

報告者: 谷村 勇輔 (産総研)

グループ	Grid Checkpoint Recovery Working Group (GridCPR-WG)
目的	ヘテロな資源からなるグリッド環境において、ユーザレベルでチェックポイントおよびリカバリを行うAPIとそれに関連するサービスレイヤを定義する。チェックポイントされたジョブは、同じマシンだけでなく異なるマシンでもリカバリされ、継続実行されることを想定する。
状況	ユースケースとシステムアーキテクチャのドラフトが出来つつある。Cactusを使った事例やRealityGridでの事例などがドキュメントに加えられた。Scopeも明確になってきたが、ドキュメント自体はもっと整理が必要である。
進捗	restartとrollbackを行う場合のAPIの使用例が提示され、データ、メタデータ、通知などのAPIがどうあるべきか、データのフォーマットはどうするか、ファイルの転送やスケジューリングについてどこまで関係するかが大雑把に議論された。そして、現状のドラフトを完成させるためのスケジュールが話し合われた。
今後	ユースケースとアーキテクチャのドキュメントを、今年の12月末までにGGFの編集者に提出し、Public Commentの段階に入る。
参加者数	15人
所感	データ転送に関して、他のWGとどう協調していくのかが注意が必要である。データフォーマットもどういうものになるか興味深い。ミーティングでは、Cactusのコードがかなり参考にされるような印象を受けた。ML、ミーティングとも議論が着実に進んでおり、予定通りドキュメントが出てくるのではないと思われる。

1

報告者: 中田 秀基 (産総研)

グループ	GridRPC WG
目的	Grid上でのRPCメカニズムのC言語APIを定義する
状況	エンドユーザAPIとミドルウェアAPIの2つを定義しようとしており、前者は前々回にサブミットされている。現在は後者の制定を行っている。
進捗	ミドルウェアAPIの引数構造体をスタックにするかベクタにするかを議論。ベクタに決定。データパーシステンシに関して座長Craig Leeがサマライズ。議論をおこなった。
今後	引数構造体のドラフトを産総研中田が9月中旬にMLに提案、データパーシステンシに関しては今後さらに議論していく。
参加者数	10名程度
所感	

2

GGF12 参加報告

JPGRID-GGF0412

会員限定

報告者: 谷村 勇輔 (産総研)

グループ	Application and Test Beds Research Group (APPS-RG)
目的	アプリケーションの開発者やユーザが、容易にグリッド技術を利用できることを目指し、グリッドの技術を使って何ができるか、できないかを両者に知らせ、また既存技術の改善点を明らかにしていく。多くのアプリケーションの事例を集め、かつAPMEの他のグループと連携してこの目標を達成する。
状況	Chairが変更になり、仕切り直しとなっているが、CharterおよびGroup名が確定していない。GGF11後にApplication RG という名前で申請したが、もっと的確な名前にするよう差し戻された。その後、Charterも変更が加えられて目的は上記のものとなり、グループ内での合意は得られている。グループ名の候補も出され、ミーティングでも名前について議論があったが、結局決まらなかった。
進捗	前回と同じように、アプリケーションの開発と実行に関して、どういうことが問題になるか会場から意見を集めた。テストベッドの信頼性や安定性、利用するたびにテストが必要になること、Firewallの問題などが挙げられた。今後、行うべきワークショップの案が6つ出され、優先順位が議論と多数決で決まった。
今後	次回のGGFでは「The Usable Grid」というワークショップを開き、システム管理者、ユーザ、コンピュータ科学者が議論できる場を設ける。また、アジア地域のグリッドのアプリケーションについて発表してもらう。
参加者数	29人
所感	少し不安な出だしたが、参加者も多くワークショップの内容次第で議論は進むと思われる。

3

GGF12 参加報告

JPGRID-GGF0412

会員限定

報告者: 小島 功 (産総研)

グループ	ASTRO-RG
目的	仮想天文台 (VO) の機能をグリッド上に実現するための RG
状況	VOの機能をどうグリッドに載せるか? について、アーキテクチャ上の議論が進んでいる。また、IVOA (国際的なVO連携機関) との対応や、関連する研究プロジェクト等についての紹介が行われている。
進捗	IVOAの課題について、GGFのグループとの対応、人間などが決まっている。(例えば、VO TableはDFDLと、VOの問い合わせ言語はDAISと関係がある、とか) 問い合わせ言語については国立天文台の大石氏が担当になっている模様。 「なぜグリッドか」「なぜGGFか」「なぜOGSAか」という部分からスタートし、OGSAの有益な機能、ない機能などをまとめていきつつある。例えば、 1. Registry 特に有益な点はない。 2. 認証、権限など: おそらくある。 3. 問い合わせ言語: 機能的に別もの といった具合。
今後	OGSAのユースケースに反映することも含め、OGSAの適用や利点について議論が進んでいくのではないと思われる。
参加者数	15人+
所感	報告者は天文の専門家ではないので詳細についてはさだかではないが、GGFと他機関との連携という意味では興味のあるものであった。 この応用のように、「必ずしも今グリッドでなければならぬ必要もない」応用や分野をどうひきつけていくか? は、これからの課題であると思われる。

4

グループ	Production Grid Management Research Group (PGM-RG)
目的	Production Gridの特徴やサービスを考え、それに向けて必要なこと、経験などをワークショップや調査などを通じてまとめ、共有する。
状況	6月に調査報告書がMLに投稿され、6人からのコメントが反映された。ただし、もう少し整理が必要であること、商業的なグリッドの事例が含まれていないことは問題である。状況も刻々と変わるので、ある時点におけるスナップショットという報告書にならざるを得ない。一方、MLは前回以降盛り上がりならず、議論が進んでいない。前回のワークショップは参加者がそれほど多くなく、今回もワークショップを企画したが、スケジュールが合わないこともあり、結局キャンセルとなった。もう一度、グループの目的を議論し直し、解散も含めた議論が必要である。
進捗	Production Gridとしてのサービスや特徴がもう1度洗い出され、どういったことに焦点を当てていくか、グループとして何を指すか議論がなされた。グループは継続となり、引き続き調査報告書やワークショップを提供していく。特定の問題は、適宜WGを設けて議論することも考える。
今後	次回のGGFで、アジア地域のNational ProjectからPGMに関するWorkshopを企画する。その次のBostonかUKで、Operation Center関連の人を集めて議論する。ML, AccessGrid, 学会などを利用したFace-to-Faceのミーティングを活用し、グループの活性化を図る。
参加者数	12人
所感	一応、明るい結論が出たが、グループが活性化していない現状は大きな問題である。

5

グループ	Grid Application Programming Interface Workshop
目的	アプリケーション開発者とGrid API設計者がAPIに関する知見を持ち寄り、“Simple API”に課せられる要件に関し議論を行う。
状況	2件のKeynote Speech, 7件のPresentation, および「自分の経験に基づいてどのようなAPIが必要としているか」に関するオープンディスカッションが行われた。 発表概要は以下のとおり。(資料は http://www.cs.vu.nl/ggf/apps-rg/meetings/ggf12.html より入手可能) (1)既存Grid Middlewareの紹介 CogKit, GAT, GridRPC, Gridway, HOCの特徴, 現状が各々紹介された。 (2)GridProjectにおいて得られた知見の紹介 Geodise Project, Reality Grid Project, e-Science projectで得られたユーザ要求, 問題点等を報告。 (3)可視化処理用APIの提案 Reality Grid及びNAREGI projectにおいて構築された可視化処理システムの紹介およびAPIの提案がなされた。
	30名
所感	1. 大半の発表は以前のGGF WSで行われたもので、新しい発表はNAREGI Projectの可視化システムの話およびComponent Based Programming環境の話くらいであった。あまりにも頻繁にWorkshopを開催しすぎてネタ切れという感がある。 2. 最後のオープンディスカッションでは、Perl, Python, Matlabのようなスクリプト言語とGrid環境との連携を実現することが有効であり、ニーズも高いという意見が多かった。このようなアプローチが重要だと主張したCoxやLaszewskiのプレゼンがうまかったという点を割り引いて考えても、実際それらの言語を使って簡単に既存のアプリケーションをGrid環境に組み込めるのであれば使いたいというユーザは多いと思われる。

6

グループ	Common Management Model WG (CMM-WG)
目的	<ul style="list-style-type: none"> OGSA の management と manageability のフレームワークを作成すること 他標準化団体とのリエゾン
状況	<ul style="list-style-type: none"> 作成していた GGF informational document を完成し GGF 編集長に提出し、間もなく GGF レビュープロセスが始まる。 上記レビューの修了後、計画通りに OGSA-WG の design team になる。
進捗	<ul style="list-style-type: none"> セッションでは進捗を説明し、フリーディスカッションを行った。GGF, DMTF, OASIS WSDM の参加者がセッション、ディスカッションに参加した。 今まで延期していた「OGSA でどの資源モデルを使うか」という重要な内容からフリーディスカッションを開始し、結局この内容はセッション全体を占めた。OGSA-WG には CIM (特に CIM v3) が十分であるとのコンセンサスがあるが、まずテストとしては OGSA Execution Management Service または Data に使用するところから始める。
今後	現在の活動(OGSA の management と manageability のフレームワーク、他標準化団体とのリエゾン)を継続
参加者数	21人
所感	<ul style="list-style-type: none"> セッション後に GGF, DMTF, OASIS WSDM の参加者と会話では、全員が「かなり良い議論であった」との事で(マシエルより)、リエゾンは順調に進みそうである。

7

- GGF8: WG born to do base manageability functionality for OGSA
- GGF9: Original work moved to OASIS WSDM
 - Started gap analysis (what is missing on manageability in OGSA, what should be done)
- GGF10: Presented management framework and glossary for OGSA
- GGF11: Completed the gap analysis

(注)

- Management: 管理対象をその環境の中でモニタリング、コントロール、メンテナンスし、内部、外部の変化に対して適切に対応する事。
- Manageability: 管理対象のManagementに有効な情報の定義であり、管理対象自体が提供する事も、何らかの別の手段で提供される事もある。
- Manageability Interfaces: Managerが管理対象に対してManagementアクションを行う為の標準化されたインタフェース。
- Manageable Entities: Manageable Interfacesを提供する事で、Managementを可能にする対象。以下のような対象がある。
 - ・物理的なものと論理的なもの
 - ・離散的なものと合成されたもの
 - ・一時的なものと恒久的なもの

8

CMM-WG Current Status (安崎 篤郎)

- GGF12: Informational document submitted to GGF editor
 - Endless thanks to Jem and Latha for final review
 - Will soon enter the 30-day public comment period
 - Please read
 - It's a part of OGSA (full version of what went in OGSA v1 document)
 - Many definitions, policies, etc. not yet divulged enough in GGF (or even in OGSA-WG)
 - Please make comments (good or bad, technical or not) in the GGF tracker

9

Resource Management in OGSA (安崎 篤郎)

- Management of the Grid resources (Functional Interfaces自体)
 - High-level resource management functionality
 - E.g., discovery, resource reservation, provisioning, monitoring and control
 - Management of the OGSA infrastructure (Functional Interfaceを構成するResourcesに対するManageability)
 - Services that compose OGSA are managed resources
 - E.g. monitoring of a registry, of caching and replication
 - Management of the resources themselves (WSRF,WSDM/CIM,SNMPの2つのレベルがある。)
 - E.g., rebooting a host, or setting VLANs on a network switch
 - Mostly out of scope of OGSA
- (注)
- Resource Model:Manageable Entitiesの抽象表現であり、スキーマと属性を表現する。
 - Resources:OGSAの文脈ではManageable Resourcesであり、Manageable Entitiesと同じ意味である。
 - Resource Management:Resourcesに対して適用される色々なManagement形態の総称である。(Managementの為のメカニズムまでは含まない。)

10

Resource Models (安崎 篤郎)

- Many models exist
 - CIM, MIBs (SNMP), JSR77 (JMX), etc.
 - Resource descriptions contain implicit models, e.g.:
 - Brokering, scheduling, job submission (Unicore Resource Schema, GLUE schema, GRIP, etc.)
 - Deployment
 - Metering
- Models have semantic and rendering
(セマンティックは出来れば一つにしたいが、レンダリングは複数可能)
- Many models will be used simultaneously in a Grid
 - Need to coordinate semantics

11

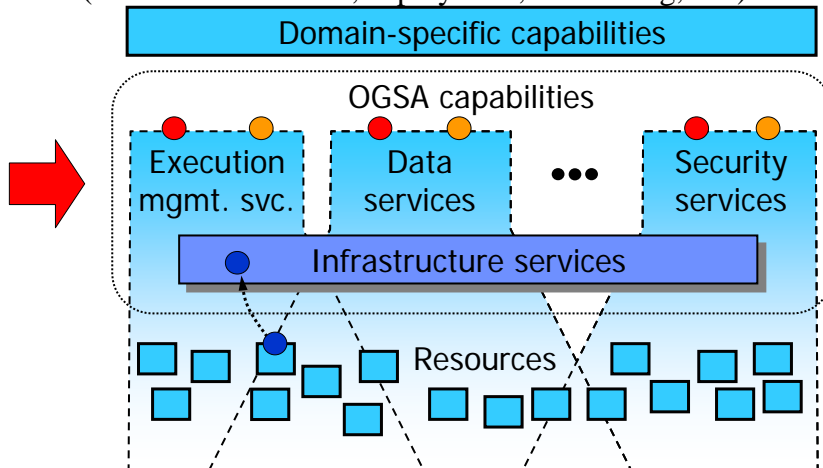
OGSA's Need: a Common Framework (安崎 篤郎)

- The OGSA-WG needs a common framework that meets the needs of Grid computing and distributed systems management:
 - Resource models (semantics)
(例えばCGS-WGのJSIMはバッチキューイングシステムのモデルであった。
Functional Interfacesに対するManageability Interfacesの為のモデルが必要である。)
 - Representation of and access to manageability (renderings)
- Many pieces already exist
 - Don't define "yet another" model. Use, compose and extend what exists.
 - Some (or most) of work done outside OGSA-WG
- But which pieces do we need? Do they fit together? How?
- How to move this work forward is still to be discussed at the OGSA-WG

12

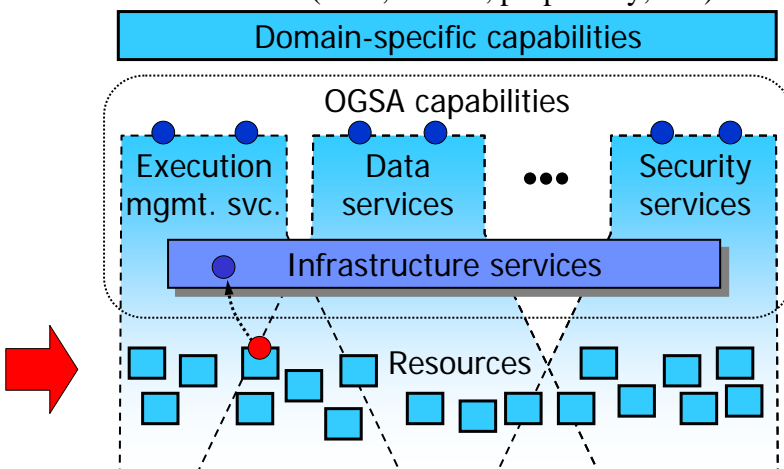
OGSA Management Framework (1) (安崎 篤郎)

- OGSA is composed of several capabilities
 - Provide high-level resource management functionality (resource reservation, deployment, monitoring, etc.)



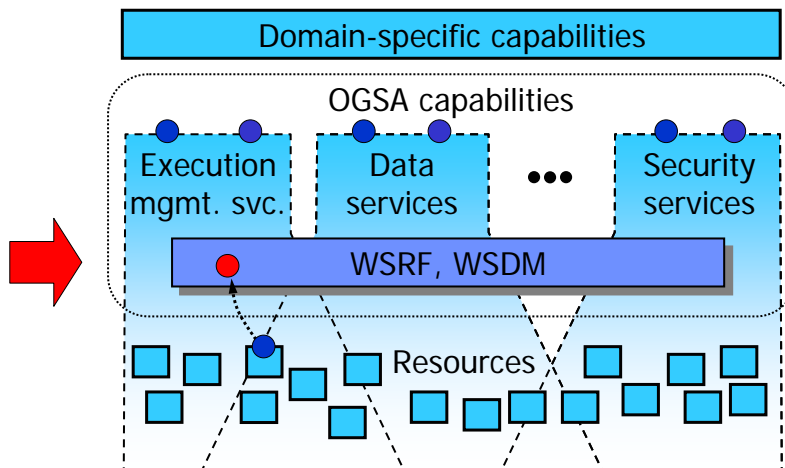
OGSA Management Framework (2) (安崎 篤郎)

- Capabilities use and manage resources to do work
 - Native manageability (reboot, set VLAN, etc.)
 - Native interfaces (CIM, SNMP, proprietary, etc.)



OGSA Management Framework (3) (安崎 篤郎)

- Infrastructure services show resources as services
 - Use Web services means for discovery, access, etc.
 - Provide common framework with basic functionality



What is WSDM? (What was CMM?) (安崎 篤郎)

- OASIS Web Services Distributed Management TC
 - Management *using* Web services (MUWS)
 - Management *of* Web Services (MOWS)
- MUWS provides:
 - a uniform, WSRF-based way to access native manageability of various types of resources
 - A set of functionalities that are of broad and general use for management, e.g.:
 - Metrics (collected values)
 - Resource states
 - Relationships among resources
 - さらにMUWS 1.0ではNotification, Discovery, Configuration, Collectionが入る予定。
- MOWS provides manageability for Web services

- Continue current work as OGSA-WG design team
 - Support the design of management and manageability in OGSA
- Liaison work with other standards bodies and GGF WGs

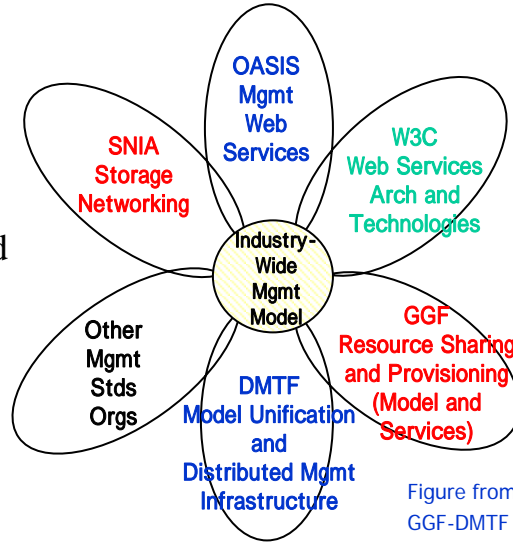


Figure from GGF-DMTF work register

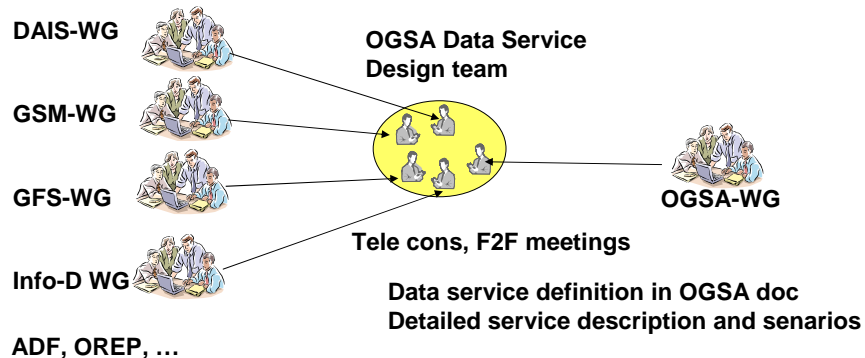
グループ	Open Grid Services Architecture Working Group (OGSA-WG)
目的	OGSAのユースケースから要件を抽出する。必要なOGSAサービスをリストアップし、優先度をつける。実際の仕様を策定するWGを特定もしくは創設する。関連するWG/RG間の活動の調整をする。さらに、関連したW3C, OASIS, WSIなどの標準化組織との関連整理と連携を行う。
状況	OGSAドキュメント(初版)と用語集の公開レビューが完了した。 ロードマップドキュメントのドラフトを公開した。 OGSAドキュメント(第2版)の内容および進め方を議論した。
進捗	OGSA-WGは7セッションを開催した。 関連WG/RGからの積極的な参加が増え、活発なアーキテクチャ議論を行った。 賛同者を増やししながら、相互接続性のある一貫したアーキテクチャを構築するために、慎重な舵取りが必要である。
今後	ロードマップドキュメントをGGF Editorに提出 (GGF13) OGSAドキュメント(第2版)および用語集をGGF Editorに提出 (GGF14)
参加者数	各セッション毎に50名から百数十名
所感	<ul style="list-style-type: none"> • OGSAドキュメントによって、まだ抽象度が高いとはいえ、OGSAの概要が明らかになったため、関連するWG/RGが具体的な、関連の仕方や問題点を見出すことが可能になった。 • その結果、良い意味での議論が複数のWG/RGとの間で始まった。

Goals for OGSA v2 document

- Refine statement of required capabilities
 - Based on work of other WGs & other standards bodies
 - Based on experience of early adopters
- Define specific interfaces (operations & behaviors) for capabilities identified in v1
 - Either in collaboration with existing WGs ...
 - Or as input to future WGs
- Series of documents
 - Delivery in one year (root docs)
 - Informational docs and recommendation docs

Design team

- Informal domain expert groups within OGSA
- May include co-chairs of other WG/RGs
- Output is included in OGSA specification



Defining a Roadmap: Challenges

- Coordination with other relevant activities
 - E.g. in the Web services community
- The danger of premature standardization
 - Without adequate experience &/or buy in from users
- OGSA design team remains fairly small
 - What perspectives have not been considered?
 - We can't do everything!
- Believing that we can define the entire future of distributed computing
 - We don't want a grander castle, we want some simple furniture

23

Goals for Roadmap

- Define priorities for OGSA interfaces, based on community input
- Identify dependencies among interfaces
- Document community requirements for timing of various interfaces
- Delivery of first draft at GGF12, final in GGF13 ('05/03)
 - First draft is in GridForge

https://forge.gridforum.org/docman2/ViewCategory.php?group_id=42&category_id=792

24

グループ	EGR-RG (Enterprise Grid Requirement RG)
目的	Enterprise グリッドに対してキーとなる技術要件と共通のアプローチを見出すこと。特にエンドユーザとの関わりが重要である。OGSA-WGやEGA、DMTFなどGGF外のグループとの連携も重要である。
状況	Charter が承認され、正式なRGとして最初の会合。 半日のWorkshop “Enterprise Grid Solutions and Deployments” を開催し、5件発表、2回のセッションを実施。
進捗	# 1 . Enterprise グリッドの定義についてMLに寄せられたコメントを元に議論。 最初のドキュメント“A survey of Enterprise Grid Deployments”を作成するにあたり、進め方について議論。三つのフェーズで実施することでほぼ合意が得られた。1) ユースケースを集める。2) 分類しカタログを作る。3) パターン分析をして共通要件を抽出する。 ユースケースとしては、1) OGSA-WGユースケースドキュメント、2) GGF12 Plenary 発表ケース、3) GGF12 Plenary で配布された451グループによるレポートなどを参考に参考にする。4) Workshop 発表ケース # 2 . Workshop で発表された内容についてのサマライズと要件の抽出を実施
今後	Enterprise グリッドの事例を集めるために毎回GGFでWorkshop を開いたらどうか。 2週毎のテレコンを実施する。曜日時間などはMLで議論する。 GGF13までにユースケースを沢山集める。
参加者数	約30名(1)、約20名(2)
所感	正式RGとして承認され、GGFのセッションでは活発に議論されているが、MLへの書き込みは依然少ない。早々にドキュメント作成に向けたプロセスを進める必要がある。

グループ	EGR-RG(ワークショップ)
目的	エンタープライズ分野におけるグリッドの動向について調査し、実験を通じ標準を策定する
状況	GGF11においてCharterの発表を行い、GGF12において今後の活動方針を決定。 ワークショップも開催し、UD、BT、Intel、ビジネスグリッド等の事例について紹介。
進捗	今回で第2回目の会合、Charter発表がメインだったGGF11と比べると意見交換も活発であり、今後の活動の具体的なアクションプランも会場から提示された。『エンタープライズ』という言葉の定義については会場においては明確な定義を保留し、各企業に対して質問状(アンケート)を送付して各企業の取り組みを分析した上で『エンタープライズ』の定義を詳細化していく。
今後	各企業に対して質問状を出し、ユースケースについて広く収拾する。
参加者数	約200名(ワークショップ)
所感	EGR-RGは日本国内のベンダー、及び研究者が多く関わっており、企業ニーズを基盤にした標準作りを目指している。実証実験においても貢献が可能であり、『技術標準作りで先行した他RGを応用技術によって追撃するRG』という図式が面白くなりつつある。しかしながら、RGが既に今後の方針を決めてしまったかのようなクローズドな雰囲気は無く、むしろ会場の議論によって更なる今後の方針が決定されている状況である。

- Introduction
 - “Aim of the Workshop” Toshiyuki Nakata (NEC Corp)
- Session 1
 - "Intra-Enterprise Grid", Jikku Venkat (United Devices)
 - "The evolution toward assured grid services", Nigel KJ Dye (British Telecom)
- Session 2:
 - "Grid Portals", Karsten Gaier (NICE srl)
 - "Business Grid Project - Objectives & Key Technical Issues - ", Nobutoshi Sagawa (Hitachi)
 - "Enterprise Grid Challenges - A Perspective", Ravi Subramaniam (Intel)
- Workshop summary, Ravi Subramaniam (Intel)

27

【BT社】

- 組織の仮想化において、VPNと同等の役割をグリッドが担うと予測
- グリッドとデータセンタを用いたユーティリティモデル(ASP)を目指している
- 通話分野からネットワーク分野へシフトしつつある
- Virtual MainframeというキーワードでポストASP時代の到来を予測している

【Intel社】

- 欧米とアジア研究拠点をグリッドポータルで仮想化し、演算リソースを共有化
- アプリケーションライセンスの取り扱いについてはグリッド対応は未だ難しい
- 他社プロジェクトより2週間先んじる事が出来れば大きなアドバンテージである
- セキュリティ、運用管理、実行環境整備、拠点間共有、リソース予約機能等

28

【United Device社】

- ・ グリッドミドルウェア開発業者
- ・ 製薬、金融等で実績が有り、PC端末数千台規模のPCグリッドが稼働中
- ・ 事務端末リプレースに対し、PCグリッドミドルウェア(GridMP)を導入
- ・ PC5台分でIAサーバ1台分の計算能力を取得
- ・ ドッキングスクリーニング実行時間が1件あたり1日かからなくなった
- ・ ノバルティスは\$0.4で\$2Mのコスト削減に成功
- ・ 世界規模の研究機関の仮想組織化に成功(欧米、つくば)

【Nice社】

- ・ イタリアのグリッドソリューションプロバイダ
- ・ <http://www.nice-italy.com>
- ・ 製造業向け解析業務支援グリッドポータルを提供
- ・ Nice Portal Fiat for LS-Dyna
- ・ DataGate for Ferrari

報告者: 小島 功(産総研)

グループ	Data Access and Integration Service(DAIS)-1,2,3
目的	グリッド上でのデータベースアクセスに関する仕様の策定
状況	ドキュメントの構成と内容がおおむねまとまり、提出に向けての問題点を一つ一つ潰している状態。ただ、いくつか重要な問題も残されている。
進捗	<p>WS-DAI(DAIS標準機能仕様)、WS-DAIR(関係データベース仕様)、WS-DAIX(XMLデータベース仕様)、Mapping Issues(WS-I,WS-RFなどの規格へのマッピング文書)の4つをまとめている。このうち、マッピングの文書がほぼ完成した。残りは、以下のように詳細な問題の検討に入っている。</p> <p>WS-DAI: 1) DBリソースのネーミングをOGSAの3レベルネーミング(Human / Abstract / Address)にあわせ、何かする必要があるか。 2) サービスから切り離されたデータセットを扱うかどうか 3) 最上位のデータアクセス手法の提供の有無(Get(), Put()など、RDBでいちいち提供する必要があるかどうか)</p> <p>WS-DAIR: 1) 仕様が自己完結かどうか。 2) Informational Propertyが更新されたときの挙動 3) SQLのRowset(結果)は更新可能かどうか、など。</p> <p>WS-DAIX: 1) XUPdate(最近活動がない)を支援すべきかどうか。 2) AddSchemaの意味づけ: データセットに束縛されるものか? など。</p>
今後	問題点の多くは決めの問題なので、おおむね次回までには文書がまとまるのではない。
参加者数	30-40人程度
所感	ようやくまとってきたという感じ。次回にはほぼ全部が提出可能になるのではないかと、今後は、今回省いた仕様(Agreementなど)や、また議論されている部分(アクセスでなく管理系)の扱いや、OGSA Design Teamの活動との整合性、連携などが問題になるが、セッション3での様々な検討など、グループの運営としておおむね良い方向性が維持されていると思う。

DAIS セッション2

DAIS機能のWS-I, WS-I with Context, WS-RF、それぞれへのマッピングを文書化している。基本的な結論としては、どれもそれなりに記述することができるとしている。一応、得失を書いている。(WS-Iだと標準的な仕様がなく、WS-RFだと現時点で確定していない、などをPro/Conとして評価)

- DAISとしてどれかを押すということはない。
- この3つを全部サポートし続けるかどうかは問題点。

所感: WS-RFに準拠するか、あるいはWS-Context/WS-CAFなどに準拠するかどうか？、あるいは併用するかどうか？などは、ここで簡単に紹介のあったWS-Enumeration/WS-Transferなどの提案なども含め、機能の比較というより、標準間対立の問題のように思われる。

DAIS セッション3 (不参加のため議事録から)

DAISのコアの機能はとっくの昔に安定しているが、周辺では他のアクティビティとの関係でまだ成熟していない。また、この仕様を広く広報し例えば実装可能なベンダなど多くの興味を引く必要がある。そのための検討点と、まとめるべき文書について。

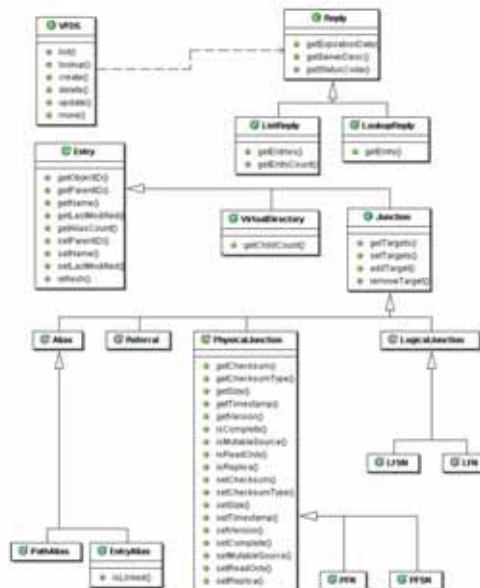
あわせて、商用システムでの動向とDAISとの比較について検討された模様。

- Microsoft SQL Server2005: すべてのスタアドプロシージャがWebサービス化可能。
- IBM(No Slide)
 - DB2: コンポーネントのすべてはWebサービス化可能
 - リソースに対するニックネームがつけられる (DAISにない機能)
- Oracle: Fully Integrated with iAS/J2EE.
 - Complex Web Serviceの支援: ebXML/RosettaNet, BPEL

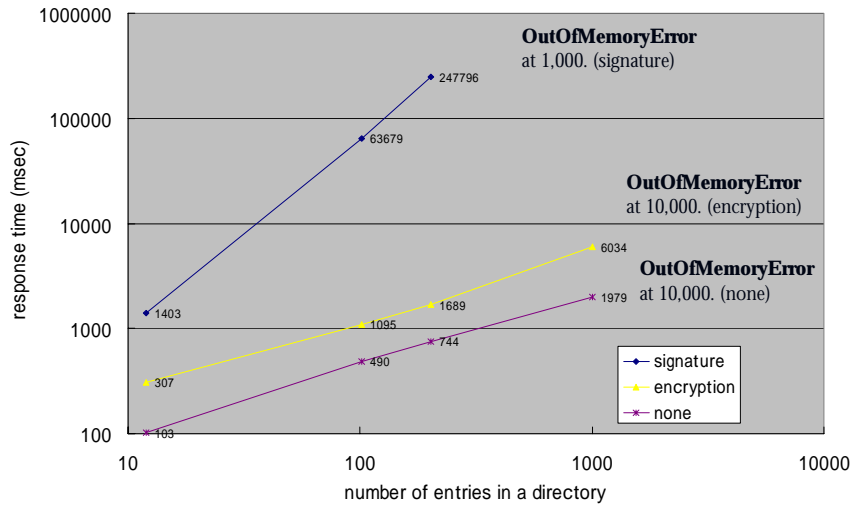
グループ	Data Format Description Language WG (DFDL-WG)
目的	バイナリやASCII/Unicodeのファイル、データストリームの構造を記述するために、XMLに基づく Data Format Description Language (DFDL) を定義する。
状況	GGF11からの進捗状況が報告された。データタイプなどの考え方がかなり明瞭化しており、例も多くある様子。 参考: http://forge.gridforum.org/projects/dfdl-wg/document/ggf-dfdl-rep-properties-proposal-basic-001.doc/en/1
進捗	DFDLの記述例などが紹介された。 Session2ではIBM社のゲストから、同社の製品である WebSphere Business Integration Brokers で使用されているDFDLの紹介も行われた。
今後	特に明確な指針は提示されなかった。
参加者数	10名程度 (#1) / 20名程度 (#2)
所感	• ADF-WG (Access To Data In Files WG) のBoFで、ADFとDFDL(他)との相関について議論が行われたらしいが、他のセッションと重なっていたため参加できなかったのが残念。

グループ	Grid File System WG (GFS-WG)
目的	グリッド上の仮想ファイルシステムの実現を目指し、1)仮想ファイルシステム・ディレクトリ・サービス(以下、VFDS と表記)の標準仕様、2)複数サービスの組合せによるグリッドファイルシステムのアーキテクチャの標準仕様を作成する。
状況	VFDS仕様のドラフト解説が行われた。また、2004年6月時点の仕様に基づいたGlobusToolkit3.2.1ベース(Java, GridService)の実装による検証結果が報告された。 アーキテクチャに関してはドラフトの解説と議論が行われた。
進捗	VFDSの検証結果の報告には下記のような懸案事項が含まれる。 <ul style="list-style-type: none"> ・ list 操作で10,000ファイルのデータをリブライするとOutOfMemoryErrorが発生する ・ 認証に GSI SIGNATURE を使用すると他の認証方式に比べて数十～数百倍の処理時間を要する ・ その他、現仕様の曖昧な点や、GridServiceで実装する上での問題点 アーキテクチャに関しては報告されたドラフトを基に、黒板を使って積極的な議論が行われたが、時間の都合により最終的な結論には至らなかった。
今後	仮想ファイルシステム・ディレクトリ・サービス仕様に関しては、報告された懸案事項等を踏まえて大幅に仕様変更される見込み。
参加者数	35名程度(#1 仮想ファイルシステム・ディレクトリ・サービス) / 15名程度(#2 アーキテクチャ)
所感	・VFDSの仕様作成者がGridServiceでの実装を行っていないこともあり多くの問題点が挙げられたが、実証結果がフィードバックされたことで、より現実的な仕様になることを期待したい。

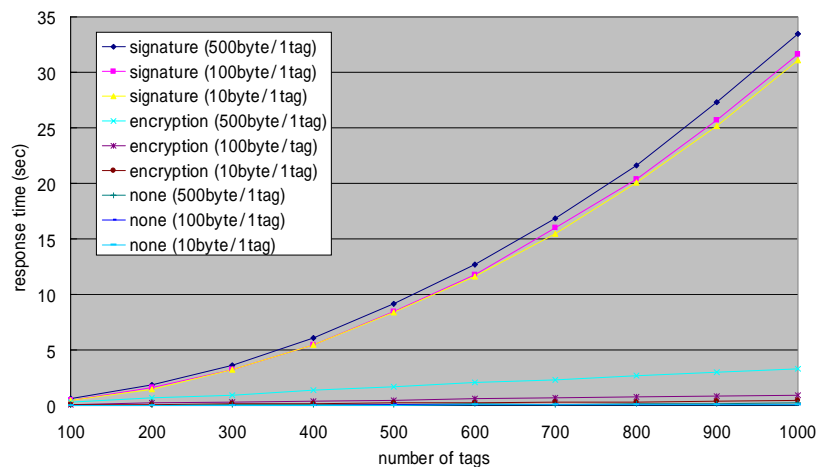
仮想ファイルシステム・
ディレクトリ・サービス
オブジェクトダイアグラム



Default heap size (64 MBytes)



Number of tags vs. Response time



One way to avoid degradation with GSI

- To improve performance, message size and number of tags should be small
- To reduce the message size and number of tags, change the WSDL interface in binary format using serialization of entry object
- Moreover, apply compression of the serialized data in gzip format

37

Content-length and response time when listing 100 entries

	in text format as usual	in binary format using serialization	in binary format with compression
GSI SIGNATURE	81 KBytes 60,000 msec	60 KBytes 260 msec	5 KBytes 140 msec
GSI ENCRYPTION	101 KBytes 530 msec	81 KBytes 280 msec	7 KBytes 130 msec
Non-authentication	80 KBytes 200 msec	60 KBytes 80 msec	4 KBytes 95 msec

- Note that the binary format is dependent on language (or serialization method)

38

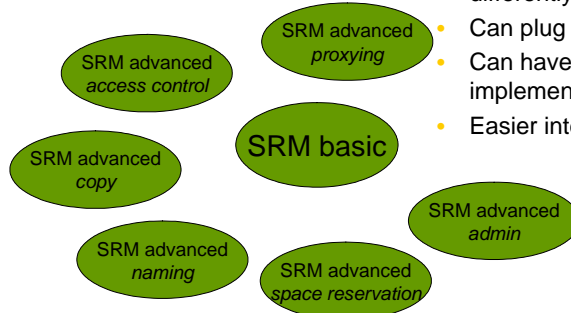
グループ	Grid Storage Management WG (GSM-WG)
目的	Grid上の共有ストレージの動的な空間割り当てとファイル管理を担うミドルウェア SRM (Storage Resource Managers) の、標準インタフェースの機能を定義する。
状況	9月初頭にパークレーで行われたワークショップの報告が行われた。 SRM インタフェース仕様の状況説明が行われた。
進捗	SRM 3 の仕様作成が始まっており、その特徴などについて解説が行われた。 ・SRM 3 では "core" と "advanced" の機能セットから成る。 ・機能の幾つかがグループ化され、Feature group となる。 ・Feature groups: Remote copy, Namespace management, Space management, Administrative status ・Future feature groups: Authorization, Accounting
今後	ドラフトを完了させてメーリングリスト上で回覧し、GGF13の前にもう一磨きかけてから、GGFDキュメントとして登録する。
参加者数	20名程度
所感	・「インタフェースに関する議論がGGFで行われないのは何故か」という質問が出ていたがもっともだと思ふ。回答は「SRMの議論はオープンなのでメーリングリストやワークショップへの参加を歓迎します」。

以下4枚のスライドは下記URLからの引用
<http://sdm.lbl.gov/gsm/docs/ggf12/GGF12-SRM.ppt>

Towards SRM 3

Peter Kunszt:

Suggestion to define not a single advanced interface but several 'feature sets'



- Basic is 'simpler' to implement
- Better version management
- Can evolve each interface differently
- Can plug modules easier
- Can have different things implemented
- Easier integration

Basic/Advanced Features

SRM feature	Basic	Advanced
Get/put multiple files	Yes	Yes
Lifetime per file (TURL)	Negotiated	Negotiated
Reserve space	Implicit	Explicit/Negotiated
Find out space assigned	No	Yes
SRM service status	Yes	Yes
User service status	No	Yes
Release file	Yes	Yes
Lifetime per space	No	Negotiated
Release space	Implicit	Yes

41

Basic/Advanced Features

SRM feature	Basic	Advanced
Multi-file streaming	Optional	Yes
Transfer protocol negotiation	Yes	Yes
Namespace (directory) support	No	Yes
File metadata discovery	Yes	Yes
File and space types	Volatile, permanent	All
Get/put from/to remote sites	No	No
Copy from/to remote sites	No	Yes, Copy&get, Put©

42

Feature Groups

1. Remote copy
2. Namespace management
3. Space Management
4. Administrative status

Future feature groups

1. Authorization – ACLs
2. Accounting

グループ	Data Access and Integration Services WG (DAIS-WG) & Grid Storage Management WG (GSM-WG) (DAIS - GSM joint session)
目的	DAIS と GSM の共通性および問題点を明らかにする
状況	
進捗	DAISから見たGSM、GSMから見たDAISを報告し、互いの特性、共通性、依存性、スコープなどについて議論された。 結果としては両者が行っていることに大きな衝突は見出されず、GSMが引き続きOGSAデータデザインチームに關与することが望まれた。
今後	
参加者数	約40名
所感	•Data Area にはWGが多数あるので、このような合同セッションで互いの位置づけを確かめることは非常に有益かつ必要なことだと思う。

グループ	information Dissemination(INFO-D)
目的	イベント通知に基づくルールベースの自動処理実行記述の仕様を定める。
状況	基本的なモデルはWG設立時に提案されており、それに基づくUse Caseと機能要件をまとめている状況。
進捗	<p>際立った進捗がない。動機付けされたUse Caseが8個紹介された。</p> <p>現時点では、Use Caseが集め切れていないため機能要件がまとめきれないのと、それをどう仕様化していくかについての方向性がまだ確定していないと思われる。</p> <p>実装面でも、当初はOGSIの上に高機能化として想定されていた部分の多くはWS-Notificationが高機能化されたために、機能的にかぶる部分(Broker的なモデルなど)が増えており、この部分でも課題が多い。</p>
今後	引き続きUse Caseを集め、機能要件をまとめて文書化・仕様化する。
参加者数	15人前後?
所感	第3者転送の記述など、基本的な機能としては有益であると思われるものの、位置づけが明確でなくなり、Use Caseを引き続き集める過程で機能的なターゲットがはっきりしなくなる可能性が高い。2回目の開催にも関わらず、参加者と進捗が増えてないのが気にかかる。

45

グループ	Transaction-RG(1のみ参加)
目的	グリッドにおけるトランザクションの機能についてのRG
状況	グリッドにおいて既存のトランザクション処理にない問題点や機能があるかどうかや、既存の道具立て(BTP,WS-Transactionなど)で十分かどうかも含めてリサーチする。
進捗	<p>(1のみ出席のため、詳細は不明)</p> <p>現在のUse Caseが紹介された。これらは、</p> <ol style="list-style-type: none"> 1. Trip Support 2. Checkpointing <p>など(ほか多数)である。</p>
今後	RGなので、Use Caseを集めていく過程でグリッド固有の問題や機能があるかどうか検討していくことになるのだと思う。
参加者数	20人+
所感	1のみ参加したので詳細は判断しかねるが、上で上げられたUse Caseのどれもが、いわば保守的なトランザクション処理の応用から抜け出していない印象が強い。参加者からも意見があったが、グリッドでのトランザクション処理の新しい部分が現時点では見えないので、さらなる検討やリサーチが必要ではないかと思われる。

46

グループ	CAOPs WG
目的	認証局の運用やPKI, 証明書などに関する標準化等について議論を行なう。
状況	<ul style="list-style-type: none"> -PDSおよびPMA CharterドキュメントはGFSGの承認待ち。 -NAREGI PKI WGがオープンソースCAソフトウェアについて発表。ソフトウェアを配布。 -Tonyがauthentication profiles documentについて議論する新たなWGの立ち上げについて、Federation BOFをGGF13で開く。 -OCSPドキュメントは80%終了。GGF13にて詳細に議論を進めることになった。
進捗	他のワーキンググループと同様に、このグループも提出したドキュメントが長い間GFSGでスタックしている。GGF13でFederation BOFを開くことなど、新たな展開が見られた。
今後	Auditingについては積極的にイニシアチブをとって貢献していきたい。また、FederationについてもAPGrid PMAとして貢献する義務がある。
参加者数	約30名
所感	APGrid PMAおよびNAREGIなど、日本、アジアからの貢献が次第に認知されるようになってきた。

グループ	OGSA AuthZ WG
目的	OGSAにおける認可(Authorization)に関する技術等について議論を行なう。
状況	新たな仕様の提案およびその検討を着々と進めている。
進捗	SAMLにObligationの拡張を入れる提案について議論をした。本WGにおいて議論を進めることについて同意を得た。
今後	今回提案のあったObligationの拡張について議論を進める。
参加者数	約30名
所感	他のOGSA関連のWG/RG、特にデザインチームとの連携の状況が気になる。

グループ	CIM based Grid Scheme WG (CGS-WG)
目的	CIMをグリッドシステムに適用するためのリファインと拡張、DMTFへの連携
状況	<ul style="list-style-type: none"> ・2004年度末にCIM v3のドラフトが出る予定であり、DMTFのワーキングドキュメントを参照しつつCIM v3をターゲットに作業を進めている。 ・MOF仕様の設計も並行して進める。
進捗	<ul style="list-style-type: none"> ・今回のGGFでは、CIMとDMTFの紹介に注力。CIM Tutorialと3セッションを費やして、CIM Overviewを実施。CIM自体の説明とDMTFの活動の紹介、およびオープンソースWBEM/CIMの概説とデモを行った。
今後	現在の活動を継続
参加者数	10-15人(セッション)
所感	<ul style="list-style-type: none"> ・未だCIMに対する理解が十分でないメンバが多いので、月曜日に開催されたCIM Tutorialは有益だったが、会場が2箇所に分散されていたこと、他のWGとぶつかったことで、参加者が数名程度と非常に少なかった。 ・今回はCGS-WGとしての単独の議題や作業がほとんど無かったため、DMTFからの積極的な協力が印象的だった。 ・GGF全体としてはグリッドのリソース記述にCIMを用いる方向であるが、反対意見も少なからずあり、CIM Overviewの際には、批判的な質問も多かった。 ・SRIMについては、Information Document v0.1が9/1に公開され、以下のステージ分けで進める方向 <ul style="list-style-type: none"> -ステージ1: スキーマ、テーブル、データタイプ -ステージ2: ルーチン、制約 -ステージ3: アクセスコントロール、式、文

49

- 11am-12:30pm, Introductions (Tom Roney) + DMTF Executive Overview (Troy Biegger) + CIM Introduction (Andrea Westerinen)
- 3:30-5pm, **CIM Overview** and Grid Service Example (**Andrea Westerinen**) + Application Management and Behavior and State (Karl Schopmeyer)
- 7:30-9pm, WBEM Architecture, Open Source and XML Renderings (Jim Davis and Karl Schopmeyer)
- チュートリアルの内容をCGS-WGの3セッションでも分割して実施。

50

Differing Aspects of a Model (安崎 篤郎)

- モデルには全く異なる側面が2つ存在する。Semantics と rendering である。
 - それぞれ、要件も制約も異なる。
- Semantics から Rendering へ
 - モデル (CIM) とはコンテンツとコンセプトである。 / 理想は単一モデルである。
 - 言語の構文、ルールがrendering である。 / 多様な rendering が可能(必要)である。 / (抽象 UML から個別 XML Schema まで)

51

Scope – The Environment AND the Element (安崎 篤郎)

- CIMのスコープは“big picture“であるが、一つの要素に限ったインプリメンテーションも可能である。
(実際には個別コンポーネントを選択して使うことになる。)
 - 必要ならば個別コンポーネントに飛び込んでも良い。
(Managementはデバイスやエージェントからの情報次第であり、デバイスやエージェントからの情報は個別コンポーネントの範囲である。)
 - 例: クリティカルデータのアクセスに20秒もかかる。 - これはサーバの問題か? ネットワーク? ストレージか? 全部か?
 - これに答える為には、各エレメントの詳細がみつようであり、さらにはエレメントと(ビジネス)目標のプライオリティの間でやり取りする情報が必要となる。
- コンフィギュレーションは多数のエレメントに跨る。この多数のエレメントでビジネス目標を達成のすのである。
 - すべてのエレメントが「大きな」(ビジネス)目標(の用語)を理解し、この目標を達成する為にどのように適応するかを理解できるようにしたい。
 - 理想は、すべての装置が同じコンフィギュレーションコマンドを理解してほしい。
 - 例: 「東京から大阪にフェールオーバーせよ」

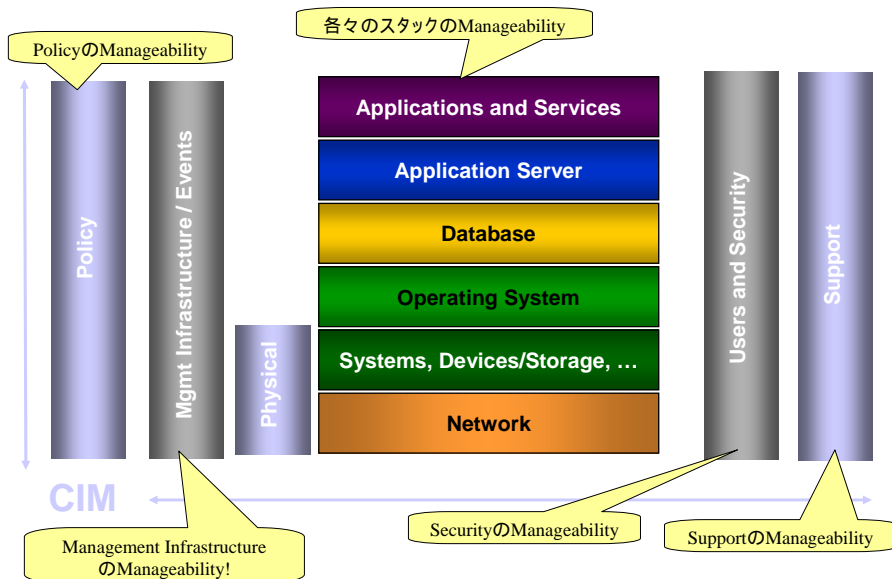
52

CIM's Coverage (1) (安崎 篤郎)

- コンフィギュレーションと汎用のManagementデータ (what is and what is desired)
 - 例: 根本原因の解析支援
- 関係
 - ...を使っている、...のコンポーネントである、...
 - 抽象化一般化しており、個別インプリメンテーションが可能である。
- 発展可能、拡張可能な(モデルの)デザインである。(標準 + プロプライエタリイ)
- データだけでなく、オペレーションもである。
 - パラメタ付きのドメイン固有オペレーション
 - 例: CreateOrModifyStoragePool
- すべての断片を一つの概念モデルに合わせている。

53

CIM's Coverage (2) (安崎 篤郎)



54

Modeling Goals (安崎 篤郎)

- 予測可能であること
 - モデルを理解すると、個別のデータの所在が維持でき(安定であり)
 - したがって予測可能となる。
- 「安定」なセマンティクスであり
 - したがって個別化、拡張が可能になる
- モデルを再利用するか、再定義するか、

55

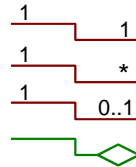
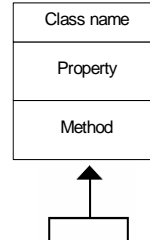
OO Concepts (安崎 篤郎)

- 抽象化
 - オブジェクトの概念境界を区別、定義する本質的な特性を確定すること。
- モジュール化
 - 概念を離散した単位に分解すること。
- カプセル化
 - 構造と動作を一体化すること。
 - 抽象化とインプリメンテーションを分離すること。
- 階層化
 - 抽象化の順位

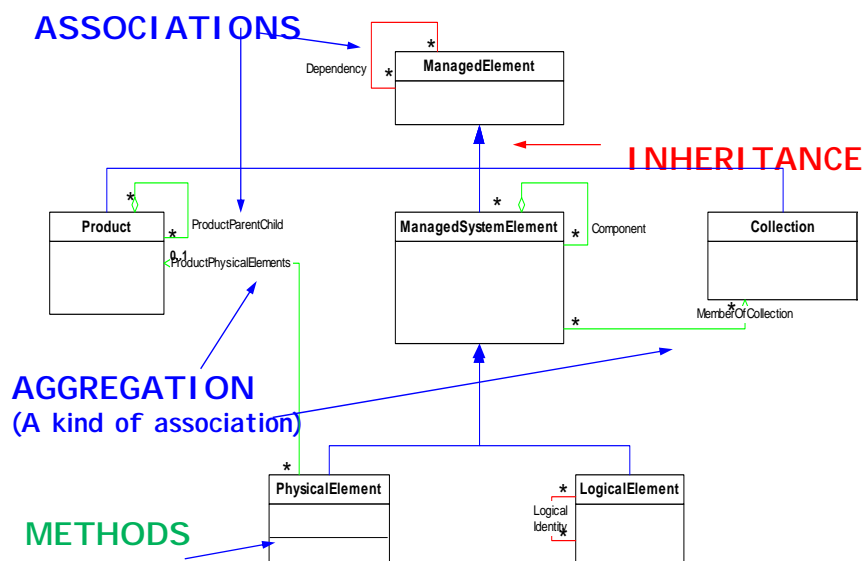
56

CIM's Elements (安崎 篤郎)

- クラス – 状態、動作、アイデンティティを持つ定義の集合
 - プロパティ
 - メソッド
- オブジェクト – クラスのインスタンス
- クラス階層 – スーパークラス・サブクラス
- アソシエーション – 関係
 - Dependency
 - Identity
 - Aggregation
 - Composition
 - And others



CIM's UML (安崎 篤郎)



CIM's MOF (Management Object Format) (安崎 篤郎)

CIM's MOF (An Abstract Rendering, Just One of the Possible

```

[Abstract, Description ( Renderings)
  "An abstraction or emulation of a hardware entity, that may "
  "or may not be Realized in physical hardware. ... ") ]
class CIM_LogicalDevice : CIM_LogicalElement
{
  . . .
  [Key, MaxLen (64), Description (
    "An address or other identifying information to uniquely "
    "name the LogicalDevice.") ]
string DeviceID;
  [Description (
    "Boolean indicating that the Device can be power "
    "managed. ...") ]
boolean PowerManagementSupported;
  [Description (
    "Requests that the LogicalDevice be enabled (\\"Enabled\\" "
    "input parameter = TRUE) or disabled (= FALSE). ...") ]
uint32 EnableDevice([IN] boolean Enabled);
. . .
};
  
```

Qualifiers

Class Name and Inheritance

Properties

Methods

CIM's Structure – Core and Common Models (安崎 篤郎)

- Infrastructure 規定
 - メタモデル、上位概念、言語の定義
- “Core” と “Common” モデル
 - Core モデルはすべての Management ドメインに適用される情報である。
 - Common モデルは個別のドメインを規定する。
 - Systems, Devices, Applications, Networks, Users, ...
 - Core モデルのサブクラス
 - モデル間でオーバーラップし相互に(アソシエーションで)参照される。
 - ベンダー拡張の奨励

Using the CIM Schema (安崎 篤郎)

- NEVER ... 「必要なクラスはなにか」
- ALWAYS ... 「Managementするものは何か？何をモデル化するのか？」
 - Who (Users と Security)
 - What (Physical Elements と Logical Elements)
 - Where (Location)
 - When (aspects of time)
 - How (Services と Service Access Points)
 - Why (ROI !)
- 1. Core または、いずれかの Common モデルにデザインポイントとなるものがあるか？
- 2. Examine the CIM の継承ツリーをみて概念の一致するものを探す。
- 3. 詳細をMOFなどで読む。
- 4. ユースケース、データフロー、CIMの中で見つけたものを元に検討を繰り返す。

報告者: 古城 隆(NEC)

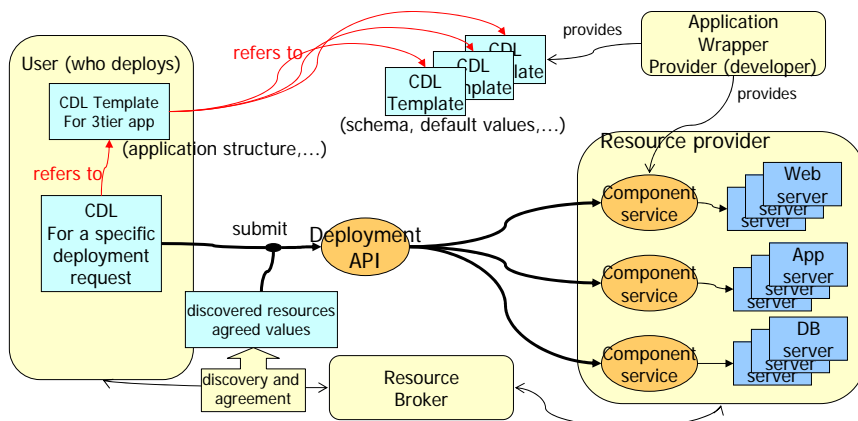
グループ	CDDL (Configuration Description, Deployment, and Lifecycle Management) WG
目的	サービスの設定方法の記述、グリッド環境におけるサービスの配備、サービスのライフサイクル(インストール化、初期化、開始、終了、再開始等)の管理に関する方法を規定する。そのために、CDDLに 関する言語、コンポーネントモデル、デプロイメントサービスAPIの仕様を定める。
状況	初日、終日のワークショップが行われ各社のプロトタイプの実装が発表になった。ただし、相互接続は されていない。IBMなどがW3Cに提出しているSolution Installation Schema, DMTF Application Lifecycle Modelなどについて当事者を招待し関係を議論した。XMLに基づく設定記述言語(CDL)、デ プロイメントサービスAPIの仕様、コンポーネントモデルの仕様に関する議論が行われ、次回GGF13ド ラフト完了を目指すことが再確認された。
進捗	ワークショップで発表された3社の実装は相互接続されていないが、それぞれ実装の側面から実現性 を実証している。HP社はWSによるサービスAPI、NECはXML-CDL、SofttricityはCDDLの利用局面 からの実装。 仕様に関しては今回はドラフトは更新されず、HPよりWSによるAPIの提案、NECによるXML-CDLの 修正案などのプレゼンと議論。
今後	記述言語(CDL)、デプロイメントサービスAPIの仕様、コンポーネントモデルのドラフト第一版をGGF13 までに発行、相互接続の実装を目標とする。
参加者数	15人(ワークショップ)、20人(セッション#1)、15人(セッション#2)、15人(セッション#3)、15人(セッション#4)、 10人(セッション#5)、8人(セッション#6)
所感	ジョブの運用において、必要不可欠な技術であり、他の技術要素(ジョブ管理、リソース割り当て等)と の違いが明確であるため、技術的な重要性が高いと考える。Solution Installation Schemaなど関係の 整理が必要。

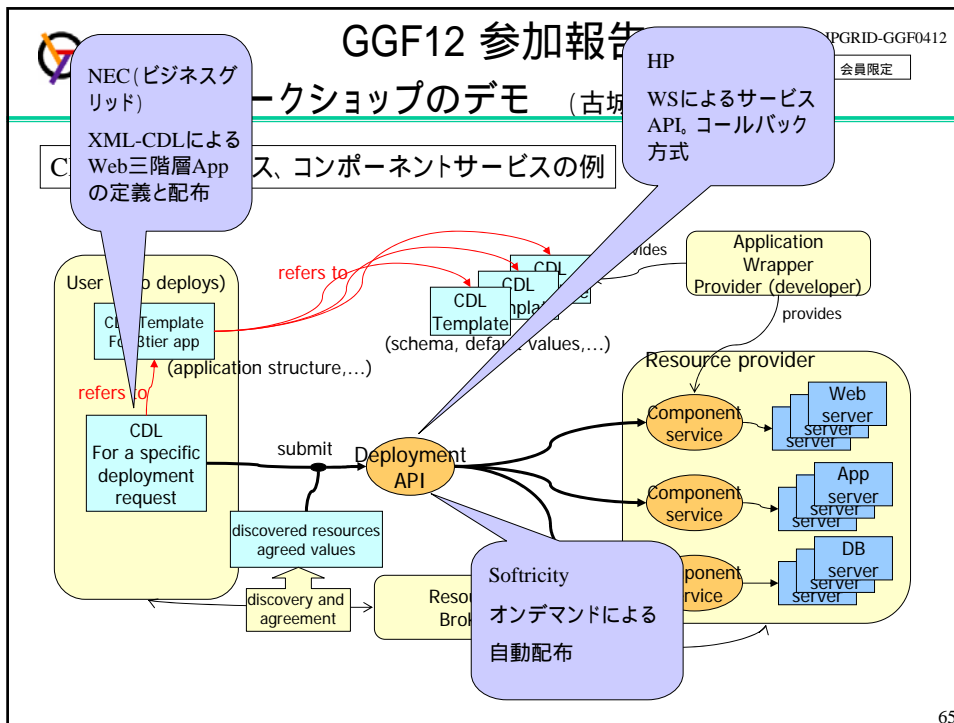
CDDLIM-WG (古城 隆)

- 配備、ライフサイクル管理を行うために、設定記述言語 (CDL)、基本サービス、コンポーネントモデルを導入している。
 - XML-CDL: SmartFrogや他の高レベルなCDLにおける相互接続性を確保するための中間言語。フロントエンドの言語として直接利用することも可能。
 - デプロイメントAPI: CDLを受け取り、解析 / 検証 / 参照解決を行った後に、サブコンポーネントに対してサービスのインスタンス化とライフサイクル管理の要求を行う。
 - コンポーネントモデル: 各々のコンポーネントの設定、ライフサイクル管理、メンテナンスを行うためのIFを提供。
- ユーザは、Application Wrapper Providerが提供するテンプレートを用いてCDLを記述し、基本サービスに投入。基本サービスは、コンポーネントサービスを用いてアプリケーションを配付。(次頁参照)

CDDLIM-WG (古城 隆)

CDL、基本サービス、コンポーネントサービスの例





GGF12 参加報告

報告者: 宮城 雅人 (NECソフト)

JPGRID-GGF0412
会員限定

グループ	Distributed Resource Management Application API (DRMAA) WG
目的	DRMS (Distributed Resource Management System) に対して、ジョブのサブミット、モニタリング、コントロールを行うAPI仕様を策定する。
状況	<ul style="list-style-type: none"> •6月にDRMAA Specification 1.0が承認され、GFD-R.22として公開された •DRMAA 1.0の各言語でのBinding仕様を策定中 (C, Java, .NET) •各言語Bindingの実装が進められている (C, Perl, Java, .NET)
進捗	<ul style="list-style-type: none"> •DRMAA (C, GridEngine)を使った画像レンダリングのデモ(dpovray)が行われた •オブジェクト指向言語のBindingについての議論 <ul style="list-style-type: none"> - DRMAA Spec 1.0はC言語を意識して書かれている - オブジェクト指向言語向けのドキュメントを作る方向になった •JSDLとDRMAAのjob templateとの関係 <ul style="list-style-type: none"> - DRMAAのjob templateにはあるが、JSDLにはない属性(e-mailなど)についての報告が行われた - DRMAAのjob templateがJSDLに置き換わるという方向は今のところないとのこと •SAGA-RGとの関係 <ul style="list-style-type: none"> - SAGA-RGの側で範囲をはっきり決める
今後	<ul style="list-style-type: none"> •各言語Bindingの仕様を完成させる •実装から得られたフィードバックを集める
参加者数	約20名
所感	DRMAA spec 1.0のオブジェクト指向言語Bindingとのギャップがはっきりしてきた。このDRMAAの経験を他のAPIに関するグループでも生かしてもらいたい。milestonesにあるWSDL-Bindingについてはあまり作業は進んでいないようである。

66

• 各言語Bindingの状況

言語	仕様 version	実装の状況
C	v0.95	GridEngine 6.0及びCondor 6.7で利用可能
Perl	-	Schedule-DRMAAc 0.81 : C実装のラッパーとして提供
Java	v0.4.2	GridEngine 6.0で利用可能
C#/.NET	v0.2	Mono上に実装しCondorで使えるようにする

67

グループ	GRAAP (Grid Resource Allocation Agreement Protocol) WG
目的	グリッド環境レベルの分散資源に対するスーパースケジューラサービスとローカールスケジューラとの間で、資源を予約し、割当てするためのプロトコルを規定すること。現在は特に資源予約やSLAを一般に指定するWS-Agreement規格の制定を目的としている
状況	前回のGGFから若干の更新の後、WG内で仕様が確定され、GGFのEditorに対し仕様が提出された。現在GGFからPublic Commentを求めるための仕様公開の準備が行われている。
進捗	WGではまずAlain Andrieuxにより提出された仕様のチュートリアルが行われた。それに並行して仕様に対するコメントが行われた。また、2番目のセッションではNECの中田から仕様に対するコメント・質問の発表が行われた。
今後	順調に行けば、仕様に対するPublic Commentを求めるフェーズに入り、その後の仕様確定までのプロセスに続く。また、今回提出された仕様はAgreementを作成する部分のみのため、Negotiationを行う部分について仕様策定が進められる。
参加者数	30人(セッション#1), 20人(セッション#2)
所感	<ul style="list-style-type: none"> 現在の仕様に関しては(細かい修正はありうるが)順調に仕様化されるものと期待している。そのためにはPublic Commentに対する反応と(「これで良い」というものでよい)実装が必要。 実装に関してはまだ存在していないが、AlainからGlobusプロジェクトで実装を行う予定があるとのコメントがあり、またNECの中田からBusiness Gridで仕様準拠のものを検討したいとのコメントがあった(他にも実装に興味を持っている参加者が2,3あった)。仕様中でWSDLが与えられているため、インタフェース自体の実装はそれほど困難ではないと考えられる。

68

Negotiation (Agreementの変更)は別仕様として分離 (WS-AgreementNegotiation)

現在策定中の仕様はAgreement作成部 (WS-Agreement)

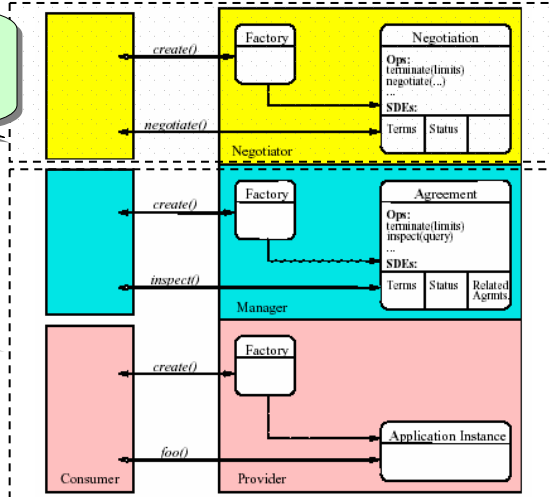
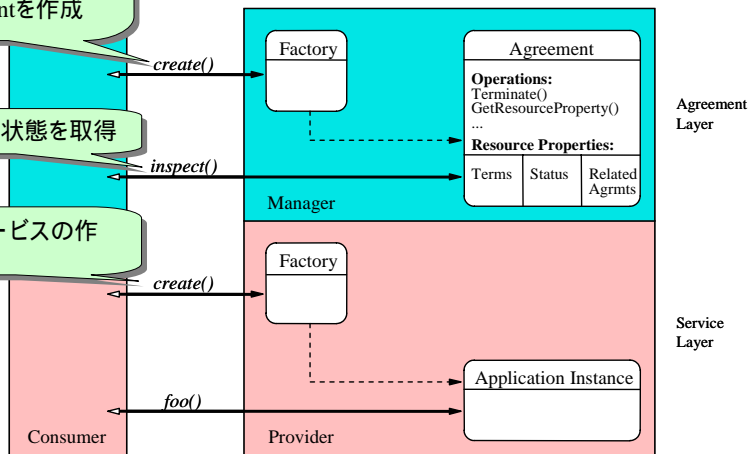


Figure 2: WS-Agreement Conceptual Layered Service Model.

- Templateの取得 (引数の制約等を記述)
- Agreement Documentを引数にAgreementを作成

- Agreementの状態を取得

- 対応するサービスの作成・利用



Agreement document

Agreement
Name
Context
Terms
Service Description Terms
Guarantee Terms

Agreement全体に関わるメタ情報

- ・ AgreementInitiator (Agreementの利用者)
- ・ AgreementProvider (Agreementの提供者)
- ・ ExpirationTime (Agreementの有効期間)
- ・ RelatedAgreements(関連するAgreement)
- ・ ...

提供するサービスに関する情報

- ・ 内容はサービスに依存(任意)
- ・ 例: ジョブ記述(プログラム名、ノード数など)

保障すべきサービス品質

- ・ QualifyingCondition(サービス品質保障の前提条件。例: 受け付けるリクエスト間隔や時間帯等)
- ・ ServiceLevelObjective(サービス保障するために満たすべき条件。例: 12MBのメモリを割当)
- ・ ...

- Templateは上記にAgreement Creation Constraintsが追加されたもの。この中にはAgreement作成時の制約が記述される(例: 予約できるノード数は8以下など)

71

- AgreementFactoryのPortType
 - Operation
 - createAgreement: Agreementの作成
 - Resource Properties
 - Template: Agreement作成時の制約(前述)
- AgreementのPortType
 - Operation
 - Terminate: Agreementの破棄
 - Resource Properties
 - Context: Agreement DocumentのContext部分
 - Terms: Agreement DocumentのTerms Compositor部分

72

- AgreementState PortType
 - Agreementの状態を取得するためのPortType
 - Resource Properties
 - GuaranteeTermStateList: Guarantee Termで指定されたQoSの状態
 - Fulfilled, violated, not determined
 - ServiceTermStateList: Serviceの状態
 - Not ready, ready, processing, completed

グループ	Job Submission Description Language Working Group (JSDL-WG)
目的	属性の一覧、属性間の関係、属性値の範囲からなるJSDLの抽象仕様を定義する。また、属性を記述する、標準XMLスキーマ定義、および、属性から、複数の既存のバッチシステムへの変換表を作成する。
状況	<ul style="list-style-type: none"> • 提案仕様(0.5.0)がGGF12でレビュー用に公開された。 • XMLスキーマと記述例の素案がGGF12に先行して、配布開始された。
進捗	<ul style="list-style-type: none"> • 5つのWGセッションが開催された。うち2つで仕様のレビューを行い、残り3つでスキーマと記述例(5つのプロジェクト)と実装デモ(別の2つのプロジェクト)を行った。プロジェクト名は次ページ参照。 • 全体として属性仕様は何点かの未解決の課題を残してほぼ安定した。個々の機能についてより理解しやすくするために説明的な記述を追加することにした。 • スキーマと記述例のレビューは時間をかけて行った。多くの問題が指摘されたが、特に階層的構造は制約が強すぎるとの結論に至った。
今後	作業はかなり前進したが、属性仕様とスキーマをGGFエディタに提出するまでには予定以上の作業が必要であるという認識に至り、このため定例の電話会議に加えて臨時会議(11月頃)を行い、GGF13でパブリックコメント開始に間に合うように仕様を完成させる予定である。
参加者数	各セッション毎に15名から25名
所感	<ul style="list-style-type: none"> • 7つのデモや記述例が提示されたことから、関連分野でJSDLに対する十分な認知があることが明らかになった。その中には彼ら独自の記述言語を放棄してJSDLを採用するプロジェクトも存在する模様。 • スケジューリング属性やワークフロー、複雑なジョブの記述をサポートするためのJSDL拡張に対しては大きな関心が寄せられている。

- JSDL demos
 - GridSAM (LeSC)
 - UniGrids (JSDL to Unicore AJO translation) (FLE)
- JSDL examples
 - Business Grid Project
 - NAREGI
 - Market for Computational Services (U. of Manchester)
 - gridMatrix (Cadence)
 - JSDL to DRMAA mapping (DRMAA/SUN)

グループ	Grid Scheduling Architecture (GSA) RG #1, #2
目的	任意のグリッドのリソースをコーディネートするスケジューリングアーキテクチャには、どのようなサービス、プロトコル、インターフェイスが必要かを洗い出す。
状況	<ul style="list-style-type: none"> •現在のコースケースドキュメントには以下のコースケースがある <ul style="list-style-type: none"> – Scheduling complex workflows in heterogeneous environment – Application-Oriented Scheduling in the KNOWLEDGE GRID – GRASP (Grid Resource Allocation Services Package) •コースケースからrequirementsを洗い出す作業が進められている
進捗	<ul style="list-style-type: none"> #1 <ul style="list-style-type: none"> •OGSA-EMSと時間が重複したためキャンセルされ、OGSA-WGのセッションへの参加が促された •OGSA-EMSデザインチームとGSA-RGのメンバーが今後Scheduling Architectureについて議論していくことに決まった #2 <ul style="list-style-type: none"> •コースケースの1つであるScheduling complex workflows in heterogeneous environmentのプレゼンテーションが行われた •required Services/Componentsの洗い出しおよび議論が行われた
今後	<ul style="list-style-type: none"> •GGF13でコースケースドキュメントのfinal draftを出す •GGF13でアーキテクチャ(requirements)ドキュメントのfirst draftを出す
参加者数	#1: キャンセル, #2: 約50名
所感	このRGのアウトプットドキュメントは他グループの仕様策定のインプットとなるよう意図されているが、例えば現在のOGSA-EMSのサービス群との間には多少の開きがある。今後活発な議論がなされることが期待される。

• Required Services/Components

- Information Service
- Job/Workflow Description
- Requirement Description
- Resource Discovery
- Reservation
- Monitoring/Notification
- Job Execution
- Migration/Checkpointing 今回追加
- Security
- Accounting/Billing
- Data Management
- Local RMS
- (...?)

グループ	Workflow Management (WFM) RG #1, #2
目的	Grid環境におけるworkflowの記述、実行について調査し、ベストプラクティスの共有を行う。また、他の団体でのworkflowに関わる活動や既存技術、ツールなどをGridコミュニティに紹介する。基本的にscientific workflowにフォーカスする。
状況	Comparing Business and Scientific Workflowsと題して両者からの発表および議論が交わされた
進捗	<ul style="list-style-type: none"> •ScienceとBusinessのworkflowの比較としてそれぞれからの発表および議論が行われた <ul style="list-style-type: none"> - Science: Building ad hoc (personal) workflows in an open world (C. Goble, University. of Manchester) - Business: A look at Industry Workflows (Darren Pulsipher, Cadence) •その他1本の発表が行われた <ul style="list-style-type: none"> - The MiGenAS pipeline: A bioinformatics Workflow Engine (Thomas Soddemann, Rechenzentrum Garching) •GGF14で開催予定のworkflowに関するワークショップのトピックについて議論された <ul style="list-style-type: none"> - GGF13ではない理由は、co-chairsがGGF13に行けないということと、EUのいくつかの最近始まったプロジェクトの成果がGGF14あたりに出るだろうということによる
今後	<ul style="list-style-type: none"> •GGF13ではセッションは行わない予定 •GGF14でworkflowに関するワークショップを開催する •Workflow記述言語とツールに関するドキュメントを作成する
参加者数	#1: 約60名、#2: 約40名
所感	今回の#1はBusinessとScienceのworkflowの比較ということであったが、Business側として発表された内容が企業内のEngineeringにおけるworkflowの話であり、結果的にはclosed (business) v.s. open (science)の比較になっていたようだ。

- Building ad hoc (personal) workflows in an open world: myGrid experiences (C. Goble)
 - myGridの中でのworkflow
 - アプリケーション: BLAST, InterProScanなど
 - Web画面でカット&ペーストを繰り返していた
 - データベースの更新 workflowの再実行
 - workflow記述言語: Scufi (Simple conceptual unified flow language)
 - Taverna : workflowの構成など
 - Freefluo : workflow enactment engine
 - 公開されているOpenなWebサービスを利用
 - ほとんどプロビジョニングはしていない
 - データの出所、関連の管理が重要
 - workflowの入力は小さいが、大量の出力がある
 - 中間結果を見たい

- A look at Industry Workflows (D. Pulsipher)
 - Business workflowの要件
 - 繰り返し可能、信頼性、観測性、ヘテロなりソースへの対応、分岐、部分実行、ロギング
 - ソフトウェア開発におけるworkflow
 - Configuration Management, Building Software, Testing Software
 - EDA (Electric Design Automation) におけるworkflow
 - Simulation, Validation, Physical Verification, File Conversion
 - ヘテロ (様々なツールのタイプ、ベンダ、プラットフォーム)
 - 300-300ステップ、50-100のツール、50%程度のflowは自動化できる

- 相違点

	Science	Business
サービス/ツールの公開性	Open	Closed
ビジネスの所有者は workflow の作成者か?	No	Yes

あくまで今回の発表の場合

- 共通点

- シンプルな workflow 記述言語が必要

81

- The MiGenAS pipeline: A bioinformatics Workflow Engine (T. Soddemann)
 - The MiGenAS pipeline
 - [Sequence] Blast ClustalW ProtDist Neighbor [Tree View]
 - BPEL4WSを採用
 - 実装が既にいくつか存在する (ActiveBPELを採用)
 - J2EEやCORBA環境との親和性が高い
 - いくつかの実装はWebサービスでないendpointに対しても使用できる
- 今回の全体の結論
 - シンプルな (abstraction levelの高い) workflow 記述言語やGUIでの workflow 構成ツールが必要

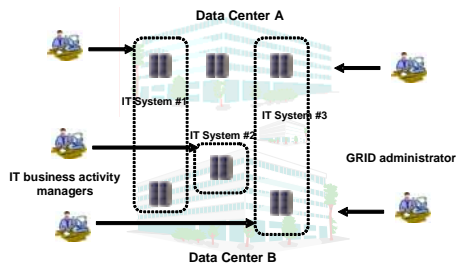
発表に使用されたスライドは <http://www.isi.edu/~deelman/wfm-rg/> から入手可能

82

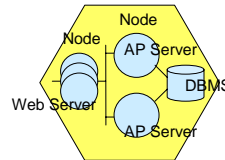
グループ	Application Contents Service (Charter Discussion BOF)
目的	グリッドアプリケーションをシステムに登録する際のインタフェースとアーカイブ形式の標準化をする事により、将来における、他システムとの相互運用性を確保する。
状況	WG設立準備中。貢献メンバを募集中。OGSA-WG face-to-face会議(8/20@San Jose)での紹介・協力依頼を実施した。2004年中のWG立ち上げ、GGF13(2005年3月)でのDraft-1作成を目指す。GGF12においてACS-WG 設立準備会(BOF)を開催し、福井(F)が概要説明と司会を行った。計画している活動概要について参加者の理解は得られた。
進捗	賛同者(米国のベンダ)2名を得て、さらに賛同候補を募集中。GGF12において、GGF CDDL-M-WG co-chairやSolution InstallationのTom Studwell氏(IBM)らと今後の標準化における協調可能性について意見交換した。
今後	メーリングリストでCharterや活動計画に関する議論を行い、早急にWG設立を目指す。
参加者数	15名
所感	<ul style="list-style-type: none"> •事前の広報や普及活動が不足だったこと、関係の深いCDDL-M-WGセッションと開催時刻が競合したことにより、期待よりも少ない参加者数であった。 •7月にW3Cに提出されたSolution Installation技術仕様について、関係者であるTom Studwell氏と直接会話して、今後の協調可能性について議論できたことは有益だった。 •今後は、関連する製品を開発するISVの参加者を巻き込んで行きたい。

Motivation/Background

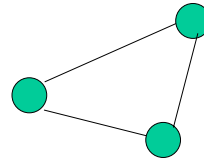
- In order to install and operate complex systems efficiently and automatically it is necessary to specify and manage as a unit a lot of different application related information.



Examples



- Three-tier systems
- Online job with a background batch operation
- Jobs with an autonomic scalability control.
- A jobs consists of geographically distributed jobs.



85

Business Grid Project

—“**Business Grid Middleware Goals and Status**”,
January 20, 2004, Hiro Kishimoto, Fujitsu Takashi Kojo,
NEC Fred Maciel, Hitachi

http://www.globusworld.org/program/slides/3c_1.pdf

- Japanese Business Grid Consortium is developing Grid middleware based on GGF OGSA, which is leveraged by many GGF WG specifications, inspiring the Application Contents Service.

86

Application Contents (1)

- Include application binaries and related information; e.g.
 - program binaries,
 - configuration data,
 - procedure descriptions for lifecycle management ,
 - requirements descriptions for the hardware and underlying middleware,
 - policy rules,
 - and anything needed to create a job instance on grid systems.
- They may be real entities or location pointers

Application Contents (2)

- Application Contents don't include:
 - Information updated by a job instance.
 - Information describes a status of a job instance.
 - Operating system and/or middleware binaries, since they are considered as a part of the resource provided by infrastructure.
- In any way, ACS doesn't interpret or execute information in the Contents, rather just manage them for use by other OGSA-services.

What is ACS for?

- Advantageous for:
 - Continuously maintained, deployable set of applications.
 - Collection of the tailor made or customized application (and data set), deployable as a unit.
 - As a basis of the automation in deployment.

- Application Contents Service:
 - Application Contents Repository Interface (ARI),
 - Application Contents Archive Format (AAF)
 - to be registered to, or exchanged between Application Contents Services.

