

Grid Programming (1)

Osamu Tatebe
University of Tsukuba

Overview

Grid Computing

- ▶ Computational Grid
- ▶ Data Grid
- ▶ Access Grid

Grid Technology

- ▶ Security - Single Sign On
- ▶ Information Service
- ▶ Data management
- ▶ Widearea Data Transfer
- ▶ Resource Management

Open Grid Forum (OGF)

- ▶ <http://www.ogf.org/>

What is Grid Technology?

- Is it a technology to connect among supercomputers and to share them?

- ▶ <http://www.itbl.jp/>

- Is it SETI@Home, UD Cancer research project, or Fight AIDS@home?

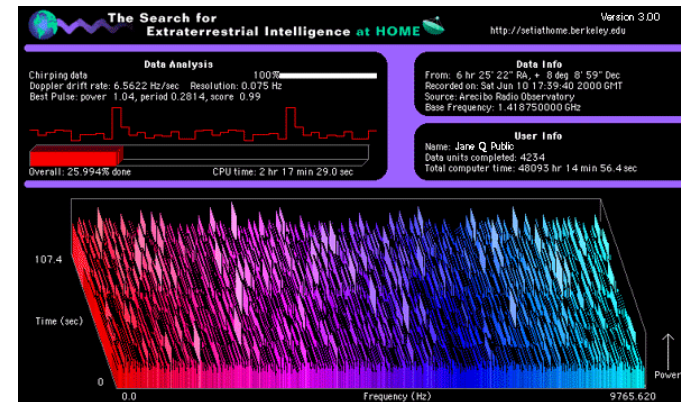
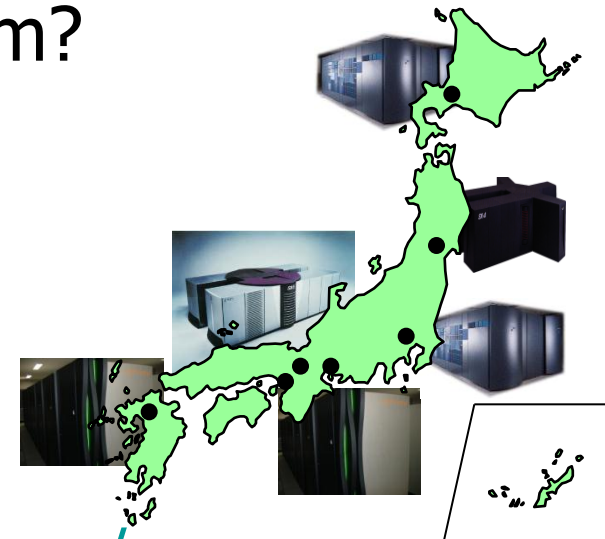
- ▶ <http://setiathome.ssl.berkeley.edu/>

- ▶ <http://members.ud.com/projects/cancer/>

- ▶ <http://www.fightaidsathome.org/>

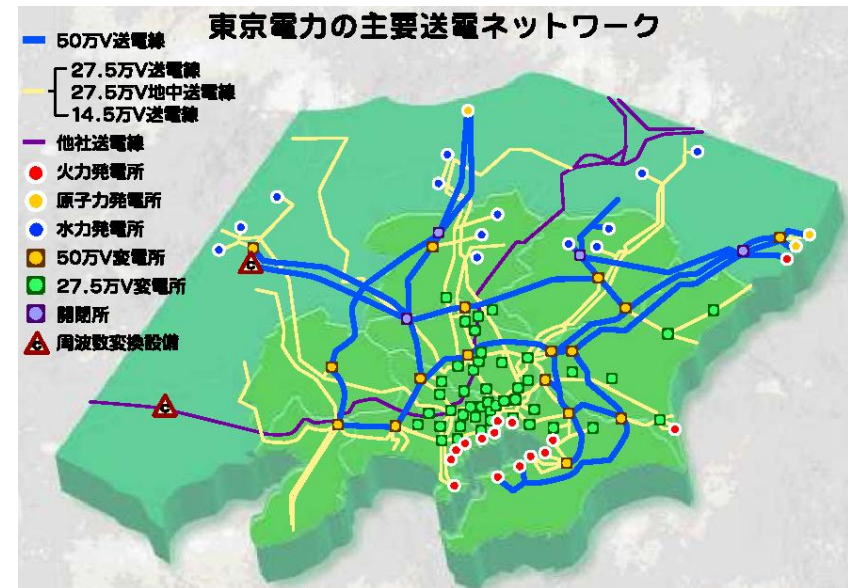
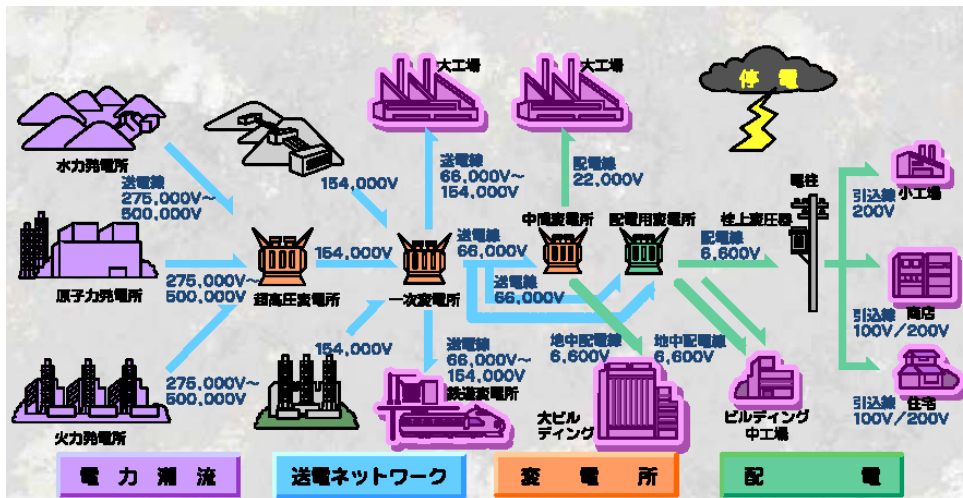
- Is it a next generation Internet technology?

- ▶ IPv6, QoS, IPsec, . . .



Grid

- Used after the middle of '90
- From similarity to Electric Power Grids
 - Electric Power Grids provides enough power, using another route in case of trouble. It is monitored, controlled, and operated.
 - Quite important invention besides power generator, electronic products

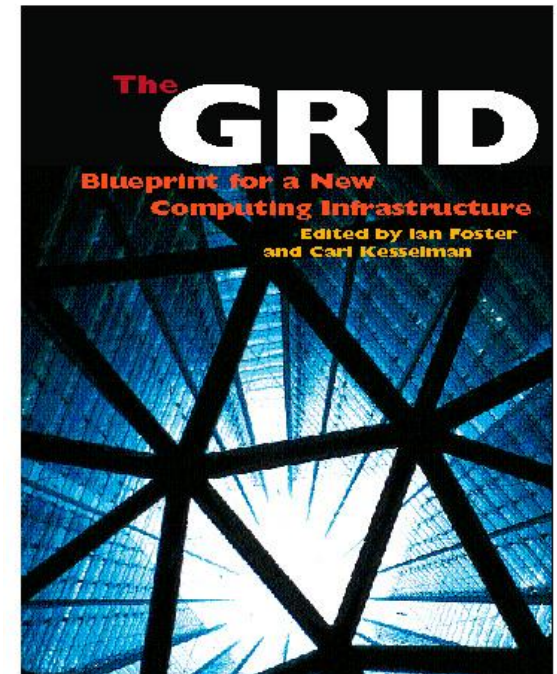


Definition of Grid in 1999

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

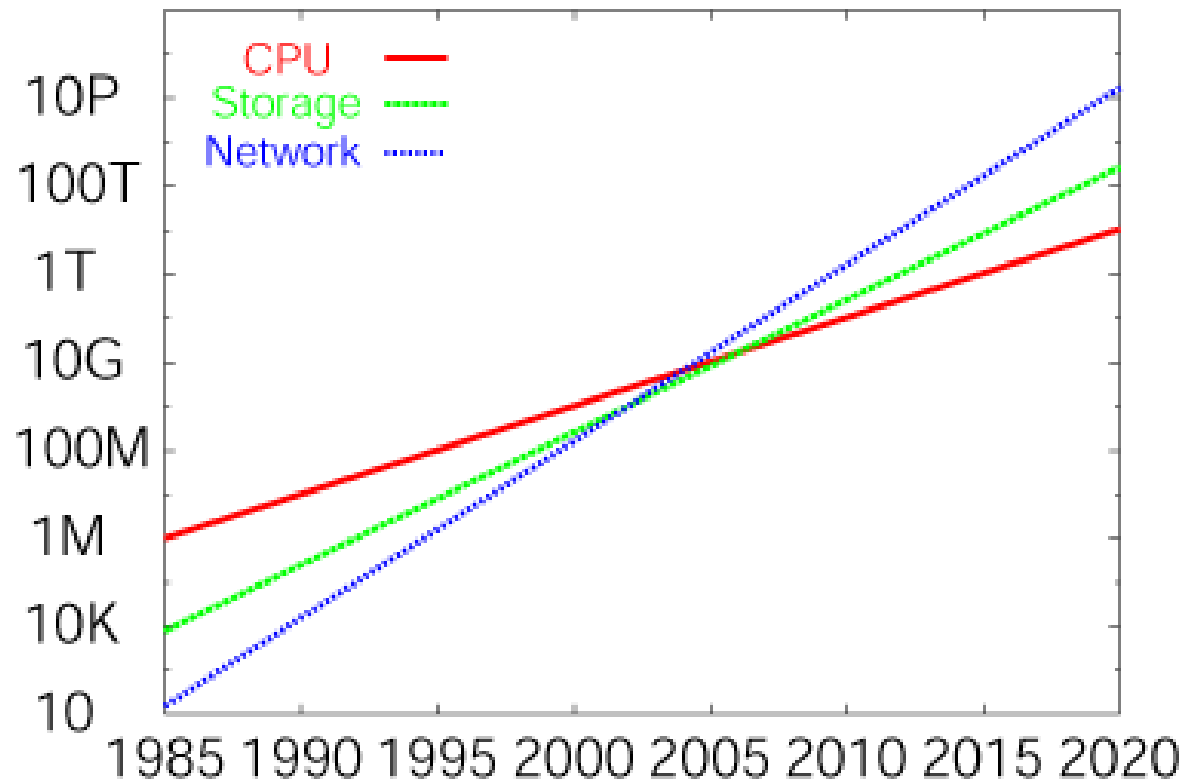
From "The GRID – Blueprint for a New Computing Infrastructure", 1999

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



Technology Trend : Grid is feasible!

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



CPU << Storage << Network

Network is free!

- 100 times in each 5 years
- We can use not only local resources, but also resources in wide area
 - ▶ Computers, storage, visualization devices, super computers, special purpose machines, experiment devices, researchers, applications, libraries, data, ...

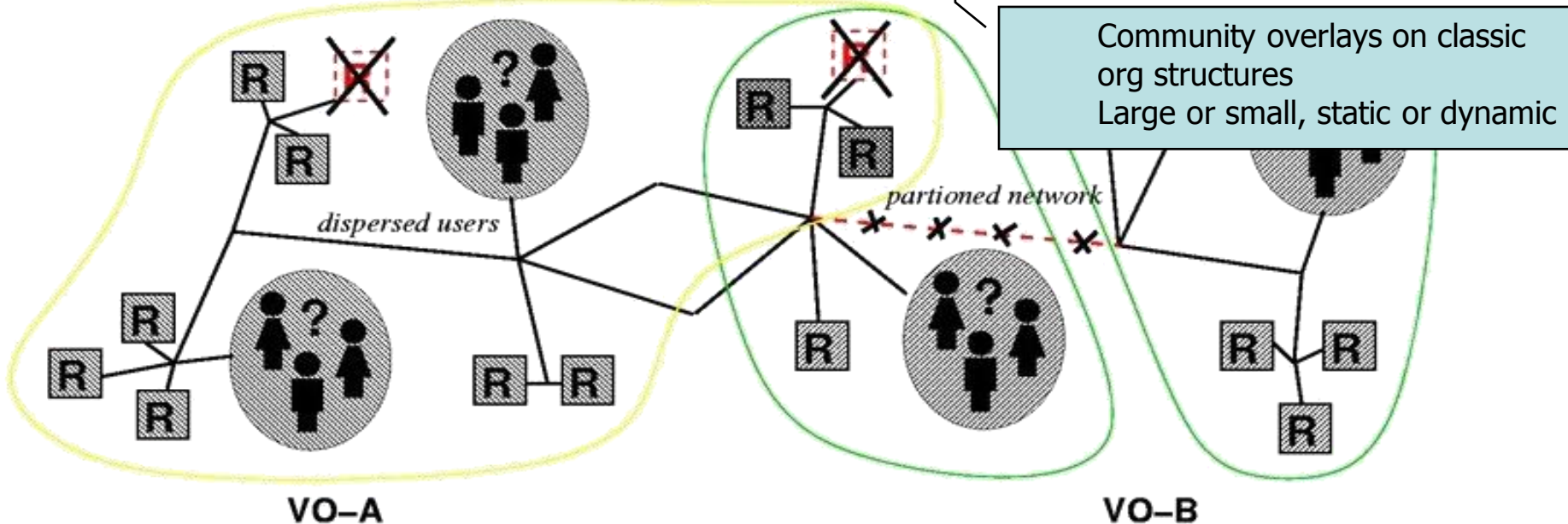
The Grid (2000)

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, ...



Virtual organization (VO) and Grid

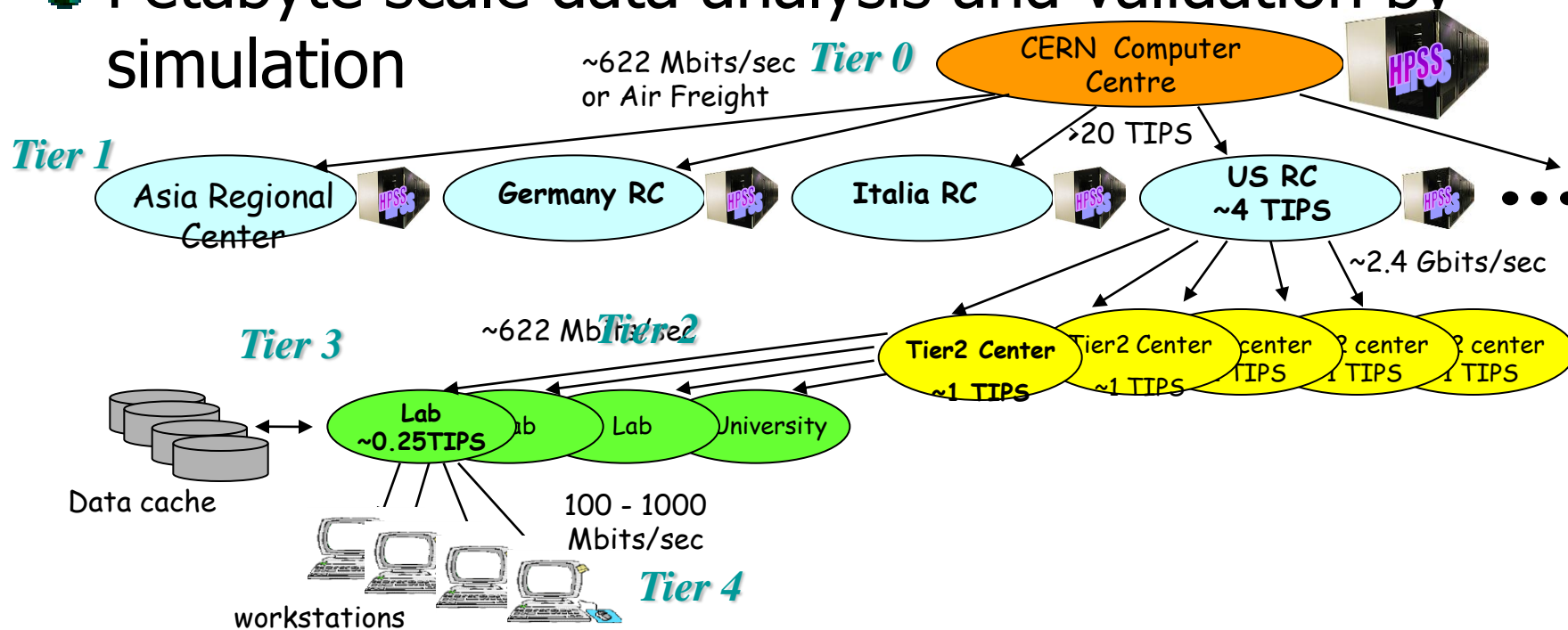
- **A set of **dynamic** and **flexible** resources**
 - ▶ Including several institutes managed independently
 - ▶ One institute may belong to several VOs
- **Large and small**
- **Secure and **controlled** resource sharing**
 - ▶ Computers, storage, sensor, experiment device, application, data, ...
- **Some restrictions**
 - ▶ Idle time, only morning, a part of resources, limited programs, ...
- **Client-server, P2P**
- **Technology to construct a VO flexibly, and to share resources securely**
 - ▶ Secure **authentication** and proper **authorization**
 - ▶ Resource **access** protocol, **discovery** protocol
 - ▶ **Fault tolerance**
 - ▶ **Common** protocol

Several scenarios

- A small VO consisting of companies A and B
- Company A has a supercomputer, Company B has a visualization device
- Both employees shares these resources securely
- A customer would like to introduce a ventilation system
- It is not clear which location is efficient to install due to a complex room structure
- Use an ASP for Computational fluid dynamics simulation, store the result in an SSP, and send it to a house company

Large scale scenario

- Large Hadron Collider (LHC) experiment
- 3000 researchers in 20 countries
- Hierarchical regional center model
- Petabyte scale data analysis and validation by simulation



Grid Architecture and standard

Requirement for Grid Technology

- Support **various security policy** required by resource providers and users
- Enough flexibility for **various resources** and sharing policy
- Scalability for **many resources, many users, many programs**
- **Dynamic resource management**
 - ▶ Dynamic extensibility of resources
 - ▶ Fault tolerance and self organization
 - Ⓢ Resource status is often changed
- Efficient execution for **large-scale data intensive computing and large-scale simulation**
 - ▶ HPC, HTC
 - ▶ Support high bandwidth and long latency
- **Standard protocol to share resources flexibly among different groups**
 - ▶ Support various resources, policies, protocols
- **Common software stack to avoid duplicate development**

Standard based Grid Architecture

Development of Standard Protocol, Standard Service

- ▶ Common access protocol to remote resources
- ▶ Based on existent protocols

Development of Grid API and SDK

- ▶ Interface for Grid Protocol and Grid Service
- ▶ Higher level of abstraction to develop applications

Success story: Internet

- ▶ HTTP and HTML
- ▶ TCP/IP, telnet, ftp, mail, . . .

Important points

● **Based on Internet Protocol, Web Services**

- ▶ TCP/IP, WSDL, SOAP, etc.

● **Define minimum services required for the Grid**

- ▶ Grid Security

- ▶ Addressing – WS-A (WS-Addressing)

@ <http://www.w3.org/Submission/ws-addressing/>

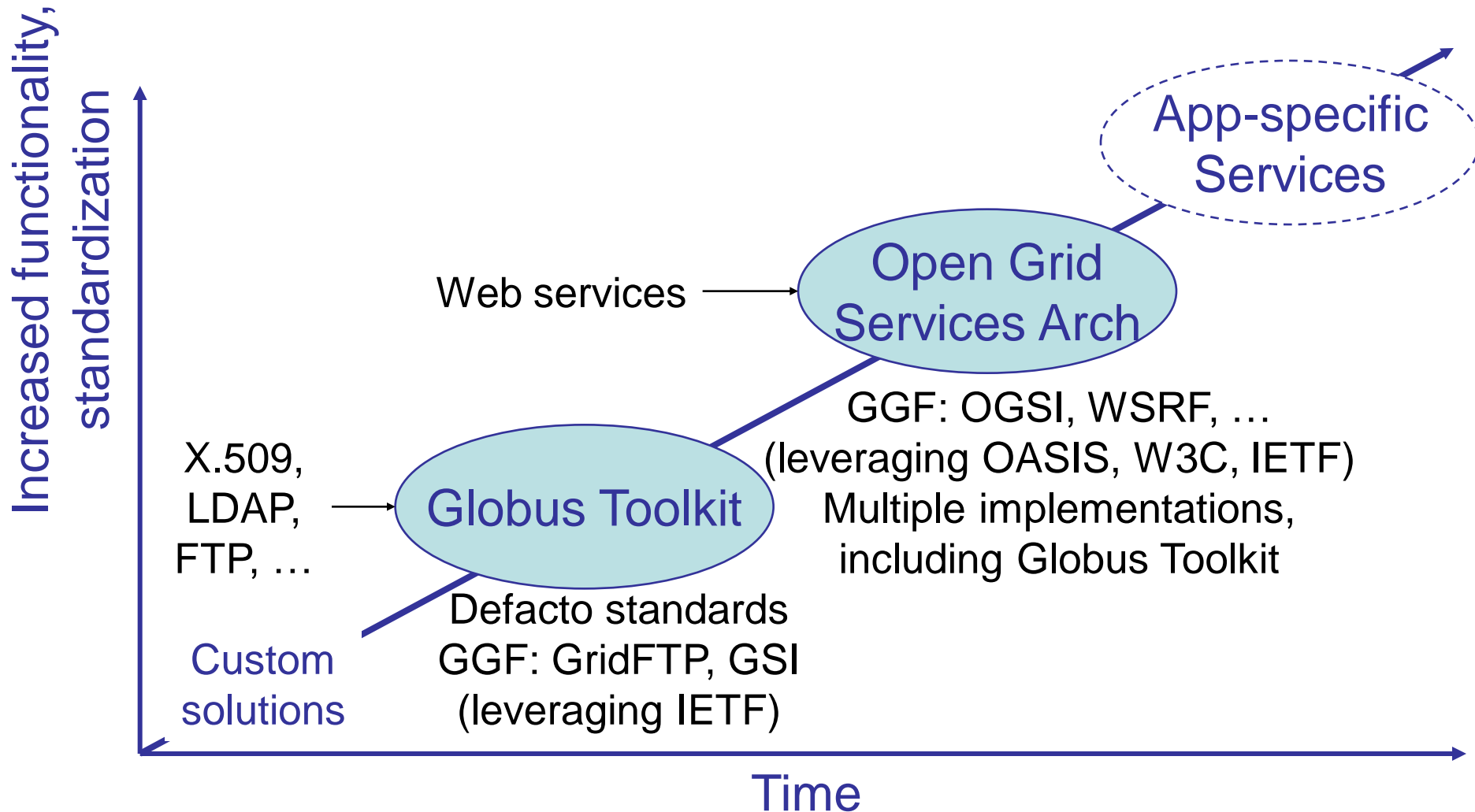
- ▶ State – WSRF (WS Resource Framework)

@ <http://www.oasis-open.org/committees/wsrf/>

- ▶ Notification – WS-N (WS-Notification)

@ <http://www.oasis-open.org/committees/wsn/>

Evolution of the Grid



Papers: Grid technology

- 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>

Papers: 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/>

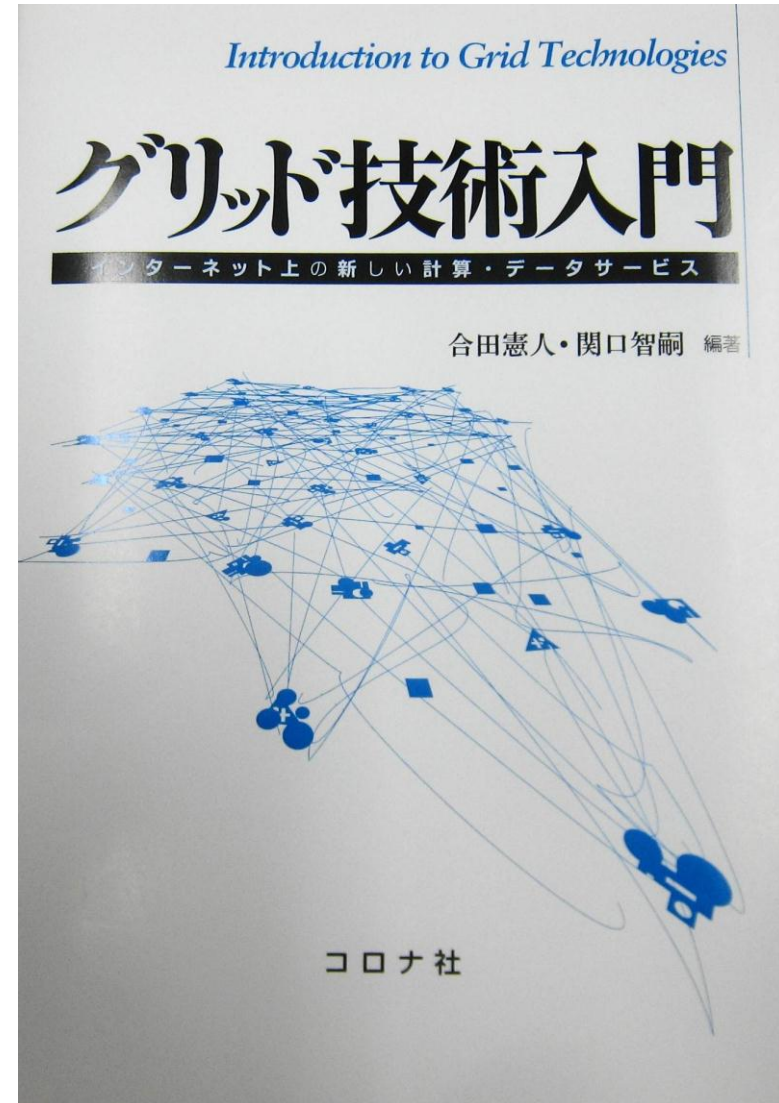
Papers: Grid Software

- 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>

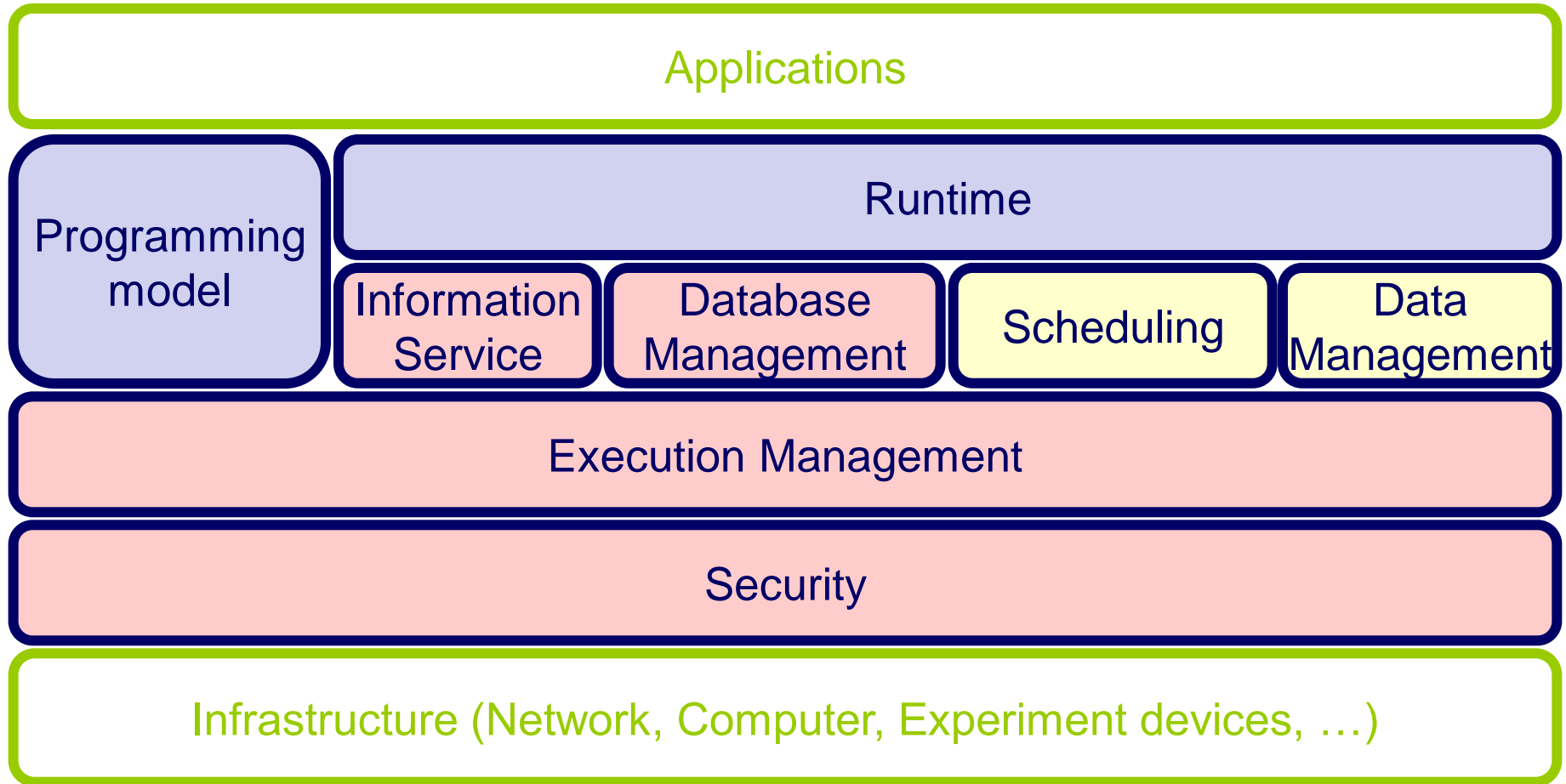
Grid Technology

Introduction to Grid Technology

- **New Computational and Data Service in the Internet**
- **Editors: Kento Aida, Satoshi Sekiguchi**
- **Corona publishing, 2008**
- **ISBN : 978-4-339-02426-5**



Grid Technology (1)



Grid Technology (2)

- Grid Security Infrastructure (GSI)
- Grid Information Service (GRIS)
- Widearea data transfer (GridFTP)
- Resource Manager (Grid inetd, GRAM)
- Aggregation of Information Service (Grid Index Information Service, GIIS)
- Resource broker (Condor-G, Nimrod-G)
- Data replica management service
- Co-allocation and co-reservation service
- Workflow management service
- . . .

Grid Security (GSI)

Single Sign On

- ▶ Access authentication and authorization by a single user authentication (pass phrase, one-time password)

Certificate delegation

Limit the delegated certificate

- ▶ Expiration, level of delegations, limited authority
- ▶ Mitigate the damage when it is stolen

Support dynamic service creation

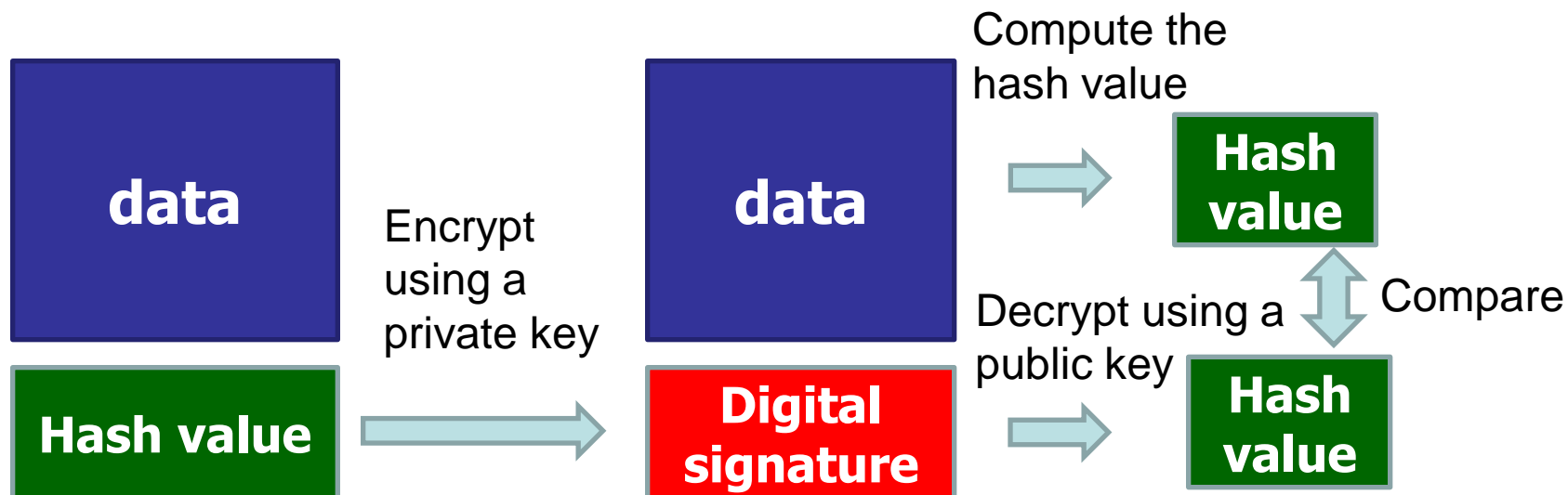
Protect a private key

Public-key Cryptosystem

- Asymmetric key cryptosystem
- A public key e and a private key d
- Plain text – $e \rightarrow$ cryptogram – $d \rightarrow$ plain text
- Computation from e to d is computationally difficult
- A public key not needed to be secret. It is easy to be provided
- Digital signature is required to authenticate a sender and to check a falsification
- Since it is often slow than symmetric key cryptosystem such as DES, it is used to send small messages such as a key of a symmetric key cryptosystem for data transfer of the rest, and credit card information
- [Handbook of Applied Cryptography](http://cacr.math.uwaterloo.ca/hac/), by A. Menezes, P. van Oorschot, and S. Vanstone, CRC Press, 1996
<http://cacr.math.uwaterloo.ca/hac/>

Digital Signature

- Ensure the integrity. The data is not altered or not falsified
- Encrypted hash value of the data
- At a receive side, compare the hash value of the data and decrypted value of the digital signature



Grid Security Infrastructure (GSI)

- **Basically public key cryptosystem + X.509 certificate + TLS (Transport Layer Security)**
- **Mutual authentication and certificate delegation using a proxy certificate**
- **Public key cryptosystem (asymmetric key cryptosystem)**
 - ▶ Public key is used to encrypt the data
 - ▶ Private key is used to decrypt the cryptogram
- **Entity (user, machine, ...) keeps a certificate signed by a certificate authority**
- **X.509 certificate includes**
 - ▶ Subject name of an entity (user ID, host name)
 - ▶ Public key
 - ▶ Issuer (Certificate Authority)
 - ▶ Digital signature signed by the CA
 - Ⓢ Ensure the certificate is issued by the CA
 - Ⓢ Ensure the Subject name
 - Ⓢ Ensure the relationship of the subject name and the public key

Certificate

Subject DN

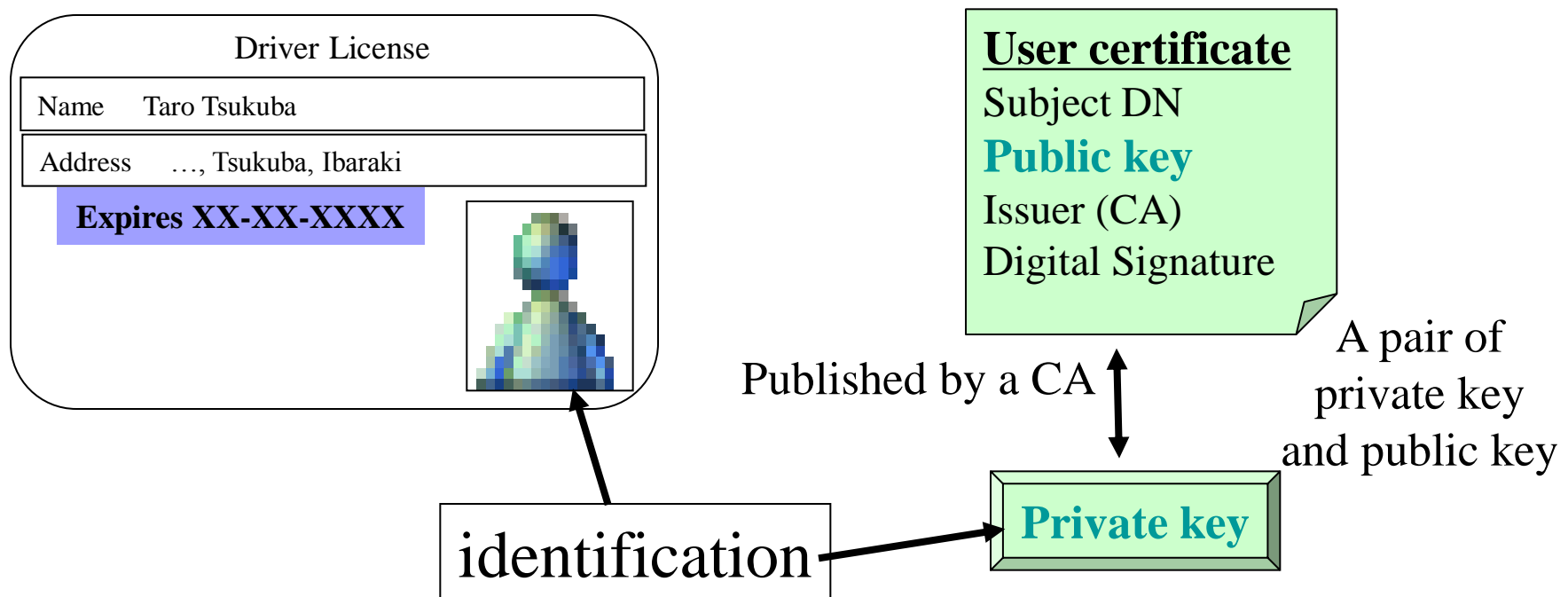
Public key

Issuer (CA)

Digital Signature

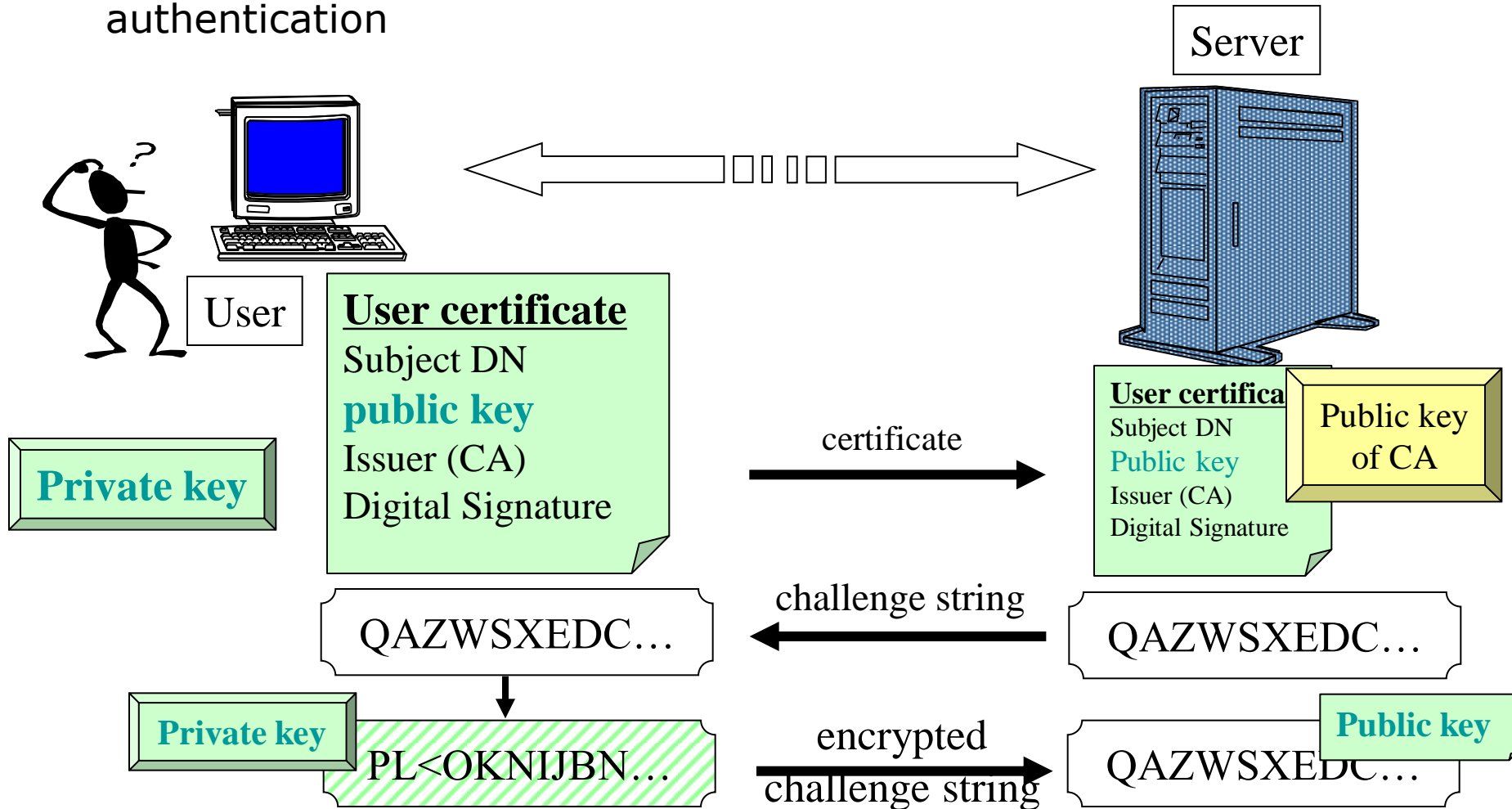
Certificate

- **It is like a drivers license. A picture, a method to identify an entity, corresponds to a private key**
- **Signed by a certificate authority**
- **Whether it is credible or not depends on the CA is dependable**



Authentication by GSI

The following example shows the user authentication, but the server will be authenticated later by the user. Thus it is called mutual authentication



Extension by GSI

🌐 Proxy Certificate Profile

- ▶ Proxy Certificate Profile based on X.509 (RFC 2459)
- ▶ restricted impersonation within a PKI based authentication system.

🌐 Extension of GSS-API (RFC 2743)

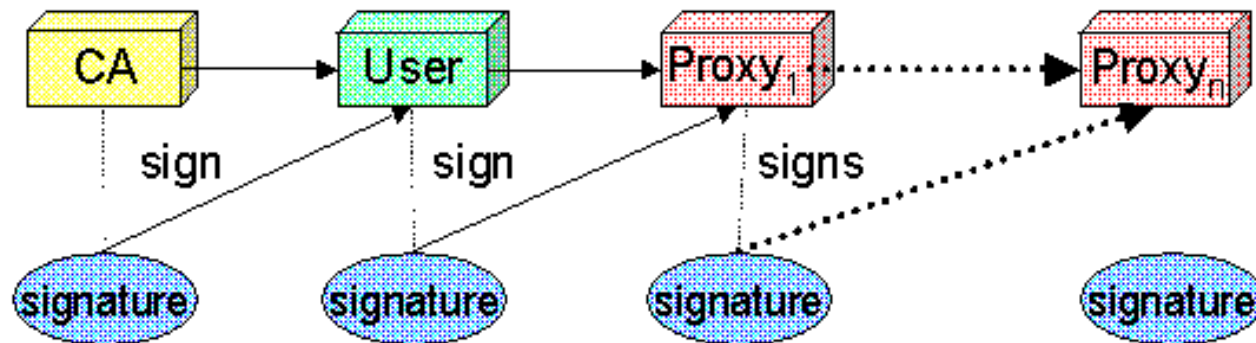
- ▶ Export and import of the credential
- ▶ Delegation at any point of timing
- ▶ Extension of Credential operation
 - 🌀 Limited delegation

🌐 Internet X.509 Public Key Infrastructure Proxy Certificate Profile

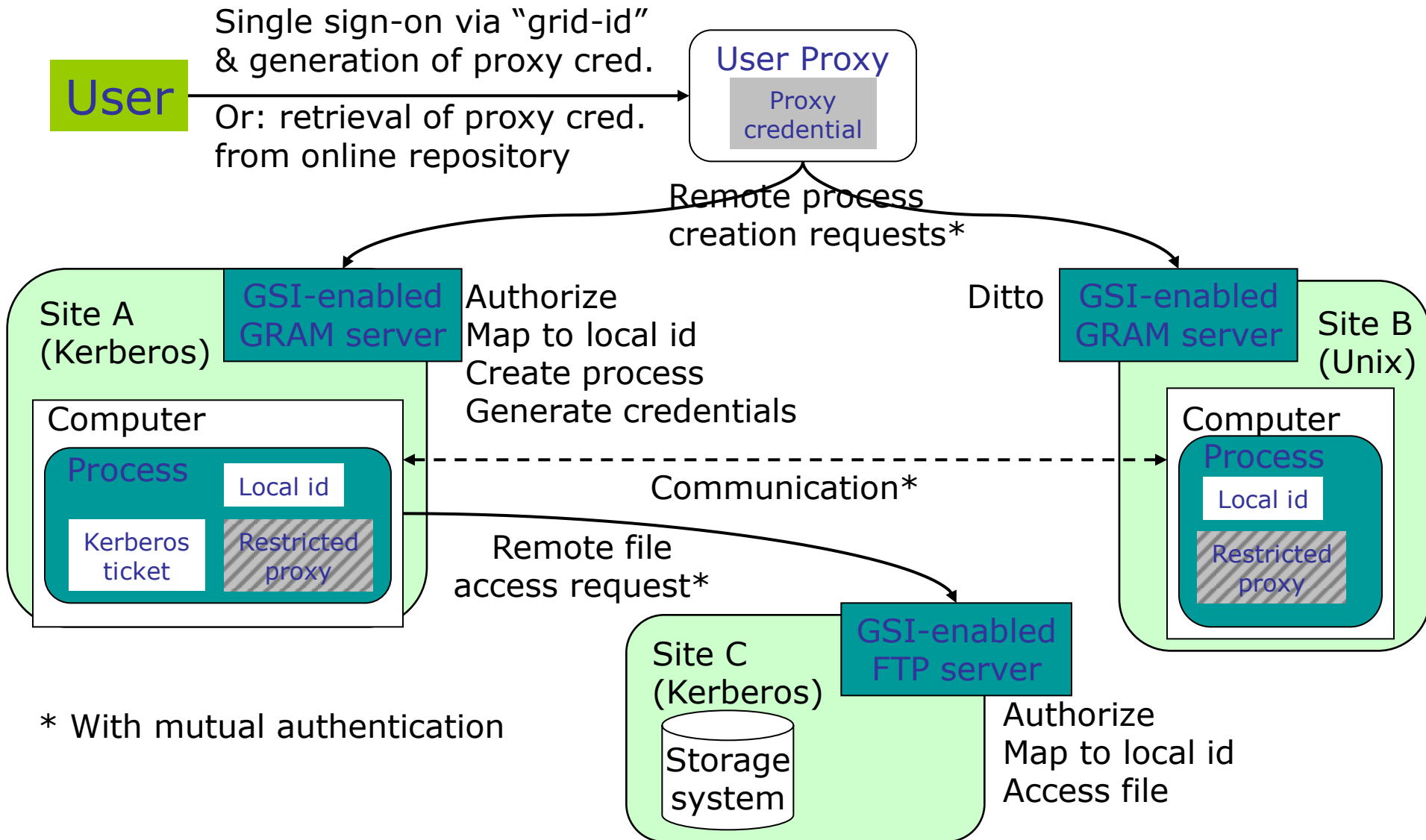
- ▶ RFC 3820 by Grid community – OGF
- ▶ GSS-API Extensions
- ▶ <ftp://ftp.rfc-editor.org/in-notes/rfc3820.txt>

Delegation of the certificate

- A pair of public and private keys are generated, and signed by a user not a CA
 - ▶ Private key is NOT transferred
- Proxy certificate can be validated by the valid user certificate



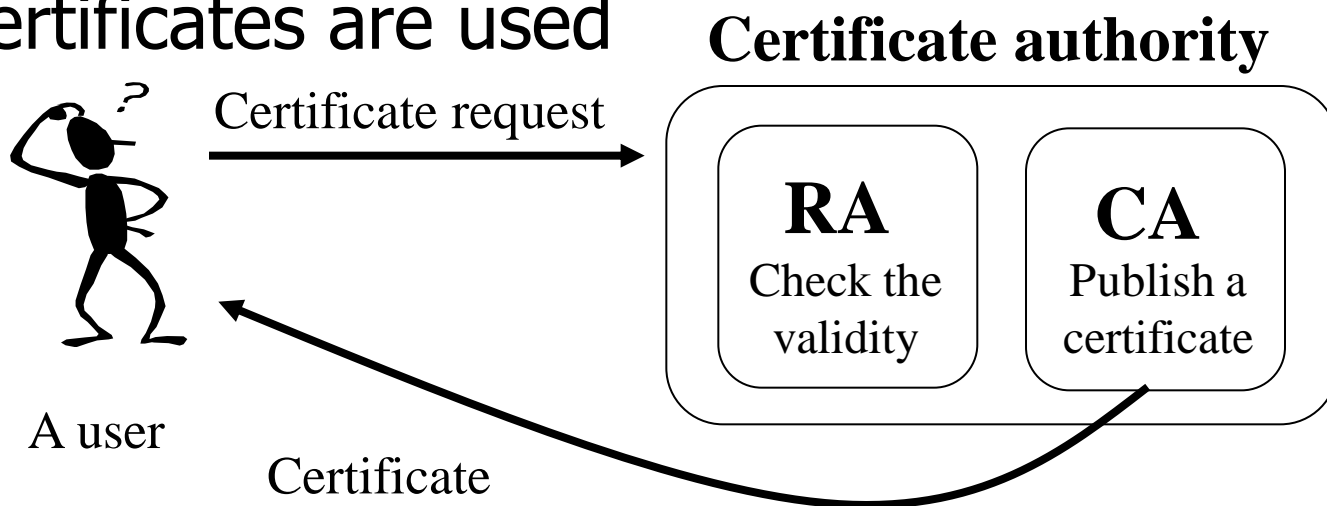
GSI in Action "Create Processes at A and B that Communicate & Access Files at C"



Certificate and Certificate Authority

🌐 Certificate authority

- ▶ A third party to publish a certificate
- ▶ Two roles: Registration Authority (RA) and CA
 - Ⓜ RA: identify users and computers
 - Ⓜ CA: publish a certificate
- ▶ No way to know how and where published certificates are used



Initial Setting for Certificate Authority (In case of Globus Toolkit)

🌐 Setup for a certificate authority

- ▶ `$GLOBUS_LOCATION/setup/globus/setup-simple-ca`
 - Ⓜ Subject DN for CA
 - ⚙ `cn=CA, ou=CS, o=Univ Tsukuba, c=JP`
 - Ⓜ Email address
 - Ⓜ Expiration date
 - Ⓜ Passphrase for a private key
 - ⚙ It is used to sign a certificate requested by a user
 - ⚙ 'space' cannot be used
- ▶ `$GLOBUS_LOCATION/setup/globus_simple_ca_CA_Has h_setup/setup-gsi -default`
 - Ⓜ The public key of the CA is stored at `/etc/grid-security/certificates`

How to obtain a host certificate

🌐 Request for a host certificate

- ▶ `grid-cert-request -host <hostname>`
 - 📍 `/etc/grid-security/hostkey.pem` (private key)
 - 📍 `/etc/grid-security/hostcert_request.pem`
 - 📍 `/etc/grid-security/hostcert.pem` (empty file)

🌐 Ask RA to identify yourself

🌐 Send `hostcert_request.pem` to CA, and ask to be signed

- ▶ `grid-ca-sign -in hostcert_request.pem -out signed.pem`

🌐 Receive the signed `hostsIGNED.pem`, and store it at `/etc/grid-security/hostcert.pem`

🌐 Display a content of the certificate

- ▶ `openssl x509 -in hostcert.pem -text`

How to obtain a user certificate

🌐 Request for a user certificate

▶ grid-cert-request

📍 ~/.globus/userkey.pem (private key)

📍 ~/.globus/usercert_request.pem

📍 ~/.globus/usercert.pem (empty file)

🌐 Ask RA to identify yourself

🌐 Send usercert_request.pem to CA, and ask to be signed

▶ grid-ca-sign -in usercert_request.pem -out signed.pem

🌐 Receive the signed signed.pem, and store it at ~/.globus/usercert.pem

Authorization by GSI

Register to Grid-mapfile

- ▶ `Grid-mapfile-add-entry -dn "/C=JP/O=Univ Tsukuba/OU=CS/OU=tatebe.net/CN=Osamu Tatebe" -In tatebe`
 - Ⓜ Add an entry to `/etc/grid-security/grid-mapfile`

Setting of GSI-enabled OpenSSH

- **Copy `$GLOBUS_LOCATION/sbin/SXXsshd` to `/etc/init.d/gsisshd`**
- **`service gsisshd start`**

Proxy Certificate and login

🌐 **Create a proxy certificate**

- ▶ `grid-proxy-init [-debug] [-verify]`

🌐 **Display the certificate**

- ▶ `grid-proxy-info`

🌐 **Login using GSI authentication**

- ▶ `gssh hostname`
- ▶ User proxy certificate will be delegated

🌐 **FTP using GSI authentication**

- ▶ `gsisftp hostname`

Papers: Grid Security

- 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>

Information Service

- **Discovery, monitoring, planning, basic mechanism for adaptive applications**
- **Various, many, dynamic, geographically distributed resources**
- **Fault tolerance**
 - ▶ Network disconnectivity and node failure are the norm not exceptions
- **Information**
 - ▶ IP address, administrator
 - ▶ CPU, OS, software
 - ▶ Network bandwidth, latency, protocol, logical topology
 - ▶ CPU load, network load, disk usage, load
 - ▶ . . .

Usage Scenario of Information Service

Service discovery service

- ▶ Find a new service

Super scheduler

- ▶ Select appropriate computational resources depending on system configuration, CPU load, ...

File replica selection service

- ▶ Choose most appropriate file copy

Adaptive application agent

- ▶ Change application behavior depending on runtime resource situation

Failure discovery service

- ▶ Find too much load, and failure

Performance monitoring

- ▶ Examine a bottleneck of performance

Requirement (1)

Distribution of information providers

- ▶ All information is old due to the distribution
- ▶ Need the confidence of the information
 - Ⓜ Timestamp, expiration date, ...
- ▶ Transfer the information as soon as possible
- ▶ Generally speaking, no need to provide consistent view of the global status
 - Ⓜ If it provides, the system does not scale to the number of providers

Focus on efficient information transfer from a single source

Requirement (2)

● Cope with failure

- ▶ Resources and network tend to fail
- ▶ Should be fault tolerant
 - Ⓞ A single failure should not prevent from collecting information of other resources
 - Ⓞ Provided information may not be complete, or inconsistent

● Information service should be distributed and not centralized as much as possible

- ▶ Increase possibility to obtain information of available resources

● Should assume failure is not an exception but the norm

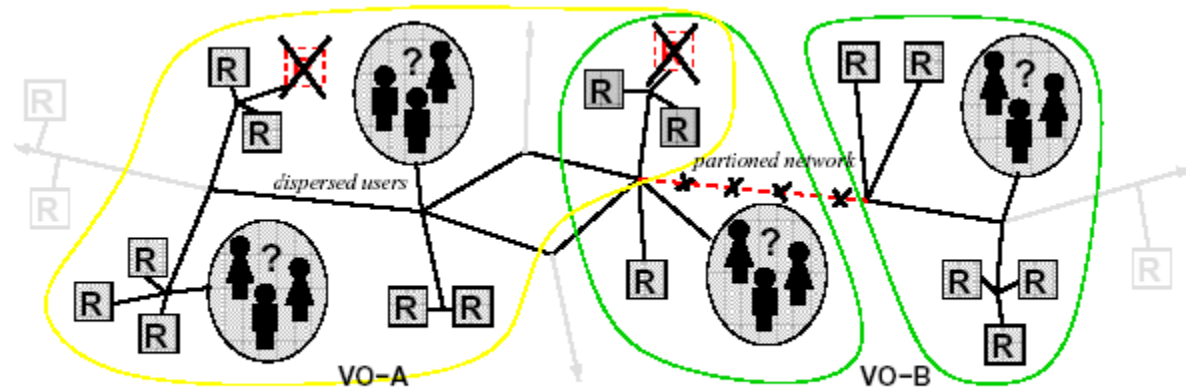


Figure 1. Distributed virtual organizations. Users in VO-A and VO-B have access to partially overlapping resources. While VO-B is split by network failure, it should operate as two disjoint fragments.

Requirement (3)

Variation of information service component

- ▶ There are various kinds of resources. Some may require a special requirement to discover and to monitor
- ▶ Various kinds of discovery and monitoring methods
- ▶ Various kinds of access policy since resources are located in several administration domains
 - Ⓜ Access control

Globus MDS Approach

Based on LDAP

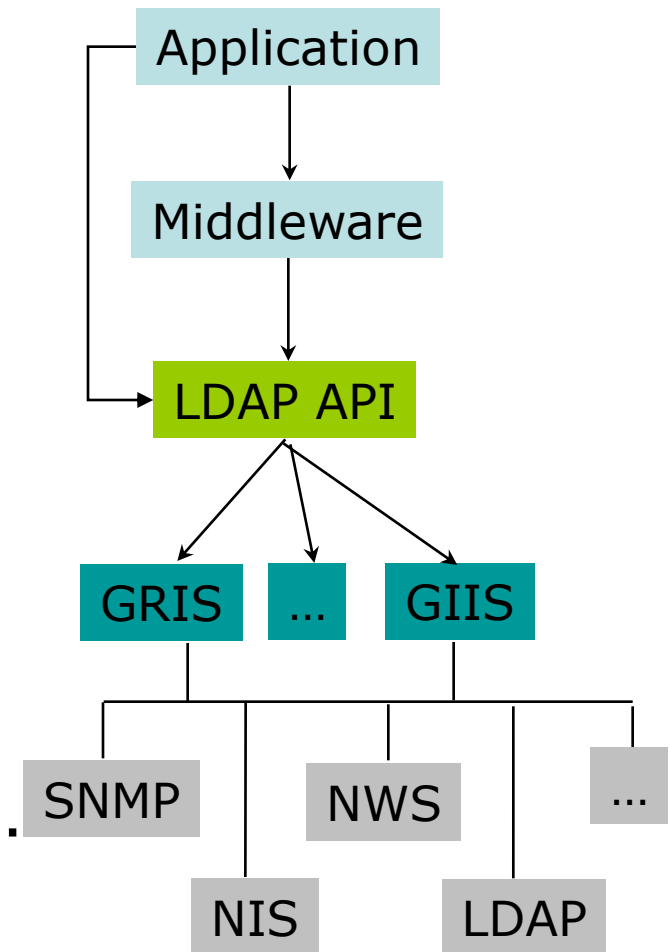
- ▶ Lightweight Directory Access Protocol v3 (LDAPv3)
- ▶ Standard data model
- ▶ Standard query protocol

Globus Toolkit schema

- ▶ Host-centric representation

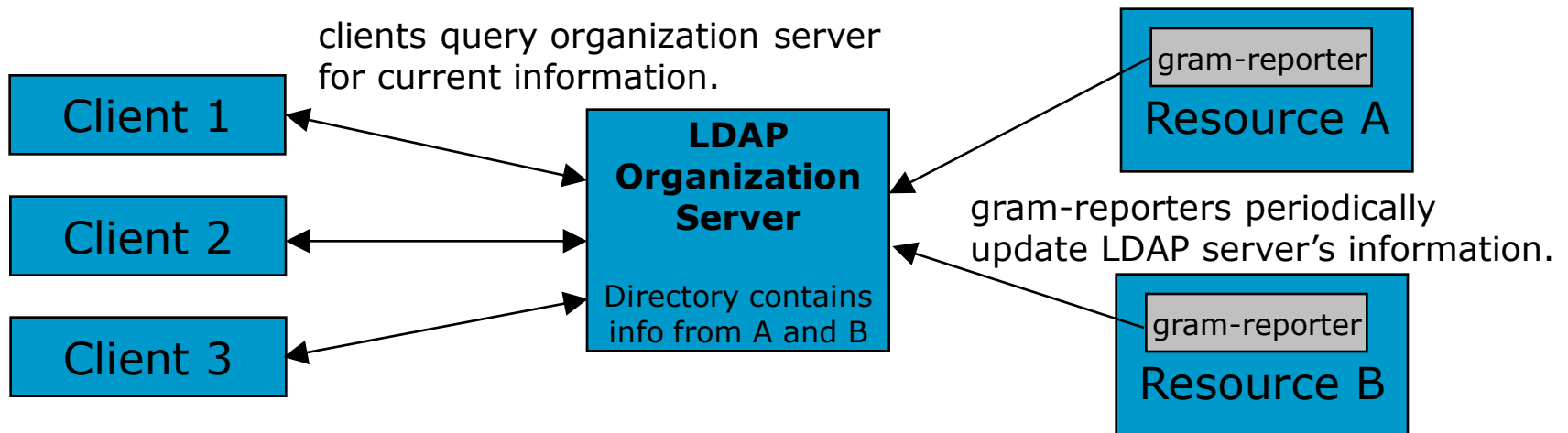
Globus tools

- ▶ GRIS, GIIS, gram-reporter
- ▶ Data discovery, publication,...



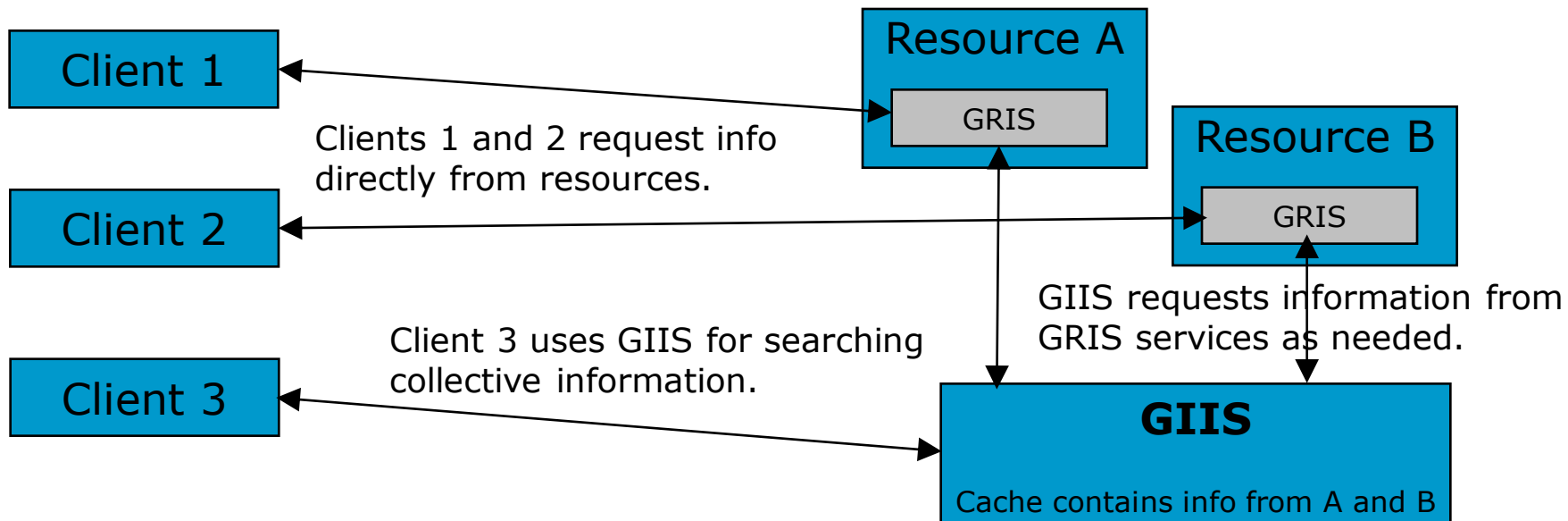
“Classic” MDS Architecture

- 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.



Component of MDS (Metacomputing Directory Service)

Grid Resource Information Service (GRIS)

- ▶ Provide the information of a single resource
- ▶ Multiple information providers can be supported
- ▶ LDAP protocol to inquire

Grid Index Information Service (GIIS)

- ▶ Provides the information collected by multiple GRIS servers
- ▶ Help to provide the information distributed by multiple GRIS servers
- ▶ LDAP protocol to inquire

Papers: Information Service

- 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>