

Grids: The Top Ten Questions

Jennifer M. Schopf
Argonne National Lab

March 10, 2003

10 Things We Hate About the Grid

Jennifer M. Schopf

Argonne National Lab

(with significant input from
Bill Nitzberg, PBS)

March 10, 2003

Overview

- Computational Grids are becoming more and more common
- Collaborations are being developed
- Governments are giving lots of money
- Globus seems to be everywhere
- Happy application scientists are no where

Things heard recently...

- “Isn’t the Grid just a funding construct?”
- “The Grid is a solution looking for a problem”
- “We tried to install Globus and found out that it was too hard to do. So we decided to just write our own....”

Things heard recently...(cont.)

- “Cynics reckon that the Grid is merely an excuse by computer scientists to milk the political system for more research grants so they can write yet more lines of useless code.” –Economist, June 2001
- “No one can really define it, everyone wants an app that can do it, and companies that claim to do it are getting a lot of interest.” – SlashDot, March 2003

Overview

- Overview of open issues in Grid computing, both technical and socio-political
- Topics:
 - User issues
 - Information
 - Security
 - Performance
 - Socio-Political
- Structure:
 - Question
 - Discussion
 - Moral- a lesson or principle contained in or taught by a fable, a story, or an event.

A grain of salt

- Many of the problems I'll discuss are in the process of being addressed by various groups
- There may be on-going work or solutions that I don't know about, I'll apologize now
- These are my opinions, not those of Argonne National Lab, the Globus Project, etc...

What is a Grid?

- Shared resources
- Coordinated problem solving
- Multiple sites (multiple institutions)

Not A New Idea

- Late 70's – Networked operating systems
- Late 80's – Distributed operating system
- Early 90's – Heterogeneous computing
- Mid 90's - Metacomputing

- Then the “Grid” – Foster and Keselman, 1999

How are Grids Different?

- Autonomy
- Heterogeneity
- Resources are more the CPU and networks
- Focus on the user

- These differences create many of the problems addressed in this talk but also make the system much more usable than its predecessors

1. Why aren't there users?

- FACT: Original Grid Forum Applications group folded because they couldn't get application developers involved (this has been started again, and is more successful, but still challenging)
- FACT: A very few applications (CMS, ATLAS, CACTUS, and a couple others) are the only ones who are really getting production use out of Grid computing

Move from sequential to parallel computing

- Parallel computing showed us that they “If you build it they will come” scenario just won’t work
- Until debuggers, fast compilers, languages, libraries, etc. the users didn’t want to use parallel machines
- Many hundreds, even thousand, of hours went into re-writing codes for parallel machines

Heroic Effort Required for the Grid

- There is the impression (right or wrong) that only heroic efforts will allow you to use a Grid
- Some re-writing of code required
- Access to resources isn't easy even once code is changed

Moral: To get users we need

- To get users we need
 - User-level tools
 - Better usage scenarios

2. Where are the user level tools?

- What a user would like:
 - “Run my job, finish by lunch”
 - “Get a data set that has these attributes”
 - “Tell me when that simulation will finish”
- Where are we today:
 - Specify exact machines, data files, explicit data transfers, etc
 - Little (or no) dynamic information or prediction

The “Ideal” Grid (F&K, 1999)

- Pervasive
- Dependable
- Consistent
- Inexpensive

How are Grids being used today?

- Grid successes are:
 - EP : Seti@home, Napster, Condor (sort of)
 - Resource selection: Genie project (NPACI)
 - Supercomputing demos
 - Limited production runs by dedicated groups of heroic users (CACTUS, ATLAS, CMS)

Moral:

- Users will only come when they have decent tools
 - simple enough for “easy” use
 - robust enough for “stupid” use
 - still allow work arounds for “hard-core” use
- But until we have basic functionality, we can't have tools

3. What about basic functionality on the Grid?

- Can't have higher level tools until you have the basic functionality
- Eg scheduling (brokers):
 - Resource discovery
 - Information access (meta information)
 - Job startup and monitoring
 - Migration, fault tolerance
 - All the trouble associated with data

Examples of basic functionality

- Ability to run a job on any system with the same command
- Ability to transfer files seamlessly
- Easy access to current dynamic information

Basic OS functions

- Process control, scheduling
- File system
- Memory management
- Security
- Accounting

Globus as an example

- Process control
 - globusrun, GRAM
 - No higher-level scheduling
- File systems –
 - Data replica work is almost a read-only FS
- Memory management
 - GASS and GridFTP for file transfers
 - mpich-g2 for communication
- Security – in a couple slides
- Accounting- open issue

Moral:

- Without basic functionality, there cannot be user level tools

4. Why don't we have usage scenarios?

- Software often doesn't do what a user wants
- One example- original replica catalogue from Globus, logical name to physical file name mapping
 - The way the developer envisioned the software being used was/is very different from how the user wants to use it

What is a usage scenario?

- Information from the user about a specific use case
- What's the right level of detail?
- What's a "general" use case?
- Who does this?
 - Application scientists and computer scientists speak different languages (eg. C. Pancake)

Moral:

- Without better communication between developers and users, the Grid cannot succeed
- Grid is about people, not just machines

Overview

- User issues
- Information
 - Getting data
 - Understanding data
- Security
- Performance
- Socio-Political
- Other issues

Information

- The Grid **IS** information
 - How do we find out about it?
 - How do we understand what it is?

5. Where do we get information from?

- Open question – how should I store the information about a Grid?
 - Globus Monitoring and Discovery Service (MDS)
 - Some implementation of the Grid Monitoring Architecture (GMA)
 - A Grid-wide data base?
- All of these are right for some of the data, no one is right for all uses

Need for Standard Interfaces

- Need for standard APIs and protocols to allow easier
 - Access to data
 - Registration of data
 - Archiving tools
- Standards for what information is available
- Standards for communication of errors

Moral:

- Without information about the Grid, it will not be usable
- This should be of primary importance to resolve

6. How do we understand information once we get it?

- Assume we have access to information about the Grid – can we use it?
- A monitoring system says “the load on machine X is Y”
- A scheduler wants to evaluate this data
- No common language for this to be communicated
 - Some effort now to come up with a common schema (GLUE schema, work with CIM in GGF) but this only touched the surface

Moral:

- Without some kind of standards or agreements, all the information in the world won't do us any good

Overview

- User issues
- Information
- Security
- Performance
- Socio-Political
- Other issues

7. How do we make Grids secure?

- Without security we can't have a Grid
- EVERYTHING needs to be secure-
 - Who can run on a machine
 - File transfers
 - What data does someone have access to (program data, system data)
 - Who can run which tools?

Security vs. Usability

- Users want security but don't want to deal with it
- Most security (including Grid Security Infrastructure (GSI)) is based on public key infrastructure (PKI)
- Users have files (public and private keys) that must be secure, use reasonable passwords, etc.

What about...

- Multiple certificates?
- Group access?
- Dynamic policy changes?
- Scalability?
- Etc., etc., etc...

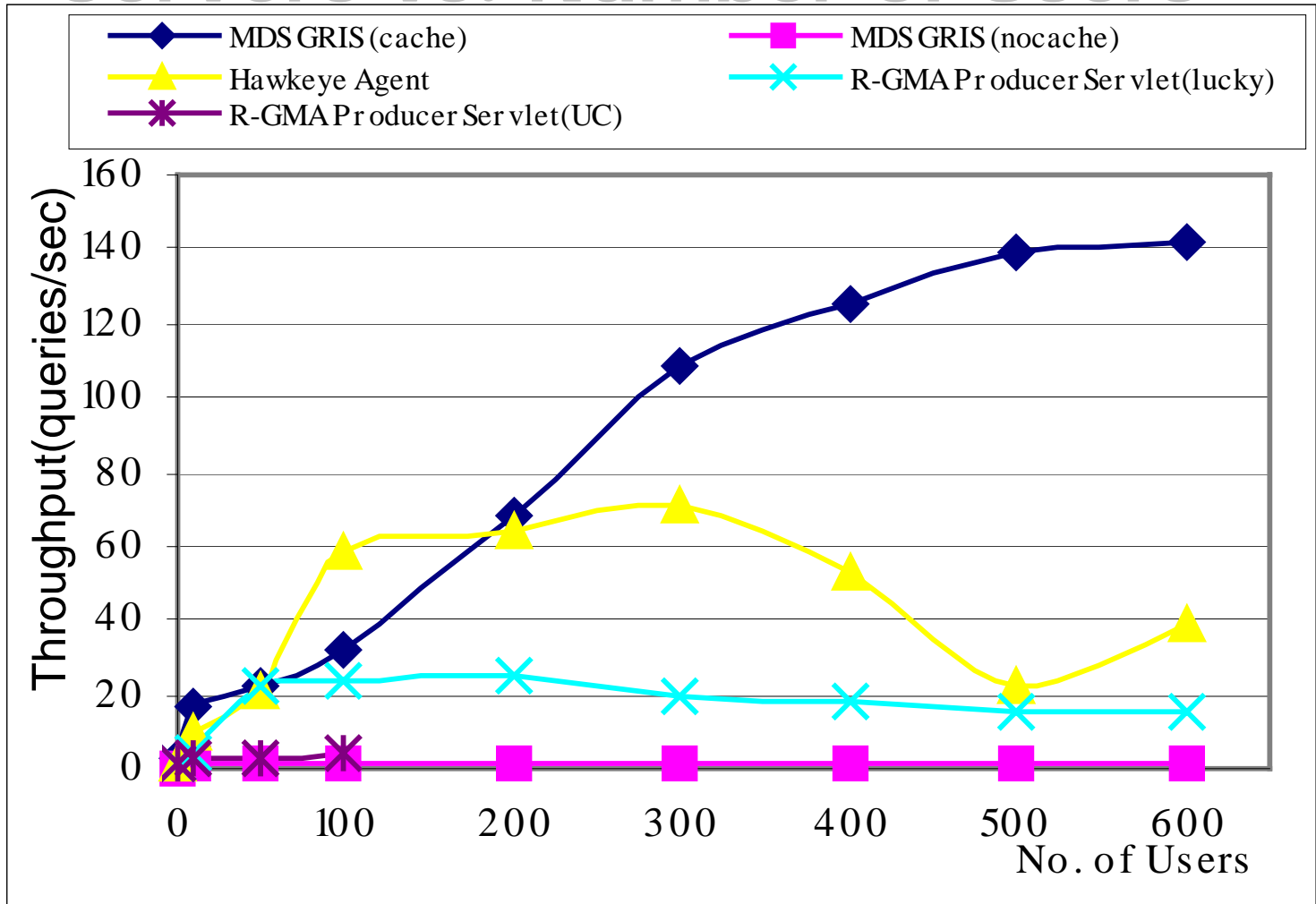
Moral:

- Until security is made easier to use, it won't be used
- Until security is made easier to manage at the group level, it won't be used
- Without security no one will really use the Grid

8. What about performance?

- It's not enough to use the Grid, it has to perform – otherwise, why bother?
- First prototypes rarely consider perf.
 - MDS1–centralized LDAP
 - MDS2–decentralized LDAP
 - MDS3–decentralized Grid service
- Often performance is simply not known

Performance of GIS Information Servers vs. Number of Users



What we found

- Performance can be a matter of deployment
 - Effect of background load
 - Effect of network bandwidth
- Performance can be affected by underlying infrastructure
 - LDAP/JAVA strengths and weaknesses
- Performance can be improved using standard CS techniques
 - Caching; multi-threading; etc.

Moral:

- Performance should be analyzed early and often
- Prototypes should be recognized as such and thrown out
- Without performance, no reason to use a Grid

9. Socio-political Issues

- Hardest problems are often not technical ones
 - Multiple administration domains means multiple policies
 - Multiple countries means multiple communication styles
 - Decisions are often made on non-technical basis

Communication is hard

- Too many people in the mix
 - Not everyone is informed of status updates
- Often hallway conversation becomes what people believe
- Too often assumptions are not verified
- Many communication styles can lead to misunderstandings

Moral:

- Ongoing efforts to continue better communication are needed to build a global community
- When in doubt – ask someone of directly!

10. Other open problems

- Where are the benefits to encourage sharing on the Grid?
- How do we educate the funding agencies about the need for the basics?
- What cost models are needed by the Grid?
 - Economic Grids may not be the answer
- What do we do about variance?

Progress

- Bill Johnston book on setting up a testbed
- Significant improvements in security infrastructure
- Basic functionality is much closer
- More funding aid for support
- Need for better-defined use cases and simpler deployment has been strengthened, as has the need for basic information and basic information services

Where are the performance metrics for success?

- No more “Grid” papers, just a footnote that states “This work was achieved using the Grid”
- Supercomputer centers don’t give a user the choice of using their machines or the Grid, that line doesn’t exist
- SuperComputing demos can be run at any time of the year

Conclusion

- Many interesting problems are left – both in terms of research and deployment issues
- Much work is being done to help address these open issues
- Next year's open issues will be very different

References

- This talk:
 - www.mcs.anl.gov/~jms/Talks (not there yet)
- Journal paper version of this talk:
 - www.mcs.anl.gov/Pubs/jmspubs.html
- Globus:
 - www.globus.org
- GGF:
 - www.ggf.org
- GMA:
 - www.ggf.org/Documents/GFD/GFD-I.7.pdf
- GLUE schema:
 - <http://www.hicb.org/glue/glue-schema/schema.htm>
- Bill Johnston
 - www.itg.lbl.gov/~johnston/Grids/ImplementingProductionGrids.v6.1.pdf

Contact Information

- Jennifer M. Schopf
- jms@mcs.anl.gov
- www.mcs.anl.gov/~jms

- Support from DOE, NASA, NSF, IBM, Microsoft

7. What do we do about variance?

- Resources on the Grid change with time
 - Bandwidth
 - CPU load
 - Disk space
 - Memory usage
 - Queue sizes

Variance – technical problem

- How do you tell if something is slow versus broken?
- How do you make a prediction?

Variance – socio-political

- Users want the same application to take roughly the same amount of time every time you run it
- Our experience – a longer running time that's more predictable is preferred to a high variance, high risk situation

Moral:

- Variance is here to live with, we need techniques to take advantage of it

8. How do we set up a Grid testbed?

- Bill Johnson, LBNL, talks about this often, based on IPG experience
 - Get the sys admins involved
 - Have a standard set-up
 - Make this a priority at the start of a project
- Accounting – open issue
- Installs are easier: from SlashDot ([www.slashdot.com](http://www slashdot.com)) last week: “Globus is f**king trivial to install (and I'm not on their staff).”
 - However many still disagree

Moral:

- Users are building testbeds, but this is still hard
- Need to have rule of thumb published for assistance with this
 - Upcoming book by Bill Johnson

