Peer-To-Peer and Grids: Synergies and Opportunities

Author: Andrew A. Chien, University of California, San Diego

Status of This Document

This document provides information to the Grid community regarding the findings and outcomes of the "Peer to Peer and Grids: Synergies and Opportunities" workshop held October 2003 at GGF9 in Chicago, Illinois. Distribution is unlimited.

Copyright Notice


Summary

The Peer to Peer and the Grid: Synergies and Opportunities workshop was held on October 7, 2003 and we had over 50 attendees. The workshop consisted of over ten presentations from panelists who are leading researchers in a wide range of areas. The presentations led to heated discussions which focused on the following questions:

- what exactly is the difference between p2p and grids?
- are these technologies converging or diverging?
- where is there real leverage in exploiting p2p technologies for grids?

There was much heated discussion and enthusiastic participation from the audience -- they spoke as much as the presenters. There was a breadth of interest to have a follow-on, but with more narrowly focused discussion.

The workshop is documented both in presentations and a very detailed set of notes at the web site: http://www-csag.ucsd.edu/P2P-Grid/

1. Document Contents

This document includes the following elements which summarize the workshop:

- the workshop agenda and schedule
- the presentations of each of the speakers
- detailed notes taken from the discussions
2. Workshop Agenda

Peer-to-Peer and Grids: Synergies and Opportunities

Workshop at GGF9  
Sunday, October 5, 2003  
Huron Room, Sheraton Chicago Hotel

Workshop Objective: To stimulate synergy between P2P and Grid communities. Each speaker will be given 15 mins to talk, followed by a heated discussion aggressively chaired by the moderator. Notes from the presentations and discussion are available.

2:00pm Introduction and Welcome (Andrew Chien, UC San Diego)

2:05pm Panel: What are the CAPABILITIES of emerging P2P that should Grids exploit? (120 mins)

Brian Cooper, Georgia Tech  
Alex Mallet, Microsoft  
Geoffrey Fox, Indiana University  
Adriana Iamnitchi, Univ Chicago  
Jim Browne, Univ Texas

Moderator: Karan Bhatia, SDSC and UCSD

Break: 25 minutes

4:30pm Panel: Are grid standards (OGSA) suitable for P2P applications? How so and how not? (120 mins)

Andrew Grimshaw, University of Virginia  
Sergio Mendiola, Oracle  
Dennis Gannon, Indiana University  
Andrew Chien, UC San Diego  
Karan Bhatia, SDSC and UCSD

Moderator: Dejan Milojicic, Hewlett-Packard

6:30pm On your own dinner and smaller group discussions

Organizers:

Andrew A. Chien, UCSD  
Karan Bhatia, SDSC and UCSD  
Dejan Milojicic, Hewlett-Packard  
Adriana Iamnitchi, U Chicago  
Ian Foster, ANL and U Chicago
3. Workshop Presentations

<included below>
Peer to Peer and the Grid: Synergies and Opportunities

Global Grid Forum 9
Chicago, Illinois
October 2003
Objective

- To stimulate synergy between P2P and Grid communities. Each speaker will be given 15 mins to talk, followed by a heated discussion aggressively chaired by the moderator.
Panels I

- What are the CAPABILITIES of emerging P2P that should Grids exploit? (120 mins)
  - Brian Cooper, Georgia Tech
  - Alex Mallet, Microsoft
  - Geoffrey Fox, Indiana University
  - Adriana Iamnitchi, Univ Chicago
  - Jim Browne, Univ Texas
  - Moderator: Karan Bhatia, SDSC and UCSD
Panels II

• Are grid standards (OGSA) suitable for P2P applications? How so and how not? (120 mins)

• Andrew Grimshaw, University of Virginia
• Sergio Mendiola, Oracle
• Dennis Gannon, Indiana University
• Andrew Chien, UC San Diego
• Karan Bhatia, SDSC and UCSD

• Moderator: Dejan Milojicic, Hewlett-Packard
Panel: The Capabilities of P2P Systems

Workshop on Peer-to-Peer and Grids: Synergies and Opportunities

Karan Bhatia (SDSC), moderator
Brian Cooper (Georgia Tech)
Alex Mallet (Microsoft)

Geoffrey Fox (Indiana Univ)
Adriana Iamnitchi (Univ. Chicago)
Jim Browne (Univ. Texas)
Hype Curve

Lots of adopters, no performance data

Early adopters

Technology proven

Lack of payoff, many depart

Napster:
1.79 M simultaneous users
2.8 B files shared/month peak

Seti@Home:
4.6 M users in over 200 countries
3.7x10^{21} FLOPS

Global Grid Forum 9, Chicago Oct 5-8, 2003
P2P Technologies and Techniques

- Routing and Overlay Networks
- Naming and Anonymity
- Search and Discovery
- Fault Tolerance
- Data Management
- High Scale
- Heterogeneous Environments
Introductions

• Brian Cooper
• Alex Mallet
• Geoffrey Fox
• Adriana Iamnitchi
• James Browne
P2P Capabilities for Grids

Brian F. Cooper

Georgia Institute of Technology
Capabilities of P2P

- Scale
- Self-supervision
- Robustness to failure
- Data management
Data management

Don’t sweat the small stuff

Query → Results

plan

index

index

index

index

index
Example: query processing

- PIER
  - Huebsch et al at Berkeley
  - Declarative queries
Example: schema mapping

- Piazza
  - Gribble et al at Washington
  - P2P data integration
Example: schema mapping

- Piazza
  - Gribble et al at Washington
  - P2P data integration
Example: data resource allocation

- SIL
  - Georgia Tech (shameless plug!)
  - Automatic (re)allocation of resources
Example: data resource allocation

- **SIL**
  - Georgia Tech (shameless plug!)
  - Automatic (re)allocation of resources
Example: data-centric routing

- Chord, CAN, Tapestry, Pastry ...
  - DHTs from lots of places
  - Route queries and data
Summary

- Write your domain-specific application
  - What data do you need?
- Data management done by P2P system
  - How to get the data you need
- More capabilities from ongoing work
  - More scale
  - Self-organizing data communities
  - High performance data processing
  - ...

Remarks on Grids and P2P Networks

PTLIU Laboratory for Community Grids
Geoffrey Fox
Computer Science, Informatics, Physics
Indiana University, Bloomington IN 47404

http://grids.ucs.indiana.edu/ptliupages

gcf@indiana.edu
What can we learn from P2P

- I learnt a lot from JXTA especially in its **routing** and **overlay network** structure.
- Naively **Grids** and **P2P networks** are roughly the same and can be thought of as services communicating by messages.
- **P2P** are quite good at **federating** heterogeneous subsystems.
  - Impacts **Protocols**, **Security** and **Interfaces**.
- They have some features that are either more or less **resilient/reliable/robust** (autonomic) than Grids depending on how you look at it.
- Join distributed **dynamic messaging** of P2P with Grids.
  - Helps **workflow** with **distributed streaming**.
- [http://grids.ucs.indiana.edu/ptliupages/publications/presentations/venicesept03.ppt](http://grids.ucs.indiana.edu/ptliupages/publications/presentations/venicesept03.ppt) has more details.
NaradaBrokering

NaradaBrokering Broker Network

Brokers ONLY process messages
Based on a network of cooperating broker nodes

- Cluster based architecture allows system to scale in size
- Grid routing topologies are open research issue?

Originally designed to provide uniform software multicast to support real-time collaboration linked to publish-subscribe to match synchronous and asynchronous interactions.

Perhaps better thought of as stream not message handler

Now has several core functions

- Reliable order-preserving “Optimized” Message transport (based on performance measurement) in heterogeneous multi-link fashion with TCP, UDP, SSL, HTTP, and will add GridFTP
- General publish-subscribe including JMS & JXTA and support for RTP-based audio/video conferencing (Major use)
- General software routing to avoid network problem
- Distributed XML topic selection using XPATH metaphor
- QoS, Security profiles for sent and received messages
- Interface with reliable storage for persistent events
NaradaBrokering and JXTA Federation

- Based on hybrid proxy that acts as both Rendezvous peer (JXTA routers) and NaradaBrokering end-point.
- No changes to JXTA core or constraints on interactions  
  • Change made to Rendezvous layer
- Peers are not aware that they interact with a Narada-JXTA proxy or Rendezvous peer.

- NB provides JXTA access to JMS-based and potentially other "centralized" resources
- Conversely gives JMS systems a P2P collaborative front-end
Laudable Features of NaradaBrokering

- Is open source [http://www.naradabrokering.org](http://www.naradabrokering.org) available now; major new release for SC03
  - Pure Java except for Microsoft Firewall penetration
- Will have end-point “plug-in” as well as standalone brokers
  - end-point is service or user-interface machine
- Will have a discovery service to find nearest brokers
- Does tunnel through most firewalls without requiring ports to be opened
- Links to NWS (Network Weather Service) style performance estimation systems
- Supports JXTA (peer-to-peer network), JMS (Java Message Service) and more powerful XML distributed native mode
- Transit time < 1 millisecond per broker
- Will have by SC03 setup and broker network administration module
Grid Messaging Substrate

Standard client-server style communication.

Substrate mediated communication removes transport protocol dependence.

Any Protocols satisfying QoS

Messaging Substrate has “intelligence” “any desired fault tolerance” “logging” etc. Hope to demonstrate with multicast fault tolerant GridFTP
Virtualizing Communication

- Communication specified in terms of user goal and Quality of Service – not in choice of port number and protocol

- Protocols have become overloaded e.g. MUST use UDP for A/V latency requirements but CAN’t use UDP as firewall will not support ……..

- A given communication can involve multiple transport protocols and multiple destinations – the latter possibly determined dynamically
XGSP Session Protocol MCU
(Peer Group) Web Architecture

Session Server
XGSP-based Control

Web Services

Media Servers
Filters

NaradaBrokering
All Messaging

NB Scales as distributed

Admire
SIP
H323
Access Grid
Native XGSP

Gateways convert to uniform XGSP Messaging

High Performance (RTP) and XML/SOAP and ..

Links Groove and Placeware Collaboration models
InterGrids Federated Grid using NB

- Build a **P2P Network** where each component (cell or Gridlet) is itself a **Grid**
- If **cell** is a single computer, reduces to using **NB** to build communication infrastructure between nodes of P2P network
- If **cell** is a JXTA peer group, then **InterGrids** includes previous federation of JXTA Peer Groups
InterGrids Mediation Architecture

- **NB** acts as a *Mediation agent* in such a *Cellular Grid*
- Using federated security model constructs a VPN like *Virtual Private Grid*
- *Mediation* includes more than routing (as in current JXTA) as can map between Interface standards
- Each *Gridlet* can use *different Service* standards
- *Services register* interfaces with mediator giving ways to map using perhaps *OGSA* as a common intermediate form
- Allows integration of *OGSI* and “*pure web service*” or *Jini* or *JXTA* based Grids where each Grid uses its natural service architecture
- Support *interoperable* (like Job Submission) and *federated* (like registry or metadata catalog) services
- Exploits *stream filtering* capability of **NB**
Autonomic Services in InterGrids

- In a Web (Grid) Service architecture, the state of any service is defined by its initial condition and all the messages (including ordering) that it receives.
  - This how shared event model of collaboration works.
- This is a “Finite State Change” model analogous to saving file and “undo” command in many editors.
- NB plus a robust store can “guarantee” to save all these messages for (all) services.
- This allows one to build both "autonomic data transport" and "autonomic services" since these services can sustain packet losses in transport and can also sustain failures of apps/brokers.
  - archived messages (previous invocations, published events etc) can be retransmitted to reconstruct state at the service or to correct a transport error.
- Anomalies in message traffic (such as a publisher or subscriber are silent) can be detected by NB and signal problems.
- We are building examples of both scenarios using GridFTP as our data transport example.
- We will build a sample autonomic visualization service with detection of failed servers and brokers.
<table>
<thead>
<tr>
<th>Functionality I</th>
<th>WebSphere MQ (formerly MQSeries)</th>
<th>Pastry</th>
<th>NaradaBrokering</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum number of nodes hosting the messaging infrastructure</td>
<td>Medium (MQ is based on the point-to-point model. There is a limit on the effectiveness of this mode in large configurations.)</td>
<td>Very large</td>
<td>Very large</td>
</tr>
<tr>
<td>JMS Compliant</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Guaranteed Messaging (Robust)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Support for routing P2P Interactions</td>
<td>No</td>
<td>Yes</td>
<td>JXTA and later Gnutella</td>
</tr>
<tr>
<td>Support for Audio/Video Conferencing &amp; raw RTP clients</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Communication through proxies and firewalls</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Support for XPath queries/subscriptions</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>end-to-end Security</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Network Performance Monitoring</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Functionality II</td>
<td>WebSphere MQ (formerly MQSeries)</td>
<td>Pastry</td>
<td>NaradaBrokering</td>
</tr>
<tr>
<td>------------------</td>
<td>---------------------------------</td>
<td>--------</td>
<td>-----------------</td>
</tr>
<tr>
<td>Workflow Support</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Support for P2P distributed caching</td>
<td>No</td>
<td>Yes (Squirrel)</td>
<td>No</td>
</tr>
<tr>
<td>Platforms or Hosting Environments</td>
<td>35 different OS/ platforms supported. Also supports the Java Platform.</td>
<td>Supported on platforms which support C# (Microsoft) or Java (Rice).</td>
<td>Platforms supporting Java 1.4 (tunneling C++)</td>
</tr>
<tr>
<td>Maturity of Software</td>
<td>Extremely mature, with very robust diagnostic information</td>
<td>Fair</td>
<td>Fair with some “production” testing</td>
</tr>
<tr>
<td>Transport Protocols Supported</td>
<td>TCP, HTTP, Multicast, SSL, SNA etc.</td>
<td>TCP, UDP</td>
<td>TCP (Blocking and non-blocking), UDP, Multicast, HTTP, SSL, RTP, (GridFTP)</td>
</tr>
<tr>
<td>Multiple transport protocols over multiple hops.</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Broker Network Design Interface</td>
<td>No</td>
<td>No</td>
<td>In Progress</td>
</tr>
</tbody>
</table>
What Are the Capabilities of P2P that Grids Should Exploit?

-Looking at Deployed Systems-

Adriana Iamnitchi
University of Chicago
Approach

- Attitude: there are three certainties in life:
  *On Death, Taxes, and the Convergence of Peer-to-Peer and Grid Computing*, Foster and Iamnitchi, IPTPS ‘03

- P2P and Grids solve the same problem: resource sharing

- But:
  - Different user communities
  - Different resources
  - Different application requirements
  - Different evolutionary paths

- How does the convergence look like?
  - Understand directions
  - Help answer the panel title
Target Communities & Incentives

Grid
- Established communities
  - Science, some industry
  - Homogeneous
  - Restricted participation
- Good behavior:
  - Implicit incentives
  - Means to enforce it

Consequences:
- Trust
- Well-defined “tax base”
- Less flexibility?

P2P
- Anonymous individuals
- No implicit incentives for good behavior

Consequences:
- No trust
- Free riding
- Implicit incentives for cheating: Seti@home, music sharing
Resources

Grid
- More diverse (in type):
  - Files, storage, computing power, network, instruments
- More powerful
- Good availability
- Well connected
- Technical support

P2P
- Computing cycles XOR files
- Less powerful
- Intermittent participation
  - Gnutella: avg. lifetime 1h ('01)
  - MojoNation: 1/6 users always on
  - Overnet: 50% nodes available 70% of time over a week ('02)
- Variably connected
- Some technical support as community effort

Consequence:
- Costly resource integration

Consequence:
- Ease of integration of new resources an early priority
Applications

**Grid**
- Often complex & involving various combinations of
  - Data manipulation
  - Computation
  - Tele-instrumentation
- Wide range of computational models, e.g.
  - Embarrassingly ||
  - Tightly coupled
  - Workflow

**Consequences:**
- Complexity often inherent in the application itself
- (Inevitably?) Complex infrastructure to support applications

**P2P**
- Some
  - File sharing
  - Number crunching
  - Content distribution
  - Measurements
- “Toy” applications only?
  - Albeit very popular “toys”!

**Consequence:**
- Complexity often derives from scale
Scale and Failure

**Grid**
- Moderate number of entities
  - 10s institutions, 1000s users
- Large amounts of activity
  - 4.5 TB/day (D0 experiment)
- Approaches to failure reflect assumptions
  - E.g., centralized components

**P2P**
- V. large numbers of entities:
  - Millions of users
- Moderate activity
  - E.g., 1-2 TB in Gnutella (’01)
- Diverse approaches to failure
  - Some centralized (SETI, ...)
  - Some highly self-configuring

<table>
<thead>
<tr>
<th>tracker</th>
<th>downloads</th>
</tr>
</thead>
<tbody>
<tr>
<td>FastTrack</td>
<td>3,394,126</td>
</tr>
<tr>
<td>iMesh</td>
<td>1,138,315</td>
</tr>
<tr>
<td>eDonkey</td>
<td>1,017,141</td>
</tr>
<tr>
<td>Overnet</td>
<td>917,504</td>
</tr>
<tr>
<td>Gnutella</td>
<td>87,007</td>
</tr>
<tr>
<td>Ares</td>
<td>11,954</td>
</tr>
<tr>
<td>Filetopia</td>
<td>2,951</td>
</tr>
</tbody>
</table>

(www.slyck.com, 10/03/’03)
Services and Infrastructure

**Grid**
- Standard protocols (Global Grid Forum, etc.)
- De facto standard software (open source Globus Toolkit)
- Shared infrastructure (authentication, discovery, resource access, etc.)

**Consequences:**
- Reusable services
- Large developer & user communities
- Interoperability & code reuse

**P2P**
- Each application defines & deploys completely independent “infrastructure”
- JXTA, BOINC, XtremWeb?
- Efforts started to define common APIs, albeit with limited scope to date

**Consequences:**
- New (albeit simple) install per application
- Interoperability & code reuse not achieved
Convergent Environment: 
Large, Dynamic, Self-Configuring Grids

- Large scale
- Weaker trust assumptions
- Ease of integration
- No centralized authority
- Intermittent resource/user participation
- Diversity in:
  - Shared resources
  - Sharing characteristics
  - Variable technical support
  - Infrastructure (sharable services)
  - Support for diverse applications

P2P
Questions?
Overview

When is a system a “grid?” When a system a P2P System?

Goals for Grid and P2P Systems
Integration Versus Incorporation
Comparison of Grid and P2P Systems
Commonality Among Grid and P2P Systems
Valuable Properties of P2P Systems
Evaluation and Metrics
Goals for Grid and P2P Systems

What are the goals to be attained by “grid” systems?

What problems do “grids” solve?

(Avoid Circular Definitions!)

What are the goals to be attained by P2P systems?

What problems do P2P systems solve?

(Avoid Circular Definitions!)
Grid-P2P Interaction Workshop

Integration Versus Incorporation

Integration

Single Specification and Standard
System = P2P-Grid or Grid-P2P

Incorporation

Specification for Interface
System = Grid System + P2P System
Grid-P2P Interaction Workshop

Comparison of Grid and P2P Systems

System State = Mapping Function(System State)

<table>
<thead>
<tr>
<th>Grid</th>
<th>P2P</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Goals</strong></td>
<td></td>
</tr>
<tr>
<td>Virtual Organization</td>
<td>Virtual System</td>
</tr>
<tr>
<td><strong>Common Knowledge</strong></td>
<td></td>
</tr>
<tr>
<td>Central State</td>
<td>Protocols</td>
</tr>
<tr>
<td>(Spec of VO)</td>
<td>Initialization + Discovery</td>
</tr>
<tr>
<td><strong>Control Mechanism</strong></td>
<td></td>
</tr>
<tr>
<td>Central</td>
<td>Fully Distributed</td>
</tr>
</tbody>
</table>
Comparison of Grid and P2P Systems - Continued

<table>
<thead>
<tr>
<th></th>
<th>Grid</th>
<th>P2P</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Default System Structure</strong></td>
<td>Static (App. LT)</td>
<td>Dynamic (App. LT)</td>
</tr>
<tr>
<td><strong>Default Application Structure</strong></td>
<td>Transient</td>
<td>Long-Lived</td>
</tr>
<tr>
<td><strong>Security Requirements</strong></td>
<td>Stringent</td>
<td>Lessened by Isolation</td>
</tr>
</tbody>
</table>
Grid-P2P Interaction Workshop

Comparison of Grid and P2P Systems - Continued

Commonality of Grid-P2P Systems

Low level implementation mechanism is ultimately a sophisticated procedure invocation with composition by control flow and direction of output
Properties of P2P Systems
Self-Organizing Behavior
Intrinsically Dynamic Structure

↓ ↓

Adaptive Optimization
Fault-tolerance
Grid-P2P Interaction Workshop

**Evaluation and Metrics**

Quality of Service – How specified and how measured?

Common Knowledge – Intrinsic versus discovered

Control – What does fully distributed control imply

Dynamic Structure – Time scales for change
Panel Session (the reverse one)

“Are Grid Standards (OGSA) suitable for P2P Apps? How so and how not”

Dejan Milojicic
Workshop on Grids and P2P
GGF9, Chicago, October 5, 2003
Participants

- Andrew Grimshaw, Virginia University
- Sergio Mendiola, Oracle
- Dennis Gannon, Purdue University
- Andrew Chien, UCSD
- Karan Bhatia, UCSD & SDSC
Structure: Objectives, Process, Radical Questions

• See how to get communities closer
• Opportunities for collaboration

• Informal process

• Radical questions
  – Can p2p apps benefit from the underlying Grid standards
  – Can p2p use Grid experience
Motivating Questions

- What does Web Service-based mean to p2p?
- What does Web v. Grid Service (statefull) mean to p2p?
- Can Grid be sufficient infrastructure for P2P apps
  - Distributed computing (✓)
  - Data access / file sharing (√/?)
  - Collaborative computing, especially at higher scale (?)
- What about P2P infrastructures?
- What OGSA Capabilities are appropriate for P2P apps
  - Ad-hoc, disconnection, reliability, anonymity, self-[manage, heal,..], lightweight, etc.
- Polarization of communities, how to evolve?
  - Grid within industry & scientific computing;
  - p2p within academia and community
P2P and Grid

October, 2003
Andrew Grimshaw
CTO & Founder
&
University of Virginia
What is a Grid System?

Gee, sounds like P2P!

A Grid system is a collection of distributed resources connected by a network.

Examples of Distributed Resources:
- Desktop
- Handheld hosts
- Devices with embedded processing resources such as digital cameras and phones
- Tera-scale supercomputers
A grid enables users to collaborate securely by sharing **processing, applications, and data** across heterogeneous systems and administrative domains for collaboration, faster application execution and easier access to data.

- **Compute Grids**
- **Data Grids**
Grids Should

- Provide access to processing, data, and applications across domains
- Incorporate heterogeneous hardware, operating systems, and system configurations
- Support secure grid-wide sharing of resources, while maintaining local, diverse security policies
- Be resilient in the face of change and system failure
- Account for resources consumed
- Provide for arbitrarily scalability
P2P – Grid – Web Services

• Are all different manifestations of the same underlying phenomenon

• Distributed Systems

• Therefore we should look to the distributed systems literature for solutions to tough problems such as
  • Naming and binding
  • Fault-tolerance and high-availability
  • Transaction support
  • Scalability
Technical Requirements of a Successful Grid Architecture

- Simple
- Secure
- Scalable
- Extensible
- Site Autonomy
- Persistence & I/O
- Multi-Language
- Legacy Support
- Single Namespace
- Transparency
- Heterogeneity
- Fault-tolerance & Exception Management

Success Requires a single integrated, object-based model as the foundation.

Manage Complexity!!
What can Grid learn from P2P?

• Scalability
• Autonomy
• Light-weight implementations
• Inclusion of desktop and smaller resources
• Intermittent operation, highly dynamic connectivity
What can P2P learn from Grid?

- **Security** – more than encryption
  - Authentication, access control, trust models, virtual organizations, cross organization interactions, etc.
- **Naming and binding** – GSH
  - Extend web services and allow the traditional distributed systems transparencies to be applied to web services and P2P applications
- **“Industrial strength” architectural support (OGSA)**
  - logging and monitoring, event models, program/service provisioning and management, etc.
- **Resource management strategies**
- **Policy negotiation**
Peer-to-Peer and Grids: Synergies and Opportunities

Sergio Mendiola
Oracle Corporation
Are these statements true?

• P2P is used to do CPU scavenging at the edges.
  – SETI@Home (This is not really p2p)
• P2P applications are based on P2P computing.
  – YIM  (YIM is not really based on P2P)
• P2P means end-user collaboration and sharing.
  – This certainly is a type of p2p application, and might be deployed using a p2p platform.
  – But p2p is also used within server infrastructure.
OGSA Requirements

- Functionality
  - Discovery
  - Data sharing
  - Virtual organizations

- Security
  - Multiple infrastructures
  - Application and Network firewalls

- Resource Management
  - CPU Scavenging
  - Load balancing

- System Properties
  - Fault tolerance
  - Self-healing
P2P Relevant areas

- Scalability *
- Connectivity
- Security *
- Decentralization *
- Group support
- ...


Scalability

• I haven’t found a requirement on scalability for Grids, in the OGSA papers.

• However this is a major concern in peer-to-peer systems.
Connectivity

- Traversing firewalls
- Dynamic IP addresses and NATs
- Network characteristics are different
- Temporal disconnects
- Ad-hoc networks (appagg-rg)
Security

- Distributed trust mechanism
- Reputation
- Sand-boxing
- Anonymity
Decentralization

- There is no particular requirement on this either.

- However, fault tolerance might be related and self-healing are related.
Group Support

- There is a good match with OGSA’s virtual organization concept.
Summary

- OGSA should support centralized and decentralized computing models.
- OGSA’s requirements today, do not include all relevant requirements for P2P.
- The earlier we address this in the OGSA WG the better.
P2P and Grid Services: Does OGSA help P2P?

Dennis Gannon
What are P2P systems?

- Instant messaging: msoftIM, ICQ, Jabber
- Distributed computation: Entropia, distributed.net, SETI@Home, United Devices
- Collaboration: Groove, Zaplet, ...
- File sharing: KaZaa, Morphius, Gnutella, Freenet
- Infrastructure: ad hoc & JXTA
How well does P2P work?

• Fantastic scaling.
  - But... Good P2P networks are power-law networks
    • A few nodes are super peers
    • Without a resource allocation model and good security they are limited to file sharing.
    • Naming and resolution services get re-invented all the time.

• Do they use web service technologies?
  - No. (but there is a lot of talk about it.)
Ask the students

- Do the grid standards like ogsi/ogsa help the cause for p2p?

No

Well … they don’t hurt

Perhaps

Yes

P2P and Grids Workshop
The Grid Service Model

Resource layer
1000s of PCs -> massive supercomputers

Open Grid Service Infrastructure (web service component model)

- Registries and Name binding
- Reservations And Scheduling
- Policy
- Security
- Event/Mesg Service
- Administration & Monitoring
- Logging
- Accounting Service
- Grid Orchestration
- Data Management Service

Grid Portals

Grid Application Factory Service

Application Factory

App Instance Instance Instance

Open Grid Service Architecture Layer

P2P and Grids Workshop
How well do Grids work?

- There are few examples that scale to more than a dozen resources.
  - This is not bad: Virtual Organizations are not always huge.
- OGSA is still being defined.
  - The model is right to provide a resource management service layer for P2P.
  - It is built on web services which should scale.
Will OGSA help P2P apps?

• Yes, it provides
  - A naming and name-resolution model that will be very important.
  - Resource management virtualization abstractions
  - An extensible architecture for building apps.
  - SLAs
  - Full OGSA will have ubiquitous services that P2P builders will love
    - Well .... Perhaps someday.

• No? possible problems
  - Will the security model scale?
  - Messaging and message routing important to P2P. OGSA is not there yet.
  - Is the software too heavy weight?
  - Too tied down to static IP?
Are Grid Standards Suitable for Peer to Peer Applications?

Andrew A. Chien
Director, Center for Networked Systems
UCSD Computer Science and Engineering
October 5, 2003
Grid Standards (e.g. OGSA)

- Discovery and Brokering
- Metering and Accounting
- Data Sharing
- Virtual Organizations
- Monitoring
- Policy
- … Multiple security solutions…
Grid Standards II

- Provisioning
- Resource Virtualization
- Optimization of Resources
- Transport Management
- Usage monitoring, SLA violation

- Goal: produce “broadly adopted framework for distributed system integration”
- “New capability, higher quality, manageability, etc.”
- Focus on things thought to be expensive!
Isn’t the synergy obvious?
Peer to Peer Applications

- Large-scale deployed
  - Distributed Cycle Scavenging
  - Distributed File Sharing
  - Distributed Gaming
  - Distributed Instant Messaging and Chat

- Not primarily from applications communities trying to get to higher levels of capability
- Not primarily “application integration” activities
Peer to Peer

• **Academic: DHT and Systems Built Atop It**
  » Distributed information services, file sharing, routing structures, distributed data management, filesystems, etc.

• **Resource Assumptions**
  » Unreliable resources, little known about properties
  » Less controlled network and resource environment
  » Dramatic scale of systems
  » Moderate “control” infrastructures
    – Monitoring, accounting, SLA’s, etc.
  » => differing initial assumptions, continuing behavior

• **Goal: produce functioning systems without control**
  » Fundamentally more flexible and robust systems
  » Early, with few specific application ”success” stories
Grids: one size fits all?

- IP Networking
- Operating Systems
- Data Management
- Water Systems
- Electric Power
- Cellular Telephony

- A Toolkit Model?

Is that what I ordered?
Grid Standards can Support Peer to Peer Applications

- Must establish a value proposition
  - How does a converged set of services enhance innovative Peer to Peer applications?
  - How does grid infrastructure (software, resource) enhance innovative Peer to Peer applications?

- Emerging focus
  - What are Peer to Peer applications?
  - Are they different from Grid applications?
    - Infrastructure and Applications

- Are they going to be integrated? Yes, but not in a single infrastructure
Are Grid Standards Suitable for P2P?

Karan Bhatia (SDSC)
NPACI Grid

- Blue Horizon (sdsc)
  - 1152 IBM Power3 processors (8 procs/node, 4GB/node)
  - 15 TB parallel GPFS file system
- Rocks Clusters (sdsc)
  - Redhat Linux-based
- HPSS Archival Storage (sdsc)
- 500 Terabyte SAN (sdsc)
- 64 cpu IBM Power4 cluster (utexas)
- 64 cpu IBM Power2-based SP (umich)
- 24 cpu IBM Power3-based SP (umich)
- 134 cpu & 256 cpu AMD Linux Cluster

NPACI Partner Sites:
- Caltech
- University of Texas at Austin
- University of Michigan
- UC Berkeley
- UC Santa Barbara
- University of Southern California
- University of Virginia

NPACI Application Thrusts:
- Molecular Science
- Neuroscience
- Earth Systems Science
- Engineering

NPACI Alpha Projects
- Monte Carlo Cellular
- Microphysiology
- Protein Folding
- Bioinformatics Infrastructure
- Scalable Visualization Toolkits
- Advanced Tomography
- Multi-Component Models
- Adaptive Computations for Fluids
GeonGrid

- Provide Data Federation
- Ontologies and data semantics
- Data Mediation
- Data Replication & Caching
- Decentralized Resources (mainly data)
- Best-effort guarantees

Geon Institutions:
San Diego Supercomputer Center
Penn State University
Geological Survey of Canada
San Diego State University
Arizona State University
Rice University
University of Arizona
University of Idaho
University of Missouri
University of Texas, El Paso
University of Utah
Virginia Tech
UNAVCO Inc.
DLESE
US Geological Survey
ESRI Inc.
Lawrence Livermore National Laboratory
P2P Requirements

- Connectivity
- Security
- Resource Variability
- Locality and Interactivity
Connectivity

- Support for NATs
  - no use of ipsec (checksums include headers)
  - Service Endpoint Rewrites
  - Proxy forwarding (push vs. pull mechanisms)
  - NAT identification
  - UPnP or IETF middlebox solutions?
- Support for Laptops (and other devices) as service endpoints
  - DHCP is here to stay (as are NATs)
  - Determine network characteristics (802.11 vs. cell vs. bluetooth)
Support richer trust models
  • Community-based trust (reputation models)
  • different trust domains (condor universes?)
  • decentralized identity establishment
  • multi-CA management
  • Support authorized, but anonymous
  • Roll-based authorization
  • Data Security/Trust
Resource Variability

- Even Servers crash!
- Decentralized infrastructure services
- Dynamic enter/exit
- Networks also crash
  - auto-reconfiguration, best effort
- Data quality and corruption
Locality/Interactivity

- Query Absolute and Relative Geographic Location
- Query Absolute and Relative Network Location
- Group Creation and Management
- Presence & Notifications
Can One Build P2P Apps on the Grid?

- Answer is ... yes, i think so.
- But, the infrastructure doesn’t help to do so.
- Is this important? ... yes!
- grids are looking more and more like P2P
4. Notes from Workshop Discussions

<included below>
I. P2P and Grid Computing; Panel 1.
   A. Karan: introduction
   B. Brian Cooper, GA Tech
      1. Data management; grids is about computation over data. Ability to specify a query and get a result; computed efficiently. Be unaware of computation of the cloud. Data management people need to be aware. Users don’t need to sweat the small stuff; can focus on domain specific application.
      2. Example 1: PIER
         Distr data; declarative queries. Query planning to determine which resources collaborate to give you the data you need. System works to pull together the data. Challenges: heterogeneity. Opportunity: leverage resources.
      3. Example 2: Piazza
         data integration; data has different schema: diff fields, relationships, constraints. Schema aren’t always the same. If no intention to collaborate; won’t interoperate. How to make mappings to collaborate productively? Hard when there are lots of sources.
      4. Example 3: SIL: Search Index?
         Another problem: distribution and heterogeneity don’t scale; there is the problem of data resource allocation and management. Strategically place indices, caches. Selfish decision yield more efficient results. Individual resources monitor their load and monitor and make and break links. If many autonomous nodes, centralized mgt system can’t assign load.
         Denial of service attacks? Yes, many people are looking at that: security, trust, denial of service. There is the question of mechanism design: how to incent nodes to take on load.
      5. Example 4: Chord, CAN, Tapestry, Pastry: data-centric routing
         Efficient application routing: how to get data to the people who need it? “Publish and subscribe” system. Register interest. DHT: Distributed Hash Tables. Efficient routing of data to people interested in it.
      6. Summary
         System offers more scale, self-organizing data communities, and high performance data processing; hidden success.
   Centralized approach won’t work for reasons of scale and organizational practices.
   What state is the community in? Research prototypes available for downloading and trial. Industrial prototypes. EII marketplace. IBM’s information integrator.
   Point well taken; take those systems and scale up.
   Use in industrial setting? Make it so easy to be a part of the infrastructure; take existing infrastructure and leverage it. Industry is concerned about how their resources are being used; security. Take existing data set and integrate into p2p grid.
Correctness of response: local relational model. Relaxed consistency; most popular of most recent… Unsolved challenge: meet everyone’s need.

Replicability: not covered by this model. not production quality; smart people are developing them.

C. Alex Mallet, Development Manager, Peer Networking Group, MS
1. p2p investment
    Shipped SDK in 7/03: adv networking pack for Windows XP. Common functionality; key capabilities. APIs for the capabilities. Serverless DNS. Security model not based on centralized authority.
    Found that many people are behind NATs. IPv6 NAT traversal (Teredo)

2. General principles
    MS is a big target

3. Key capabilities
   a. Name registration and resolution
      Publish end points on home PCs. Serverless.
      Multiple endpoints on a machine (multiple services). People don’t have static IP addresses. Dynamic endpoints.
      (1). Easy way to map friendly names to unfriendly names (mechanically produced)
      (2). Other initiatives in this problem: classic problem.
         (a). I3C: LSIV
         (b). GGF:
         (c). Handle.net: CNRI
   b. Data Distribution:
      (1). Continuum between efficient and robust; happy medium. Rooted tree sometimes.
      (2). Multiple transports
      (3). Disconnected case
   c. Persistent Peer Groups
      Rejoin when back online. Membership restrictions. Similar to the “Groove” model. Similar, though Groove doesn’t have the notion of insecure groups.
   d. Security
      Signing and validation; people sending data have authorization.
      Make sure people could plug in their own security measures.
   e. Management and Deployment
      Real world experience. Need mgt and debugging knobs. Hard: few insights into the system. Non-sophisticated users. Install and “do the right thing.”
   f. Programming Model: still trying to figure out. If you can’t explain it, you certainly can’t test it. Wound up with a best effort.
   g. Miscellaneous

4. Key capabilities?
5. Questions: not built on .net? No. Like Pastry: name resolution is similar but not built on Pastry. Reliable delivery: provide none. SRM: scalable reliable multicast. Provide folks with our version of key capabilities. Not trying to sell it!

D. Geoffrey Fox, Indiana U
1. Focus on messaging
3. JXTA plays a prominent role. Interoperation between peer groups. JMS: Java Message Service. NWS: Network Weather Service.
5. Only 30 minute notice of need to give a presentation.

E. Adriana Iamnitchi, U Chicago, What are the Capabilities of P2P that Grids Should Exploit?
1. Looking at deployed systems
2. Certainty: convergence of p2p and grid computing
3. Comparison of the communities
4. Comparison of the resources
5. Statement of similarity in Grid 20 years ago (multiple parallel efforts) with P2P today. Difference is that the user base is not mature. Greece: can point out the app’s with large computation requirements; p2p has not.
6. Question: p2p called grids and grids called p2p? Hype engine!
7. Convergent environment
   a. Normal path is for Grids to grow in scale
   b. What applications have that many people who are interested in them?
   c. Interesting parameters to connect grids and p2p rather than separate them.

F. Jim Browne, U Texas
2. Neither have clear goals.
3. Suggest fundamental differences: way to put them together. Work towards an integration? Unsure? Incorporation: standards for grids; p2p; clean I/F between to leverage strength of each. Claim that there are fundamental differences. Both are instances of distributed systems. Specification of assumptions that underlie grid and p2p; not easy. Know a little about distributed systems.
5. Commonality: low level implementation mechanism is ultimately a sophisticated procedure invocation with composition by control flow and direction of output.

G. Discussion:
2. What type of grid are you trying to build? Scaling may not be that critical for some applications?
3. Changing: production grids and those to be: definitely a notion of having more users and decentralization; limits to current technology. P2P has advantages to address those problems. People here recognize that their grid systems must deal with these problems.
4. Consider the distinction between consumers and providers; predominantly one role or another? predominantly both? Scientific data sets and music files. Hard time understanding, if having a hard time doing computation, why should I share cycles? Tragedy of the commons. Radically asymmetric users: 10% of users consuming large portion of resources.
5. Understand what is needed to build the system? Learn from other solutions.
6. P2P begins with decentralization and no centralized control.
7. Not imaginative chaos. There are means of control. P2P are not anarchic. Control
doesn’t look like a business organization.
8. People who are building and deploying P2P are not purists about decentralization.
Directories make things easier; super nodes. Purists aren’t going to get far.
9. Difference between grids and p2p: p2p is not just distr control; it’s distr data. Where
to find this chunk of data. Grids: where to find a service. Data generated by service
can be cached and moved around the network.
10. Big area of data grids; look a lot like p2p.
11. Need both: some resources which are by nature centralized, e.g., particle accelerator.

II. P2P and Grid Computing: Panel 2
A. Andrew Grimshaw; founder of Avaki
1. P2P: included small devices such as Palms, phones. Grid: started with
supercomputers and working their way down.
2. OGSI is a spec; OGSA is a collection of interesting ideas.
3. Policy description and negotiation
4. P2P hasn’t done “naming and binding” well. XML is something everyone can agree
upon.
5. Byzantine security models in the wild and woolly Internet. Peer devices may be
arbitrarily malicious. Certification levels; “web of trust.”
6. “Simple and lightweight;” construct from that something highly functional. Neuron
analogy.
B. Sergio Mendiola, Oracle
1. P2P and Grids: Synergies and Opportunities.
2. Knob from centralized to decentralized.
3. Scalability: assumption of Web Services scalability
4. Things are in the OGSA in nascent form. Just a vehicle for other existing
technologies?
5. No requirement for decentralization, so self-healing.
6. Word “scale” and “scaleability” only appear once in the entire OGSA document.”
Stated by a co-author of the document. It is an issue.
C. ?, Bloomington, University of Indiana
1. Suns JXTA: another model to build P2P
2. Grid community has scale envy when it looks at P2P. A few nodes are super peers.
“Power-law networks.” Successful P2P evolves to this state. Where does control lie.
Clumped in an interesting way.
3. Chien: contradiction to fixed degree, homogeneous.
4. Limited applications: file sharing.
5. Lot of talk about Web Services and P2P but no evidence of action.
6. Question: do grid standards like OGISI/OGSA help the cause for P2P.
   a. No
   b. Well, don’t hurt
   c. Perhaps
   d. Yes, but when thought of in a different way.
7. What can be leveraged? OGSA: ubiquitous distributed services to allow us to think
of OGISI and resources as a simple computer. Chien: distr operating system.
8. Rich layer of services to enable a richer set of P2P applications.
9. Not convinced that Grids are scalable. Built on Web Services that are designed to
scale.
10. Grid: Virtualizing resource management. P2P doesn’t do that at all. Grid looking at
SLAs carefully.
11. Tests of XML signatures in SOAP. Would hate to do too many chains of signature
validation.
12. Think that most Grid s/w is too heavyweight. Static IP isn’t going to work.
D. Andrew Chien, UCSD
1. List of focus areas for OGSA. Very telling of mindset. Management, control, ownership mentality. New capability, perhaps higher quality and manageability.
2. Focus on things thought to be expensive.
3. Not clear that this is the right list for P2P! Focus may not be right. Synergy may not be so obvious.
4. P2P applications: two clusters: people with large scale deployed systems: cycle scavenging, etc. Up and running for years. “Make big complex things that you are stuck with, work together productively.” Second half of P2P world: academic. Not pursuing 100sK users but DHT and systems built atop it. Not confused with folks running the large scale systems. Characterized by assumptions different than grid assumptions.
5. Moderate control infrastructure, and that’s being kind. Can’t predict and control. Marked contrast to “monitoring, accounting, SLAs, etc.”
6. P2P: functioning systems without control; early in the evolution with a few specific applications success stories.
7. Question: hit on a key difference; corporate IP is critically important. Corporations won’t risk their IP. Agree.
8. Grids: one size fits all. Infrastructures that are widely shared; not homogeneous. Globus has been successful because it started with the toolkit.
9. PlanetLab analogy.
10. Converging, parallel, diverging? Nature of applications?
11. They will have to talk together, but not in a single infrastructure!

E. Karan Bhatia, SDSC
1. Grid app’s: NSF TeraGrid. Teragrid is not P2P. Homogeneous clusters of machine; huge networking component. 4-5 sites. Very administratively centralized structure.
2. NPACI Grid: more heterogeneous: equipment and institutions. PRAGMA (sp?) Grid: pacific rim. GeonGrid: geo sciences community: resources are data. Has to be decentralized. Best-effort. Very autonomous; themes of P2P. Distributed and autonomously managed. 20 different institutions; scaling to 100. Grid systems are looking for some subset to be more like P2P.
3. Requirements:
   a. NATs and DHCP (laptops)
   b. User interface? Not part of the grid? Data is located on my laptop; want the results on my laptop. Want to be part of the grid to run my app.
   c. No: just a user.
   d. Laptop has to have a service end point.
   e. Just a certificate

4. Security
5. Value proposition isn’t clear; understand the value prop for my systems, though.

F. Discussion
1. OGSA discussions focusing on Byzantine settings? Not in the security groups; punted to the security group. They are not looking at Byzantine: highly structured, Certificate based.
2. Where to take the authorization group that we just started. Requirements for policy language. Conceptually, running far ahead. Delegate rights, but no way to implement; no common language to express. Standardize on a policy language. Need in grid and P2P. Understand statements about trust, rights, and delegation of rights.
3. Need conventions about the semantics.
4. XML-based.
5. Concept of Agreements: machine is reliable, degree of security. Defining the profiles of machines. Not just security; wide range.
6. Discussion is premature. Taking products, stovepipe solutions. OGSA standards. Mismatch. None of OGSA are in place. What is the GGF trying to solve? Help to P2P: where to embed standards. P2P between machines, not services. Other modes to establish environment.
7. Andrew: agree that it’s early. Problem: would be too late if we waited until that is 
true. Raise people awareness that there are underlying forces that will lead to their 
being less useful.
8. Analogy: goal for the ARPANET: communication in the presence of large number of 
faults. Contrasted to the telephone company. Grids look like telephone companies. 
P2P look like the Internet. They will get together. Need now to identify some 
structure in the P2P community to get them to work with Grid. Some folks in 
corporate world using P2P and Grid; see if they wouldn’t be interested.
Documents imply that Grid Services are Web Services. Web Services Distributed 
Management WG. Are Web Services standards applicable to P2P.
10. Searched the web looking for answers to that question: P2P using Web Services.
11. Web Services are too top heavy for P2P.
12. Disagree: boils down to RPC; means using XML over SOAP; all the tooling that is 
coming out, e.g., .net. WSDL itself is a useful standard. SOAP isn’t bound to its 
lower level comm – but it is. Web Services isn’t necessarily the wrong thing for P2P, 
but it isn’t there.
13. Alan: propose that P2P and Grid are totally incompatible because Grid is build on 
Web Services.
14. Web Services aren’t that widely adopted yet.
15. Been looking at implementing more general computations using P2P. Web Services, 
if not so cumbersome, would be perfect. One thing that could be done: define 
middleweight and lightweight of RPC as a standard for P2P comm.
16. GSR doesn’t have to be a binding.
17. Also want to send code to execute. Very easy thing to do. Not rocket science.
18. Andrew: preserve a toolkit approach. Subset the architecture: take this piece. You 
want one real small piece.
19. Get some success on a small piece. NSF sponsored project: P2P and Web Services 
computation.
20. Grid service handle and resolution to grid services references; very general. Scale 
where referencing; GSH for every doc in the world. Very interesting push to a 
different type of scale. P2P community with a notion of handle and handle 
resolution. PlanetLab. Conflicting security model.
21. Couple of things missing in handle resolution. Resolution itself. Depends on name 
space. Plans on the side: on line global registries. Work to be done to address that 
issue. Scalable name resolution. How would it help the P2P if that were available.
22. Name resolution and binding is something that everyone needs.
23. Andrew: benefit of interoperating with those guys. Incentives for interoperation 
before adoption.
24. Incentive would be there if part of a toolkit and part of a global name.
25. Problem of Web Services being top heavy may disappear over time. Plan for the 
probable future.
26. Andrew: listed foci of OGSA that are perceived to be expensive. Accounting and 
resource management really important in huge hard drives and bandwidths.
27. Accounting isn’t just because things are valuable but for auditing. Looking ahead 10 
years is a good idea, but some things are needed now: naming, security.
28. End-to-end connectivity isn’t guaranteed: IPv6 isn’t being adopted, NATs. Scale and 
decentralization will be more of a problem.
29. P2P will be forced into Web Services; will be non-standard.
30. As you scale up, you have to assume fault tolerance. Tremendous relief, don’t have 
to compute MTTF.
31. Andrew: when things fail, have to assume that they will fail in arbitrary and 
Byzantine ways.
32. Literature assumed fail-stop. Unix doesn’t fail-stop. Can’t afford the overhead of 
Byzantine model. 80% of our code is exception handling.
33. Looking 10 years ahead: top edge goes up the scale; bottom moves down the scale. Greater heterogeneity. Web browsing on cell phones. Don’t push the argument to heavyweight things. History says it will. Increased scale and heterogeneity. Push down to smaller devices and more ubiquitous.

34. PDAs now are like the machines we used to do. Big crypto on those is really expensive. Crypto will migrate to devices.

35. Grids have tended to focus on middle to high end. P2P will be focused on the range.

36. Andrew: note that moving down is harder than moving up.

37. Smart dust: half megabyte memory; MHz processor.

38. PDAs won’t be part of the grid; they offer no grid service.

39. Imagine that the grid are all the devices here in the room (laptops). No, because you can disconnect now and go home. Suppose a new phone with camera and memory: does offer services: access control device, same security model with phones as with anything out there. From security perspective, since devices are data sources.

40. Looked at app’s and use cases of grids and p2p. Two categories: large scale, scientific. Mass consumer file sharing. Insufficient treatment of business applications. How businesses are affecting this debate?

41. In industry today, businesses are using grid. Integration or access to data between or within organizations. Share info in real time. Contractors who aren’t necessarily trusted. Life sciences: huge amount.

42. OGSA: motivations are enterprise level applications. Very large concern. If not there, they’ll walk. 80% of OGSA working people are industrial.

43. No mid-level app’s for P2P is not clear that businesses wanted to invest resources in building infrastructure. Hope will help by the emergence of toolkits, MS sol’n. Can’t speak to the grid stuff.

44. If not using standards like CORBA, why use standards like Web Services? “They’ll see the light!”

45. Don’t think they will use web services. P2P trust themselves and not others. Will go for lowest common denominator that they trust, sockets. Avoid admins and thought police.

46. Andrew: zero incremental cost.

47. P2P larger than pirates. They don’t want to get caught. Not the determining factor for legitimate P2P.

48. Enterprises want to keep their own domain.

49. If build a system that is secure: anonymity, trust. Need a security infrastructure. Not too many sol’ns there.

50. Difference between grid and p2p is the amount of money they have to spend. Any good P2P enterprise applications?

51. Collaboration technology has a P2P flavor. Pick up Web Services component. Starting to do that.

52. Common standard.

53. When US commercial people talk with potential customers about Web Services and P2P? Our customers couldn’t care less. Sol’n to a problem. Some care about Web Services and OGSA, but I swear they don’t know what they’re talking about (OGSA).

54. Andrew Chien: interoperability; long term tech strategy; interoperability and manageability. Manageability and integration with existing security infrastructure.

55. Sun person: JXTA group has applied to IETF. JNGI.

G. One Minute Statements

1. Andrew: life is complex; real systems are complex. Myriad ways to fail. Construct abstractions; OGSA allows us to build layers systems and hide complexity.

2. To convince P2P users to use Web Services will be hard; toolkit is a good approach.

3. Grid community – before going to P2P comm. to tell them what they should do – is let their hair down and rock and roll with them; learn what they’re problems are. Then start to build P2P, then rest of the world will care.

5. Advantage in both ways. Sol’ns to specific problems can help. Don’t have to solve the problem. Get your stuff out faster.

H. Question: propose actionable items?
   1. Grid research oversight committee: strongly suggest that this group put together to write GGF info docu based on observations from this workshop.
   2. Send notes to Cees
   3. Andrew C: solicit ideas and proposals.
   4. P2P WG in GGF. Members to participate to seed some ideas.
   5. Do want to I/F with these groups; question of structure. Requirements, e.g., scalability.
Author Information

Andrew A. Chien, Professor at University of California, San Diego
achien@ucsd.edu

Glossary

Recommended by not required.

Intellectual Property Statement

The GGF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the GGF Secretariat.

The GGF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this recommendation. Please address the information to the GGF Executive Director.

Full Copyright Notice

Copyright (C) Global Grid Forum (date). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the GGF or other organizations, except as needed for the purpose of developing Grid Recommendations in which case the procedures for copyrights defined in the GGF Document process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the GGF or its successors or assigns.

This document and the information contained herein is provided on an “AS IS” basis and THE GLOBAL GRID FORUM DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.”