供するプロセスアカウントパッケージのログファイルから取得する。OS が Linux の場合の具体的な取得方法としては、前者については、proc ファイルシステムへのアクセス、後者については、psacct パッケージのログファイルからの取得が、それぞれ挙げられる。(c) はアプリケーションごとに指定されたファイルの内容を提供する。

4.2 ネットワーク資源の情報収集

NRM では、一般に管理するドメイン内のネットワークの状態を監視し、健全性を確認するためのネットワーク管理システム (NMS) が常に動作している。また、NMS は NRM と連携し、個々のユーザの要求したパスのセットアップが正常に行われ、帯域が保証されていることを確認するためのモニタリングも行って

いる。よって、ネットワーク資源におけるモニタリングエージェント (MA) では、NMS の収集するモニタリング情報が格納されている NRM 内部の DB にアクセスすることで、情報を取得する。取得する情報には、認証情報、予約に関する情報、資源モニタリング情報がある。

NMS のモニタリングでは、NRM はパスのセットアップ時 (予約開始時刻) に NMS にこのパスのモニタリングを依頼し、パスの切断時 (予約終了時刻) にモニタリングの終了を通知する。NMS はモニタリング情報をモニタリング時刻と共に DB に順次格納する。モニタリング周期は、ネットワーク機器への負荷や情報の更新周期を考慮して、NMS で決定する。MA は図 5 に示す IC の資源モニタリングインタフェースから受け取った予約 ID およびユーザ情報をもとに、DB から NMS の収集したモニタリング情報を取得することができる。

NMS のモニタリングは、指定されたパスの両端のネットワーク機器のインタフェースに対して行う。NMS がモニタリング情報として収集する情報の内、DMS に提供する情報はインタフェースのリンクの状態 (Up/Down) とパケット統計情報に限定する。物理インタフェースが VLAN 等で複数の仮想インタフェースで共有される場合は、パスが使用する仮想インタフェースを対象にする。パケット統計情報には、入出力のパケット数と転送量があり、パケット数には転送パケット数、破棄パケット数、エラーパケット数がある。帯域保証が行われている場合、対象となるパスの両端のインタフェースにおける入力パケット数と出力パケット数が同じになると考えられる。よって、パケット数に関する統計情報を検査することで帯域保証を確認することができる。また、入力側の破棄パケット数が 0 でない場合は、計算機資源や他のネットワーク資源などドメインの接続点で予約帯域以上のパケットが転送されて来たことを示している。

NMS が端点のインタフェースにアクセスして各種情報を取得する手段として、SNMP (Simple Network Management Protocol)¹²⁾ が一般に使用できる。SNMP のモニタリング情報は標準的な MIB (Management Information Base)¹³⁾ に定義されている項目なので、ほぼ全てのネットワーク機器から取得できる。ただし、物理インタフェースではなく VLAN 等の仮想インタフェースの情報が取れるか否かは、ネットワーク機器に依存する。SNMP で取得できない場合には、ネットワーク機器のリモートコンソールによる CLI (Command Line Interface) を利用する。

4.3 拠点間情報の収集

拠点間のスループットや遅延およびその揺らぎの測定はネットワーク資源の情報に属するが、拠点間のパスが 1 つのネットワークドメインから提供されると

は限らないため、NRM でのモニタリングでは測定できない。よって、確保した各 CRM で予約により動的に構成される計算機のリストを共有させ、拠点内の 1 ノードから異なる拠点内の任意のノードに対して拠点間情報を取得する機能を CRM に付加する。ユーザが DMS に対して拠点間情報を要求した場合は、CRM 内であらかじめ収集されている拠点間情報を提供する。拠点間情報は ping 等により測定する。

5. 関連研究

MonALISA (MONitoring Agents using a Large Integrated Services Architecture)¹⁴⁾ は Ganglia や MRTG (Multi Router Traffic Grapher)¹⁵⁾ 等のモニタリングツールで収集された情報をレポジトリに格納し、クライアントインタフェースから提供する。認証は行うものの、開示する情報の認可は行っていないため、同一レポジトリにアクセスするユーザは全ての情報にアクセス可能である。レポジトリおよびレポジトリの情報を用いた各サービスへのインタフェースは、ウェブサービスに基づいている。

Inca¹⁶⁾ は TeraGrid¹⁷⁾ におけるユーザレベルグリッドモニタリングシステムであり、エージェントによりユーザの権限で情報を収集・提供する。しかしながら、複数サイトの情報を集中管理しており、サイト外に収集した情報が流出する点、サイト毎に様々なポリシーを定義して細粒度のアクセス制御を行わない点、システム管理者へのモニタリング情報の提供を目的としている点で本研究と異なる。

いずれのシステムも、全ユーザに対して同一のモニタリング情報を提供するものであり、複数ドメイン環境で動的に確保される資源に対し、ドメインごとの開示ポリシーで各ユーザに情報を提供することはできない。

6. まとめ

本研究では、資源予約により動的に構成される複数管理組織を跨る資源群を利用するユーザに対し、利用資源のモニタリング情報を収集し、各管理者の開示ポリシーに基づき資源情報を提供する、階層型分散資源モニタリングシステム (DMS) を提案した。DMS では、先行研究である XACML による認可を行う情報システムを改良して階層的に構成させ、GridARS コアロケーションフレームワークを用いて資源予約との連携を実現する。また、本稿では DMS における計算機およびネットワーク資源のモニタリング情報の収集・提供方針について述べた。今後は本研究の設計をもとに、DMS の開発を進める。

謝辞 本研究の一部は、情報通信研究機構 (NICT) の委託研究「ダイナミックネットワーク技術の研究開発」により実施した。

参考文献

- 1) 竹房, 中田, 工藤, 田中, 関口: 多様な資源を事前予約で同時確保するためのグリッドコアロケーションシステムフレームワーク GridARS, 情報処理学会論文誌コンピューティングシステム (ACS20), Vol.48, No.SIG18, pp.32-42 (2007).
- 2) G-lambda: <http://www.g-lambda.net/>.
- 3) EnLIGHTened Computing: <http://enlightenedcomputing.org/>.
- 4) HARC: The Highly-Available Robust Co-allocator: <http://www.cct.lsu.edu/harc.php>.
- 5) Thorpe, S. R., et al.: G-lambda and EnLIGHTened: Wrapped In Middleware Co-allocating Compute and Network Resources Accross Japan and the US, *Proc. GridNets2007* (2007).
- 6) T. Moses(ed.): *eXtensible Access Control Markup Language (XACML) Version 2.0*, OASIS (2005).
- 7) 竹房, 中田, 柳田, 工藤, 田中, 関口: WSRF に基づく情報サービスの XACML によるアクセス制御, 情報処理学会論文誌コンピューティングシステム, Vol.1, No.2, pp.180-192 (2008).
- 8) Sergio Androozzi(ed.): *GLUE Specification v.2.0 (OGF Working Draft)*, OGF (2008).
- 9) S. Graham and A. Karmarkar and J. Mischinsky and I. Robinson and I. Sedukhin(ed.): *Web Services Resource 1.2 (WS-Resource)*, OASIS (2006).
- 10) S. Graham and D. Hull and B. Murray(ed.): *Web Services Base Notification 1.3 (WS-BaseNotification)*, OASIS (2006).
- 11) Ganglia Monitoring System: <http://ganglia.info/>.
- 12) J.D. Case and M. Fedor and M.L. Schoffstall and C. Davin: Simple Network Management Protocol(SNMP) (1990).
- 13) K. McCloghrie and M.T. Rose: Management Information Base for Network Management TCP/IP-based internets: MIB-II (1991).
- 14) Legrand, C., Newman, H. B., Voicu, R., Cirstoiu, C., Grigoras, C., Toarta, M. and Dobre, C.: MonALISA: An Agent based, Dynamic Service System to Monitor, Control and Optimize Grid based Applications, *Proc. CHEP2004* (2004).
- 15) MRTG: <http://www.mrtg.org/>.
- 16) Smallen, S., Olschanowsky, C., Ericson, K., Beckman, P. and Schopf, J.M.: The Inca Test Harness and Reporting Framework, *Proc. the IEEE/ACM SC2004 Conference* (2004).
- 17) TeraGrid: <http://www.teragrid.org/>.