

---

# グリッドにおけるセキュリティ

産業技術総合研究所 グリッド研究センター

田中 良夫



# 話の内容

---

- **グリッドにおけるセキュリティの肝**
  - ▶ セキュリティに対するグリッド特有の要求事項
- **グリッドシステムにおけるセキュリティの実現例 (PKI / 証明書を使った認証を中心に)**
  - ▶ UNI CORE
  - ▶ Globus Toolkit
  - ▶ GRIP (UNI CORE & Globus)
  - ▶ Grid Portal
- **現状、動向および課題**



---

# グリッドにおけるセキュリティの肝



## なぜGridにおけるセキュリティは難しいか

- 利用される資源は高価なもので、解こうとする問題も繊細な場合が想定される
- 利用する資源はしばしば管理体制の異なるドメインに置かれる
  - 各資源が独自のポリシーや手続きを持っている
- 1度の計算に利用される様々な資源は巨大で動的に変動し、あらかじめ予測することができない可能性がある
  - 単なるクライアント・サーバではなく、委譲が必要
- 広い範囲で利用可能かつ適用可能でなければならない
  - 標準的、良くテストされ、良く理解されたプロトコルを幅広いツール群に取り込む



# Gridにおけるセキュリティの必要条件

## ユーザの視点

- 1) 使いやすい
- 2) Single Sign-on
- 3) アプリケーションが動く  
ftp, ssh, MPI, Condor, Web, ...
- 4) ユーザに基づく信頼モデル
- 5) Proxies/agents (委譲)

## 資源の所有者の視点

- 1) ローカルなアクセス制御の指定
- 2) 検査、課金など
- 3) ローカルシステムとの統合  
Kerberos, AFS, license mgr.
- 4) 資源の保護

## 開発者の視点

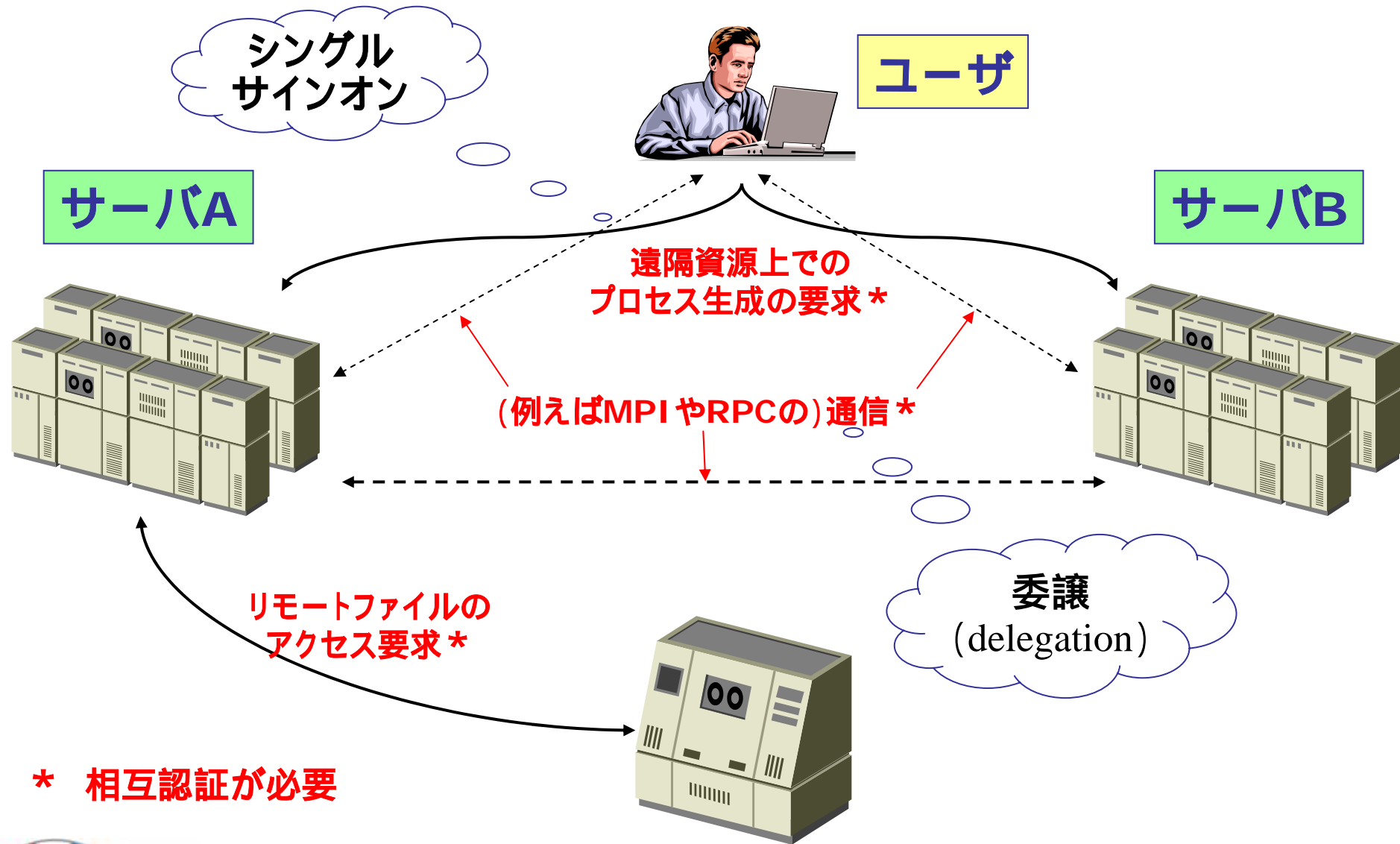
認証、柔軟なメッセージ保護、柔軟な通信、委譲などを備えたAPI/SDK

様々なセキュリティ機能を直接呼び出す (例: GSS-API)

あるいはより高レベルなSDKに組み込まれるセキュリティ:

例: GlobusIO, Condor-G, MPICH-G2, HDF5, など

# Single Sign On と delegation



# 最近の言い方

## three key functions in a Grid security model

---

### Multiple security mechanisms

- ▶ VOに参加する各組織の(複数の)セキュリティ機構を利用

### Dynamic creation of services

- ▶ 管理者の干渉なしに、ユーザが新しいサービスを動的に生成できる

### Dynamic establishment of trust domains

- ▶ ユーザとリソースの間だけではなく、VOにおけるリソース間の信頼(trust domain)も確立する
- ▶ それらのtrust domainは複数の組織にまたがり、動的に適応する

Von Welch, et.al., Security for Grid Services, HPDC-12, 2003



# 最近の言い方

## Security Challenges in a Grid Environment

---

### ● The Integration Challenge

- ▶ 既存のセキュリティ技術を(相互)利用、統合

### ● The Interoperability Challenge

- ▶ 複数のドメイン、Hosting Environment同士が協調するために、複数のレベルでのinteroperabilityが必要

#### ◎ Protocol level

- ✦ メッセージ交換の機構が必要(SOAP/HTTPなど)

#### ◎ Policy level

- ✦ 各サイト(party)が(相手方に望む)ポリシーを指定できる
- ✦ 表現されたそれぞれのポリシーは互いに理解できる

#### ◎ Identity level

- ✦ ドメインをまたいでのIdentityの確立
- ✦ 物理的なIdentityの確立ではなく、(各ドメインにおける)IdentityとCredentialとのマッピング

Nataraj Nagaratnam, et.al., Security Architecture for Open Grid Services  
GWD-I (draft-ggf-ogsa-sec-arch-01)



# 最近の言い方

## Security Challenges in a Grid Environment (続き)

### ● The Trust Relationship Challenge

#### ▶ 動的かつユーザによって制御(生成、管理)されるグリッドサービスにおける問題(挑戦)

- ◎ Identity and authorization
  - ✦ サービスを実行する際のidentity, privilegeを制御する
- ◎ Policy enforcement
  - ✦ サービスに対するポリシーの制定、制御
- ◎ Assurance level discovery
  - ✦ セキュリティのレベルを知りたい
  - ✦ Privacy, virus protection, firewall usage, VPN, etc.
- ◎ Policy composition
  - ✦ ポリシは(1つのリソースオーナーではなく)複数のソースにより生成
- ◎ Delegation

Nataraj Nagaratnam, et.al., Security Architecture for Open Grid Services

GWD-I (draft-ggf-ogsa-sec-arch-01)



# Grid Security Requirements

---

- Authentication
- Delegation
- Single Logon
- Credential Lifespan and Renewal
- Authorization
- Privacy
- Confidentiality
- Message integrity
- Policy exchange
- Secure logging
- Assurance
- Manageability
- Firewall traversal
- Securing the OGSA infrastructure

Nataraj Nagaratnam, et.al., Security Architecture for Open Grid Services

GWD-I (draft-ggf-ogsa-sec-arch-01)



---

# グリッドシステムにおける セキュリティの実現例

UNI CORE  
Globus Toolkit (GT2 & GT3)  
GRIP – UNI CORE&Globus  
Grid Portal

謝辞： UNICOREのスライドは下記の方々、プロジェクトにご提供いただきました。

David Snelling (Fujitsu Europe)

Ralf Ratering (Pallas)

Philipp Wieder (Research Centre Jülich)

GT3のスライドの多くは中田秀基氏(産総研)にご提供いただきました。

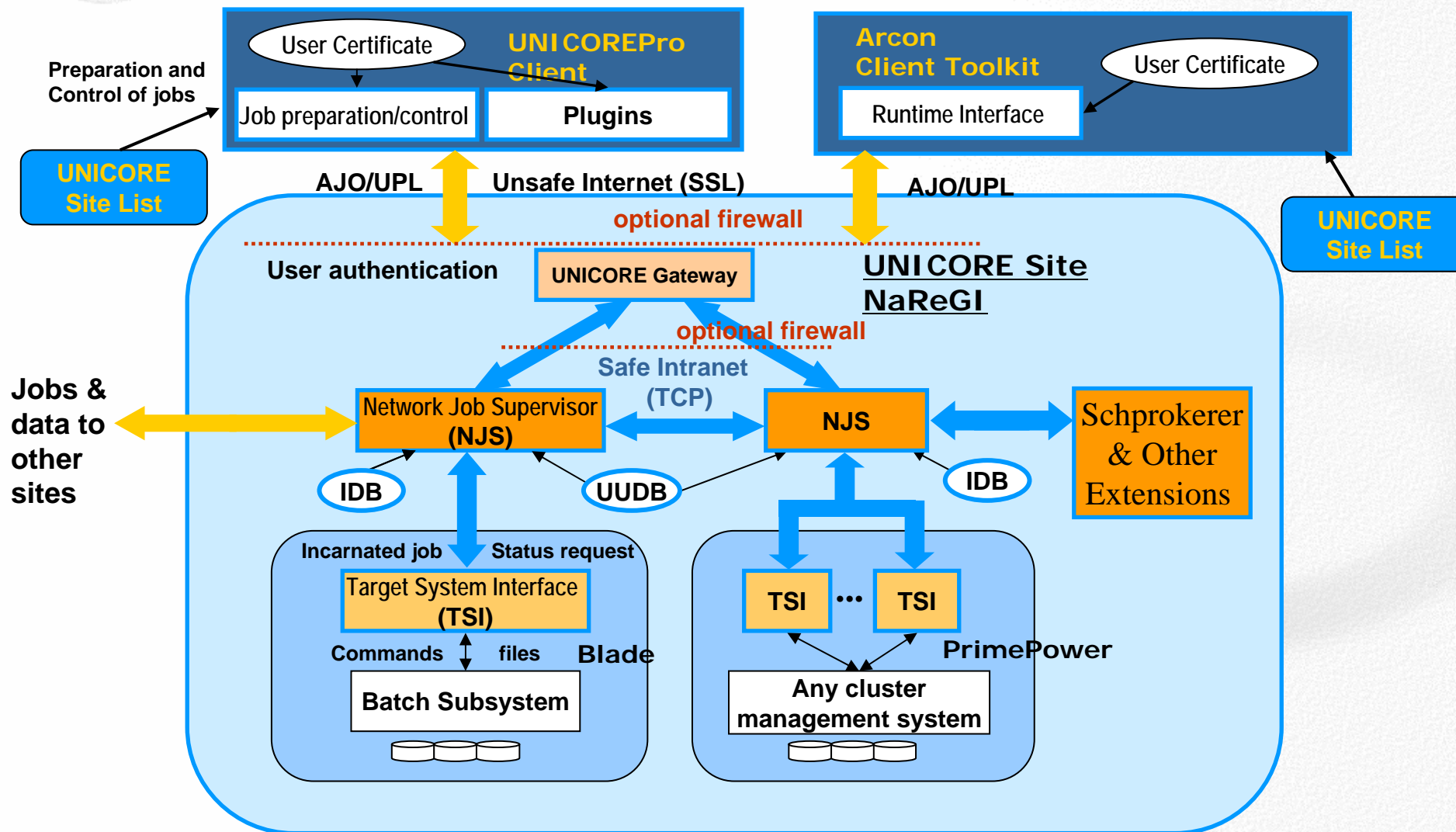


---

# UNI CORE



# Architecture

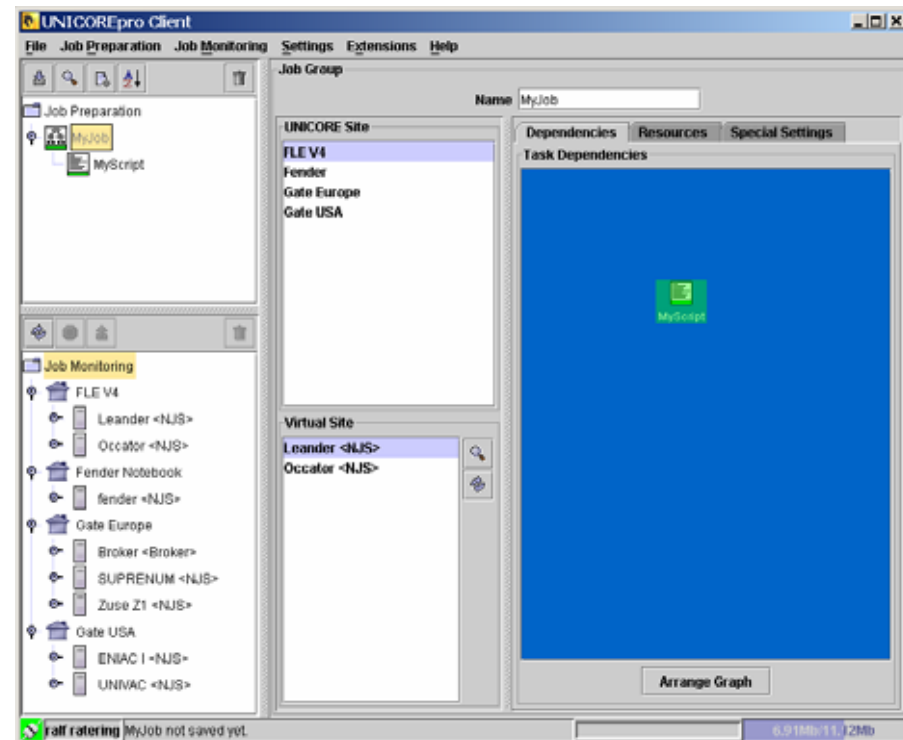


QuickTime™ and a GIF decompressor are needed to see this picture.

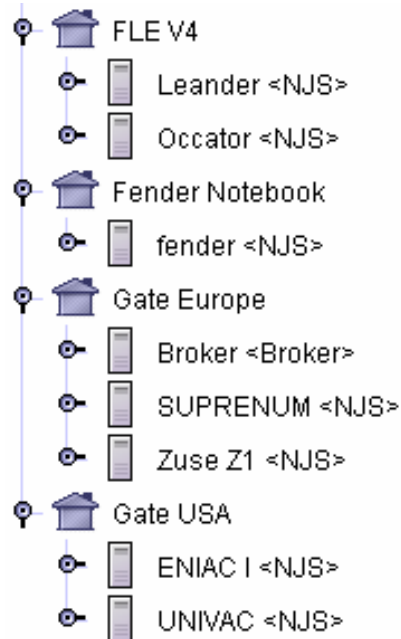
# The UNICOREpro Client



- Graphical Interface to UNICORE Grids
- Completely written in Java
- Open Source under Pallas Community License
- Job Preparation, Monitoring and Control
- Complex Workflows
- File Management
- Certificate Handling
- Integrated Application Support



By Courtesy of Ralf Ratering



UNICORE Sites:

Gateway installed at site



Virtual Sites:

NJS (Network Job Supervisor)

Configure your own Grid:



← Gateway  
← addresses

<http://www.unicorepro.com/unicoreSites.xml>

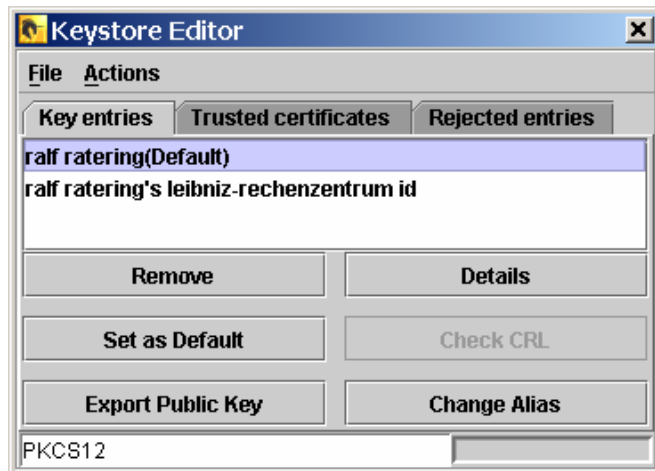
[./naregi/naregiSites.xml](http://naregi/naregiSites.xml)

By Courtesy of Ralf Ratering

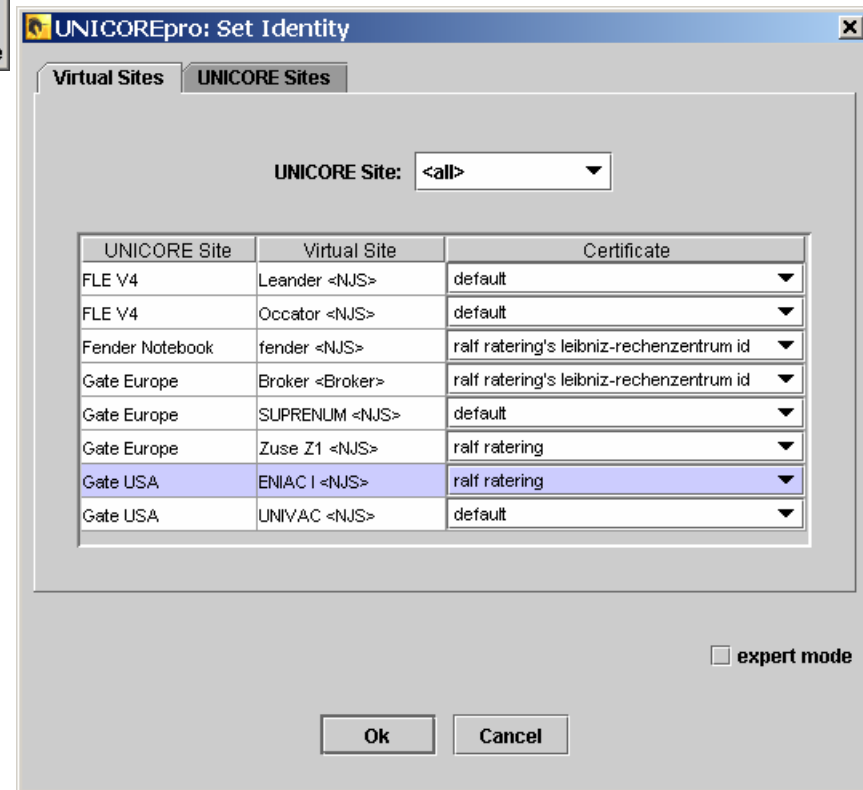
# Authentication: User certificates



Unlock keystore at startup



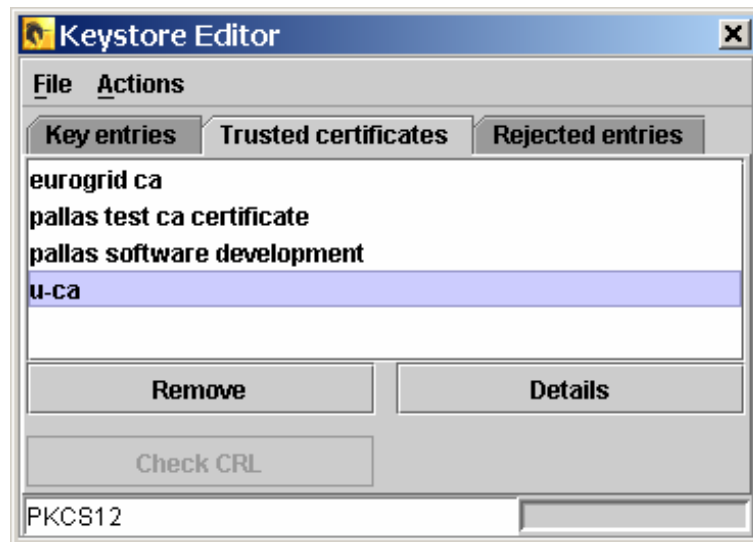
Key entries: Who am I?



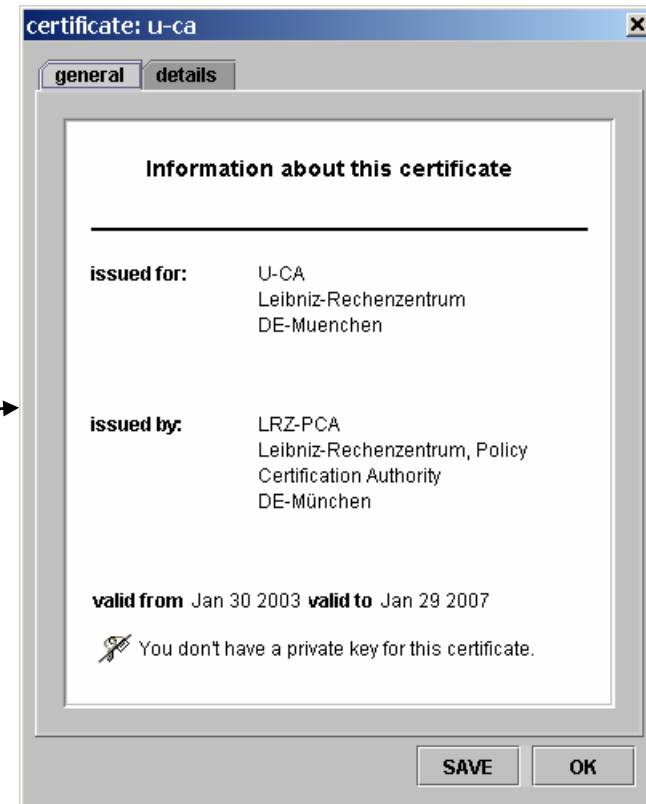
Using different identities

By Courtesy of Ralf Ratering

# Authentication: Trusted entries



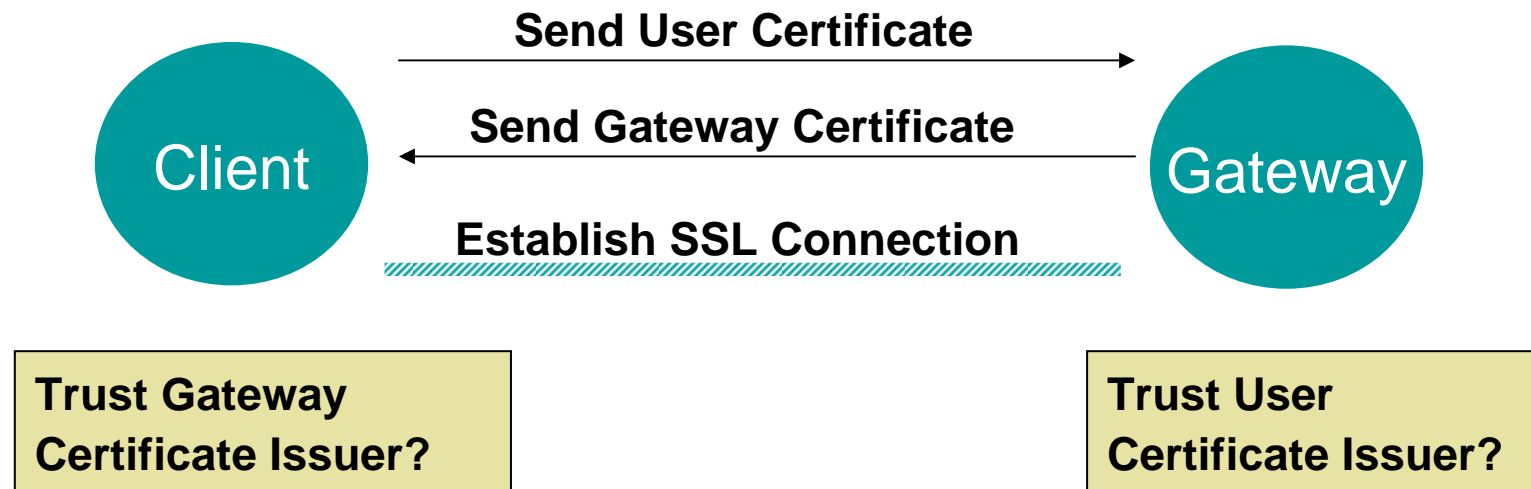
Trusted certificates:  
Whom do I trust?



View details  
about certificate

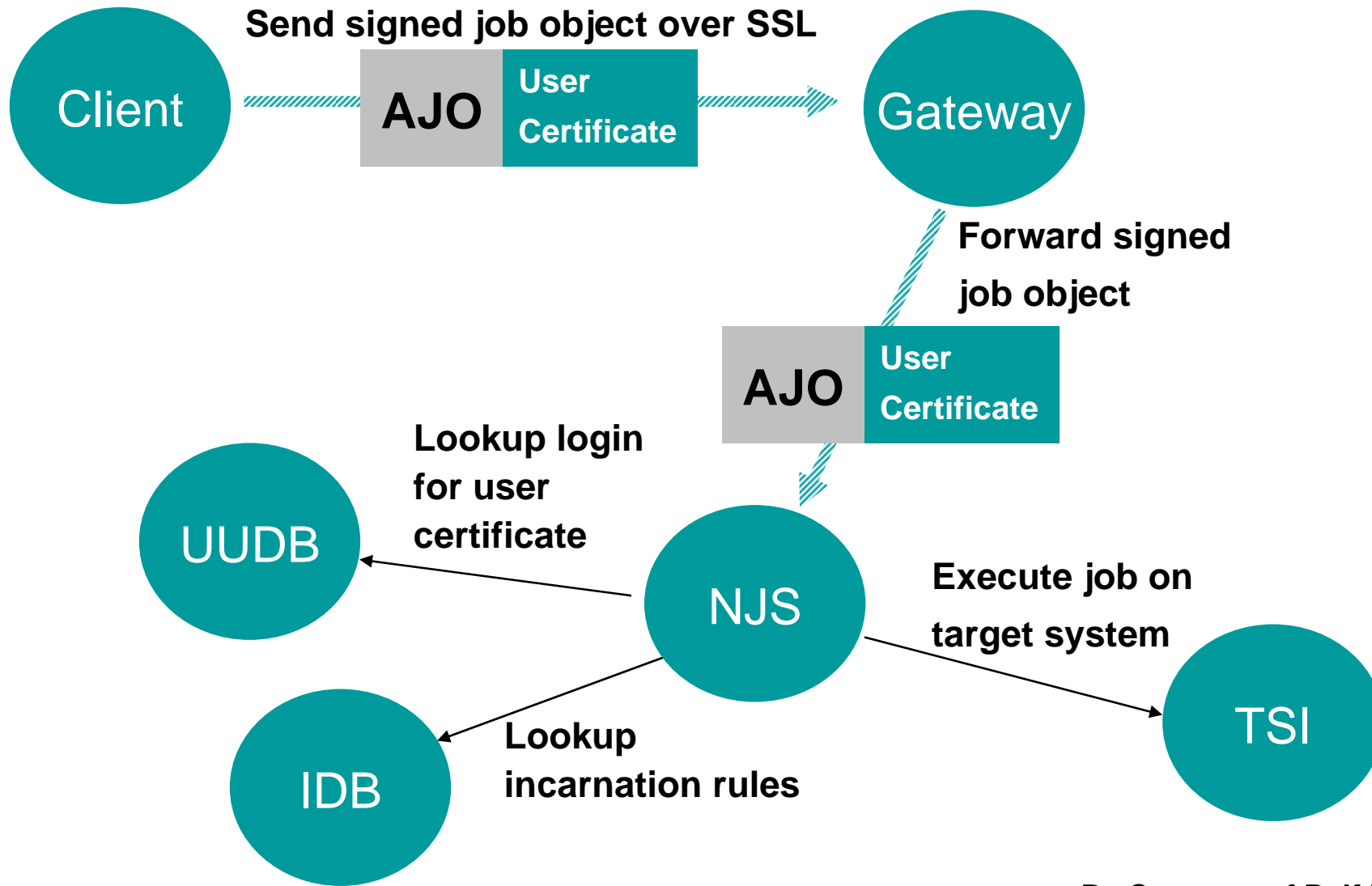
By Courtesy of Ralf Ratering

# Authentication: How does it work?



By Courtesy of Ralf Ratering

# Authentication: How does it work?

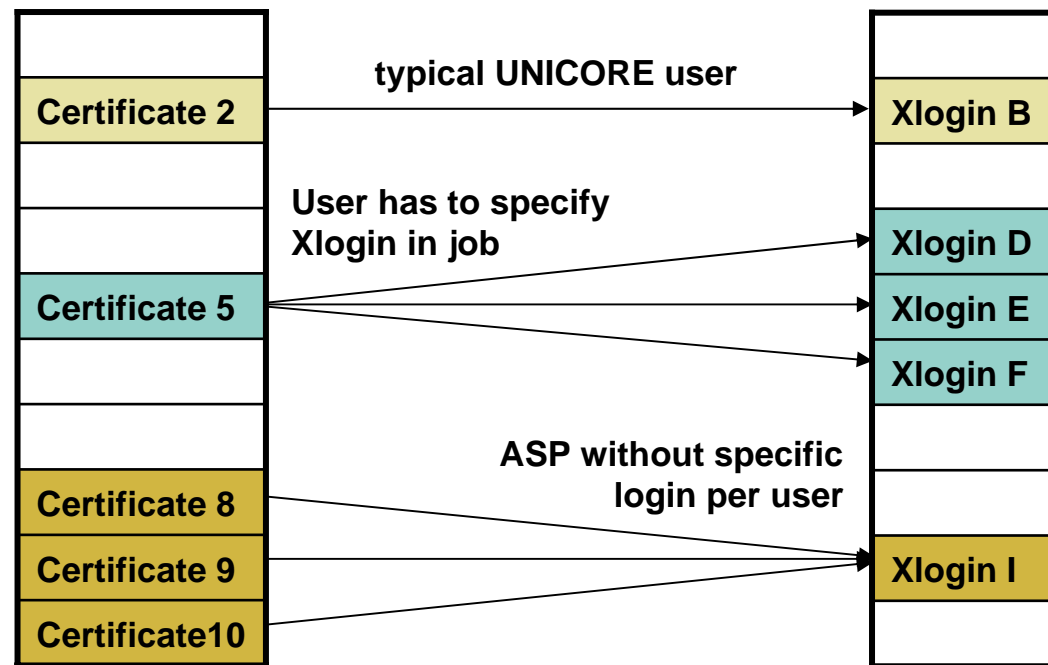


By Courtesy of Ralf Ratering

# Certificate Mapping in UUDB



- Certificates are mapped in the UNICORE User Database (UUDB) to UNIX logins



By Courtesy of Ralf Ratering

## UNICORE – 補足 –

---

- クライアント、サーバいずれもインストールディレクトリの下に認証局の証明書等を置くようになっている。
- 複数の証明書を同時に(サイトごとに)利用可能
- CRLのチェックは以下の条件を満たした場合行われる
  - ▶ プロパティファイルで指定 (gw.check\_crlsがtrue)
  - ▶ クライアントの証明書のV3 extension “CRL Distribution Point”がセットされている
- ジョブに署名されている
- Delegationの機能はない(現在実装中)

---

# GT2



# GT2におけるセキュリティ(GSI)

- 基本は公開鍵暗号 + X.509証明書
- 公開鍵暗号(非対象鍵)
  - ▶ 秘密鍵はデータの暗号化に利用される
  - ▶ 公開鍵は秘密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)は認証局によって発行された証明書を保持していなければならない。
- 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”
    - ⊗ Subject名の保証
    - ⊗ 公開鍵とsubject名との対応の保証
    - ⊗ 証明書が認証局から発行されていることを認める

## 証明書

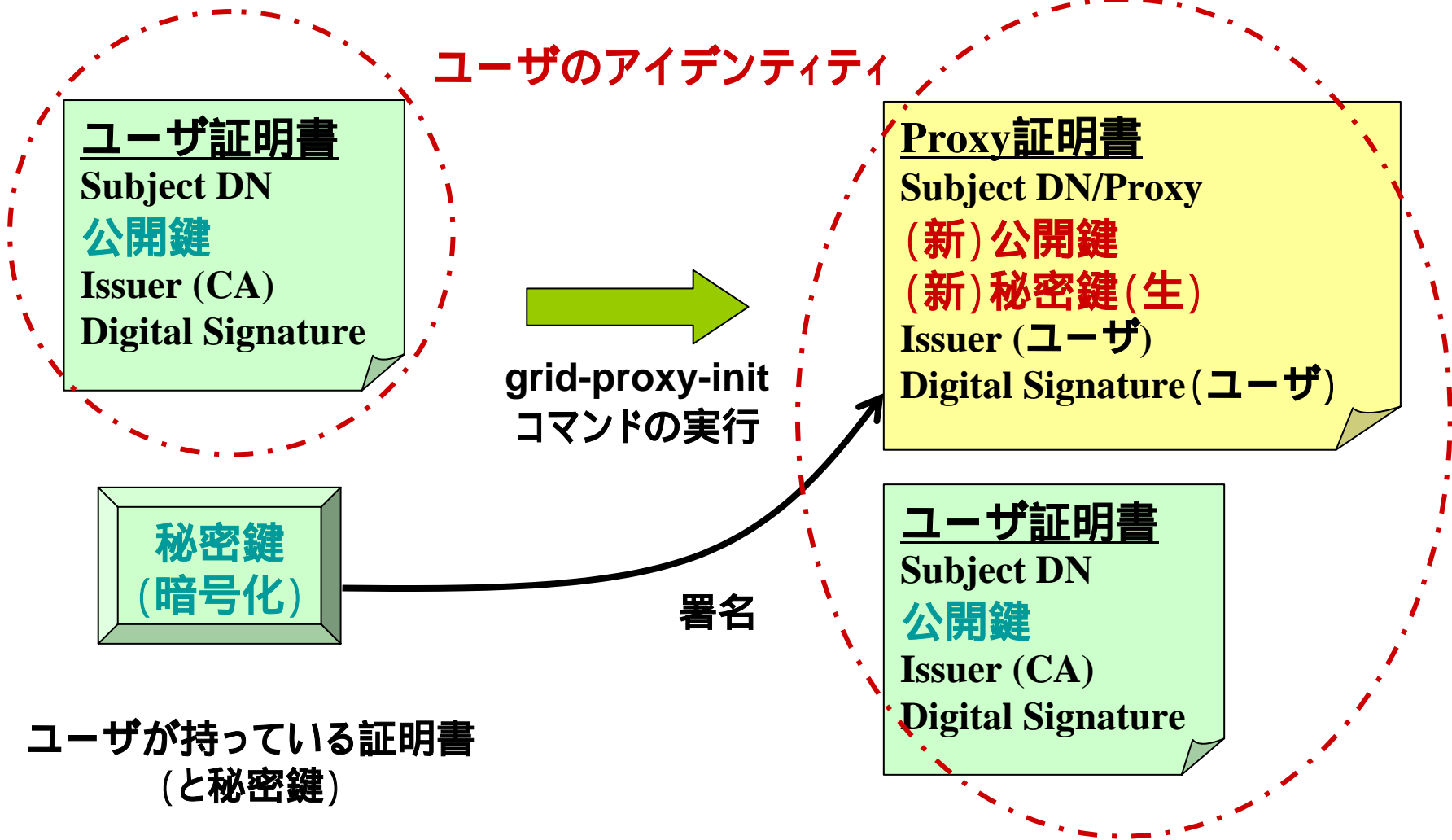
Subject DN

## 公開鍵

Issuer (CA)

Digital Signature

# Proxy証明書



# Proxy証明書 (続き)

## ● 性質

- ▶ X.509 End Entity証明書(EEC) または他の Proxy証明書(PC)により署名されている
- ▶ 他のPCに署名できる(EECは駄目)
- ▶ 新しい(独自の)秘密鍵と公開鍵を持つ
- ▶ PCに署名をしているEECのidentityを基にしたidentityを持つ
- ▶ X.509の拡張

## ● 作り方

- ▶ 新しい公開鍵と秘密鍵のペアを作成
- ▶ それらの鍵を元にProxy証明書のCSRを作成
- ▶ EECまたは他のPCの秘密鍵により、CSRに署名

● Delegationの場合は最初の2つのステップが委譲先で、最後のステップが委譲元で行われる



# Delegation

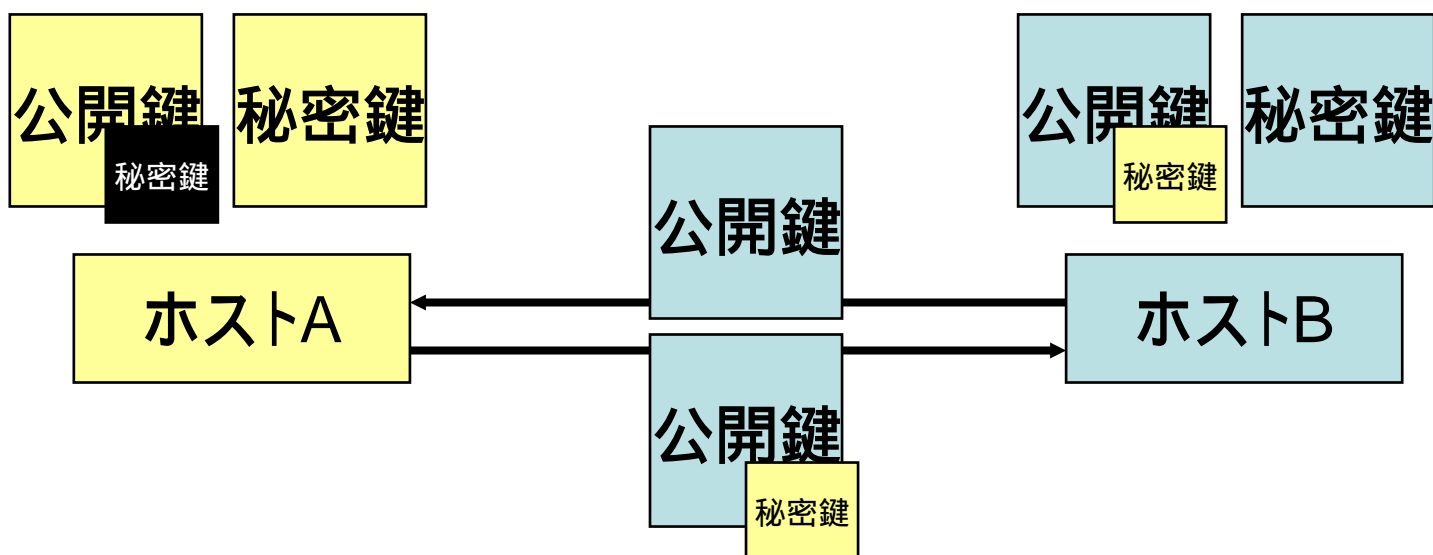
---

- ユーザプロキシの遠隔生成
- 新しい秘密鍵と、元の鍵によって署名された X.509 証明書が生成される
- 遠隔プロセスがユーザの代理として振舞える
- パスワードや秘密鍵をネットワークを介して送らずにすむ

# Delegationの実装

## ● プロキシ証明書の遠隔生成

- ▶ 証明書は、秘密鍵とサインされた公開鍵のペア



# GSI のまとめ

---

- ユーザはかならず証明書を取得しておく。
- 秘密鍵は大切に保管
- 認証が必要なリソースにアクセスする場合は、事前に Proxy 証明書を作成する。
  - ▶ (globusの場合) grid-proxy-init コマンドの実行
  - ▶ 仮想的なログイン
  - ▶ grid-proxy-init コマンドを実行したマシン(クライアントマシン)上に Proxy 証明書は作成される
- あとは Proxy 証明書 + delegation の機能により、single sign on + 隅々までの認証(安全性)が実現される。

## GT2- 補足 -

---

- 認証局の証明書(hash\_value.0)は
  - ▶ /etc/grid-security/certificates/
  - ▶ \${GLOBUS\_LOCATION}/etc/certificates/
  - ▶ ~/.globus/certificates/あたりに置くようになっている
- 複数の証明書を同時に(サイトごとに)利用することはできない
- CRLは認証局の証明書と同じ場所に置いておく
  - ▶ hash\_value.r0
  - ▶ CRLの期限が切れていると、その認証局に発行されている証明書はすべて無効とされる
  - ▶ CRLを自動的に更新する機能はない
- ジョブへの署名はない
- GRAMでDelegationされるのはLimited Proxy

---

# GT3



# GT3のセキュリティ

---

## ● GT2と変わっていないところ

- ▶ User and Service Credentials
  - Ⓜ X.509証明書 & Proxy証明書
  - Ⓜ grid-proxy-init
- ▶ Resource Authorization
  - Ⓜ grid-mapfile
- ▶ Application Interfaces
  - Ⓜ GSSAPI

## ● 変わったところ

- ▶ Proxy証明書のフォーマット
  - Ⓜ impersonation proxy 証明書
  - Ⓜ independent proxy 証明書
- ▶ Web Service Protocol
- ▶ Improved resource security model
- ▶ Removal of network services from trust model

# GT3 セキュリティ

---

## ● トランスポートレベル **メッセージレベル**

▶ 実際にやり取りされるプロトコルはほぼ等価

◎ 暗黙裡に行われていた通信をSOAP通信のレベルに持ち上げている

▶ メッセージの中継が可能に

▶ 証明書のデレゲーションなどはGT2と同じ

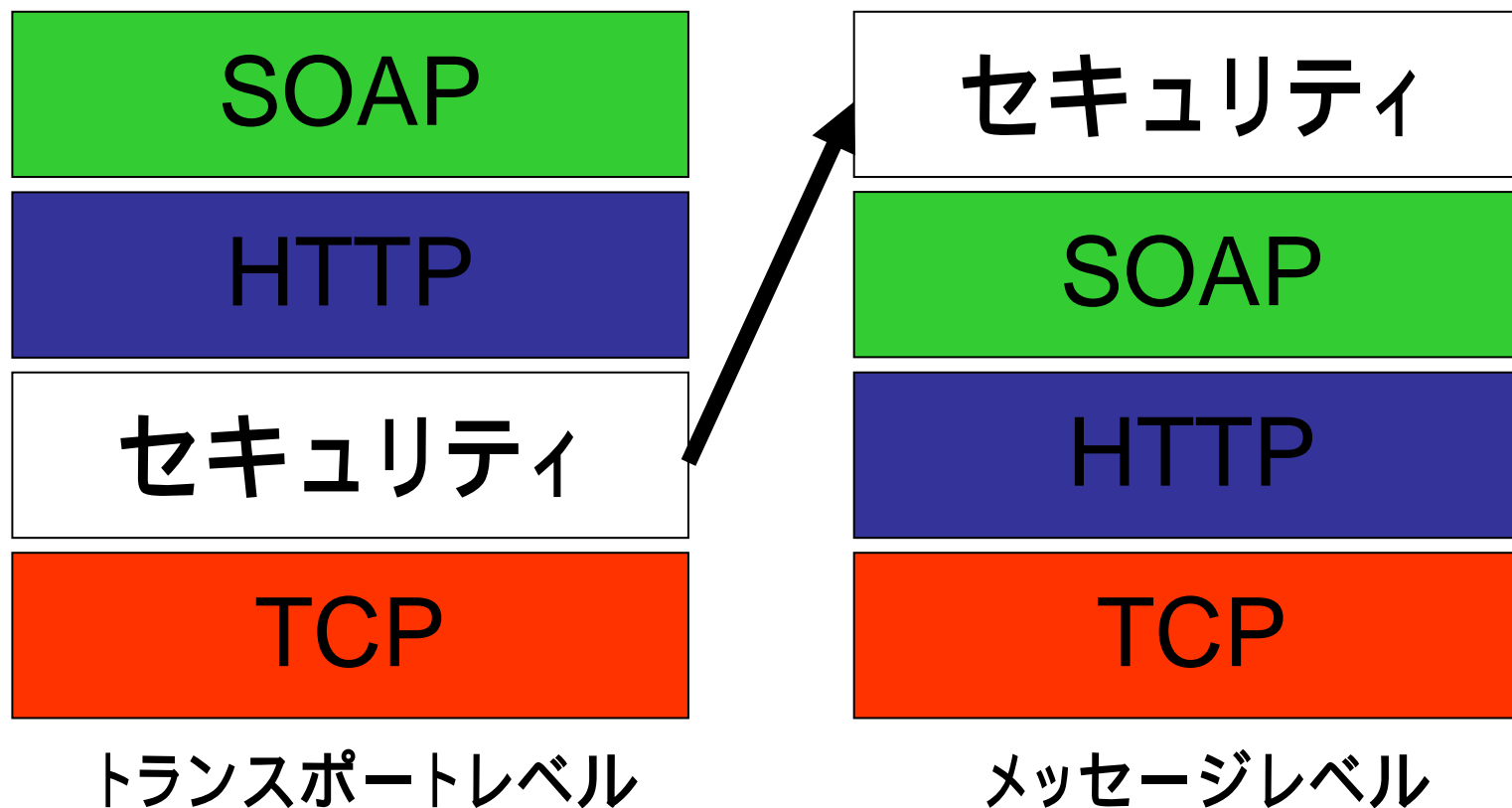
## ● WS-SecurityとSecure Conversation Serviceを使用



By Courtesy of Hidemoto Nakada



# トランスポートレベル vs メッセージレベル



By Courtesy of Hidemoto Nakada

# トランスポートレベル vs メッセージレベル

---

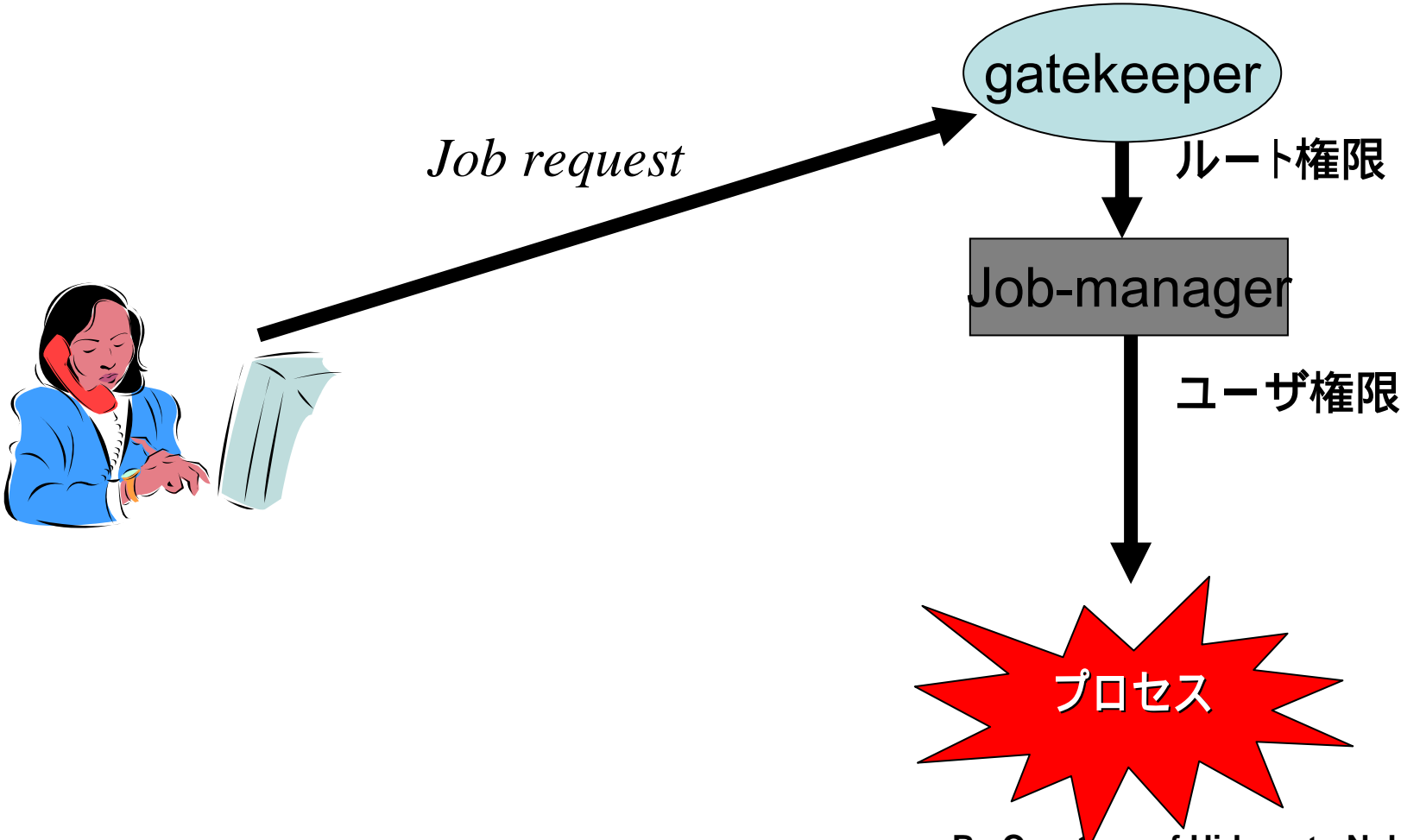
## ● トランスポートレベル

- ▶ ソケットライブラリ以下にセキュリティ機構を隠蔽
- ▶ 高速、プログラマの負荷が小さい
- ▶ × 通信のリダイレクトが難しい

## ● メッセージレベル

- ▶ SOAPメッセージレベルで通信を暗号化
- ▶ 通信のリダイレクトが容易
- ▶ × 低速

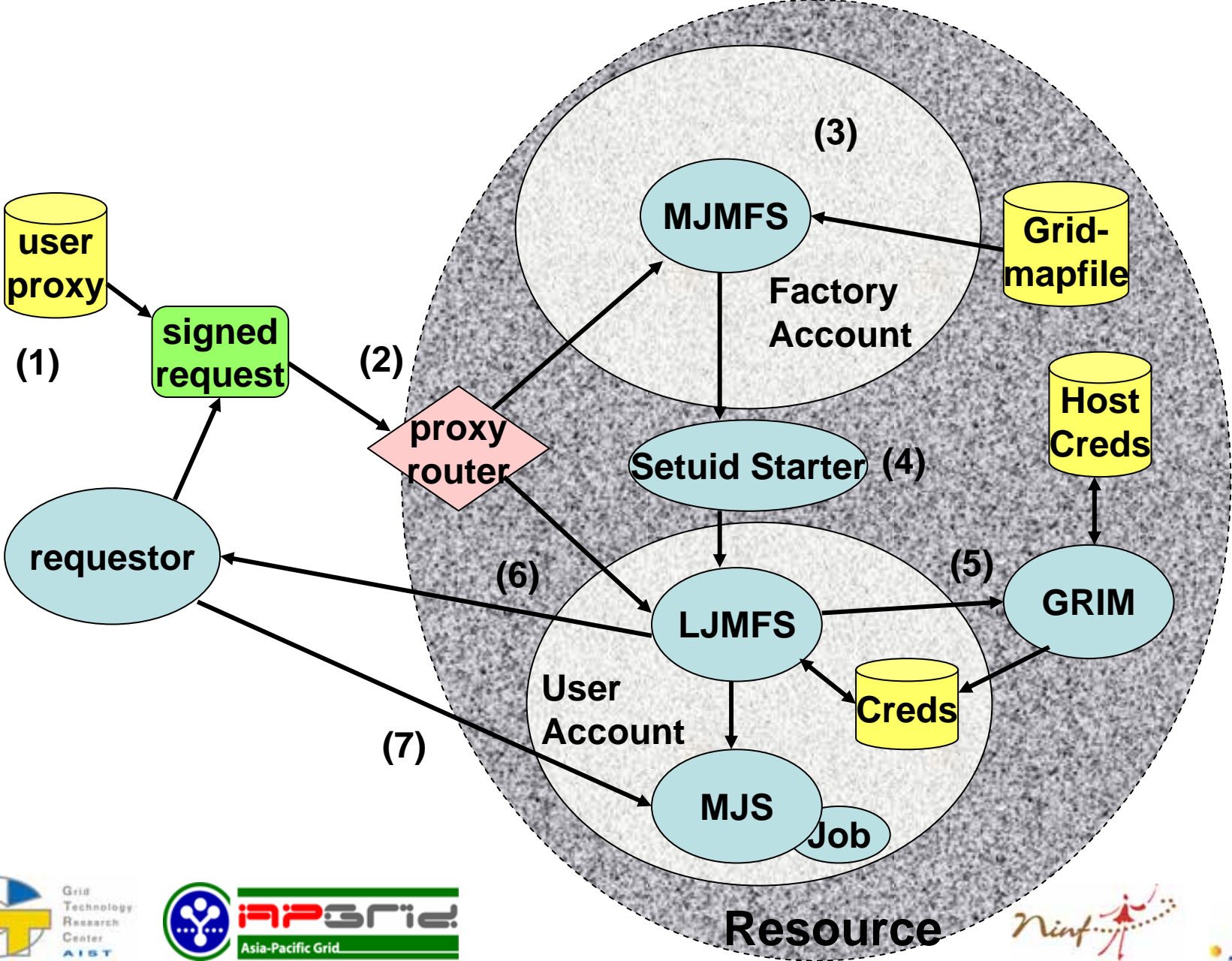
# GT2のGRAM



By Courtesy of Hidemoto Nakada



# GT3のGRAM



---

# GRIP





## Security Basics

- Public/private key infrastructure to establish connections
- X509v3 certificates (incl. extensions)

### UNICORE:

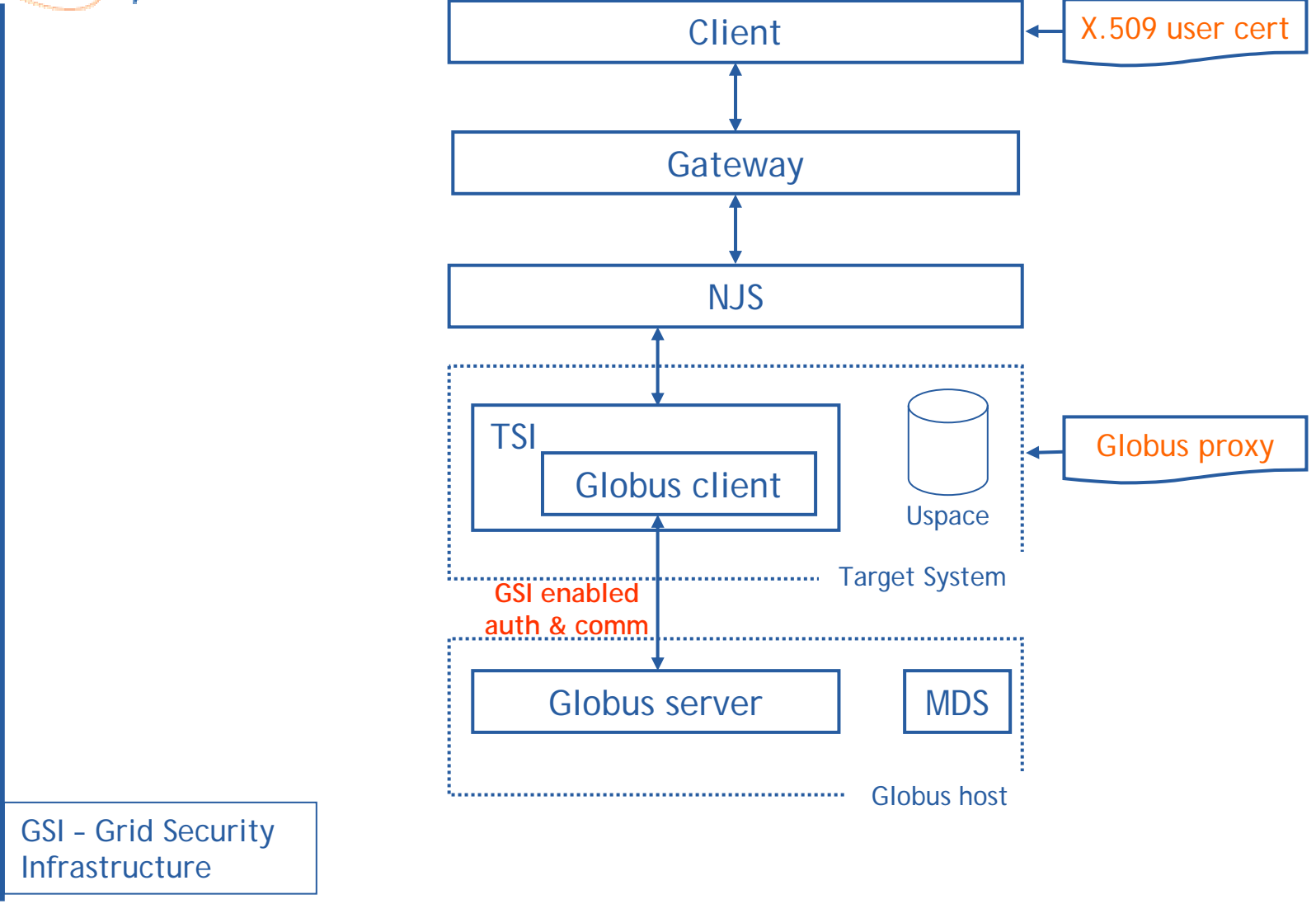
- End-to-end security, jobs signed
- Keys & certificates are stored in a keystore at the client side

### Globus:

- Transitive trust, proxy certificates
- Keys & certificates are stored on the file system



# Interfacing Globus through UNICORE



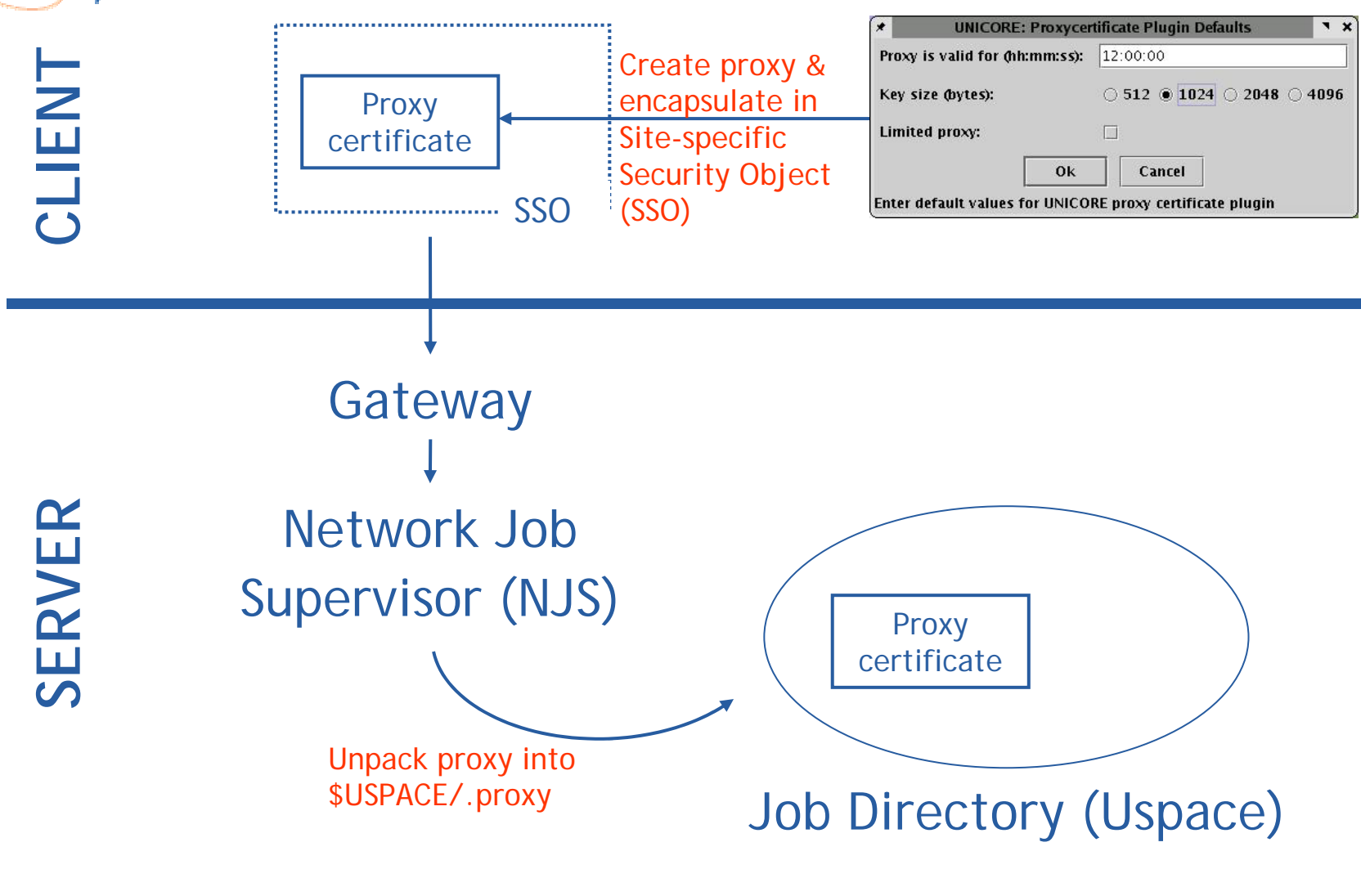
By courtesy of Philipp Wieder



## Security Interoperation

- Proxy Certificate Plugin generates a proxy from the UNICORE user's private key
- The proxy certificate is transferred to the user's Uspace
- Proxy used for every task involving GSI enabled authentication & communication
- Configure Globus client (TSI) to use proxy
- Configure Globus server to trust signing CA

Details next slide ...



---

# Grid Portal



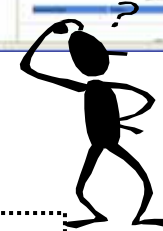
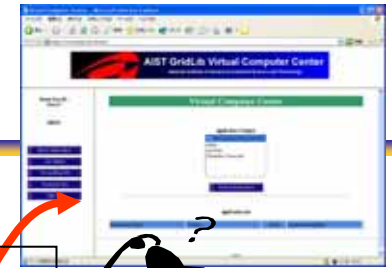
# Gridポータルにおける認証

---

- Proxy証明書をWebサーバ上に作成する方法に応じたいくつかのアプローチ
  - ▶ (旧)Grid Port (NPACI HotPage)
  - ▶ MyProxy
  - ▶ AI ST Grid Lib Portal
  - ▶ AI ST Grid PSE Builder



# GridPort / HotPage (old version)



SSL connection

Grid Port

user

globusrun

invoke  
grid-proxy-init

HTTP server  
+  
Servlet  
(Apache +  
Tomcat)

証明書

秘密鍵

証明書

秘密鍵

証明書

秘密鍵

証明書

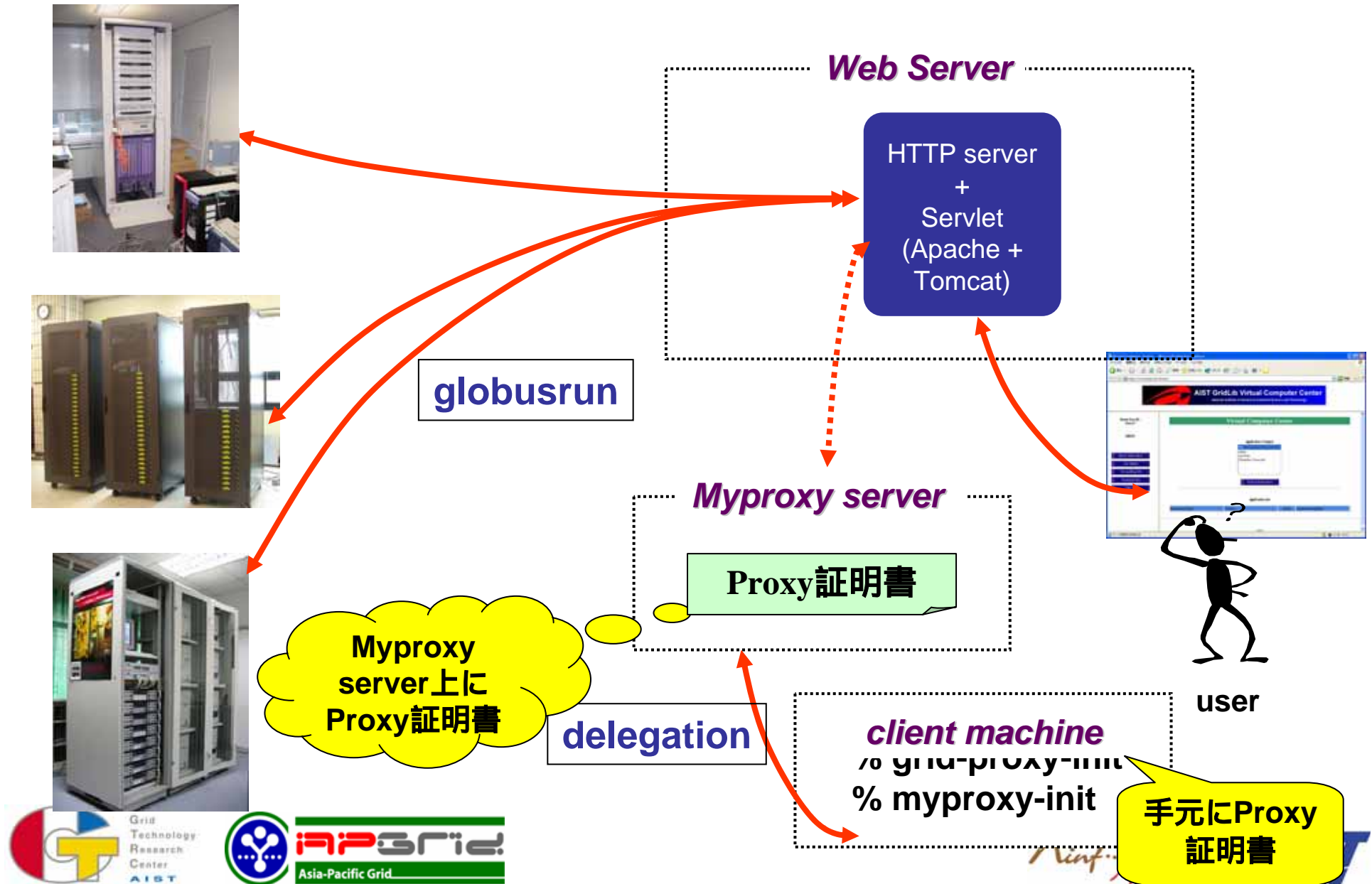
秘密鍵

ユーザはWebブラウザから秘密鍵を復号するためのパスフレーズを入力し、grid-proxy-initコマンドを実行

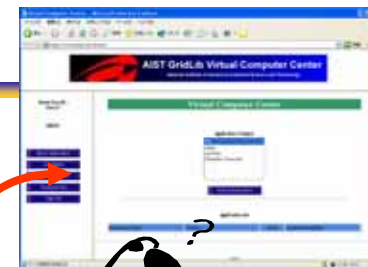
すべてのユーザの秘密鍵と証明書はWebサーバ上に置かれている



# MyProxyサーバを使った方法



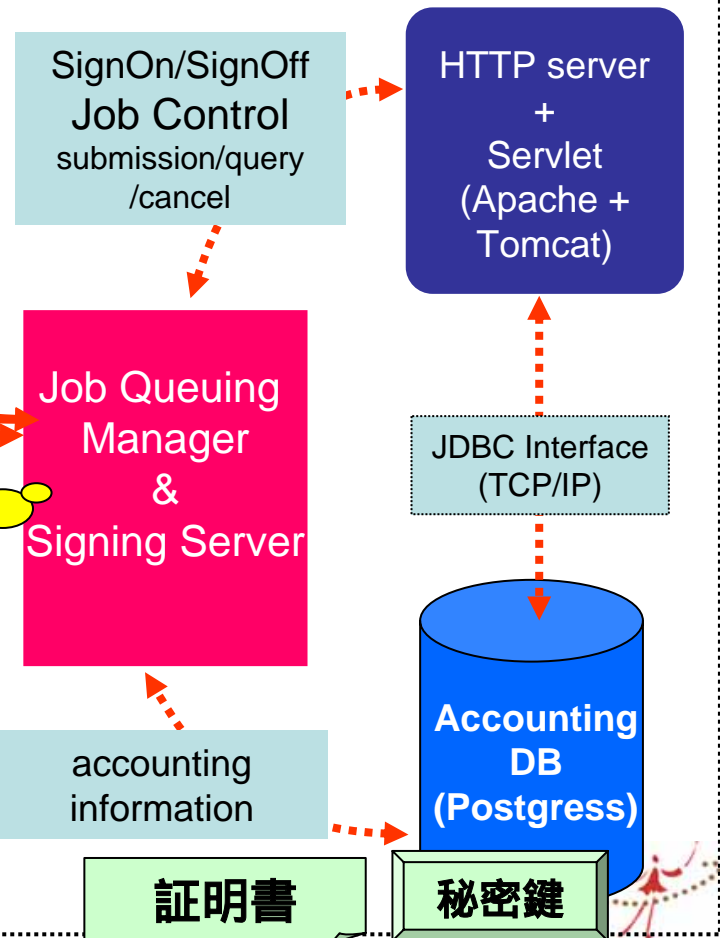
# AIST GridLib Portal



user

client auth.

## AIST GridLib Portal

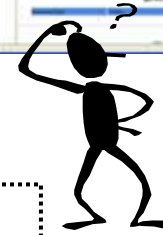
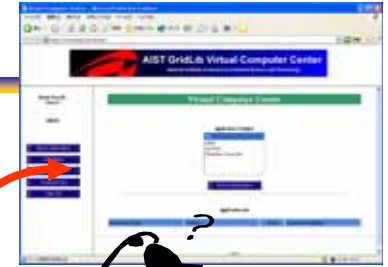


globusrun

Signing Server  
がgrid-proxy-  
initを内部で実行  
(現状はシングル  
アカウント)



# Grid PSE Builder



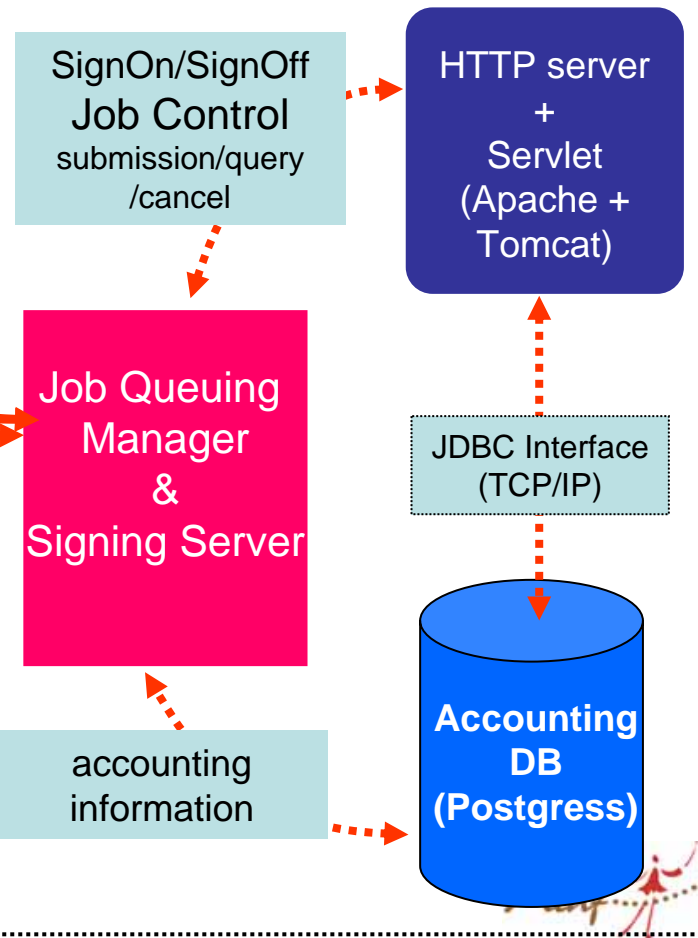
user

client auth.

AIST GridLib Portal

クライアントと  
Webサーバとの  
間でdelegation  
が行われる  
(JWS/Applet)

globusrun



# 現状、動向および課題

---

- OGSA Security Architecture?
- Proxy証明書でいいのか？
- 認証以外にもたくさん問題
- Federation
- 技術よりポリシー



# 参考文献

## ● OGSA SEC

- ▶ Nataraj Nagaratnam, et.al. “Security Architecture for Open Grid Services”, GWD-I (draft-ggf-ogsa-sec-arch-01), <http://www.ggf.org/ogsa-sec-wg/>
- ▶ Fank Siebenlist, et.al. “OGSA Security Roadmap” GWD-I (draft-ggf-ogsa-sec-roadmap-01), <http://www.ggf.org/ogsa-sec-wg/>

## ● UNICORE

- ▶ T. Gross-Walter, et.al. “An Analysis of the UNICORE Security Model”, <https://forge.gridforum.org/projects/ggf-editor/document/GFD.18/en/1>
- ▶ UNICOREpro Client User Guide

## ● Globus Toolkit

- ▶ Steve Tuecke et.al. “Internet X.509 Public Key Infrastructure Proxy Certificate Profile”, GWD: draft-ggf-gsi-proxy-04, [http://www.gridforum.org/2\\_SEC/GSI.htm](http://www.gridforum.org/2_SEC/GSI.htm)
- ▶ Von Welch et.al. “Security for Grid Services”, In Proceedings of HPDC-12, 2003
- ▶ Pearlman, L. et.al. “A Community Authorization Service for Group Collaboration”  
In Proceedings of International Workshop on Policies for Distributed Systems and Networks, 2002

