

The TeraGrid

Applied Grid Infrastructure that is
Persistent and Reliable



Pete Beckman
Charlie Catlett

Argonne National Laboratory
University of Chicago

Acknowledgments:

The TeraGrid is Big... Really Big...

- Director: Rick Stevens, ANL/U Chicago
- Executive Director: Charlie Catlett, ANL/U Chicago
- Chief Architect: Pete Beckman, ANL/U Chicago
- Fran Berman, SDSC
- Jim Pool, Caltech
- Rob Pennington, NCSA
- Ian Foster, ANL/U Chicago
- Mike Levine, PSC
- Ralph Roskies, PSC
- Site leads: JP Navarro (ANL), Mark Bartelt (CIT), J. Ray Scott (PSC), Tom Cockeril (NCSA), Phil Andrews (SDSC)
- + 40 or 50 more....



Goals For The 2 Hour Presentation

- Fun
 - ◆ This presentation will be the best
- Practical
 - ◆ No over-hyped claims (starting now)
 - ◆ No vaporware or claims how Grid computing will reinvent the Internet
- Comprehensive
 - ◆ The practical design, construction and operation of large-scale Grids for scientific computation
 - ◆ Not about technology (WSRF, Dagman, which service to run on what port, or why to choose Myrinet over Quadrics)



Yes, Grid Hype has made Everyone's Job Harder

- **NYT:** “Grid computing, a concept that originated in supercomputing centers, is taking a step toward the mainstream: Sony will announce today that it will use the technology to accelerate its push into the emerging market for online games with thousands of players at a time.”
- **Financial Times:** “Grid computing involves yoking together many cheap low-power computers to create a system with the high processing power typical of a large supercomputer, at a fraction of the price.”

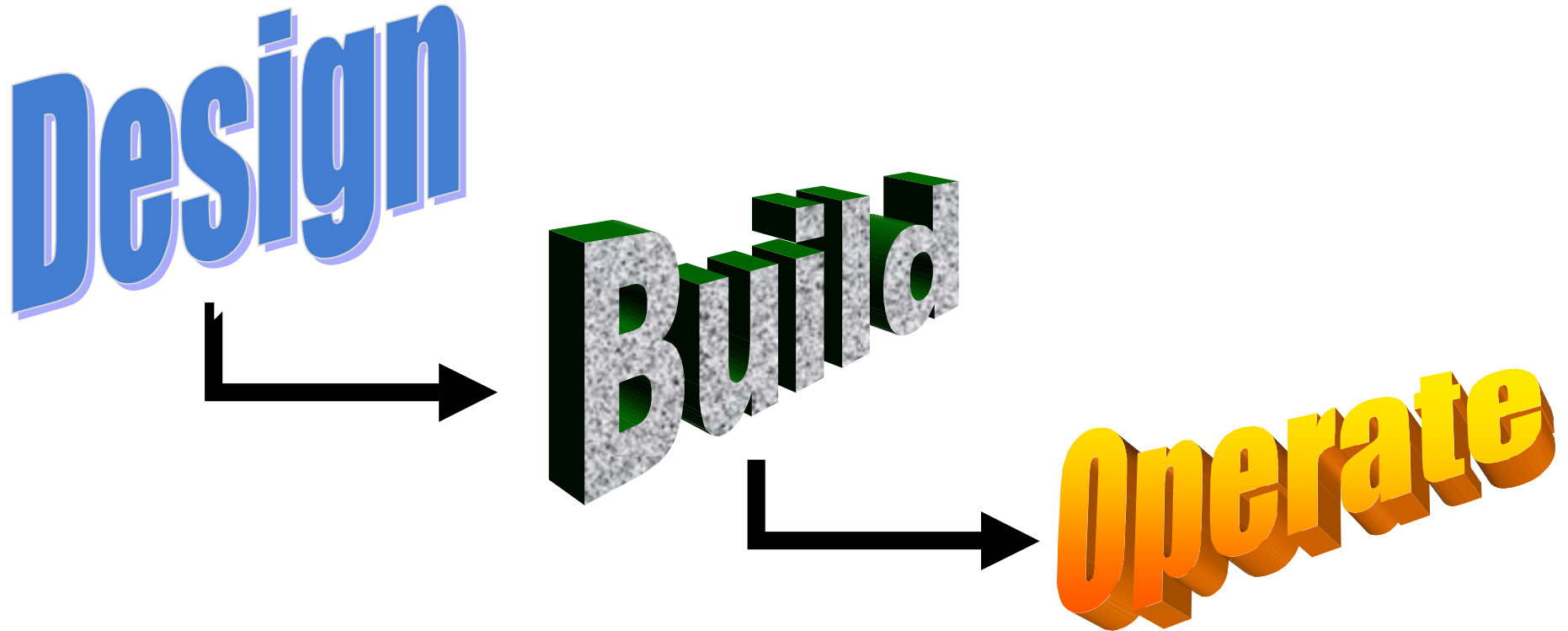


Hype Hype Hype

- **Technology Review:** “The coming explosion of activity could create a world of interlinked computer Grids – a development dwarfing the Internet boom of the 1990s.”
- **Dan Farber@ZDnet:** “On my hype meter, grid computing receives a rating of 6.5 on a scale of 10”
- **Gartner Report:** “Biometrics, Web services and grid computing are the most hyped technologies”



Forget The Hype: Applied Grid Infrastructure



- Architects, Builders, Operators
 - ◆ Passing from role to role over time

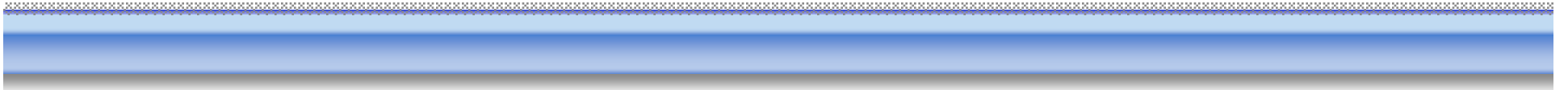


Argonne/U Chicago

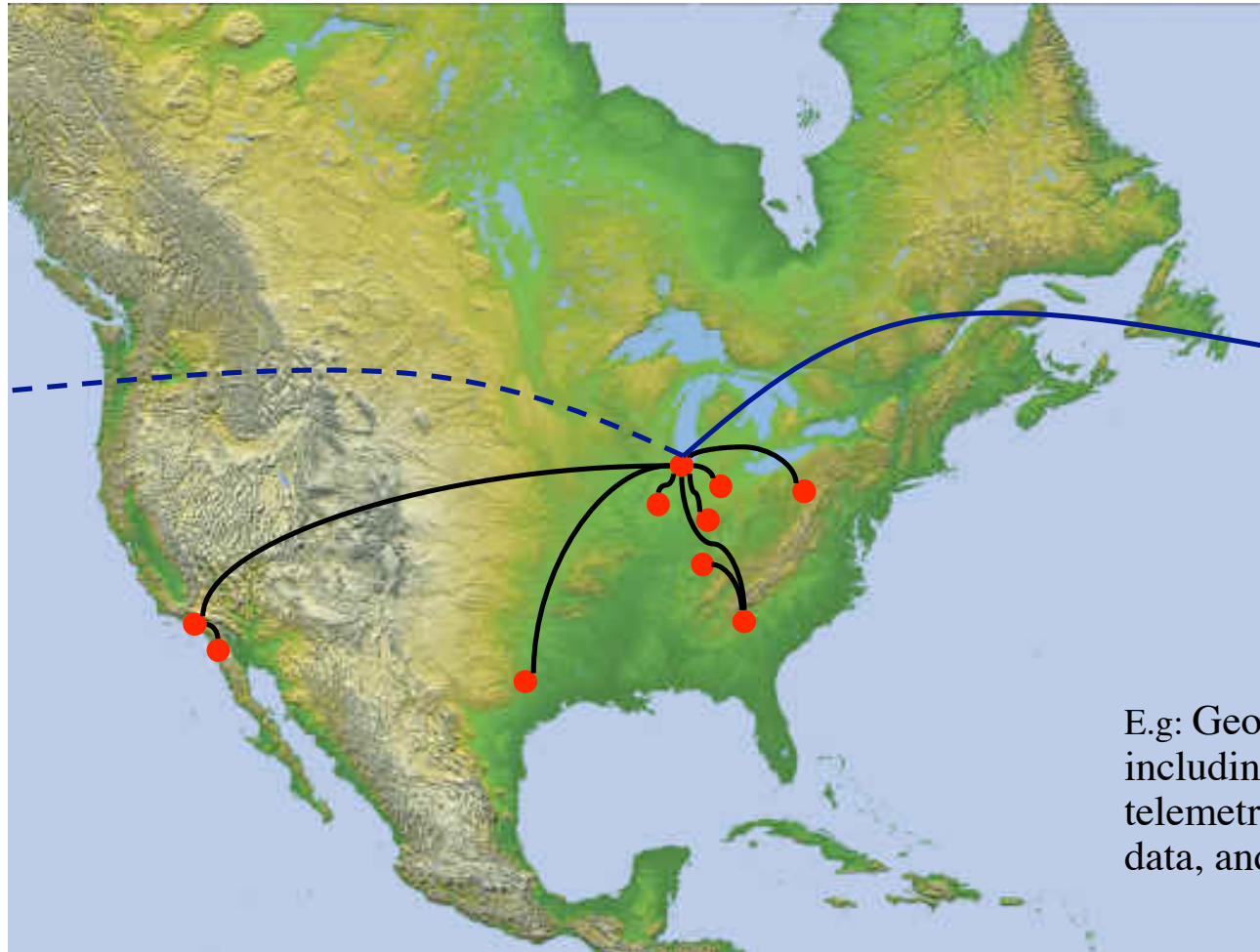
Pete Beckman <beckman@mcs.anl.gov>
Charlie Catlett <cec@uchicago.edu>



What Is The TeraGrid?



TeraGrid Vision: A Unified National HPC Infrastructure that is Persistent and Reliable



- Largest NSF compute resources
- Largest DOE instrument (SNS)
- Fastest network
- Massive storage
- Visualization instruments
- Science Gateways
- Community databases

E.g: Geosciences: 4 data collections including high-res CT scans, global telemetry data, worldwide hydrology data, and regional LIDAR terrain data

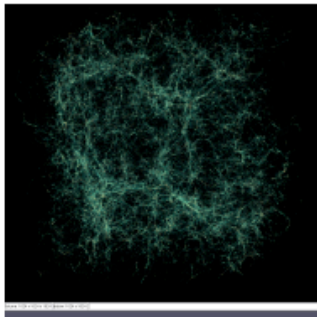


Argonne/U Chicago

Pete Beckman <beckman@mcs.anl.gov>
Charlie Catlett <cec@uchicago.edu>

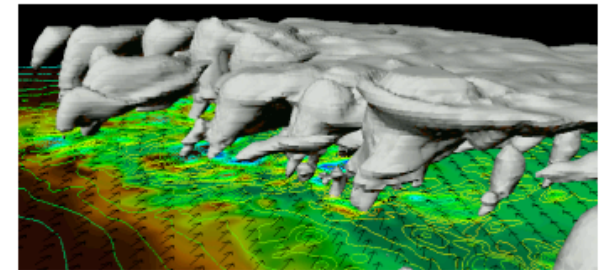
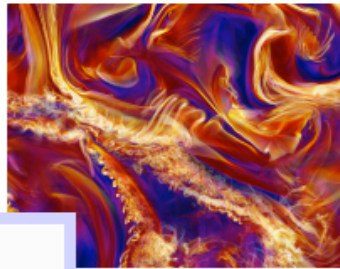


Enabling Scientific Discovery Across a Broad Spectrum of Science

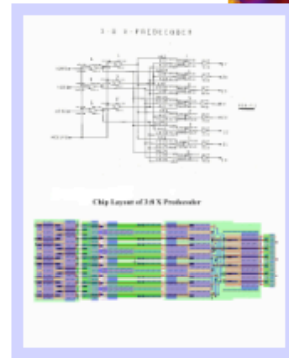


ENZO
(Astrophysics)

PPM
(Astrophysics)



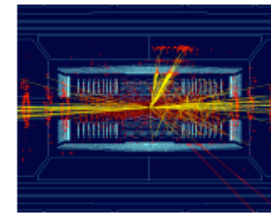
MEAD (Atmospheric Sciences)



GridSAT
(Computer Science)

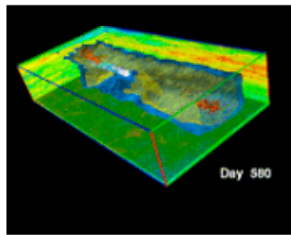


AtlasMaker
(Astronomy)

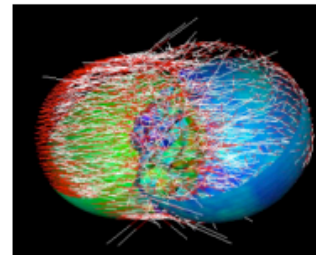


**CMS/
GriPhyN**
(Physics)

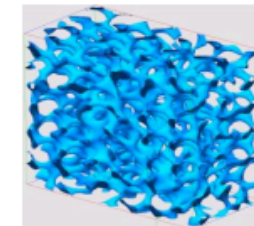
GAFEM
(Ground-water modeling)



Encyclopedia of Life
(Biosciences)



VTF
(Shock Physics)



TeraGyroid
(Condensed Matter Physics)

BioCoRE (Biomedicine)
Biological Collaborative Environment



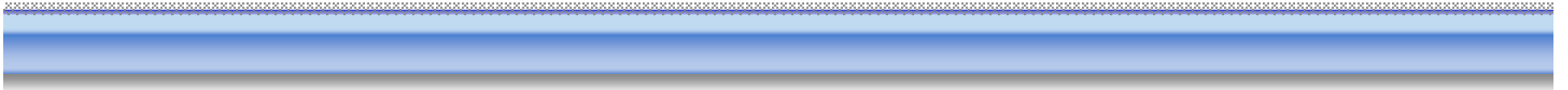
Argonne/U Chicago

Pete Beckman <beckman@mcs.anl.gov>
Charlie Catlett <cec@uchicago.edu>



Was That Just Hype?

What IS The TeraGrid?



Distributed Computing Infrastructure =

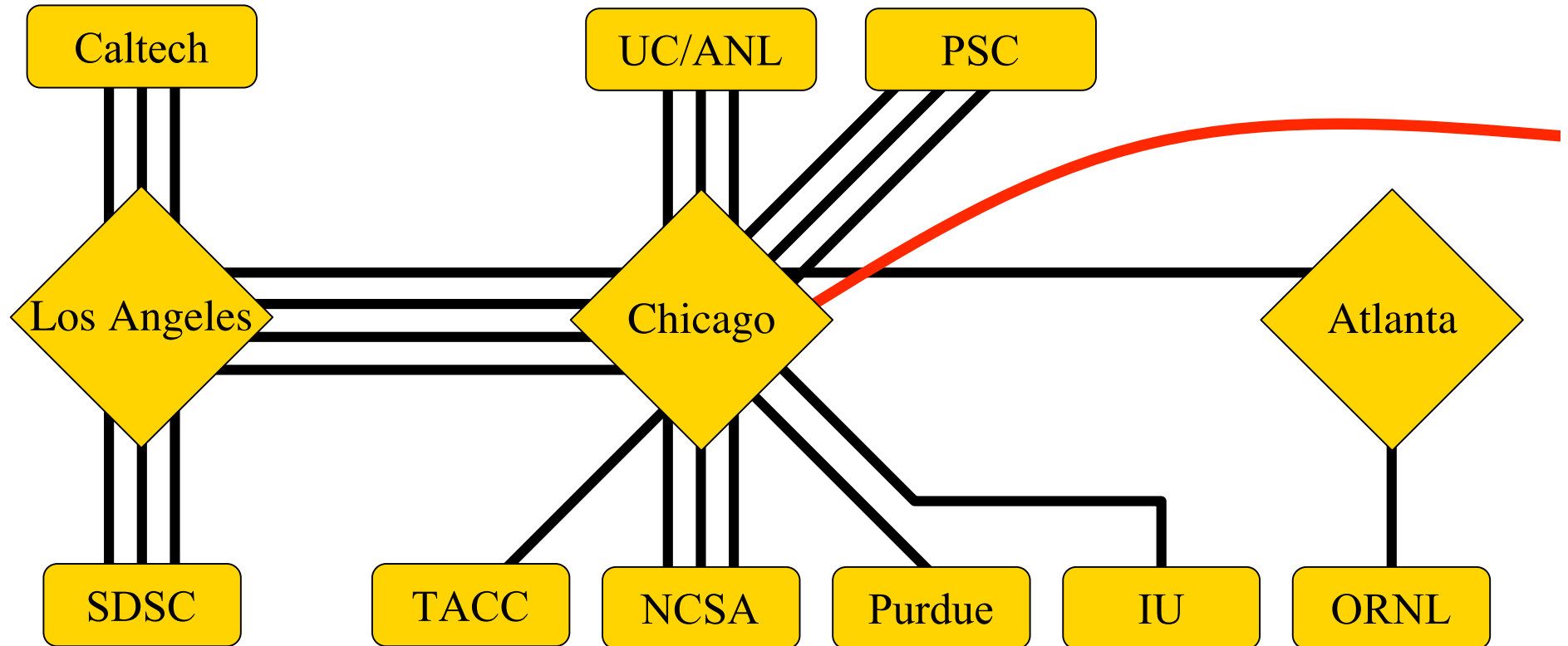
Raw Materials + Policy + Software + Services

Resources and Services (33TF, 1.1PB disk, 12 PB tape)

	UC	Caltech	IU	NCSA	ORNL	PSC	Purdue	SDSC	TACC
Computational Resources	Itanium2 (0.5 TF) IA32 (0.5 TF)	Itanium2 (0.8 TF)	Itanium, IA-32 (2.1TF) Power4+ (1TF)	Itanium2 (10TF)		TCS (6TF) Marvel (0.3TF)	Heterogeneous (1.7 TF)	Itanium2 (4 TF) Power4+ (1.1 TF)	IA-32 (5.2 TF) Sun (Vis)
Online Storage	20 TB	170 TB	6 TB	230 TB		150 TB		540 TB	50 TB
Archival Storage			150 TB	1.5 PB		2.4 PB		6 PB	2 PB
Networking (Gbps to hub)	30 Gbps CHI	30 Gbps LA	10 Gbps CHI	30 Gbps CHI	10 Gbps ATL	30 Gbps CHI	10 Gbps CHI	30 Gbps LA	10 Gbps CHI
Database & Data Collections			Yes	Yes			Yes	Yes	Yes
Instruments			Yes		Yes				
Visualization	Yes		Yes			Yes	Yes		Yes
Dissemination	Yes	Yes	Yes	Yes		Yes		Yes	Yes



Current TeraGrid Network



Resources: Compute, Data, Instrument, Science Gateways



Argonne/U Chicago

Pete Beckman <beckman@mcs.anl.gov>
Charlie Catlett <cec@uchicago.edu>



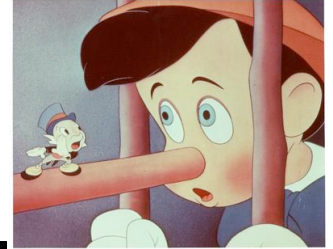
The Difficult Parts

- Software
- Services
- Policy



Why Building a Production Grid is Hard

- We perpetuate a great big fat lie:
 - ◆ Technology solves problems
 - “Grid Technology builds collaboration...”
- Truth: Grid technology is raw, it only enables new procedures and operational methods
 - ◆ Installing “Globus technology” does not create a Grid, just like installing Apache does not create an e-commerce site
- An enormous amount of architecture, design, and process management is required to convert technology into a solution for users



Part 1

Design: What is it Supposed to do?

Design



Argonne/U Chicago

Pete Beckman <beckman@mcs.anl.gov>
Charlie Catlett <cec@uchicago.edu>



The Art and Science of Design

- Who are the consumers?
 - ◆ HPC Users
 - ◆ Funding agencies
- What are their expectations?
 - ◆ Grid hype, overselling, “and then a miracle occurs”
 - ◆ The art of negotiation...
 - Not covered in this talk, but can provide professional consulting services on an hourly basis
- What is your budget?
- What technologies will you choose?
 - ◆ Free software, \$\$\$ SW/support, build it from scratch



Job #1: Create Virtual Organization for Participants

Single, Distributed Team

Software

- Testing, QA, Verification
- Software Stack
- Advanced Networking
- Data Services
- Viz Services
- Accounting SW

Developer's Org

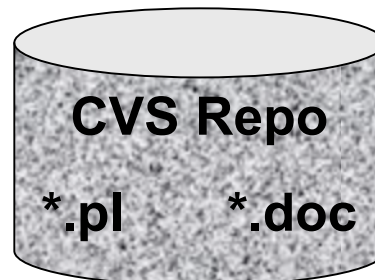
- Exec. Mgmt Committee
- Directors (Exec & Eng)
- Site Leads
- Working Group Leads
- Engineers

User-visible Org

- Operations Center
- User Services
- Accounting

It is not your code, it
is our code

It is not your doc, it
is our doc



**One repo to rule them all,
one repo to find them
one repo to bring them all
and in the grid-world bind them**



Argonne/U Chicago

Pete Beckman <beckman@mcs.anl.gov>
Charlie Catlett <cec@uchicago.edu>



Collaboration Structures: (strange: most are not Grid based...)

- Repository and version control: CVS
- Problem tracking: Bugzilla
- Mailing lists: Mailman
- Document Library: scripts
- Project Plan: MS Project
- Real-time conferencing: AG
- Software Repository: CVS

- Integrated Solutions? Savannah?
SourceForge?





- Cooperative Agreement:
 - ◆ The “Virt Org Bylaws”
 - How will decisions be made?
 - How will money be spent?
 - What are the approval processes?
 - How will resources be shared and billed?
 - How are disputes resolved?
 - What is Virt Org *not* responsible for?
 - ◆ Critical for real collaboration
- Mgmt & Org Chart
 - ◆ Director, Chief Architect
 - ◆ Site Leads, etc

Question:

**From a User's Perspective,
What Features are Needed
From A Distributed
Computing Environment
(Grid?)**

*(You may NOT mention the name of any
software packages, only capabilities/features)*

TeraGridness!!

What “Grid” means in 23 languages

- Parameter sweep interfaces
- Collab viz environments
- Viz steering tools
- Global, sync file space
- Resource queue and broker
- Client-side tools for apps
- Workflow/dependency tools
- Directory services/discovery
- Differentiated pricing
- Meta/Co scheduling
- MP between resources
- Remote database access
- Advanced reservations
- Non-CPU resource pricing
- Fast data movement tools
- Data streaming instruments
- Remote batch/interactive viz/rendering
- Remote data read/write
- End-user portal development tools
- Domain-specific community portals
- Single signon / delegation
- Unified accounting and single help desk
- User portals for projects



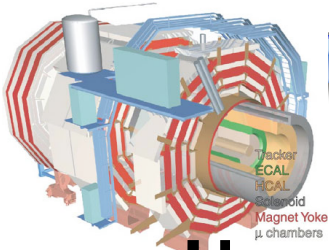
Design

- User Survey
- Assessment of roadmap and capabilities
- Analysis of technologies
- Gap analysis
- Policy framework
- Construction



Many Software Needs

Real User Examples:



User: Compact Muon Solenoid at CERN's Large Hadron Collider

- Workflow/dependency tools
- Meta/Co scheduling
- Remote data read/write
- Unified accounting and single help desk

User: Atlasmaker / Palomar-Quest

- MP between resources
- Fast data movement tools
- Remote data read/write
- Single signon / delegation



What Users Say:

Teragrid must be:

- Persistent and reliable
- Deliver useful features
- Easy to use
- Inclusive, leveraging many projects and strengths
- Powerful
- Not only for the biggest users

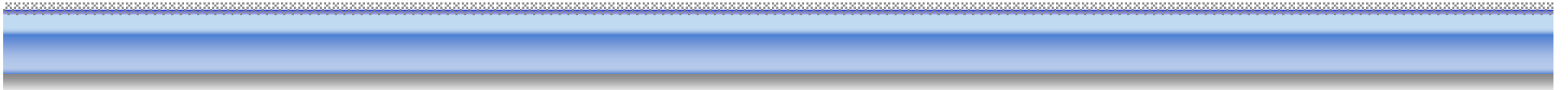


For The Consumer: It Must Be Easy³

- Even the smallest barrier to sharing prevents collaboration
- “Technologists call it revolutionary... users call it unusable”
- Benefits to collaboration less tangible
- Examples:
 - ◆ Dialing a few extra telephone digits
 - ◆ Sharing between laptops (using email)
 - ◆ Gopher / FTP / Archive verses the Web



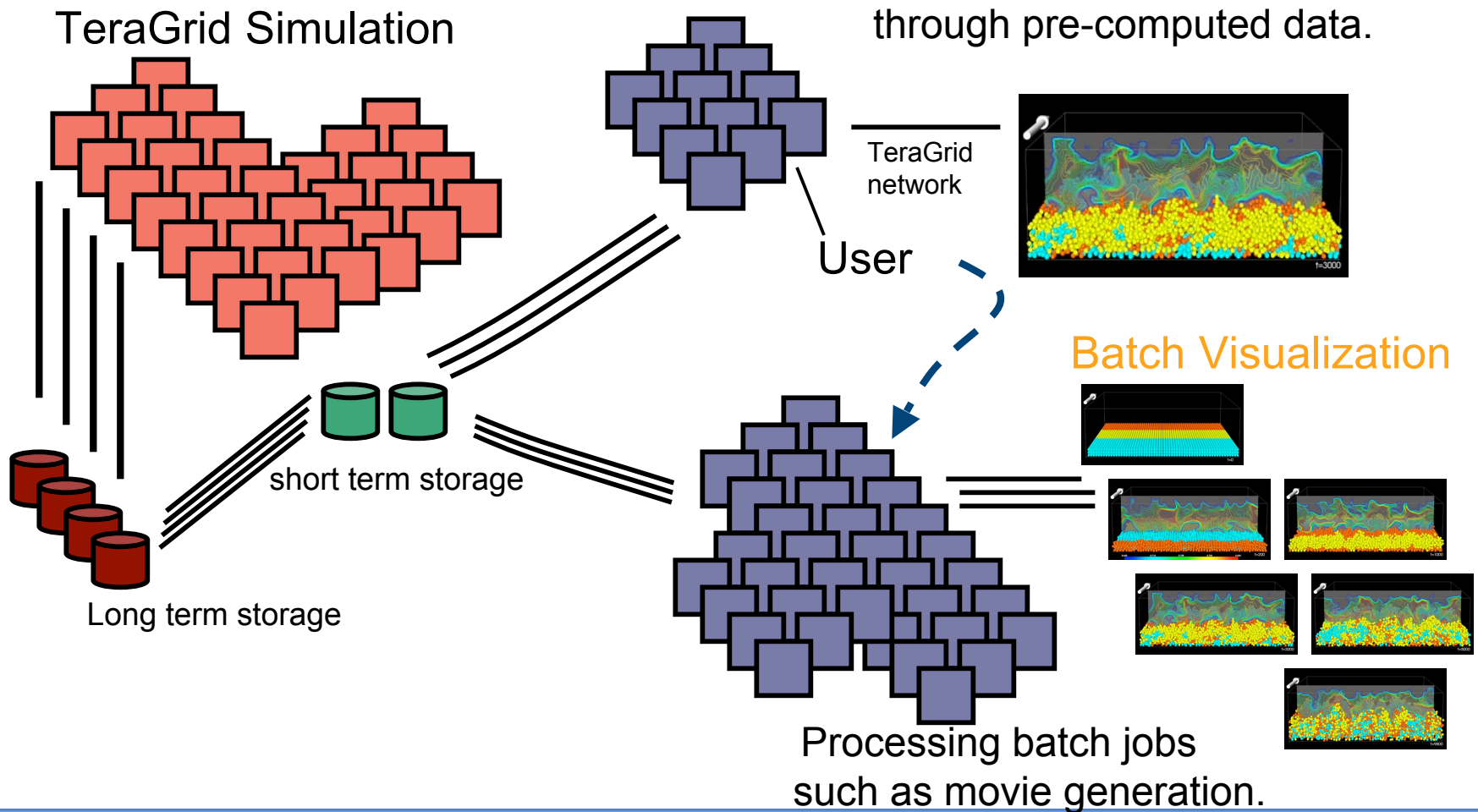
**...But Support Advanced
Features**



Two Types of Loosely Coupled Visualization

Interactive Visualization

Computationally steering through pre-computed data.



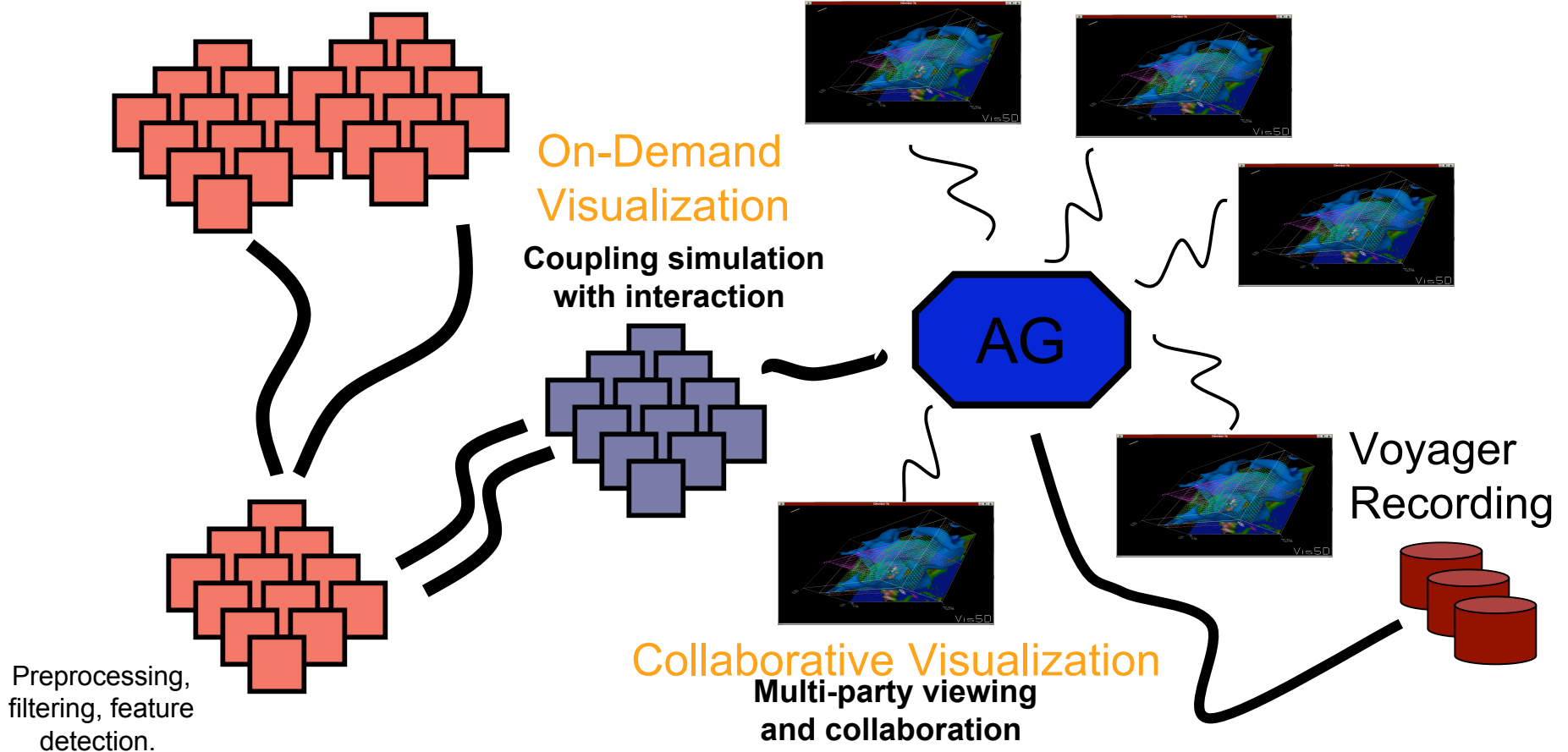
Argonne/U Chicago

Pete Beckman <beckman@mcs.anl.gov>
Charlie Catlett <cec@uchicago.edu>



On-Demand and Collaborative Visualization

TeraGrid Simulation



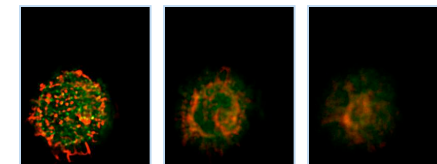
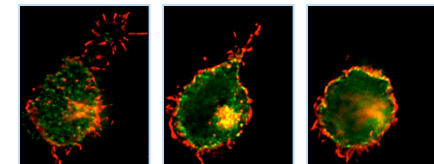
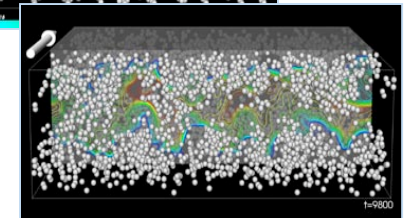
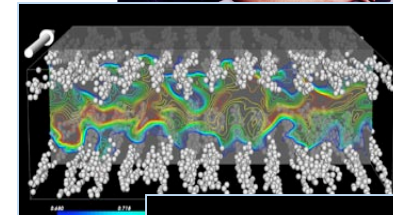
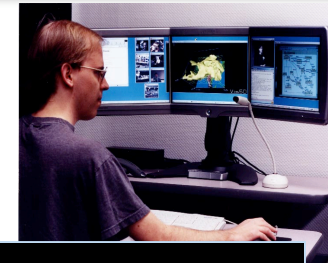
Argonne/U Chicago

Pete Beckman <beckman@mcs.anl.gov>
Charlie Catlett <cec@uchicago.edu>



Visualization – Sample Use Cases

App	Modality	Tools	Data
Collaborative Analysis of Atmospheric Simulations	Collaborative, Interactive Analysis.	AG 2.0 Grid VTK MPICHG2	GridFTP in. AG distribution
Interactive Visualization of Time-Dependent CFD Data	Collaborative, Interactive	AG 2.0 Grid VTK VisBench	GridFTP in. AG distribution
Volume Rendering for Production Visualization	Batch Rendering of Movies	NPACI Scalable Visualization Alpha Project Volume Render. Radiosity renderer. APST, PBS.	Dynamic, on-demand use of SRB.



Argonne/U Chicago

Pete Beckman <beckman@mcs.anl.gov>
Charlie Catlett <cec@uchicago.edu>



**Policies: Persistent,
Reliable, Easy to Use...**



TeraGrid Unified Policies, Common Currency

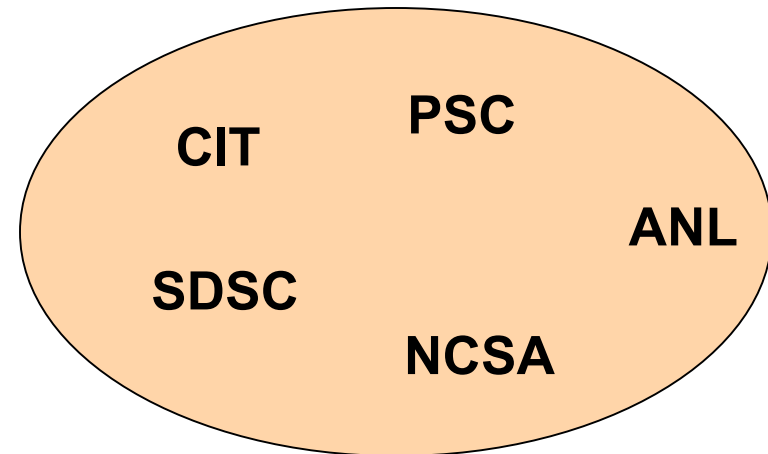
- ◆ One help desk
 - ◆ One allocation request
 - ◆ One account interface
 - ◆ *One user certificate*
 - ◆ *One accounting currency* (R/S allocations)
 - ◆ One set of user policies
 - ◆ One documentation set
- Result:
 - ◆ Improved usability, attractive development target
 - ◆ Unified networked resources more valuable to community (Metcalf)



“TeraGrid Roaming”

It must be easy, it must be easy, it must be easy
Nearly eliminate the barrier to entry

- Develop application at ANL, run at NCSA
- Run at CalTech with data from SDSC
- Run large job across all sites
- Unified accounting and billing
- Predictable levels of service



Launching a new service is hard

- Enormous investment
- Ubiquitous
- Easy
- Paying Customers

(remember Iridium?)



Argonne/U Chicago

Pete Beckman <beckman@mcs.anl.gov>
Charlie Catlett <cec@uchicago.edu>



A Grid Hosting Environment

An SLA for a Virt Org that hosts other Virt Orgs and Grid Applications

Example:

Web Hosting Env.

PHP, Perl, Python scripting.
MySQL, FrontPage
100 POP accts, 100MB disk
SMTP, IMAP & Webmail
US\$49 per year

Special Capabilities

Experimental math libraries
Unique storage system
Large shared memory arch....

TeraGrid Hosting Env.

Single Contact: help@teragrid.org
Unified Ops center
Certified Software Stack
MPICH, Globus
GridFTP, BLAS, Linpack,
Atlas, SoftEnv, gsi-ssh
\$110 Million
\$TG_SCRATCH, ...

Classic Unix-like Environment

■ /bin/sh, bin/cp, /bin/lis, Unix file system & tools, dev tools (make, compilers) etc



Argonne/U Chicago

Pete Beckman <beckman@mcs.anl.gov>
Charlie Catlett <cec@uchicago.edu>



Allocation & Accounting

- A single “Euro” or “Service Unit” for requesting time and tracking usage
- “Roaming” allocations may be spent anywhere
- “Specific” allocations to particular resources
- Every resource will provide “fair share” usage for R & S allocations
- Conversions between machines based on Linpack
- Refund policy



Scheduling & Differentiated Service

- Two basic user-specified job variables
 - ◆ Time for job to complete (parallelism, priority in queue, etc)
 - ◆ Reliability (scavenging, preemption, real-time-controls, etc)
- Varied Cost
 - ◆ Service Units charged for the job
- Differentiated services that will be built:
 - ◆ Support for on-demand computation
 - ◆ Support for backfill of cheaper jobs
- Technology Integration Required:
 - ◆ Cross-site organization of jobs, shared job attribute schema, pricing strategies connected to resource management, accounting, etc



TeraGrid Software Requirements

- A social contract with the user:
 - ◆ LORA: Learn Once, Run Anywhere
- Precise definitions:
 - ◆ Services
 - ◆ Software
 - ◆ User Environment
- **Reproducibility**
 - ◆ Standard configure, build, and install
 - ◆ Single CVS repository for software



Question:

What techniques are used to guarantee interoperability between software components and services at participating organizations?

Concepts for Production-Quality Grid Systems

- Formal specifications of supported environment and “operational” must exist
- A set of independent tests should check for correctness
- A human should not be “in the loop”
- Testing should include performance, and be archived over time for trend analysis
- Scalability: Adding new sites should require minimal resources



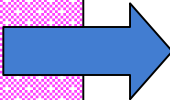
Joining The TeraGrid is Straightforward

- Identify resource (compute, database, instrument)
- Participate in Allocations process
- Integrate Account/Accounting software
- Support coordinated environment
 - ◆ Protocols, services, and software to support Distributed Computing, change management processes
- Participate in Operations Process
 - ◆ Automated testing, accept trouble tickets, account requests, certificates
- Join Security Infrastructure
 - ◆ Sign Security Memorandum, participate in reporting, vulnerability & risk analysis, etc.
- Have fun! Great Science!



Joining The TeraGrid is Straightforward

- Policy**
- Identify resource (compute, database, instrument)
 - Participate in Allocations process
 - Integrate Account/Accounting software

- Software**
- Support coordinated environment
 - Protocols, services, and software to support Distributed Computing, change management processes
- 

- Policy**
- Participate in Operations Process
 - Automated testing, accept trouble tickets, account requests, certificates
 - Join Security Infrastructure
 - Sign Security Memorandum, participate in reporting, vulnerability & risk analysis, etc.
 - Have fun! Great Science!



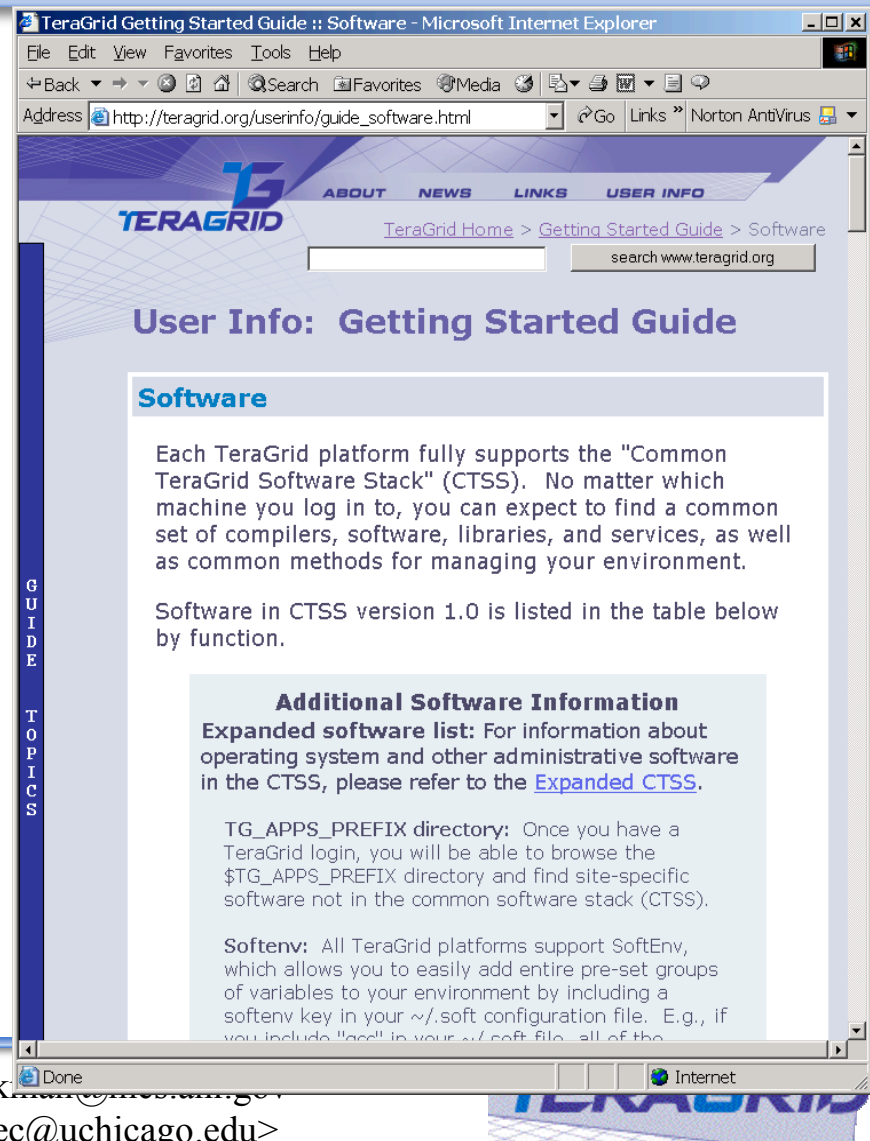
How TeraGrid Creates a Unified Infrastructure

- Start with Production HPC environments “Supercomputing Classic”
- Computer Scientist: First, do no harm
 - ◆ All TG capabilities must be **additional capabilities**, and **not replace or interfere** with existing projects and systems (inclusive!)
- Capabilities are prioritized based on maximizing impact to user community
- Tight coordination and change management so users have persistent and reliable systems



Coordinated TeraScale Software & Services (CTSS)

- CTSS Provides a single, unified set of interoperable components and services that define the TeraGrid's Hosting Environment and enable "TeraGrid Roaming"



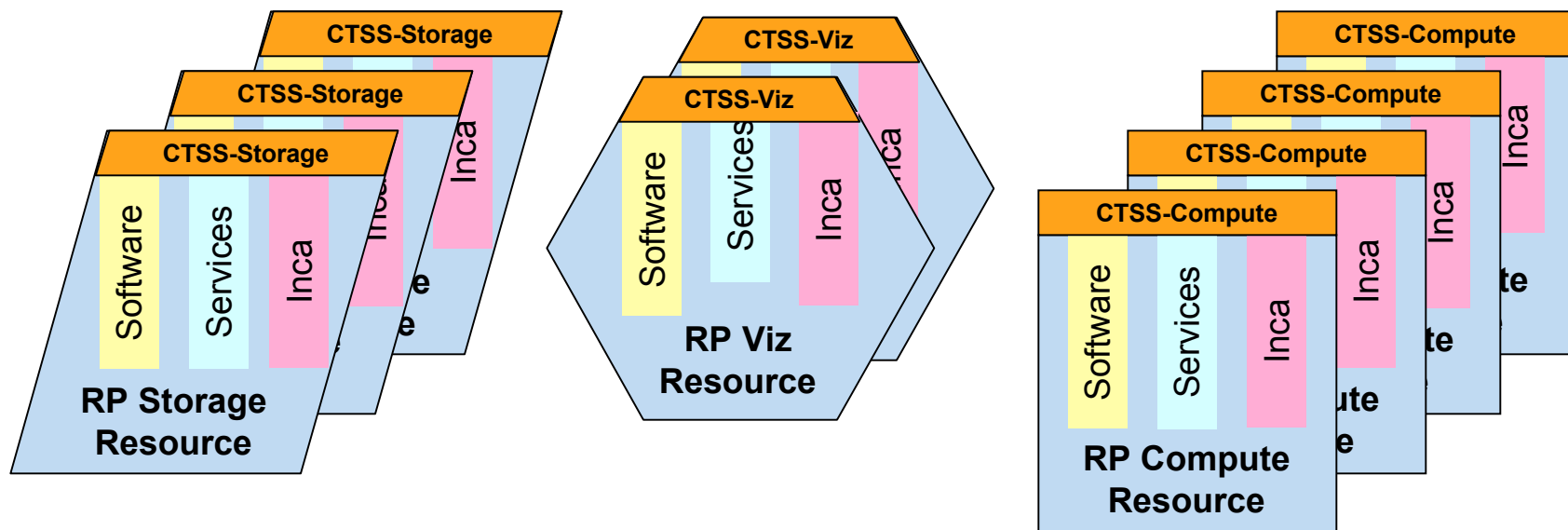
Coordinating Services and Software

- Users want middleware capabilities that “just work” across the unified infrastructure
 - ◆ GridFTP, SRB, Condor-G, MyProxy, etc
- **CTSS** coordinates protocols, services, and software to provide applications **reliable** and **persistent** infrastructure
- **Inca** provides production level quality assurances



Persistent, Reliable, TeraGrid Software

CTSS: A Single Shared Grid-enabled Infrastructure



- ◆ Learn Once, Run Anywhere: Users assume WE will manage the complexity
- ◆ “Grid Hosting” permits application groups to reliably target TeraGrid to host grid applications suites which could be made available in many ways:
 - portals, community maintained applications, grid services, etc.

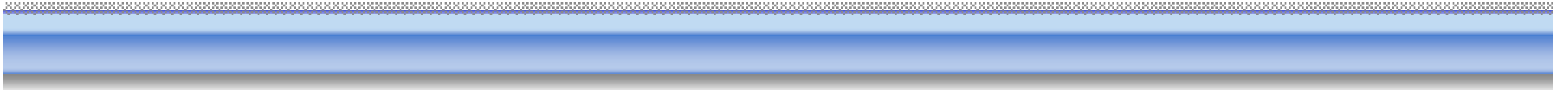


Argonne/U Chicago

Pete Beckman <beckman@mcs.anl.gov>
Charlie Catlett <cec@uchicago.edu>

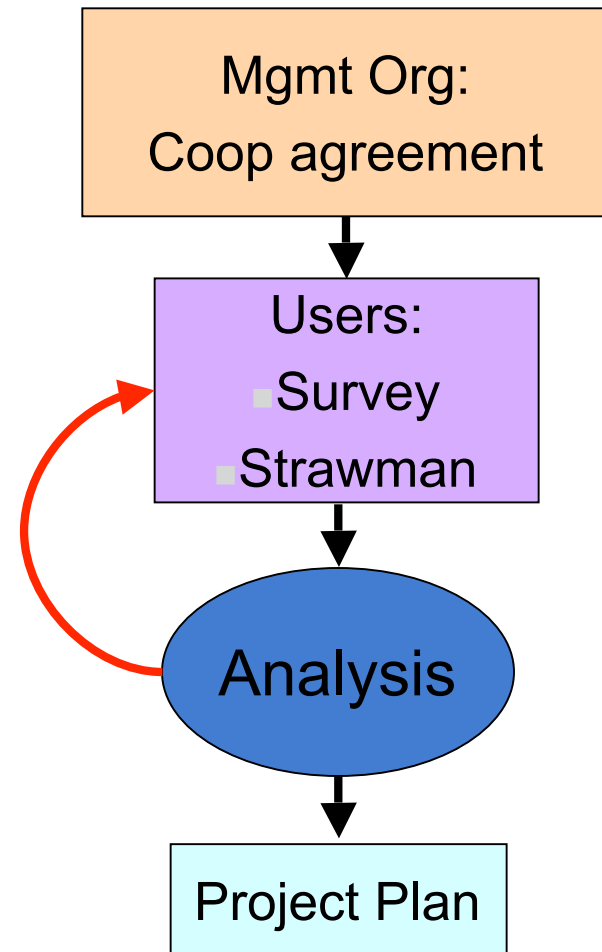


Putting The Plan Together



Develop a Project Plan

- With *Strawman User Guide* and budget, pick technologies
 - ◆ Free software, \$\$\$ SW/support, build it from scratch
 - ◆ How much of the strawman user guide can be implemented?
 - ◆ Which features must be dropped?
- Iterate with users & management
- Develop project plan
 - ◆ Staffing
 - ◆ Timelines
 - ◆ Contingency plans
 - ◆ Expansion plans early... what is non scalable from day 1?



Work Breakdown Structure

- Seven Project Areas
 - ◆ Resources (Hardware Acquiring and Deployment)
 - ◆ Software & Services Environment
 - ◆ Network
 - ◆ Operations and System Support
 - ◆ Applications and User Services
 - ◆ Project Management and Service Policies
 - ◆ Expansion
- Integrates current status and progress
 - ◆ A “reset” injecting lessons learned
 - ◆ Useful to plan additional sites



Design & Operation Document Guide

- **Cooperative Agreement**
- **Mgmt & Org Chart**
- **User Survey**
- **Strawman user guide**
- **Flagship apps**
- **Project Plan**
- **Security Policies**
- **Elevator Overview**
- **QA / Test / Accept Plan**
- **Primer**
- **Risk Mgmt Plan**
- **Accounting practices & policy**
- **Data practices & policy**
- **Customer service: Help desk, trouble tickets, SLAs**
- **Work and task list**
- **Real user guide**



Other Useful Documents

- Elevator pitch overview (5 slides)
 - ◆ **Everyone** in the project should be able to describe and represent the project clearly
- QA / Test / Acceptance Plans
 - ◆ Ongoing, **independent** QA and testing is vital
- Risk management
 - ◆ Go / No-go decision points
 - ◆ Alternative technologies
 - ◆ Contingency budgets & plans



Step 2

Build: Integrate and Develop

Build



Argonne/U Chicago

Pete Beckman <beckman@mcs.anl.gov>
Charlie Catlett <cec@uchicago.edu>



Software Development Principles For Building The TeraGrid

- Drive development with applications
- TG Software & Services and environment is homogeneous across all sites **except where a difference is clearly justified to the users and driven by their requirements**
- Every package is versioned and the build/install/config parameters reproducible
- CTSS packages have specific versions and tests; change is carefully managed
- A Test Harness is constantly working to insure stability and conformance to the TG Hosting Environment
- After successful deployment on the TestGrid, new components are tested on the ProdGrid



Begin Software Development

- During the design phase, you should have determined:
 - ◆ Freely available software
 - ◆ \$\$\$ SW/support
 - ◆ Software that requires development
- Two implementation teams, two kinds of people, two different cultures
 - ◆ “Classic sysadmin”
 - Install freely available or \$\$ SW
 - ◆ “Programmer”
 - Comfortable with CVS, design specifications, design reviews, user interface design



Culture Clash Warning...

- Sysadmins:
 - ◆ Totally demand driven: “Which emergency do I handle first today?”
 - ◆ “I don’t have time for software engineering”
 - ◆ “I wrote this really ugly, un-maintainable, 500 line Perl program to temporarily solve our problem, isn’t it so cool!”
 - ◆ “Version control? I can remember all the details, I don’t need a version system, and besides, I don’t have time to learn that and maintain it”
 - ◆ Underestimate time: “A weekend of hacking should do it”
- Programmers:
 - ◆ Comfortable with timelines and budget
 - ◆ Understand project plans, technical reviews, documentation, etc
 - ◆ Live and die by versioning and build systems
 - ◆ Think sysadmining is easy, even though they have never done it
 - ◆ Overestimate time: “32.9 man-years to create the specification”



REPRODUCABILITY!!!!

- To maintain the Service Level Agreement and consistent software stack (CTSS) requires versioning and reproducible results
- **The TeraGrid continues to loose this battle**
- SysAdmins:
 - ◆ “I know secret magic to setup and configure the machines, and it is so complicated it would be nearly impossible to automate, and besides, I don’t have time”
 - ◆ Therefore, SysAdmins will spend most of their time painfully duplicating, by hand, and with human errors, their work



Reproducibility: What We Need

- ALL source code, patches, and build scripts go into repository
 - ◆ Examples:
 - 1.2.5..9-gcc, 1.2.5..9-intel, mpich-1.2.5..10-gcc, mpich-1.2.5..10-intel
- Automated (non-human) steps for:
 - ◆ Configuring source (patches, config), building source, building package, installation, configuration, post-installation
- Automated process for rebuilding all packages
- (similar to how Linux companies build a distro)
- For many (most?) sysadmins, they have never used CVS (or the discipline of versioning) on a daily basis



Make the Repository Part of the Culture

- If it is not in the repository, it does not get included in status reports or press releases
- Code cannot be installed on machines unless it is in the Repo first
- PPT slides go in the repository
- Web pages are checked into the repository
- TortoiseCVS is used on Windows desktops



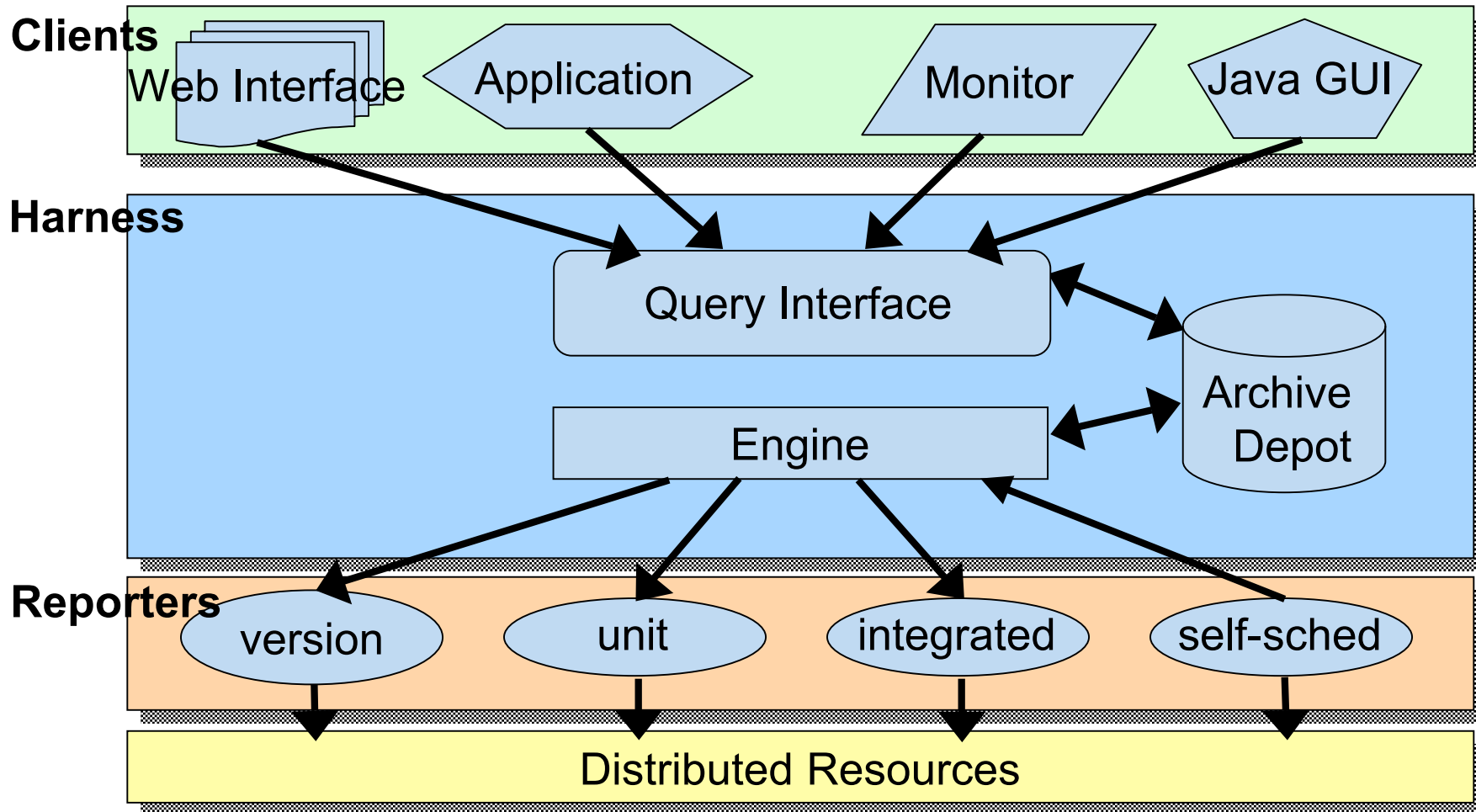
Testing: Part of the Culture

- All programs that are part of the software stack must have:
 - ◆ Version Reporter (for automated collection)
 - ◆ Unit Test (the Self Test for packages)
 - ◆ Integrated Test (test that confirms proper integration with the other system components)
- Automated test harness
- Historical archive of test results for mining
 - ◆ Comparing performance
 - ◆ Finding components that fail often





Inca: A Test Harness Framework for Builders



A Single Language For Reporters

```
<?xml version="1.0" encoding="UTF-8" ?> <!-- Generated
  by Turbo XML 2.3.1.100. Conforms to w3c
  http://www.w3.org/2001/XMLSchema -->
<xsd:schema
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified">
<xsd:element name="INCA_Reporter">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="INCA_Version" />
<xsd:element ref="localtime" />
<xsd:element ref="gmt" />
<xsd:element ref="ipaddr" />
<xsd:element ref="hostname" />
<xsd:element ref="uname" />
<xsd:element ref="url" />
<xsd:element ref="name" />
<xsd:element ref="description" />
<xsd:element ref="version" />
<xsd:element ref="INCA_Input" />
<xsd:element ref="body" />
<xsd:element ref="exit_status" />
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="gmt" type="xsd:string" />
<xsd:element name="localtime" type="xsd:string" />
<xsd:element name="hostname" type="xsd:string" />
<xsd:element name="ipaddr" type="xsd:string" />
<xsd:element name="uname" type="xsd:string" />
<xsd:element name="url" type="xsd:string" />
<xsd:element name="name" type="xsd:string" />
<xsd:element name="version" type="xsd:string" />
<xsd:element name="INCA_Version" type="xsd:string" />
```

```
<xsd:element name="description" type="xsd:string" />
<xsd:element name="exit_status" nillable="true"
  fixed="0">
<xsd:complexType mixed="true">
<xsd:choice>
<xsd:element ref="message" minOccurs="0" />
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="message" type="xsd:string" />
<xsd:element name="body" abstract="true" />
<xsd:element name="ID" type="xsd:string" />
<xsd:element name="INCA_Input">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="input" minOccurs="0"
  maxOccurs="unbounded" />
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="input" />
<xsd:element name="verbose" type="xsd:integer"
  substitutionGroup="input" />
<xsd:element name="help" substitutionGroup="input">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="yes" />
<xsd:enumeration value="no" />
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
</xsd:schema>
```



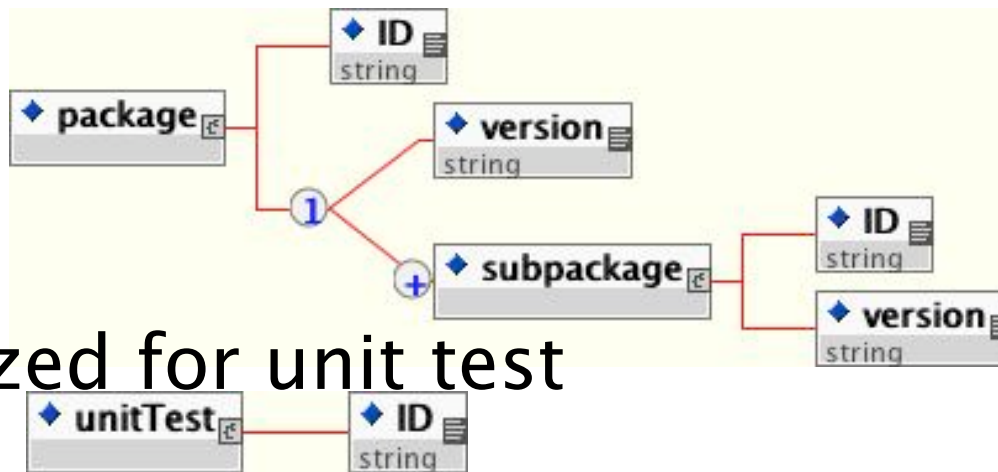
Argonne/U Chicago

Pete Beckman <beckman@mcs.anl.gov>
Charlie Catlett <cec@uchicago.edu>



Low-level Schema Only Provides XML Bus For Moving Snips of Info

- Specialized for reporting Version information

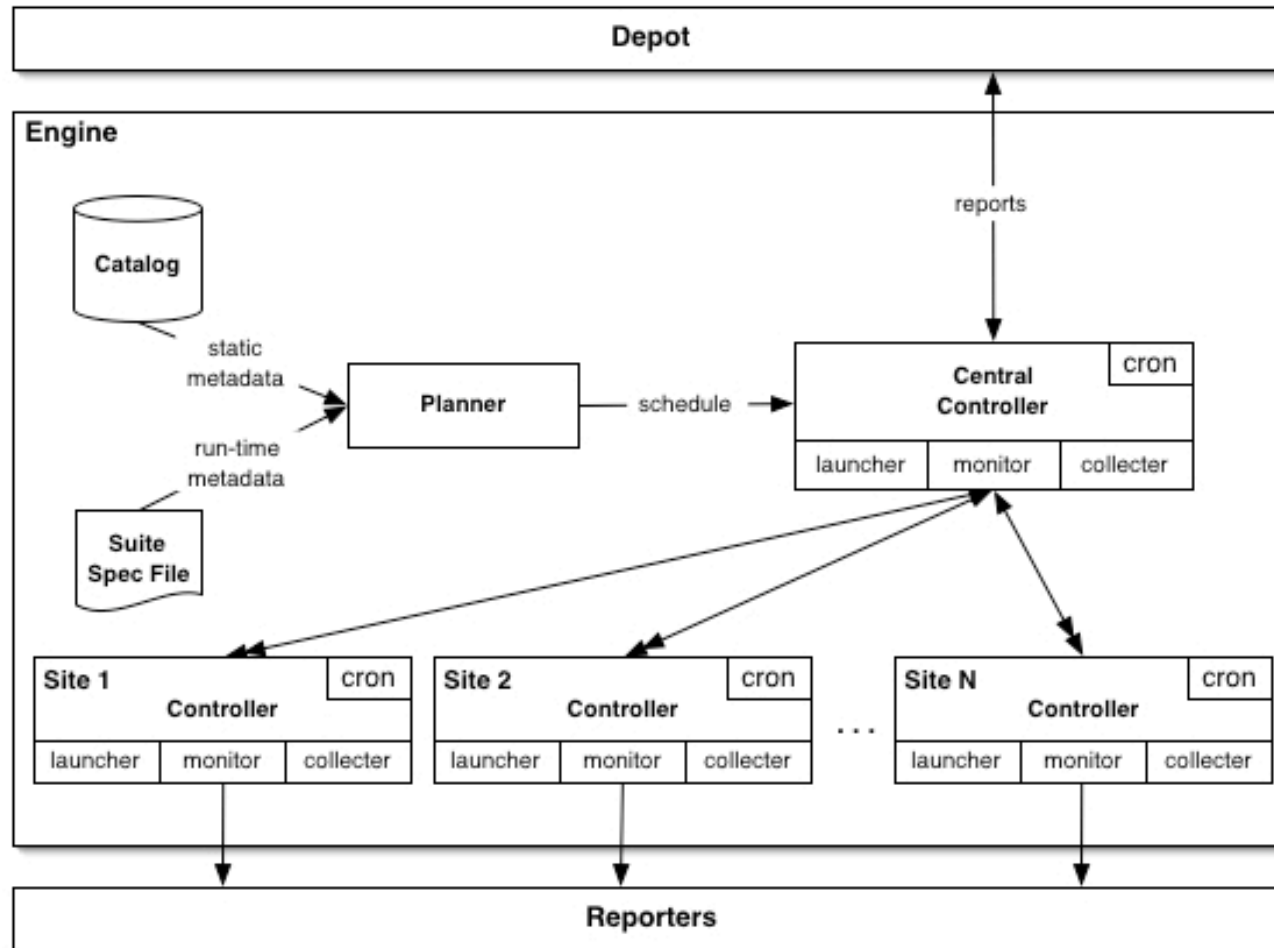


- Specialized for unit test

- Example:

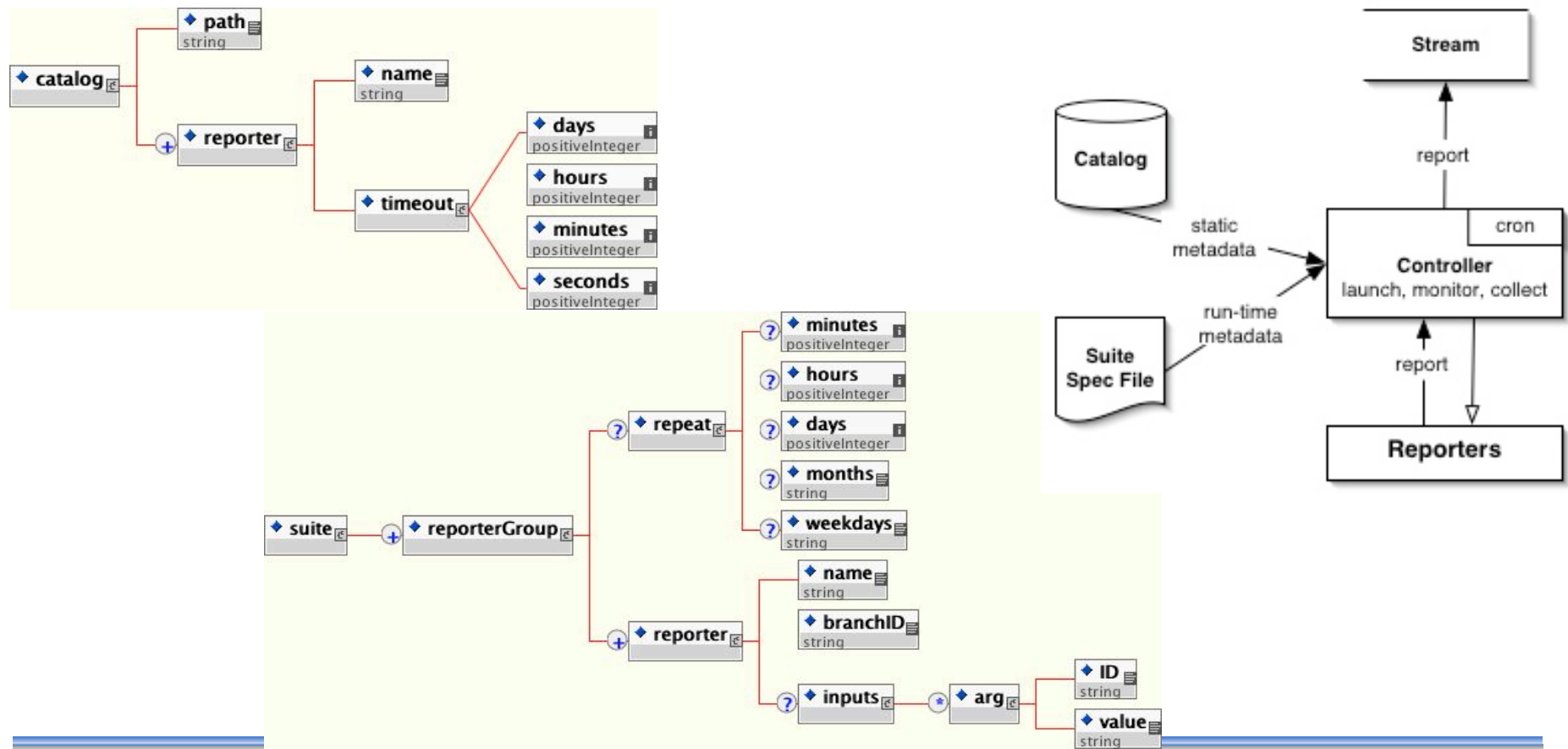
```
<gsissh> <ID>gsissh</ID> <version>3.4p1</version> </gsissh>
```

The Test Harness



Language for Controlling the Test Harness

- Test suite specification & execution control



Inca - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Media Print Mail

Address <http://repo.teragrid.org/inca/html/environmentStatus.html> Go Links Norton AntiVirus

TERAGRID TeraGrid Home > Inca

Default User Environment

Last Updated: Wed Jan 14 01:28:35 2004 CST

The following table shows the environment variables that should be defined in the user's default environment and their status at each of the sites. The resource's soft.db entry is compared to the output of /usr/bin/env

Variables	anl-ia64	anl-viz	caltech-ia64	ncsa-ia64	psc-gs1280	psc-tcs	sdsc-datastar	sdsc-ia64
GLOBUS_LOCATION	available	available	available	available	available	missing	available	available
GLOBUS_PATH	available	available	available	available	available	missing	available	available
GM_HOME	available	available	available	available	n/a	n/a	n/a	available
HDF4_HOME	available	available	available	available	available	missing	missing	available
HDF5_HOME	available	available	available	available	available	missing	missing	available
INTEL_HOME	available	available	available	available	n/a	n/a	n/a	available
MKL_HOME	available	available	missing	available	n/a	n/a	n/a	available
MPICH_GM_HOME	available	available	available	available	n/a	n/a	n/a	available
SASL_PATH	available	available	available	available	available	missing	available	available
SRB_HOME	available	available	available	available	available	missing	missing	available
TG_APPS_PREFIX	available	available	available	available	missing	missing	missing	available
TG_CLUSTER_GPFS	n/a	n/a	available	available	n/a	n/a	n/a	available
TG_CLUSTER_HOME	available	available	available	available	available	missing	available	available
TG_CLUSTER_LUSTRE	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a
TG_CLUSTER_PFS	available	available	available	available	missing	missing	missing	available
TG_CLUSTER_PVFS	available	available	available	n/a	n/a	n/a	n/a	available
TG_CLUSTER_SCRATCH	available	available	available	available	missing	missing	available	available
TG_NODE_SCRATCH	available	available	available	available	missing	missing	available	available

The following tables shows the path variables that should be defined in the user's default environment and their status at each of the sites. Each row in the table represents a modification to the path from that key. The resource's soft.db entry is compared to the output of /usr/bin/env

LD_LIBRARY_PATH	anl-ia64	anl-viz	caltech-ia64	ncsa-ia64	psc-gs1280	psc-tcs	sdsc-datastar	sdsc-ia64
globus	available	available	available	available	available	missing	missing	available
globus-2.2.4-gcc-r1	available	available	available	available	available	missing	missing	available
gm	available	available	available	available	n/a	n/a	n/a	available
gm-2.0.8-r1	available	available	available	available	n/a	n/a	n/a	available
hdf4	available	available	available	available	available	missing	missing	available
hdf4-4.1.5-r1	available	available	available	available	missing	missing	missing	available

Internet

Inca in Practice...



Argonne/U Chicago

Pete Beckman <beckman@mcs.anl.gov>
 Charlie Catlett <cec@uchicago.edu>



Grid-wide Change Management

- Set up TestGrid / development platforms
- Changes to production machines must be handled by Change Management process
 - ◆ Functional on TestGrid
 - ◆ Rationale for upgrading production machine
 - ◆ Test on production machine by sysadmins
 - ◆ Trial by limited number of users
 - ◆ Move to full production status



Project Plans and Task Lists

- Create long-range plans and milestones for entire project
- Create detailed task lists for 8 – 10 week horizon
- Working groups (teams) must be responsible for weekly reporting on their milestones
- Weekly “architecture” meetings used for:
 - ◆ Discussing the 8–10 week detailed tasks
 - ◆ Synchronization and dependencies between groups
 - ◆ Failures in the system



Technical Reviews and All Hands Meetings

- Plan for frequent mini-reviews of architecture, status, and technologies (8 – 10 week horizon)
- Quarterly careful review of technical achievements and designs.
 - ◆ Results used for quarterly reports
 - Weekly reports very helpful
 - ◆ Risk assessment and go/no-go decisions
- Semi-annual “All Hands Meeting” to synchronize all the Grid participants



Step 3: Operate and Extend

Operate



Argonne/U Chicago

Pete Beckman <beckman@mcs.anl.gov>
Charlie Catlett <cec@uchicago.edu>



SLAs and why this matters

- Sites must report how well they are participating in “the Grid”
- Funding always follows performance
- Definitions are very very important....



TeraGrid SLA Measurements (Under Construction)

- Resource Up
 - ◆ Local jobs running, completing
 - ◆ (Machine could be off the net)
- Development Up
 - ◆ Compilers, home directories, libraries, etc
 - ◆ (Job queues and Grid services could be down)
- Grid Resource Up
 - ◆ Ahh.... This is hard



SLA: Grid Resource is UP

- All of the following “services” must be “UP” at a site:
 - ◆ Remote Globus job launch (gatekeeper, auth)
 - ◆ GridFTP transfers
 - ◆ Condor-G
 - ◆ SSH-GSI
 - ◆ ...
- A Service is up if and only if at least one remote site can use it.
 - ◆ Yes.. This is flawed. Ideas?
 - ◆ That’s why we are here?



Security

- Ugh.
- Let's talk about recent problems first



Security

- Policy:
 - ◆ Security Memorandum of Understanding
 - ◆ Certificate Management Authority Policy
 - ◆ Baseline security requirements for all TG sites (draft)
- Incident Response
 - ◆ Security Contact List (incident and non-emergency), 24 Hour Hotline for real-time coordination
 - ◆ Incident Response Flowchart, Playbook
 - ◆ Security Incident Report Form
 - ◆ Security Incident Communication Plan with NSF (still evolving)
- Secure Communication
 - ◆ Secure Website to exchange and track critical information
 - ◆ Secure PGP email between all sites
 - ◆ Secure (secret) weekly conference calls limited to a pre-approved participant list
- Planning and Organization
 - ◆ TG Site Specific Risk Assessments
 - ◆ Developing official vulnerability evaluation and response form & process
 - ◆ Developing plan for testing security: (red-team probing, scans, etc.)
 - ◆ Security Reviews



Coordinating Security

US-CERT
UNITED STATES COMPUTER EMERGENCY READINESS TEAM

Secure Portal User Login

ATTENTION:
This is a restricted system
Unauthorized use of this system is prohibited.

The use of this system is authorized for users authorized by the system owners. This includes all networks, and (specifically including) may be monitored for security purposes, including use is authorized to verify security and operation of the system, to prevent unauthorized dissemination of information recorded, copied, or otherwise transmitted, including personal information, or sent over the system, or monitored for security purposes.

This system does not authorize dissemination of information recorded, copied, or otherwise transmitted, including personal information, or sent over the system, or monitored for security purposes.

Use of this system is authorized for users authorized by the system owners. This includes all networks, and (specifically including) may be monitored for security purposes, including use is authorized to verify security and operation of the system, to prevent unauthorized dissemination of information recorded, copied, or otherwise transmitted, including personal information, or sent over the system, or monitored for security purposes.

Collaboration Tools

- Secure Messaging
- Library
- Find Users
- Calendar
- Online Briefings
- Forum Discussions (15 new)
- Survey Wizard
- TaskTrac (1 new)
- Chat
- WebPort
- Alerts

Admin Tools

- Update Profile/Preferences
- My Groups
- Suggestion Box
- Report Problems
- Log Out

TeraGrid Security Portal

Security Event	Event Name	Severity/Importance
Alerts	nothing to report	
Vulnerabilities	nothing to report	
Exploits	nothing to report	

Click on a button to interact with the US-CERT Watch

MAIL **FORUMS**

NEW Portal-Wide Forums

- [Malware Code Analysis](#)
- [Emerging Threats](#)
- [Vulnerabilities](#)
- [Incident Response](#)

DHS/US-CERT Daily Unclassified Briefing as of 7/1/04

Over the preceding 24 hours, there has been no cyber activity which constitutes an unusual and significant Security, National Security, the Internet, or the Nation's critical infrastructures.

Watch Synopsis: The LSASS exploit code for Windows XP has been perfected for some malicious viruses as the recent versions of the Korgo IRC Worm prove. The Watch still expects that other exploits for MS vulnerabilities will be perfected and used in the future.

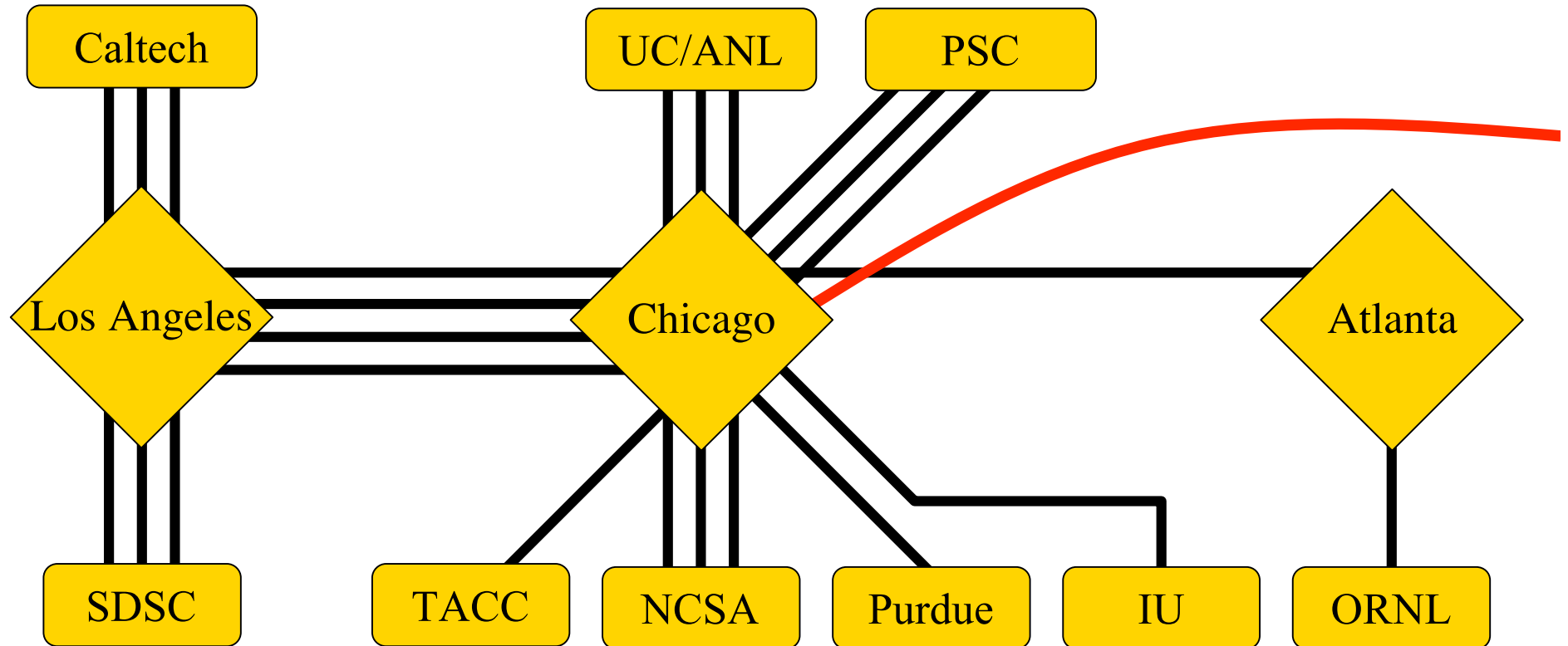
Security Event	Event Name	Severity/Importance	Severity/Importance
Malicious Activity	Nothing to report		
Vulnerabilities	Nothing to report		1 Minimal: The security event is of minimal impact on National Critical Infrastructure and our strategic partners.
Exploit Code Released	Nothing to report		
Virus Activity	W32.Korgo.F	2 Low	2 Low: The security event is of low impact on National Critical Infrastructure and our strategic partners.
Internet Scans	No abnormal scanning patterns reported	1 Minimal	1 Minimal: The security event is of minimal impact on National Critical Infrastructure and our strategic partners.

Argonne National Laboratory
UNIVERSITY OF CHICAGO

Argonne/U Chicago

Charlie Catlett <cec@uchicago.edu>

Current TeraGrid Network



Resources: Compute, Data, Instrument, Science Gateways



Argonne/U Chicago

Pete Beckman <beckman@mcs.anl.gov>
Charlie Catlett <cec@uchicago.edu>



Hub Operations and Policy

- Juniper Connections
 - ◆ Available Hub Router Capacity
 - Chicago: 17 x 10 Gb/s
 - Los Angeles: 20 x 10 Gb/s
- Hub Policies
 - ◆ Minimum 10 Gb/s connections
 - Otherwise Abilene likely to be sufficient
 - <10 Gb/s wastes limited slot capacity at hubs
- Backplane Policies
 - ◆ Not a general traffic network
 - ◆ Hubs and border routers centrally managed
 - ◆ Resources directly connected (i.e. at layer2) to border routers
 - No intervening LANs, firewalls, etc.
 - Implies one site per connection to hub



Backplane Operations Policies

- Initial (ramp-up) Policies
 - ◆ Use of individual lambdas for experiments as warranted
 - E.g. recent storage over SONET experiments
 - ◆ Experimenting with MPLS for reserved bandwidth
- Steady State Policies
 - ◆ Backplane typically will operate at 40 Gb/s
 - ◆ Option to use 10 Gb/s (25% resource) for experiments
 - Available for scheduling, time granularity tbd based on experience
 - Additional hardware and funding will be required for equipment capable of interconnecting to lambda



Unified Support

- Single 1-800 number that moves between sites
 - ◆ 12 hrs at NCSA
 - ◆ 12 hrs at SDSC
- Single page for requesting allocation
- Unified trouble tix system
- Working groups that can work directly with users:
 - ◆ Performance Evaluation
 - ◆ User Services
 - ◆ Networking



Portals

- Simplified mechanisms for single-point status
 - ◆ Current accounting
 - ◆ System status (as needed by user)
 - Network
 - Queue
 - Projected downtime
- Domain-specific portals for job submission, mass storage, etc, can be built by their respective communities
 - ◆ We are a Grid Hosting Environment, we don't want nor need to build user-level portals... they can do it!



Operations Monitoring

- There are many types of status info users:
 - ◆ Developers
 - ◆ Users
 - ◆ Operators
- Operations must construct set of status monitors from basic tools to meet their needs:
 - ◆ Clumon / Ganglia
 - ◆ Inca
- Set of actions and procedures to take based on status information
- Planning for uptime / downtime

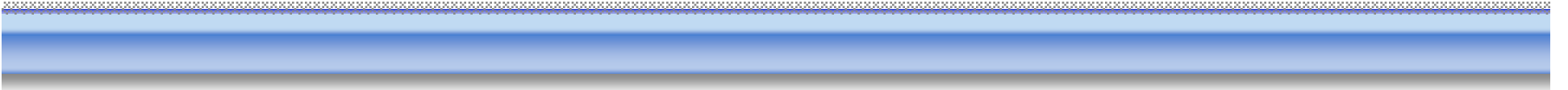


User Interactions

- Continued reprioritization of features
- Up-to-date project information
- Project-wide training team
 - ◆ Creating SC03 / SC04 Tutorial
- User workshops
- Review of best practices: what worked and what failed when working with users?



Summary



Summary (1/2)

- Virtual Organization & Cooperative Agreement
- Management and Decision Making
- Design -> Build -> Operate
- Eliminate hype, be practical, get user commitment early
- Collaboration technology required from day 0
- Installing and understanding Globus is the least of your concerns
- Integration and engineering is costly, budget for it



Summary (2/2)

- Construction
 - ◆ Reproducibility
 - ◆ Sysadmin & Soft Eng cultures
- Operations
 - ◆ Training, running, support (it must be easy!)



Remember What The Users Want

- Parameter sweep interfaces
- Collab viz environments
- Viz steering tools
- Global, sync file space
- Resource queue and broker
- Client-side tools for apps
- Workflow/dependency tools
- Directory services/discovery
- Differentiated pricing
- Meta/Co scheduling
- MP between resources
- Remote database access
- Advanced reservations
- Non-CPU resource pricing
- Fast data movement tools
- Data streaming instruments
- Remote batch/interactive viz/rendering
- Remote data read/write
- End-user portal development tools
- Domain-specific community portals
- Single signon / delegation
- Unified accounting and single help desk
- User portals for projects



The End



Argonne/U Chicago

Pete Beckman <beckman@mcs.anl.gov>
Charlie Catlett <cec@uchicago.edu>

