グリッドプログラミング環境（１）

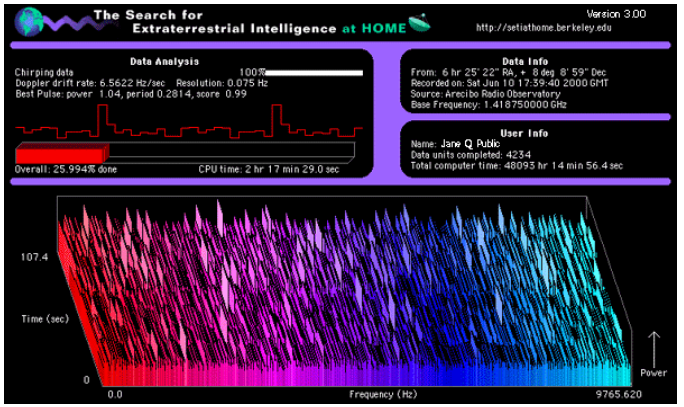
建部修見
筑波大学システム情報工学研究科
コンピュータサイエンス専攻

概要

- **グリッド技術とは？**
  - 計算グリッド
  - データグリッド
  - アクセスグリッド
- **グリッド技術とその要素技術**
  - 単一認証技術
  - 高速広域データ転送技術
  - 情報サービス
  - 資源管理
- **オープングリッドフォーラム (OGF)**
- **グリッド基本ソフトウェア**
  - Globus
- **グリッドプログラミング**
  - コンポーネントモデル
  - MPI
  - GridRPC

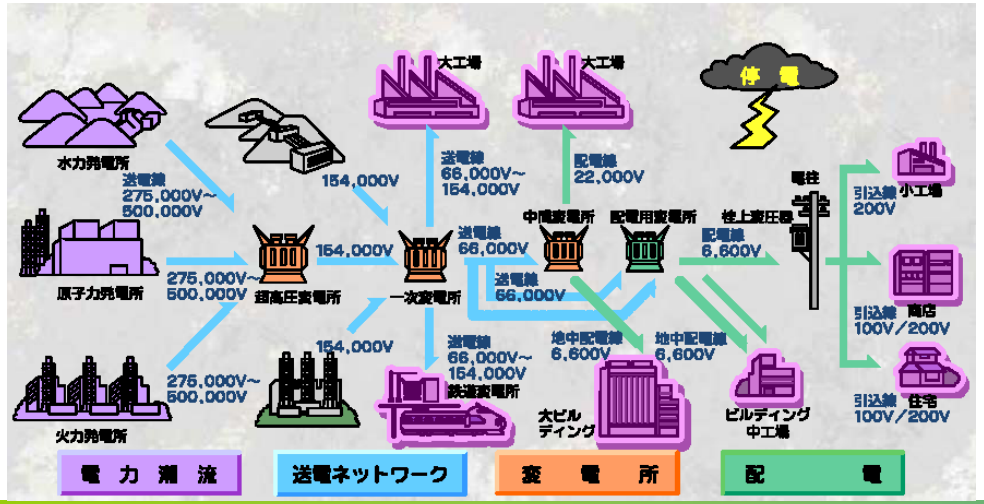
グリッド技術とは？

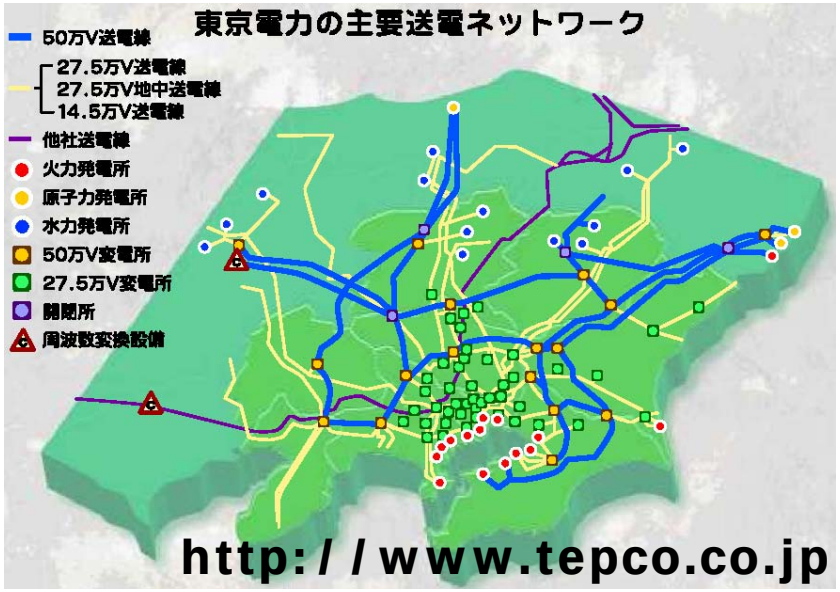
- スーパコンピュータを高速ネットワークで接続して共有すること？
  - http://www.itbl.jp/
- SETI@Home、UD Cancer research project、Fight AIDS@homeなどのこと？
  - http://setiathome.ssl.berkeley.edu/
  - http://members.ud.com/projects/cancer/
  - http://www.fightaidsathome.org/
- 次世代インターネット技術？
  - IPv6, QoS, IPsec, . . .



グリッド

- 90年半ばに作られた言葉
- 電力の送電線網（Electric Power Grids）との類似性
- 送電線網は過不足のない電力供給、トラブル時における別ルートの確保など重要な役割。監視され、制御され、運用される。
- 発電装置、電気製品だけではなく送電線網は重要な発明！
- 現在の計算基盤は20世紀初めの送電線網のなかった電力の状況と似ている





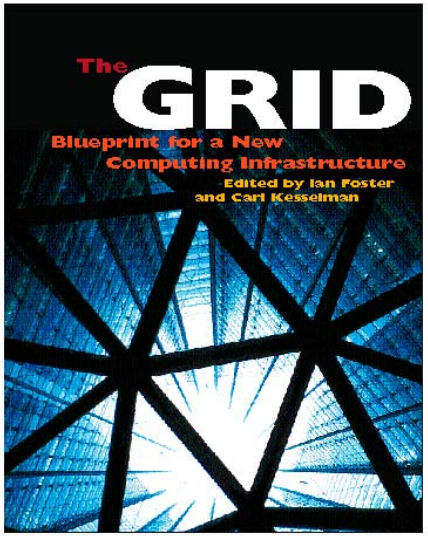
グリッドの定義（1999年）

- **計算グリッドとは，高性能計算能力へのアクセスのためのソフトウェア，ハードウェアである**
  - 信頼できる，一貫した，広範囲の，安価な

A computational grid is a hardware and software infrastructure that provides dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities

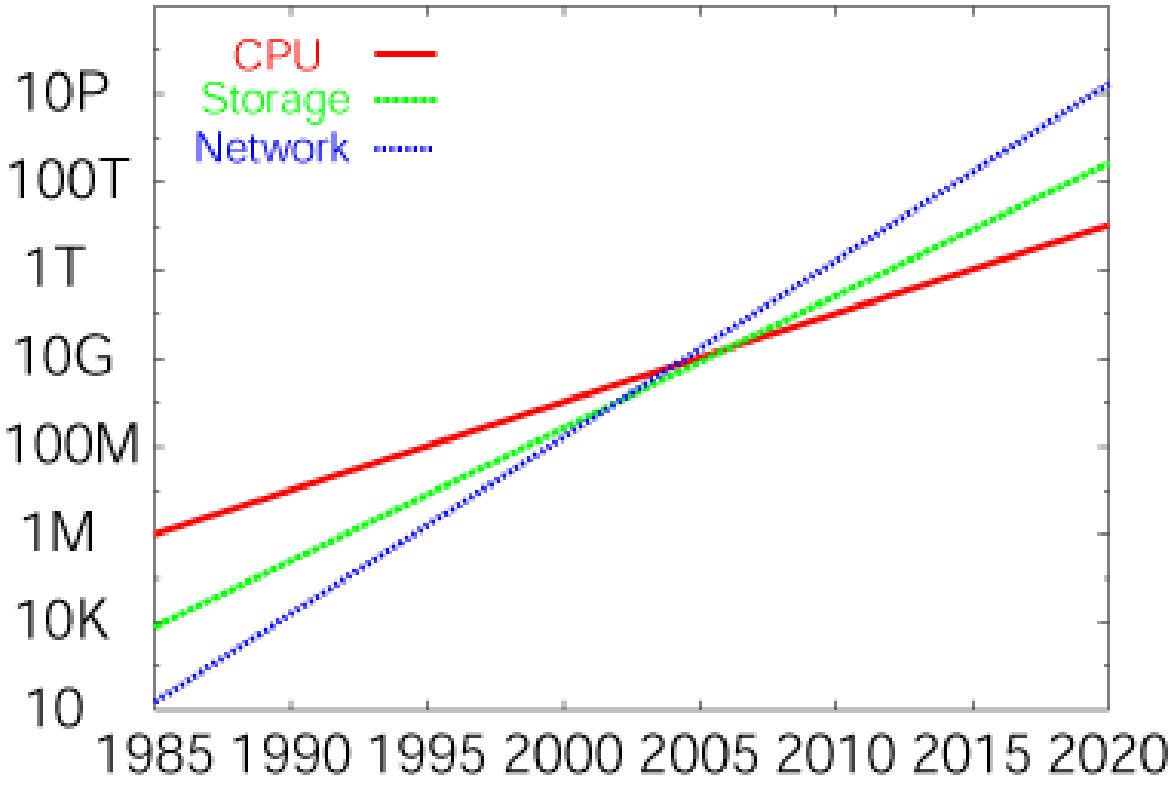
From “The GRID – Blueprint for a New Computing Infrastructure”, 1999

http://www.mkp.com/grids/



今がそのとき：技術トレンド

- CPU speed doubles every 18 months (Moore’s law)
- Storage capacity doubles every 12 months
- Network speed double every 9 months



技術トレンドのスナップショット

- **2005**
  - CPU 10 Gflops
  - Network 16 Gbps
  - Storage 1000 Gbytes
- **2010**
  - CPU 100 Gflops
  - Network 1600 Gbps
  - Storage 32000 Gbytes

CPU << Storage << Network

ネットワークがふんだんに、ただになる！

- 5年で100倍
- 手元の資源にしばられない
- 広域に存在する計算資源、ディスク資源、可視化装置、ベクトル計算機、専用計算機、（大規模、広域分散）実験測定機器、研究者、アプリケーション、ライブラリ、データなどなど利用することが可能となる！

The Grid Problem (2000)

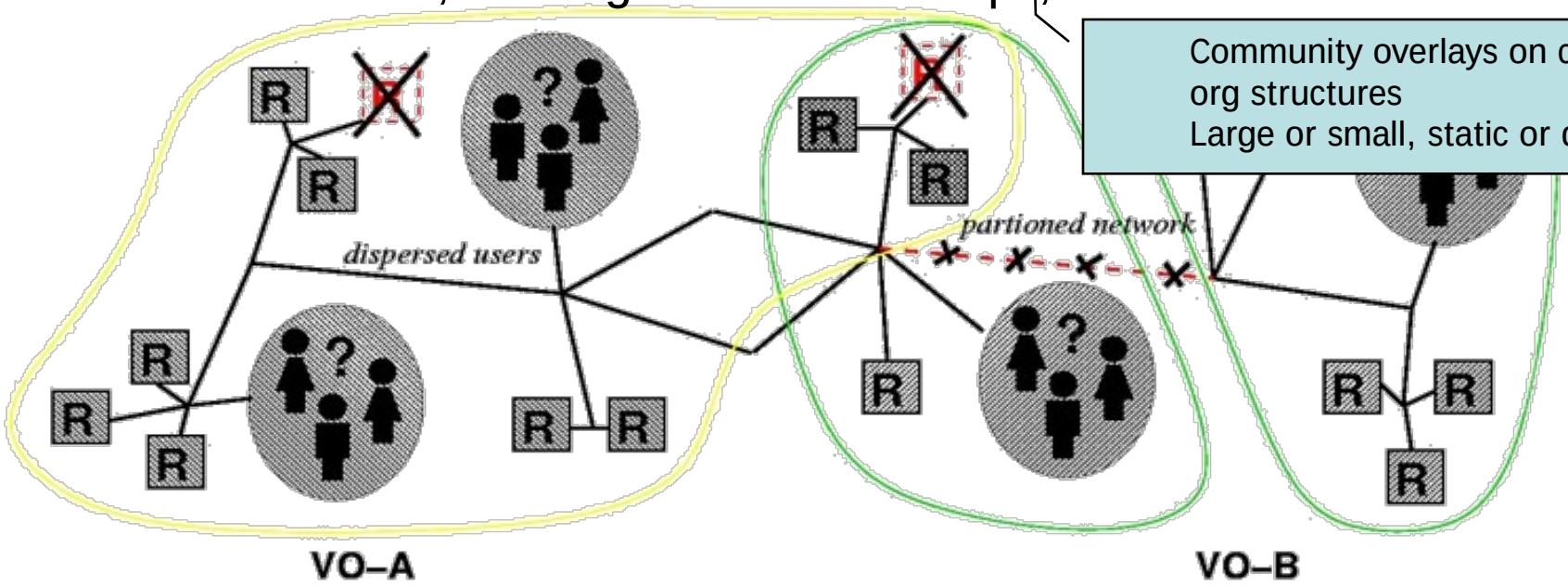
Computers, storage, sensors, networks, ...
Sharing always conditional: issues of trust, policy, negotiation, payment, ...

Beyond client-server: distributed data analysis, computation, collaboration, ...

Resource sharing & coordinated problem solving in dynamic, multi-institutional virtual organizations

- Communities committed to common goals
  - Assemble team with heterogeneous members & capabilities
  - Distribute across geography and organization
  - Assuming the absence of central location, central control, omniscience, existing trust relationships, . . .

Community overlays on classic org structures
Large or small, static or dynamic



Slide courtesy by Ian, Carl, and S. Sekiguchi

仮想組織とグリッド技術

- **動的で柔軟な資源の集合**
  - 独立に管理されている複数の組織を含む
  - 一つの組織が複数の仮想組織に所属可能
- **大規模、小規模**
- **安全に制御された資源共有**
  - 計算機、ディスク、センサ、実験装置、アプリケーション、データなど
- **資源共有は条件付なことも**
  - 空いている時間だけ、午前中だけ、一部分の計算機だけ、実行するプログラムの制限などなど
- **共有の形式は、クライアント・サーバだけではなくP2Pなども**

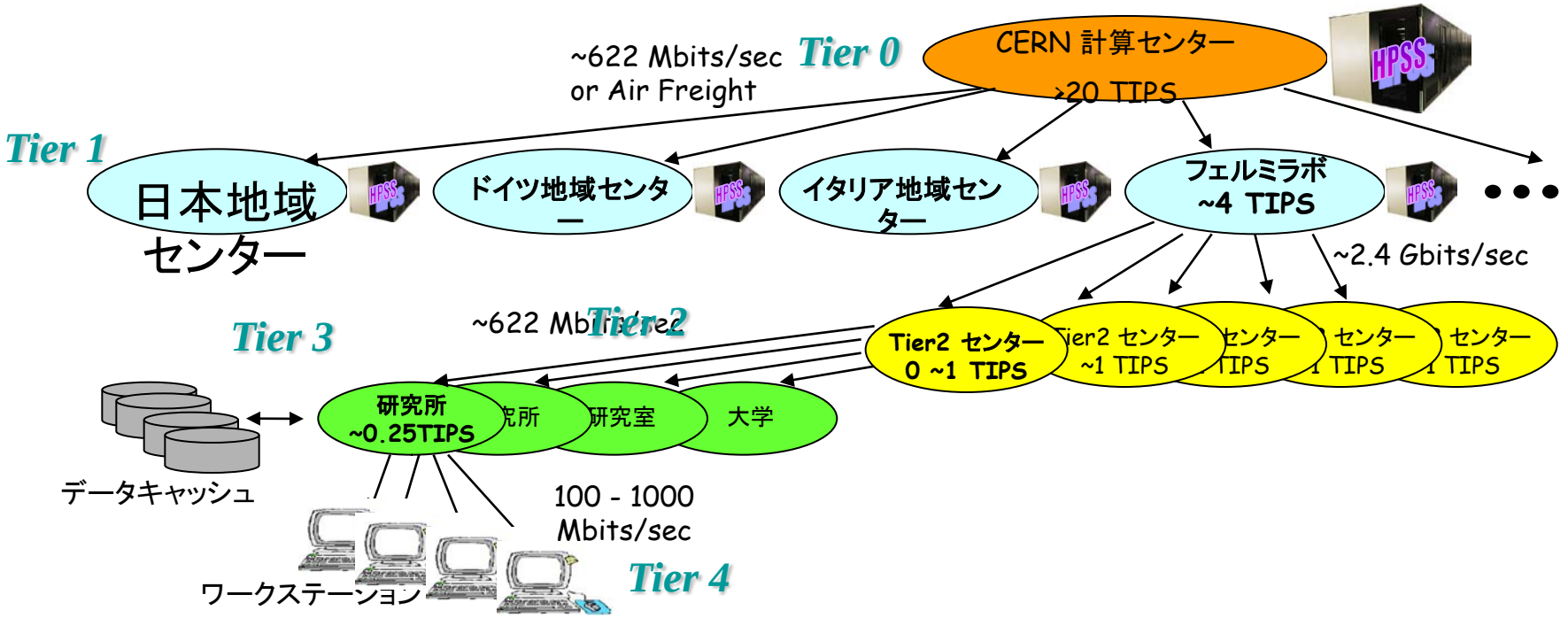
- **柔軟に仮想組織を形成し、安全に資源を共有するための技術**
  - 安全な**認証**と適切な**権限**の付加
  - 資源の**アクセス**方式、**発見**方法
  - **耐故障性**
  - **相互運用性**、**共通**のプロトコル

想定されるシナリオの例（1）

- 会社A、Bによる小さな仮想組織
- 会社Aのスーパコンピュータを利用してシミュレーション
- 結果を会社Bの可視化装置で可視化
- それぞれの社員が安全に共有して解析する
- 部屋の密閉性が高く、換気システムの導入を検討
- 部屋が入り組んでいるため、どこに取り付けると効率的か明らかではない
- ASPを利用して、流体シミュレーション。結果をSSPに格納し、住宅会社に転送。

想定されるシナリオの例(2)

- 欧州CERNによるLHC加速器実験。20カ国3000人規模の研究者と階層的な地域計算センターからなる仮想組織。年間数ペタバイトの実験データ解析およびシミュレーションによる検証を行う。



Grid Architecture and standard

グリッド技術に対する要求

- 資源の所有者と利用者に関する多様なセキュリティとポリシに対応
- 多種多様の資源、共有の形式に対し十分な柔軟性をもつ
- 多くの資源、多くの利用者、多くのプログラムに対しスケールする
- 動的な資源の管理
  - 資源の動的拡張性
  - 耐故障性、自己組織化
    - 資源の動的な変化は日常茶飯事
- 大容量データ処理、大規模計算を効率的に実行する
  - HPC、HTC
  - 高バンド幅+高遅延対応
- 異るグループが柔軟に資源共有するための相互運用性
  - 多種多様の資源、ポリシ、方式に対応
  - 例えば、個人認証の方式、通信方式、資源の記述方式
- 開発の重複を防ぐための共通基盤サービス
  - 例えば、遠隔プログラム起動のための、ツールやアプリケーションに依らない共通のサービス、プロトコル
  - 例えば、運用コストのかかる認証局

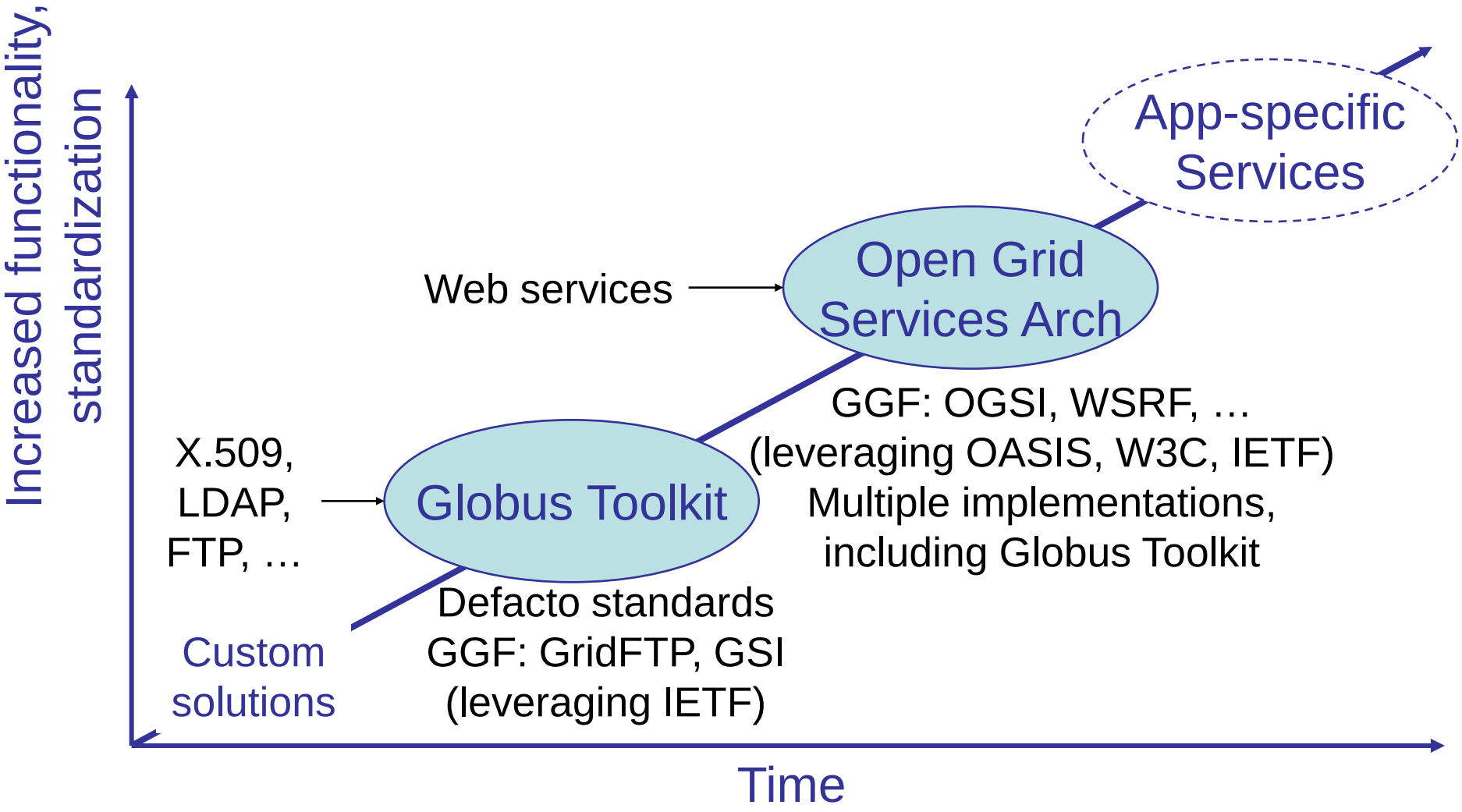
標準に基づくグリッドアーキテクチャ

- **標準プロトコル、標準サービスの開発**
  - 遠隔資源への共通のアクセス方式
  - 既存のプロトコルに基づく
- **グリッドAPI，SDKの開発**
  - グリッドプロトコル、サービスへのインターフェース
  - アプリケーション開発に対するより高いレベルの抽象性を提供
- **成功例：インターネット**
  - HTTP and HTML
  - TCP/IP, telnet, ftp, mail, . . .

重要なポイント

- **Internet Protocol，Web Servicesに基づく**
  - TCP/IP, WSDL, SOAP, etc.
- **グリッドに必要な最小限のサービスを定義**
  - Grid Security
  - Addressing – WS-A (WS-Addressing)
    - http://www.w3.org/Submission/ws-addressing/
  - State – WSRF (WS Resource Framework)
    - http://www.oasis-open.org/committees/wsrf/
  - Notification – WS-N (WS-Notification)
    - http://www.oasis-open.org/committees/wsn/

参考：Evolution of the Grid



参考文献:グリッド全般

- Ian Foster, Carl Kesselman. Computational Grids. In The Grid: Blueprint for a Future Computing Infrastructure, Morgan-Kaufmann, 1999. http://dsl.cs.uchicago.edu/papers/gridbook_chapter2.pdf
- I. Foster, C. Kesselman. The Grid 2: Blueprint for a New Computing Infrastructure, Second Edition, ISBN 978-1-55860-933-4, 2003. http://www.mkp.com/grid2
- I. Foster, C. Kesselman, S. Tuecke. The Anatomy of the Grid: Enabling Scalable Virtual Organizations.. International J. Supercomputer Applications, 15(3), 2001. http://www.globus.org/research/papers/anatomy.pdf
- I. Foster, C. Kesselman, J. Nick, S. Tuecke. The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration.; June 22, 2002. http://www.globus.org/research/papers/ogsa.pdf

参考文献：Web Services

- Web Services Addressing, http://www.w3.org/Submission/ws-addressing/
- Web Services Resource Framework, http://www.oasis-open.org/committees/wsrf/
- Web Services Notification, http://www.oasis-open.org/committees/wsn/

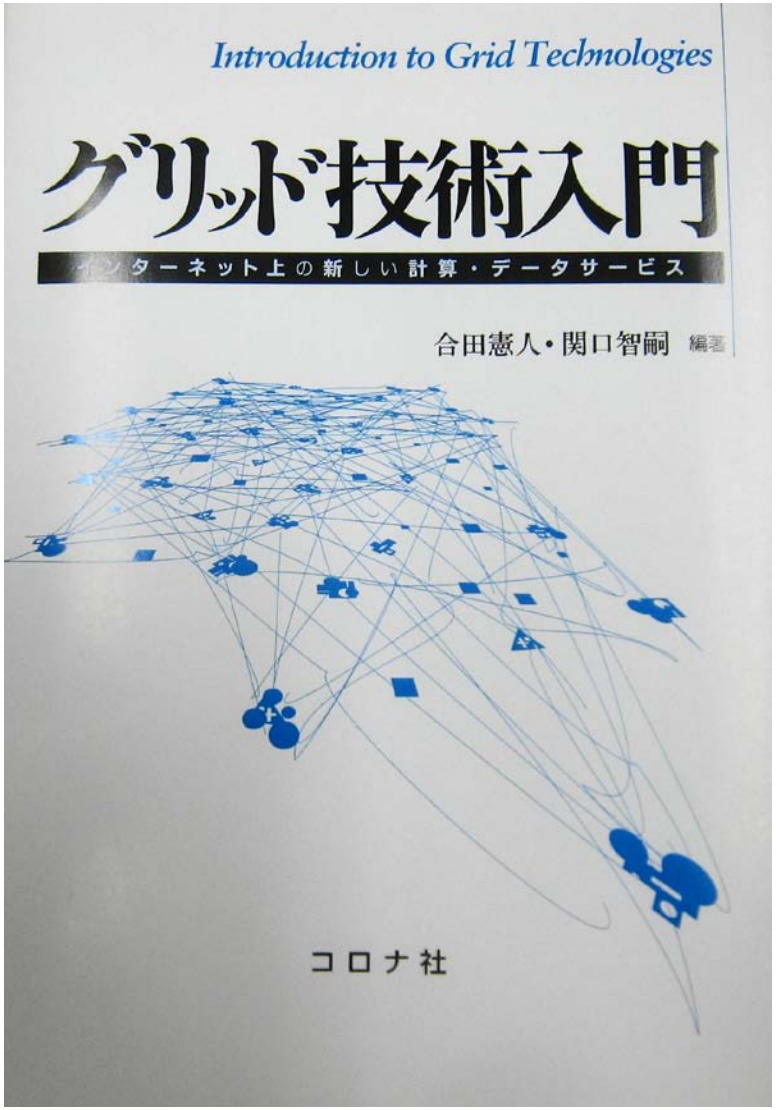
参考文献:グリッド基本ソフトウェア

- Ian Foster and Carl Kesselman. Globus: A Metacomputing Infrastructure Toolkit. International Journal of Supercomputer Applications, 11(2):115-128, 1997. ftp://ftp.globus.org/pub/globus/papers/globus.ps.gz
- Andrew Grimshaw, Michael Lewis, Adam Ferrari, and John Karpovich. Architectural Support for Extensibility and Autonomy in Wide-Area Distributed Object Systems. University of Virginia CS Technical Report CS-98-12, June 1998. http://www.cs.virginia.edu/~legion/papers/CS-98-12.ps

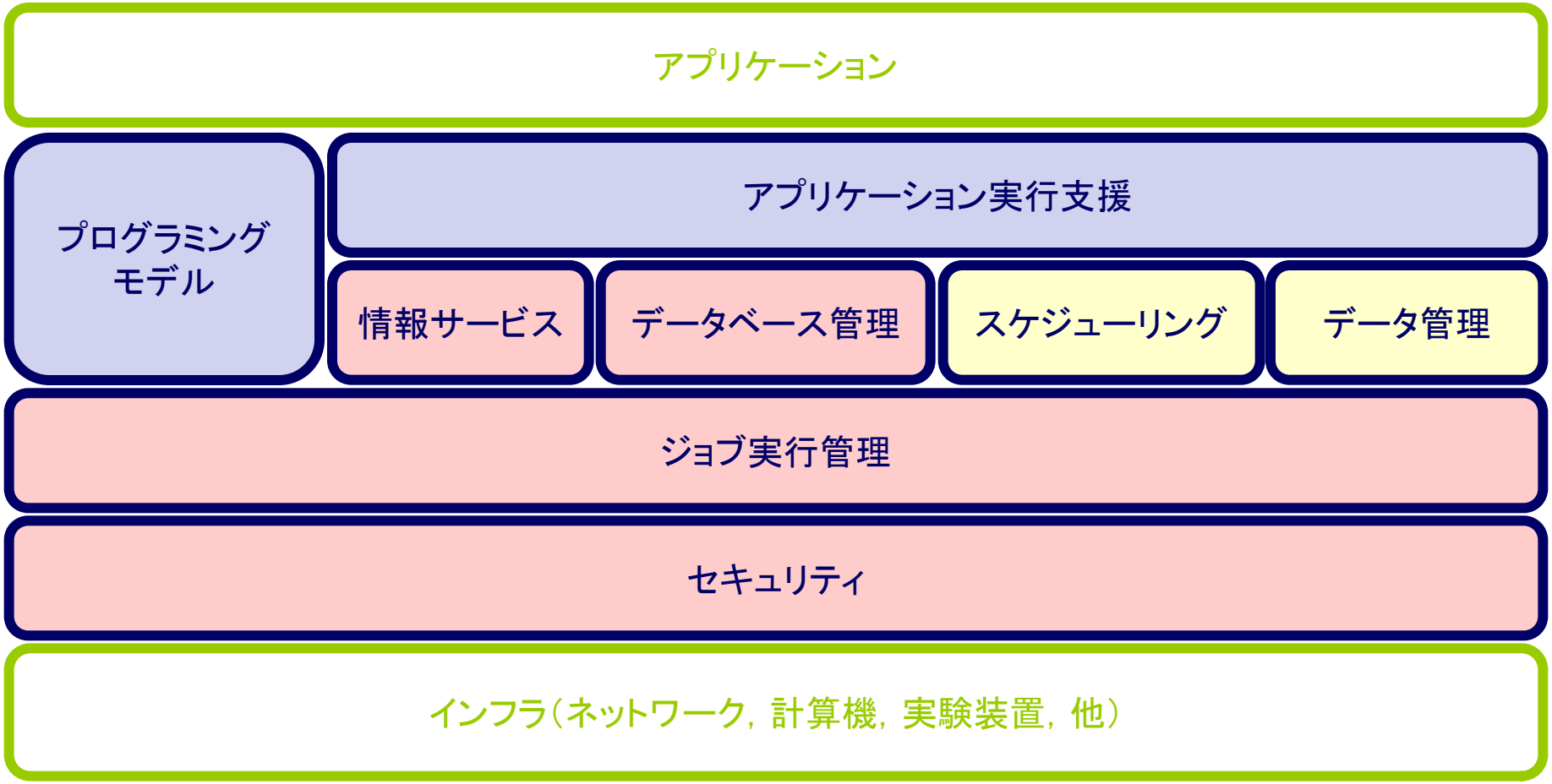
グリッド要素技術

グリッド技術入門

- **グリッド技術入門**
  - インターネット上の新しい計算・データサービス
- **合田憲人，関口智嗣編著**
- **コロナ社，2008年**
- ISBN：978-4-339-02426-5



グリッドの要素技術（1）


アプリケーション
プログラミング
モデル
アプリケーション実行支援
情報サービス
データベース管理
スケジューリング
データ管理
ジョブ実行管理
セキュリティ
インフラ（ネットワーク，計算機，実験装置，他）

グリッドの要素技術（2）

- グリッドセキュリティ (Grid Security Infrastructure, GSI)
- ローカル情報サービス (Grid Information Service, GRIS)
- 広域高速データ転送 (GridFTP)
- 資源管理マネージャ (Grid inetd, GRAM)
- 情報サービスの集約 (Grid Index Information Service, GIIS)
- リソースブローカ (Condor-G, Nimrod-G)
- データ複製管理サービス
- コアロケーション、コリザベーションサービス
- ワークフロー管理サービス
- . . .

グリッドセキュリティ（GSI）

- **単一認証技術**
  - 単一のユーザ認証（パスフレーズ、one-timeパスワード入力）により数十、数千、数万の資源に対するアクセス認証を行う
- **証明書の委譲**
- **委譲された証明書の制限**
  - 有効期間、委譲の深さ、機能の制限
  - 証明書の悪用による被害を軽減
- **動的なプロセス、サービス生成に対応**
- **秘密鍵の保護**

公開鍵暗号系 (public-key cryptosystem)

- 非対称暗号系とも
- 公開鍵eと秘密鍵d
- 平文 – e → 暗号文 – d → 平文
- eからdの計算は計算量的に非現実的
- 公開鍵は秘密にする必要はないため、提供が容易
- 発信者認証、改竄発見のためには、電子証明が必要
- DESなどの対称暗号に比べ遅いため、引き続くデータ転送のための対称暗号の鍵の送信、クレジットカード情報など短いメッセージに利用される

- Handbook of Applied Cryptography, by A. Menezes, P. van Oorschot, and S. Vanstone, CRC Press, 1996
  http://cacr.math.uwaterloo.ca/hac/

電子署名（digital signature）

- 受け取った情報が改竄されていないことを保障する
- 公開鍵暗号においては、情報のハッシュ値を秘密鍵で暗号化して付加する
- 受信側は情報のハッシュ値と付加されたハッシュ値を公開鍵で復号化したものと比較する

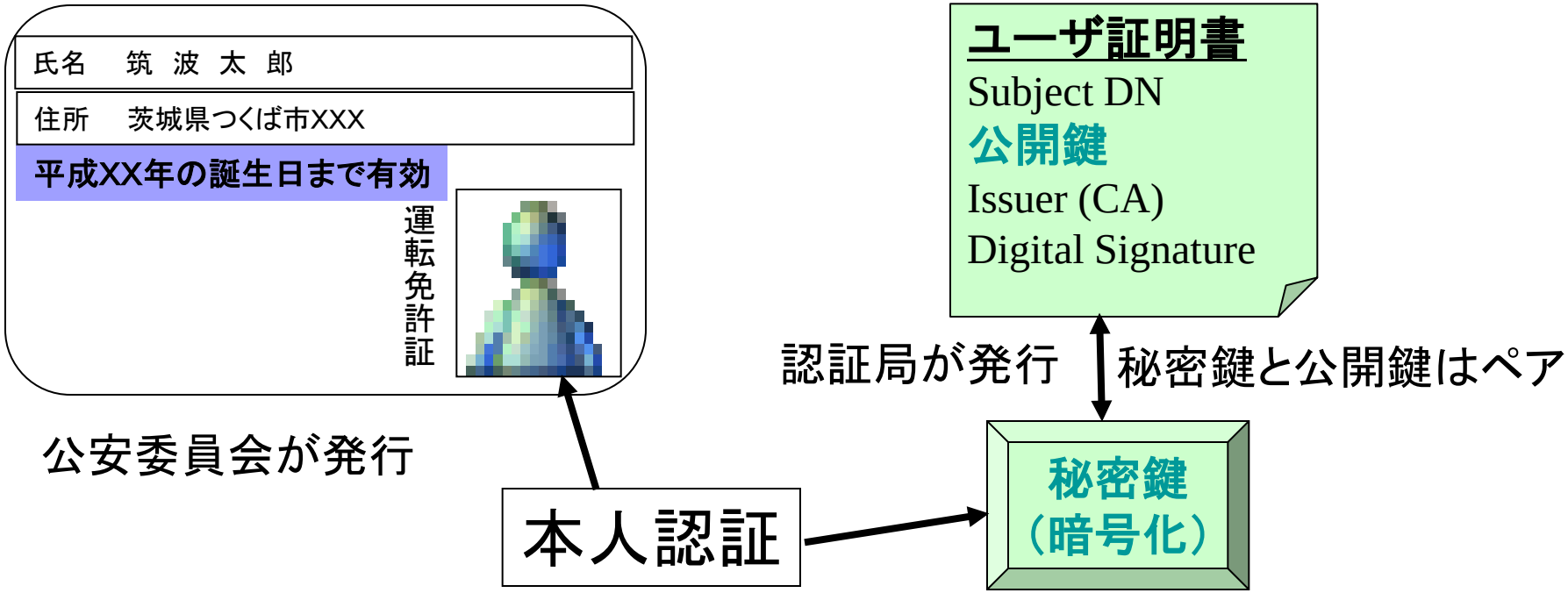
Globus Toolkitにおけるセキュリティ(GSI)

- **基本は公開鍵暗号＋X.509証明書＋SSL (Secure Socket Layer)**
- **相互認証と代理証明書による証明書の委譲**
- **公開鍵暗号（非対称鍵）**
  - 公開鍵はデータの暗号化に利用される
  - 秘密鍵は公開鍵で暗号化されたデータの復号に用いる
- **認証を受けるエンティティ（ユーザ、計算機等）は認証局によって署名された証明書を保持している**
- **X.509 証明書は次のものを含んでいる:**
  - エンティティのsubject名 (user ID, host name)
  - その公開鍵
  - 証明書に署名している認証局（CA)のID
  - 認証局(CA)からの“署名”
    - 証明書が認証局から発行されていることを認める
    - Subject名の保証
    - 公開鍵とsubject名との対応の保証

証明書
Subject DN
公開鍵
Issuer (CA)
Digital Signature

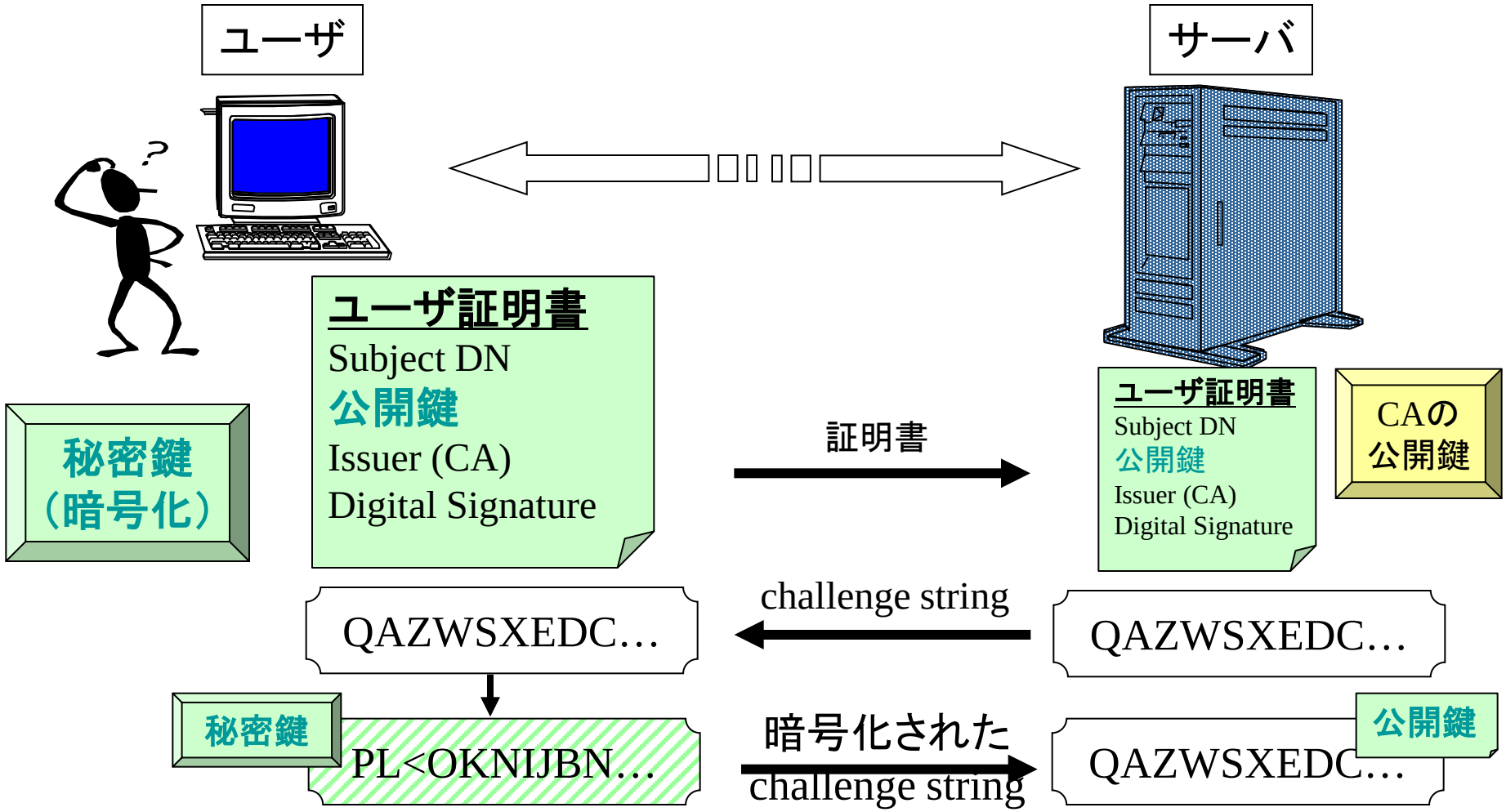
証明書

- **証明書**
  - 運転免許証などと同じようなもの。免許証における写真（本人であることを確認する手段）が、秘密鍵に相当
  - 認証局により署名される
  - 証明書が信用してもらえるかどうかは（サービスを提供する）相手に依存



GSIにおける認証

以下の例はユーザ認証であるが、この逆方向の認証も行われ、相互に認証する



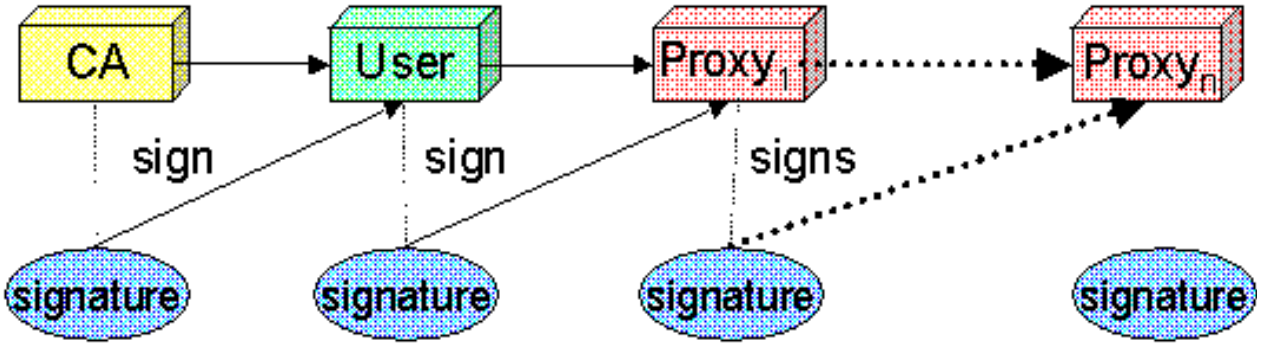
GSIによる拡張

- **Proxy Certificate Profile**
  - X.509（RFC 2459）に基づくProxy Certificate Profile
  - restricted impersonation within a PKI based authentication system.
- **GSS-API（RFC 2743）の拡張**
  - Credentialの輸出、輸入
  - 任意のタイミングの委譲
  - Credentialの操作の拡張
    - 限定されたcredential

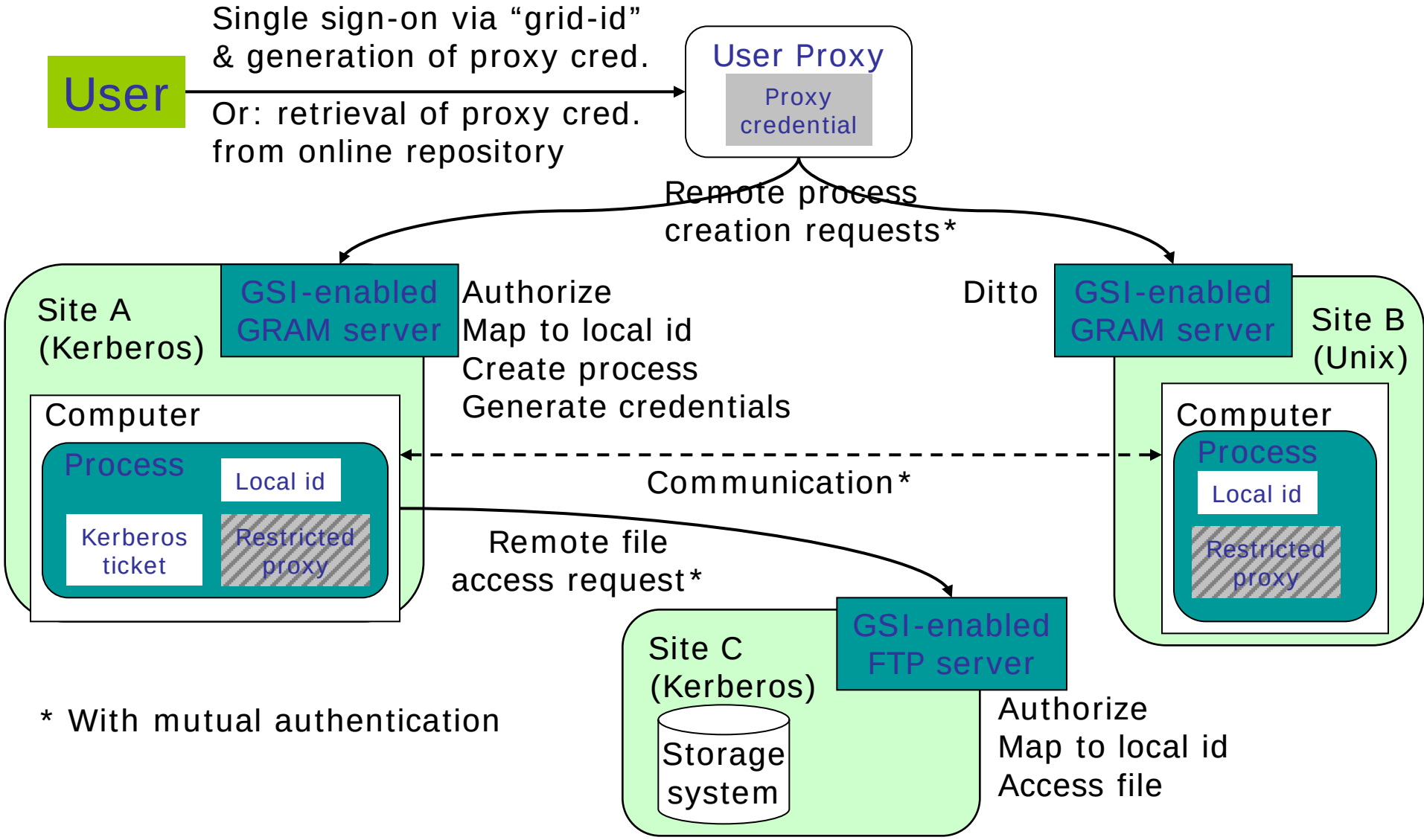
- **Internet X.509 Public Key Infrastructure Proxy Certificate Profile**
  - RFC 3820 by Grid community – OGF
  - GSS-API Extensions
  - ftp://ftp.rfc-editor.org/in-notes/rfc3820.txt

証明書の委譲

- 新たに秘密鍵、公開鍵を生成し、（CAではなく）所有者により署名される
  - 秘密鍵は転送されない
- 代理証明書と所有者の証明書を受け取り、代理証明書および所有者の証明書の正当性を確認する

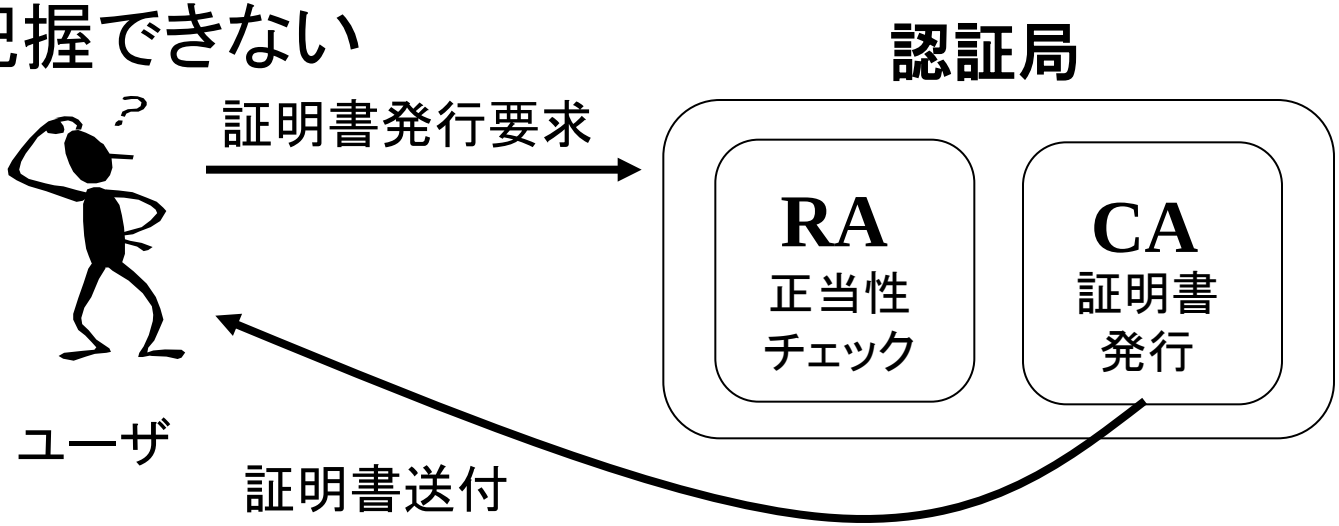


GSI in Action "Create Processes at A and B that Communicate & Access Files at C"

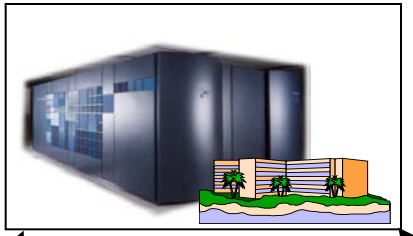


証明書と認証局

- **認証局**
  - 証明書を発行する第三者組織
  - RAとCAの2つの役割が必要
    - RA: ユーザ、計算機の正当性をチェック
    - CA: 証明書を発行
  - 発行した証明書がどこでどのように使われるかは把握できない



認証局の仕事



CPSの
作成

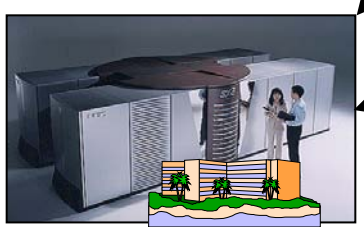
証明書の配布

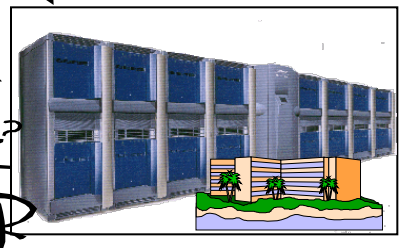
証明書取得ツールの配布

認証局の
証明書作成

認証局

証明書取得
ツールの作成





CSRの受付
正当性確認
証明書発行、送付
管理（期限切れ通知等）

認証局の設定 (Globus Toolkitの場合)

- **認証局 (certificate authority)の設定**
  - $GLOBUS_LOCATION/setup/globus/setup-simple-ca
    - CAのSubject DNの入力
      - cn=CA, ou=CS, o=Univ Tsukuba, c=JP
    - Emailアドレス
    - 有効期限
    - 秘密鍵のパスフレーズ
      - 電子署名に利用する
      - スペースは使えない
  - $GLOBUS_LOCATION/setup/globus_simple_ca_CA_Hash_setup/setup-gsi -default
    - /etc/grid-security/certificatesにCAの公開鍵を保存

ホスト証明書の取得

- **ホスト証明書の発行要求**
  - grid-cert-request -host <hostname>
    - /etc/grid-security/hostkey.pem（秘密鍵）
    - /etc/grid-security/hostcert_request.pem
    - /etc/grid-security/hostcert.pem（空ファイル）
- **RAに確認してもらう**
- **hostcert_request.pemをCAに送付し，電子署名してもらう**
  - grid-ca-sign -in hostcert_request.pem -out signed.pem
- **署名されたhostsigned.pemを受取り，/etc/grid-security/hostcert.pemに保存**
- **証明書の内容を表示**
  - openssl x509 –in hostcert.pem -text

ユーザ証明書の取得

- **ユーザ証明書の発行要求**
  - grid-cert-request
    - ~/.globus/userkey.pem（秘密鍵）
    - ~/.globus/usercert_request.pem
    - ~/.globus/usercert.pem（空ファイル）
- **RAに確認してもらう**
- **usercert_request.pemをCAに送付し，電子署名してもらう**
  - grid-ca-sign -in usercert_request.pem -out signed.pem
- **署名されたsigned.pemを受取り，~/.globus/usercert.pemに保存**

GSIによる認可の設定

- **Grid-mapfileに登録**
  - Grid-mapfile-add-entry -dn “/C=JP/O=Univ Tsukuba/OU=CS/OU=tatebe.net/CN=Osamu Tatebe” -ln tatebe
    - /etc/grid-security/grid-mapfileにエントリを追加

GSI-enabled OpenSSHの設定

- **$GLOBUS_LOCATION/sbin/SXXsshdを/etc/init.d/gsisshdにコピー**
- **service gsisshd start**

代理証明書の生成とlogin

- **代理証明書の生成**
  - grid-proxy-init [-debug] [-veriry]
- **確認**
  - grid-proxy-info
- **GSI認証によるLogin**
  - gsissh hostname
  - ユーザ証明書が委譲される
- **GSI認証によるftp**
  - gsisftp hostname

参考文献:グリッドセキュリティ

- Ian Foster, Carl Kesselman, Gene Tsudik and Steven Tuecke. A Security Architecture for Computational Grids. Proc. 5th ACM Conference on Computer and Communication Security, 1998. ftp://ftp.globus.org/pub/globus/papers/security.ps.gz
- Eshwar Belani, Amin Vahdat, Thomas Anderson, and Michael Dahlin. The CRISIS Wide Area Security Architecture. Proc. USENIX Security Symposium, January 1998. http://now.cs.berkeley.edu/WebOS/papers/uss.ps

情報サービス

- **発見、モニタリング、計画、適応的アプリケーションのための基本的なメカニズム**
- **多様、多数、動的、地理的分散した資源**
- **故障の対応**
  - ネットワーク不通、ノード故障は例外ではなく規則
- **情報の種類**
  - IPアドレス、管理者
  - CPU、OS、ソフトウェア
  - ネットワークバンド幅、遅延、プロトコル、論理トポロジ
  - CPU負荷、ネットワーク負荷、ディスク使用量、負荷
  - . . .

情報サービスの利用例

- **サービス発見サービス**
  - 新しいサービスの発見
- **スーパスケジューラ**
  - システムコンフィギュレーション、CPU負荷などより最適な計算資源を選ぶ
- **ファイル複製選択サービス**
  - 最適なファイルコピーを選ぶ
- **適応型アプリケーションエージェント**
  - 実行時の資源状況によりアプリケーションの振る舞いを変化
- **故障発見サービス**
  - 過負荷、故障の発見
- **性能診断**
  - 性能の低い原因を診断

要求事項(1)

- **情報提供者の分散**
  - 分散しているため、全ての情報は古い
  - 情報の信頼度が必要
    - タイムスタンプ、有効期限など
  - 可能な限り早く、効率的に伝える
  - 一般的に大域的状態の一貫したビューを見せる必要はない
    - 提供者の数にスケールしない

単一ソースからの状態情報の効率的な伝達に焦点

要求事項(2)

- **故障の対策**
  - それぞれの資源、ネットワークは故障しやすい
  - 頑強である必要がある
    - どれかの資源の故障が他の資源の情報収集を妨げない
    - 部分的な情報、一貫性ない情報の可能性もある
- **情報サービスは、可能な限り分散、非集中化させる必要がある**
  - 利用可能な資源の情報を得る可能性を増大
- **故障は例外ではなく、ルールだという仮定のもとに構築する必要がある**

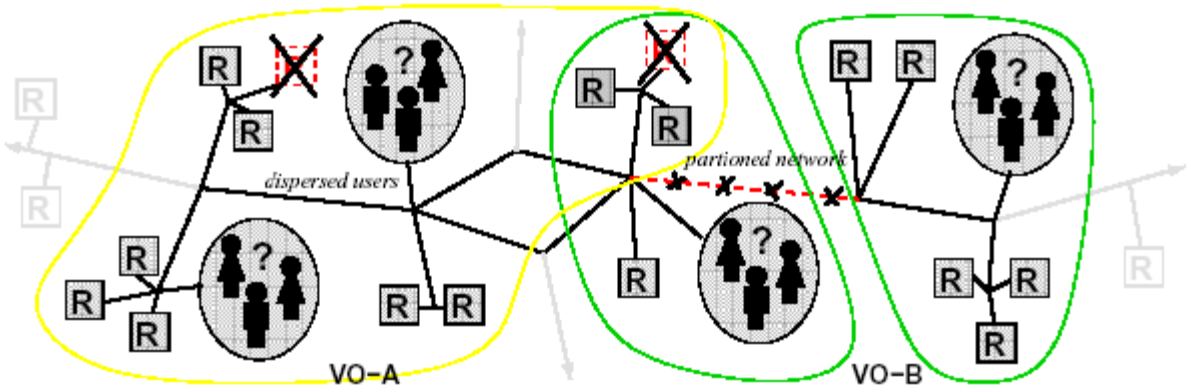


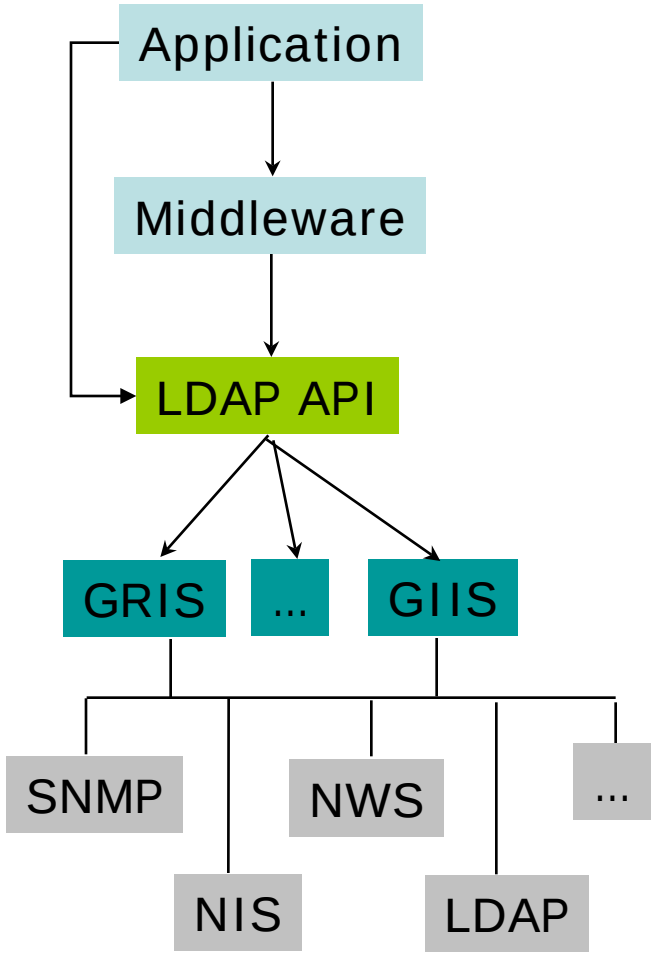
Figure 1. Distributed virtual organizations. Users in VO-A and VO-B have access to partially overlapping resources. While VO-B is split by network failure, it should operate as two disjoint fragments.

要求事項(3)

- **情報サービスコンポーネントの多様性**
  - 多くの、多種多様な資源があり、発見、モニタリングに対しそれぞれ独特な要求があるかもしれない
  - 十分な発見、モニタリング手法を準備する必要がある
  - 複数の管理領域に含まれるため、多種多様なアクセスポリシがある
    - アクセス制御

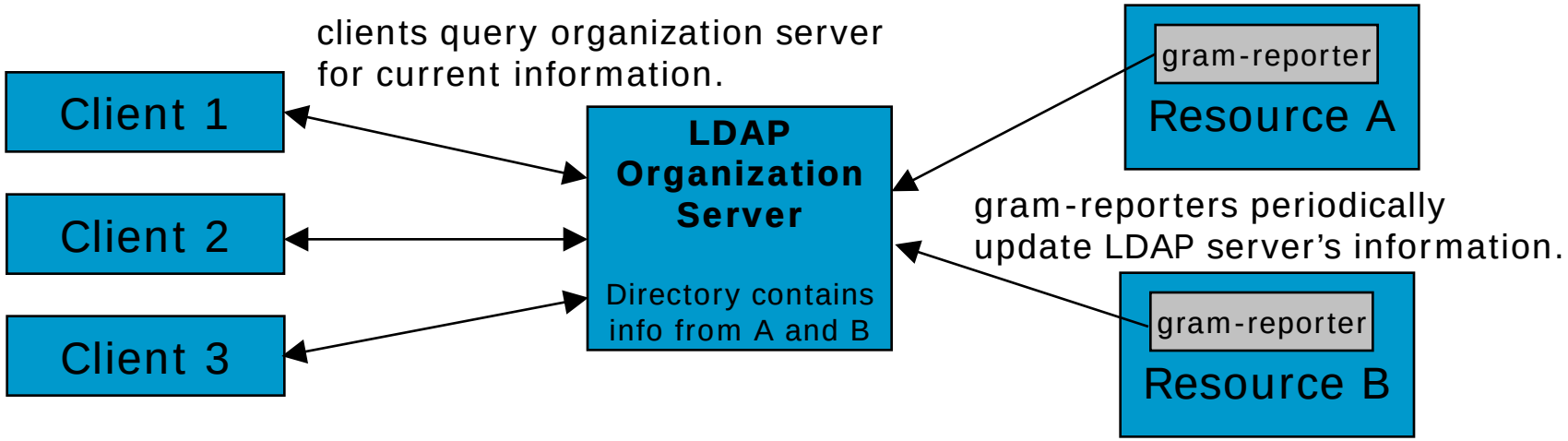
Globus MDS Approach

- **Based on LDAP**
  - Lightweight Directory Access Protocol v3 (LDAPv3)
  - Standard data model
  - Standard query protocol
- **Globus Toolkit schema**
  - Host-centric representation
- **Globus tools**
  - GRIS, GIIS, gram-reporter
  - Data discovery, publication,...



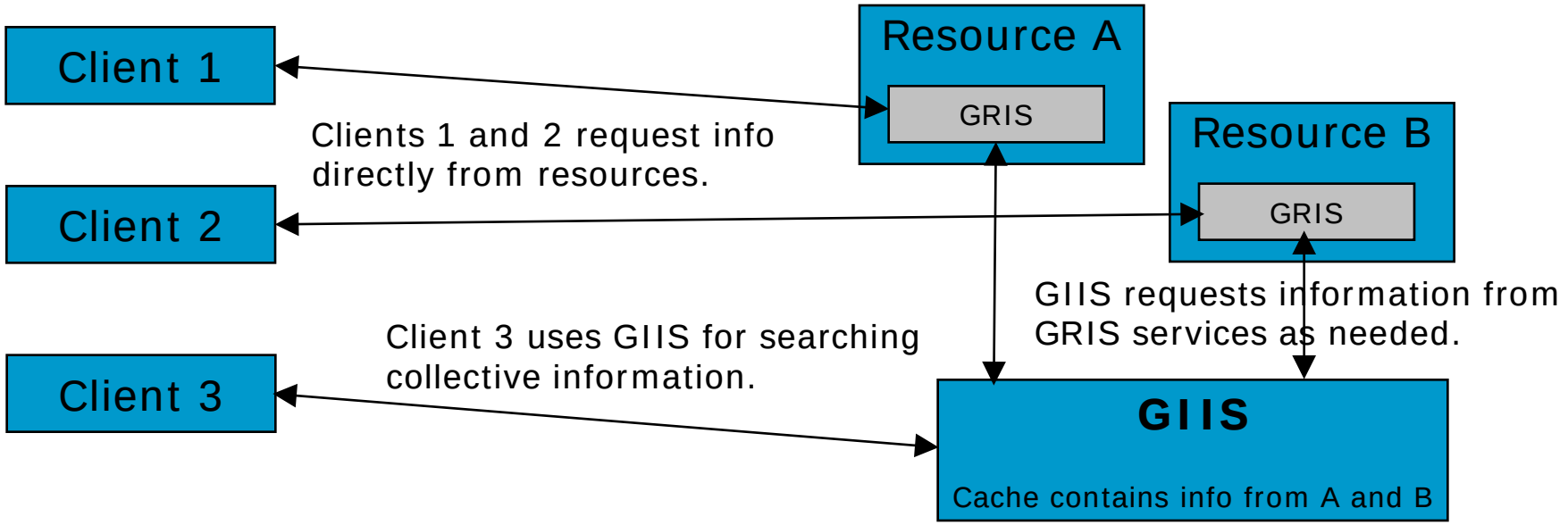
“Classic” MDS Architecture

- Resources push information into a central organization server via regular updates (globus-gram-reporter), where it can be retrieved by clients.
- Regular updates don’t scale as the number of resources grow rapidly. Commercial LDAP servers are optimized for “read” requests, and can’t handle frequent “write” requests.
- If organization server is unavailable, no information is available.



“Standard” MDS Architecture (v1.1.3)

- Resources run a standard information service (GRIS) which speaks LDAP and provides information about the resource (no searching).
- GIIS provides a “caching” service much like a web search engine. Resources register with GIIS and GIIS pulls information from them when requested by a client and the cache as expired.
- GIIS provides the collective-level indexing/searching function.



MDS (Metacomputing Directory Service) の構成要素

- **Grid Resource Information Service (GRIS)**
  - 特定のリソースに関する情報を提供
  - 複数の情報プロバイダをサポートするように設定可能
  - 問い合わせにはLDAP プロトコルを用いる
- **Grid Index Information Service (GIIS)**
  - 複数のGRISサーバで集めた情報を提供
  - 複数のGRISサーバに分散した情報を効率的に問い合わせることを支援
  - 問い合わせにはLDAP プロトコルを用いる

参考文献:情報サービス

- K. Czajkowski, S. Fitzgerald, I. Foster, C. Kesselman. Grid Information Services for Distributed Resource Sharing. Proc. Tenth IEEE International Symposium on High-Performance Distributed Computing (HPDC-10), IEEE Press, August 2001. http://www.globus.org/research/papers/MDS-HPDC.pdf