open middleware infrastructure institute

Real Users – Real Requirements: 25 Conversations with UK eScience Projects

Jennifer M. Schopf

Argonne National Laboratory

UK National eScience Centre

Steven Newhouse, OMII

My Definition of Grid Computing

- Resource sharing
  - Computers, storage, data, sensors, networks, …
  - Sharing always conditional: issues of trust, policy, negotiation, payment, …
- Coordinated problem solving
  - Beyond client-server: distributed data analysis, computation, collaboration, …
- Multiple administrative domains
  - Multi-institutional "virtual organizations"
  - Community overlays on classic org structures
  - Large or small, static or dynamic

Growth

- Computational Grids are becoming more and more common
  - 10,000+ downloads of Globus software
  - 3,000+ downloads of OGSA-DAI
- Collaborations are being developed
  - EGEE uses VDT (based on NMI)
- Governments are giving lots of money
  - 250M pounds of UK eScience funding
- **Great need for requirements gathering from users**

Need For Data

- Grid projects need to better understand what users need
  - More and more Grid projects
  - Success stories still few and far between
- Steven Newhouse and I spoke with 25+ UK eScience projects and attended several additional meetings to gather requirements data (July ’04)

We Met With:

- Oxford Security Workshop
- Networking for Non-Networkers Workshop (NNFN)
- Grid Service Workshop
- R. Baldrock, NeSC, Mouse Atlas
- M. Baker, Portsmouth, OGSI testbed
- R.Baxter, EPCC eDIKT
- N. Chue Hong, EPCC, OGSA-DAI
- D. Colling, IC, GridPP2
- T. Cooper-Chadwich, Southampton, gYacht
- S. Cox, Southampton, GeoDise
- M. Daw, Manchester, AG
- W. Emmerich, UCL, eMinerals & OGSI testbed
- M. Ghanen, UCL, DiscoveryNet
- M. Giles, Oxford, gViz
- S. Lloyd, Oxford, eDiamond
- C. Goble, Manchester, MyGrid & Integrative Biology Project
- J. MacLaren, Manchester, UoM Broker
- A. Martin, Oxford, ClimatePreciction.NET
- M. McKeown, Manchester, OGSI:lite and WSRF:lite
- S. Pickles, Manchester, TeraGyroid & GRENADE
- A. Porter, Manchester, Reality Grid
- A. Rector, Manchester, CLEF
- M. Rider, Manchester, eViz
- R. Sinnott, Glasgow, BRIDGES
- L. Smith, EPCC, QCDGrid
- T. Sloan, EPCC, INWA
- L. Yang, B. Yang, NeSC, AI Workflow

User Characteristics

- We met with
  - Current application developers with some Grid or Web Services experience
  - Those with software that might be of broader use or interest
  - Those who have expressed dissatisfaction with current tools
- Note: Users – *not* system administrators
- Note: Mostly on project specific Grids, NOT larger deployments (TG, LCG, TeraGrid)

What We Found

- Need for Training
- Security
- Functionality
  - Jobs
  - Data
  - What isn't mentioned
- What tools should look like
- Infrastructure/Operations

Training

- Grid vision still needs to be sold –
  - "What do these tool give me over SSH, scp?"
  - "What if I don't want to stop using my magnifying glass to read x-rays?"
- Still need basic common practices to be written: for user, developers *and* admins
  - Web service basics
  - Firewalls
  - Builds and packaging
  - How do I make my service secure?
- No surprise: Communication is still a large unsolved problem in Grid computing

Security: What's Old News

- If it isn't easy users aren't interested
- All users hate firewalls, all system administrators love them
- Anonymizing data is hard
- Still need a lot of information sharing:
  - How fire walls interact with Grid/Web services
  - Security audits

Security: What's Surprising Us

- User focus on need for data integrity not authentication/authorization
  - Time and again this was mentioned
- Delegation seems to be the next big question
  - GT2-style delegation needed in a services world
  - No one has an agreed upon solution yet
- Way to verify that your security is secure

Questions?

- Need for Training
- Security
- Functionality
  - Jobs
  - Data
  - What isn't mentioned
- What tools should look like
- Infrastructure/Operations

Functionality

- Prediction of Pete Beckman (TeraGrid):
  - All users want is SSH, scp, and top

Functionality

- All users talk about is job submission and file transfer capabilities
- When asked about trouble spots they also want tools to tell how jobs are progressing
- Some user developed tool "add-ons", but strongly tied to project domain and narrowly scoped
  - Eg. Viz tools, replica management policy tracker, data format translations
- Many other tools/functionality/services considered farther out, but simply don't seem to be on the 6 month horizon for the users we've spoken with

Job Submission: No Surprises

- Want simple, dependable "run my application" interfaces
  - This was identified at GF1!
- Only resource discovery is "small"
  - How many nodes have a matlab license?
  - NOT: which cluster should I use?

Job Submission: Urgent Needs

- Tools to understand errors while a job is running- something stopped, where and why?
  - TeraGyroids' use of SSH for debugging
  - Need for Global Job Unique ID
- What to do when a job fails-
  - Resubmit or ignore?
  - Workflow issues
  - Steering

File Transfer

- People seem pretty happy with GridFTP
- Some reliability (RFT) would fill out rest of use
  - This needs 3rd party transfer (delegation)
- Some projects starting to work with provenance issues, access to databases, replication
- Still issues with performance and making sure background infrastructure is all as it should be – more later

What (These) Users Aren't Talking About…

- Notification – except for job progress tracking
- Registries or resource discovery
- Reservations, brokering, co-scheduling, other advanced scheduling techniques
- Job migration, checkpointing
- Accounting and pricing (but we're talking with users, not admins so far)
- Data migration
- Instruments

…And Why We Think This Is

- A gap still exists between the computer science research and tool building community and the average user
- Large difference between short term needs and long term planning-
  - Most users are still trying for basic functionality and dealing with today's hurdles
  - Most researchers are looking at the greener pastures a few years out

User's View of Tools

- Users have strong opinions on tools – what a surprise!
- Mostly known problems, but the prioritization of certain aspects wasn't known

Tools Should Be

- Vertical solutions
  - End to end use cases, not horizontal pieces that don't work together
- Simple
  - One job, one tool – think unix!
  - Work easily for the 80% case, and rest is possible if needed
- Ease of use/installs
  - Bundle all together so you have entire use together
  - Don't reinstall things I already have
- Acknowledgement that there may not be ONE best tool

Composable Functionality

- Lego blocks of basic functions
- "Shims" to fit between where needed
  - API mismatches
  - Data translation
  - Interfaces to legacy code
  - SOAP Lab (wraps command line to look like SOAP)

Interfaces

- Need simple APIs at the user level
  - eg. SAGA-RG (but then we knew this)
- User API might sit a layer above standard tool APIs to mitigate upgrade effects
  - eg. HiCog
- The API a user sees and the API the infrastructure not only can but should be different – different goals

Tool Environments

- Tools need to fit in with existing "user comfort zone"
  - Biologists like Perl
  - CFD folks like MatLab
  - HEP (EGEE) are used to Python
- This is a sys admin's and tool developer's nightmare – but for usability it's a must

This Talk

- Need for Training
- Security
- Functionality
  - Jobs
  - Data
  - What isn't mentioned
- What tools should look like
- Infrastructure/Operations

Builds and Upgrades

- Need for a reproducible build
  - Hands off process, works every time
  - Verification tools
- Need better understanding of effects of upgrades, or else users don't want it
  - What will change, what will be affected
- Tools are being used "off label"
  - Tool for usecase A in common use for usecase B
  - Scalability becomes an issue
  - New/different functions needed

Understanding System Stability

- Need for basic tools to verify functionality and performance
- "I can't transfer my files today"
  - What's broken?
  - What changed?
  - How do I fix it?
  - Why couldn't someone find this before me?
- Strong needs for quality assurance tests on all platforms – clusters, networks, AG, etc.

System Tests

- Often system tests don't look like current applications
  - Tests for firewall functionality don't include checks for all ports in current use
  - System benchmarks don't look like "my" application
  - Ping tests aren't enough to assure that a GridFTP transfer will work
- Need for better testing, verification- for the user, and even by the user!

"WebMD" for Grid Applications

- Basic diagnostics needed for users
- Q - "I'm trying to transfer a 1 gig file between A and B and can't"
- A- "Is your cert ok? Here's how to check"
  Y/N, if Y…
- A – "Is the route between you're hosts up? Here's how to run traceroute for your system…"

What We Found

- Need for Training
- Security
- Functionality
  - Jobs
  - Data
  - What isn't mentioned
- What tools should look like
- Infrastructure/Operations

Current Gaps For Users (In No Particular Order)

- Training and education, esp. security
- Delegation for web services
- Job tracking
- Dependable builds
- Wrappers for usability
- Composability of functionality
- Verification and instability analysis
- User-oriented diagnosis tools

The Point

- We can’t say it any more simply
- Grid tool developers must continue to talk and interact with application scientists – without them, we are nothing

Towards this end…

- As a continuation of the Performance Workshop
  - June ’05 NeSC
- Encouraging discussions between performance tool builders and applications scientists
- BOF at SC

How To Find Us:

- Jennifer M. Schopf
  - jms@mcs.anl.gov
  - jms@nesc.ac.uk
- Steven Newhouse
  - s.newhouse@omii.ac.uk

- Thanks to all the users for their frank discussions!

Longer version of the paper available from www.nesc.ac.uk/technical_papers/UKeS-2004-08.pdf