UABgrid: Practice and Experience

Enis Afgan, Purushotham Bangalore, John-Paul Robinson
University of Alabama at Birmingham (UAB)
Birmingham, AL

Presented at OGF22, Boston, MA
Feb 26, 2008.

Overview

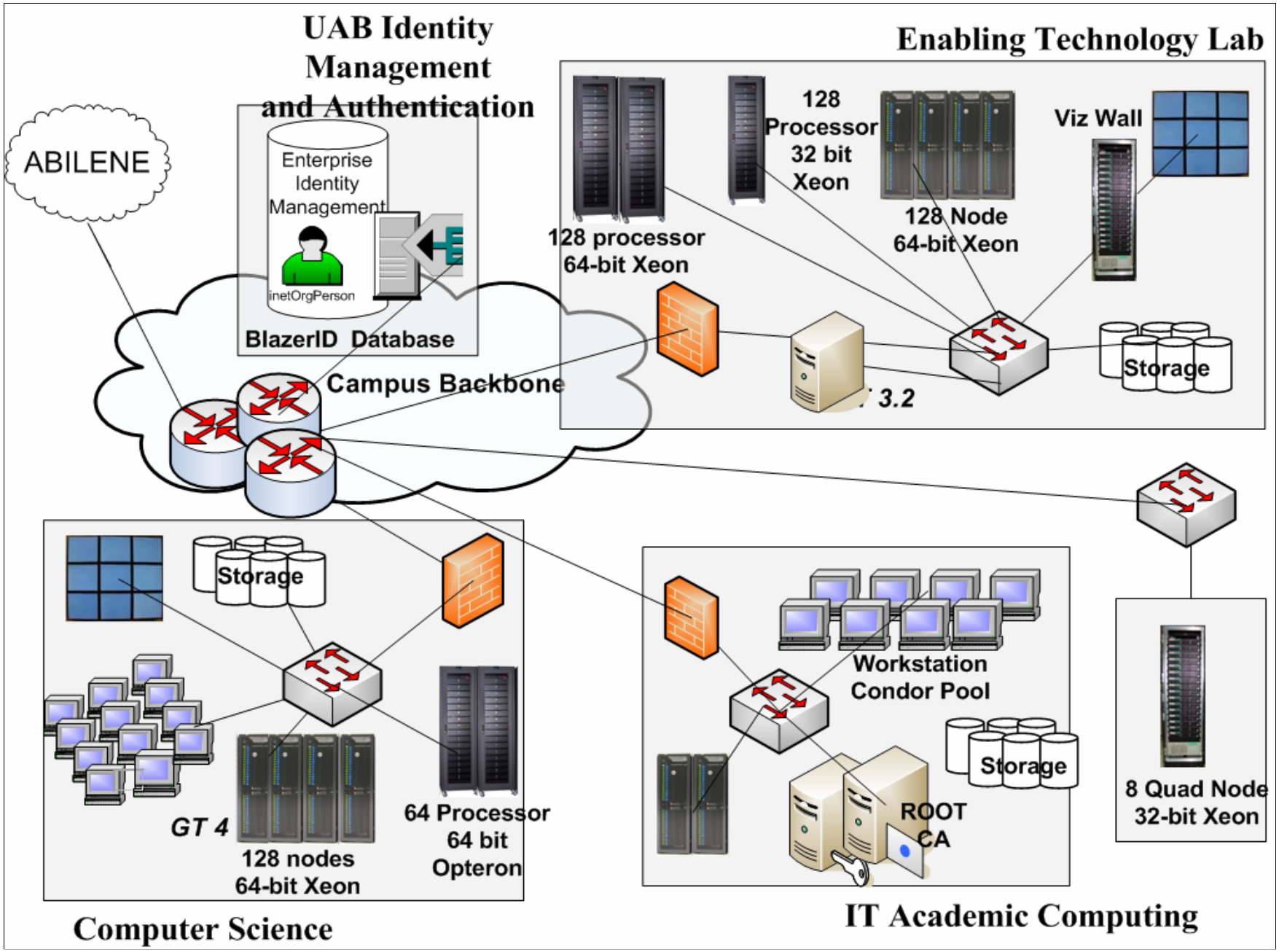
- UABgrid infrastructure
  - Participants
  - Status
  - Goals
- Identity management
  - Leveraging BlazerID & InCommon
- Applications for UABgrid and beyond
  - Dynamic BLAST
  - R

What is UABgrid?

- A distributed computational infrastructure for research and education that provides a common interface to access distributed resources
- Collaboration between academic and administrative IT units at UAB
- Leverages InCommon identity services for consistent identity across resources
- Support research collaboration through autonomous virtual organizations

UABgrid Partners

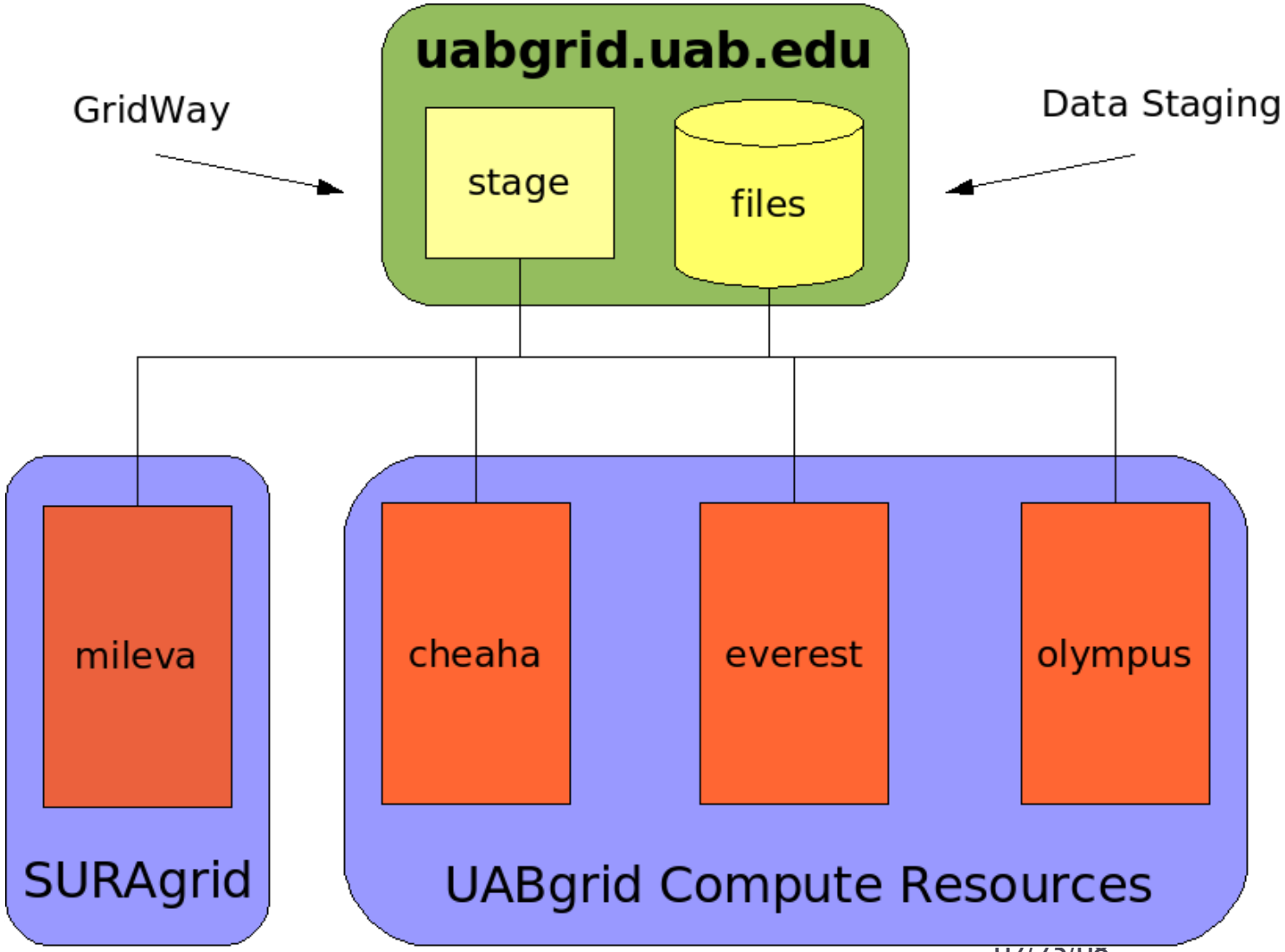
- High Performance Computing Services, IT infrastructure services:
  - 128 CPU AMD cluster
- Department of Computer and Information Sciences:
  - 256 CPU EM64T cluster with Infiniband; 64 CPU Opteron cluster;
- UAB Shared Computing Facility located in Engineering:
  - 256 CPU EM64T cluster with gigabit Ethernet; 128 CPU Xeon cluster;
  - IBM Bluegene - 2040 CPUs


UAB Identity Management and Authentication
Enabling Technology Lab
ABILENE
Enterprise Identity Management
inetOrgPerson
BlazerID Database
Campus Backbone
128 Processor 32 bit Xeon
128 processor 64-bit Xeon
128 Node 64-bit Xeon
Viz Wall
Storage
3.2
Storage
GT 4
128 nodes 64-bit Xeon
64 Processor 64 bit Opteron
Workstation Condor Pool
Storage
ROOT CA
8 Quad Node 32-bit Xeon
Computer Science
IT Academic Computing

UABgrid status

- Globus Toolkit 4 based infrastructure
- Leverages institutional identity management investment
- UABgrid access options:
  - Command-line interfaces
  - Programmatic interfaces through GridWay
  - Web interface using GridSphere
- Current UABgrid user groups:
  - Biostatistics
  - Bioinformatics

UABgrid Compute Architecture



UABgrid goals

- Maximize use of university's investment in computational resources
- Minimize administrative effort involved in campus-wide resource sharing
- Leverage investments in Identity Management, WebISO, Directory services, and Network infrastructures
- Provide a GridWay based entry point and abstract grid as LRMs abstract a cluster -> a meta-cluster
- Provide a direct link to regional and national grids (e.g., AlabamaGrid, SURAgrid, TeraGrid)
- Unified interface to compute resources in the region
  - 'Cluster in a chip'

Utilizing UABgrid

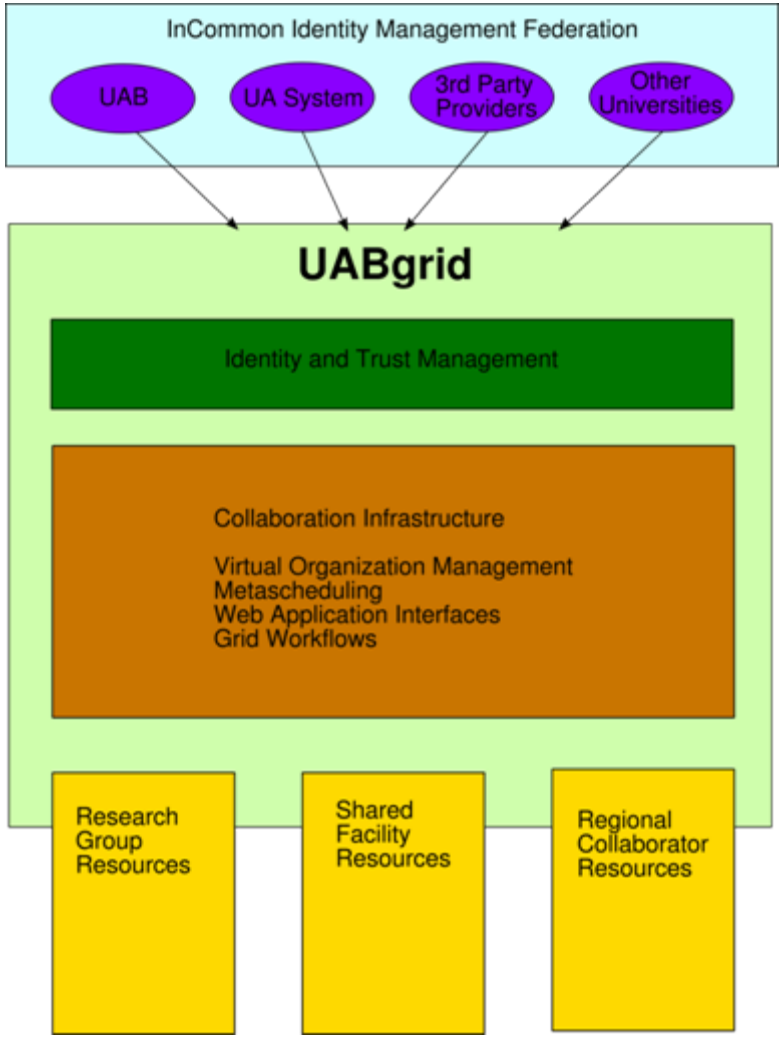
- Research
  - Grid testbed
  - Grid-enablement of applications (BLAST, R)
  - Meta-scheduling
  - GAUGE – application development environment through modeling framework
  - GADE – grid application execution framework
- Education/teaching (training)
  - Graduate course in grid computing since Fall 2003
  - HPC and grid bootcamps

Identity management

Leveraging BlazerID & InCommon «

Conceptual architecture

- Leverages IdM investments via InCommon
- Provides collaboration environment for autonomous VOs
- Supports integration of local, regional, and national resources

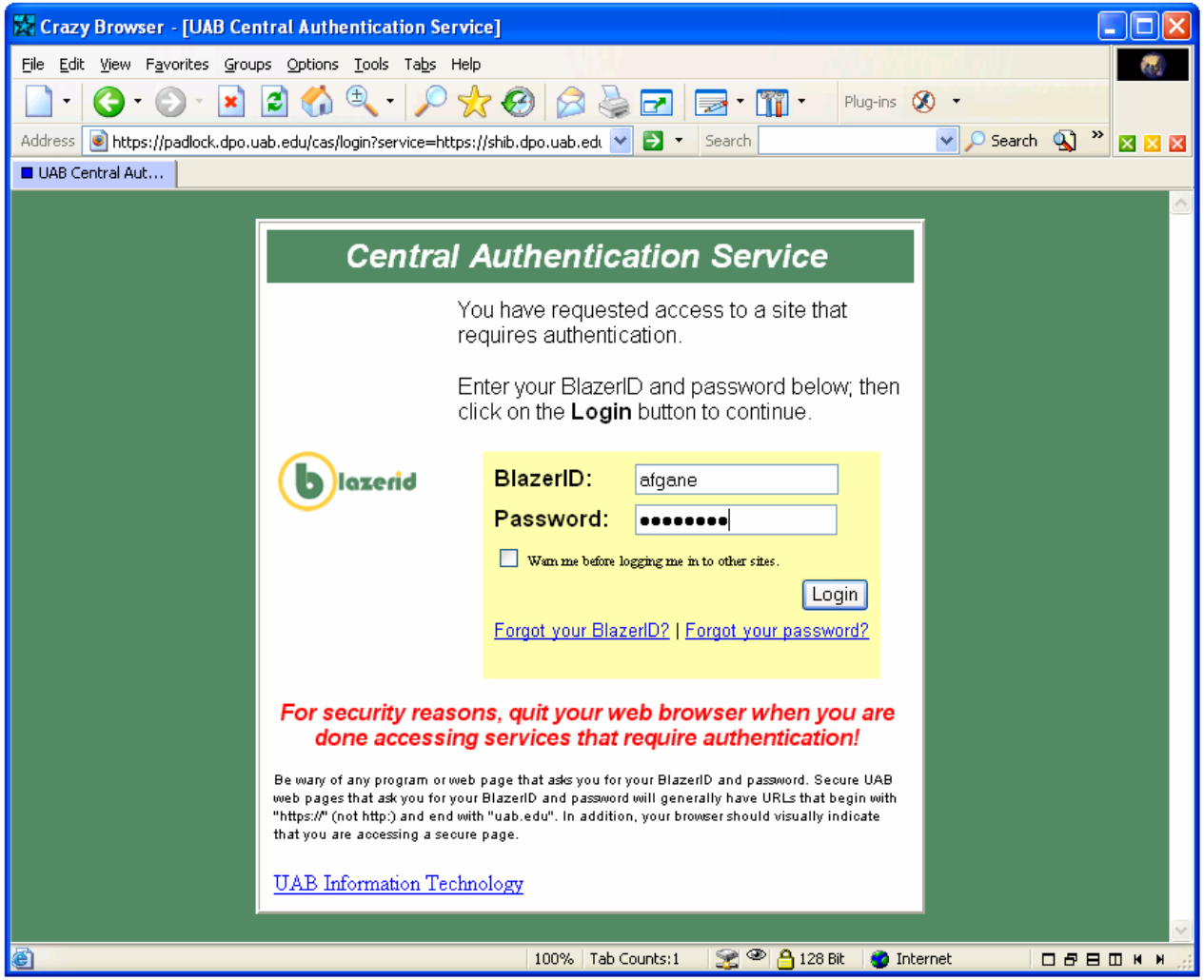


Trusting users

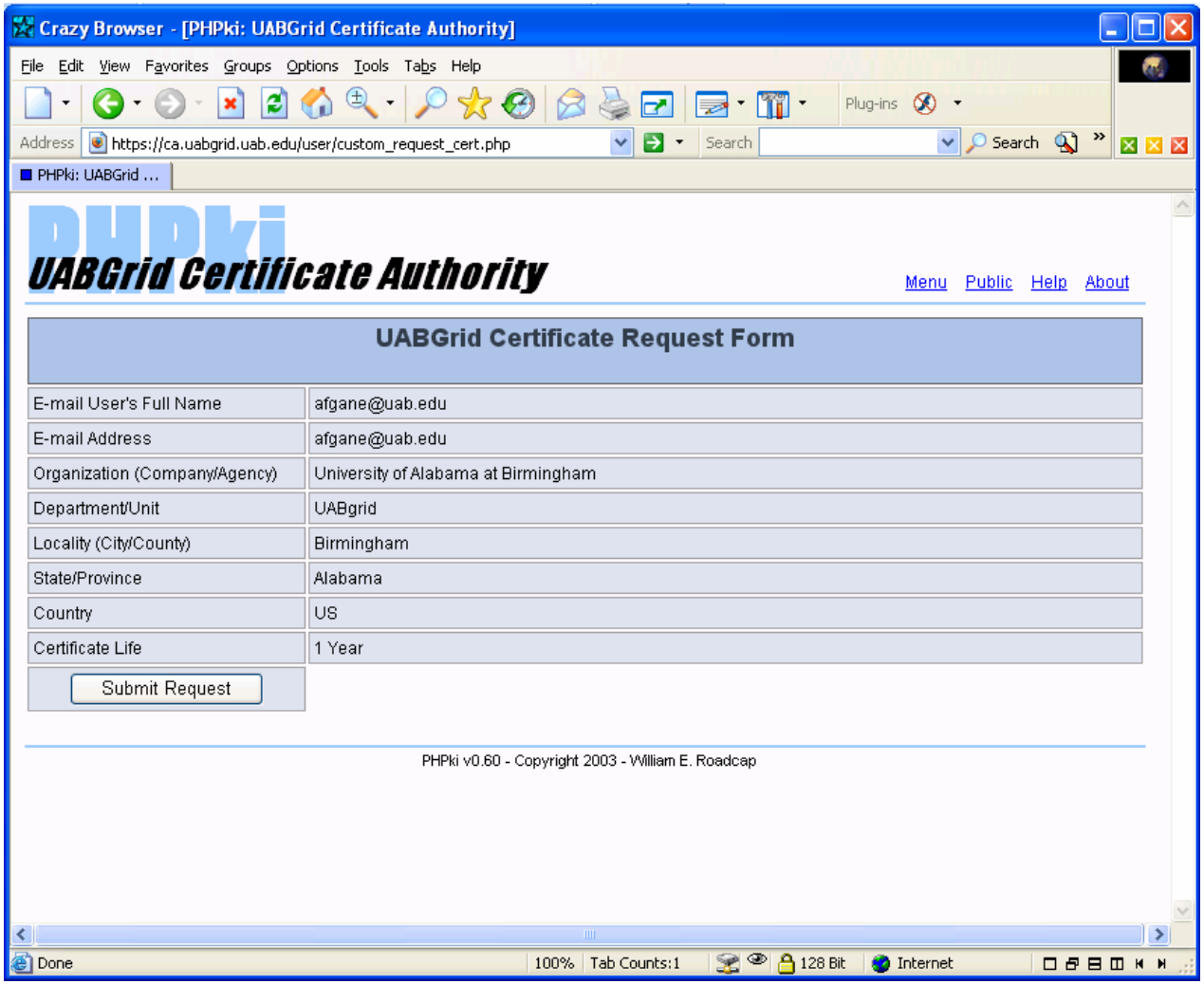
- UABgrid trusts InCommon members to determine user identities
  - Members of UAB community can use BlazerID based authentication
  - Non-UAB members use home institution in InCommon or an open registration provider like openidp.org or protectnetwork.com
- Leverages these identities to grant access to collaboration environment
- Registered users may join VOs
- VO membership and role attributes influence resource authorization

Point your browser to https://ca.uabgrid.uab.edu/user and log in using your UAB BlazerID or preferred InCommon ID

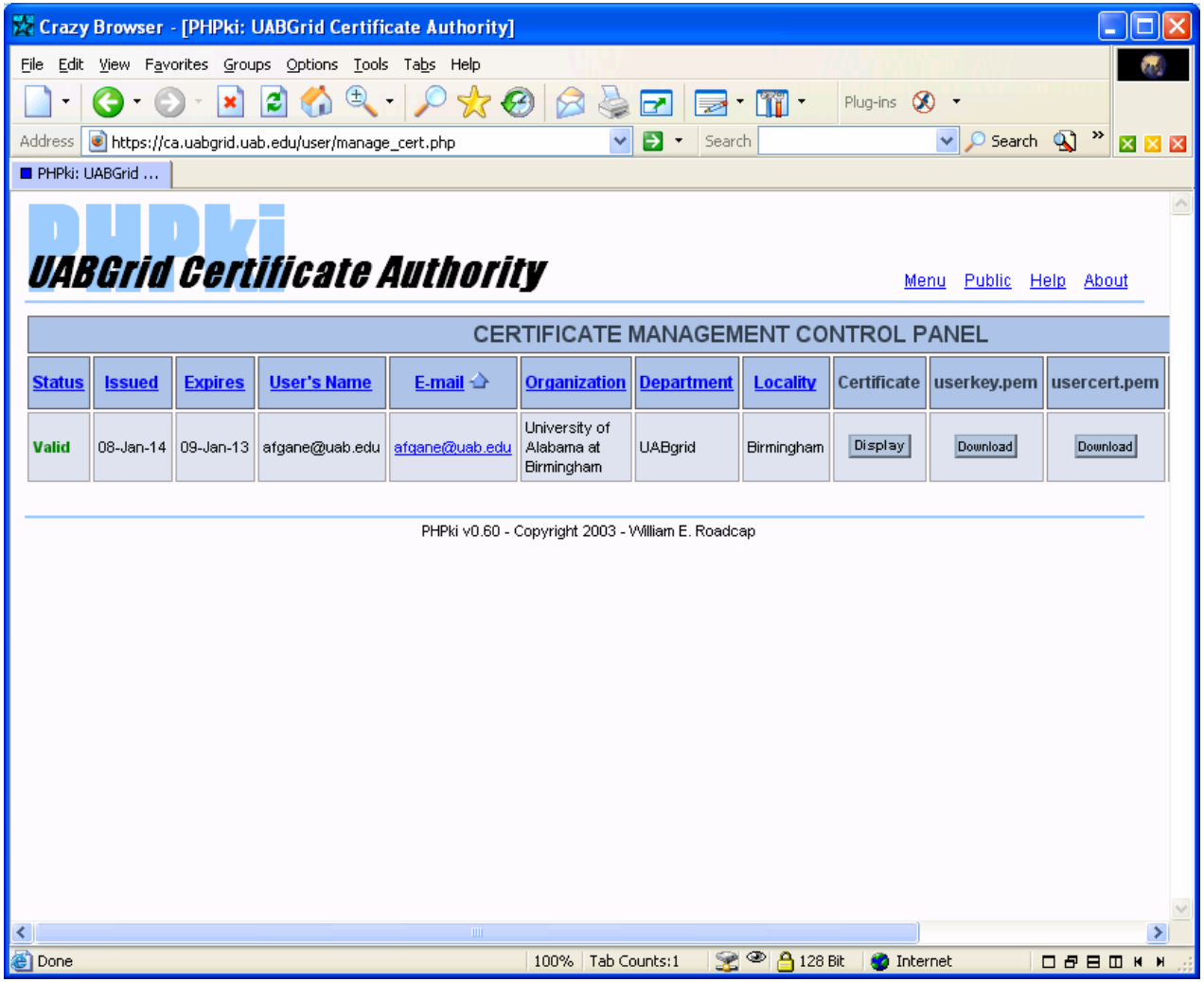
Login using standard BlazerID



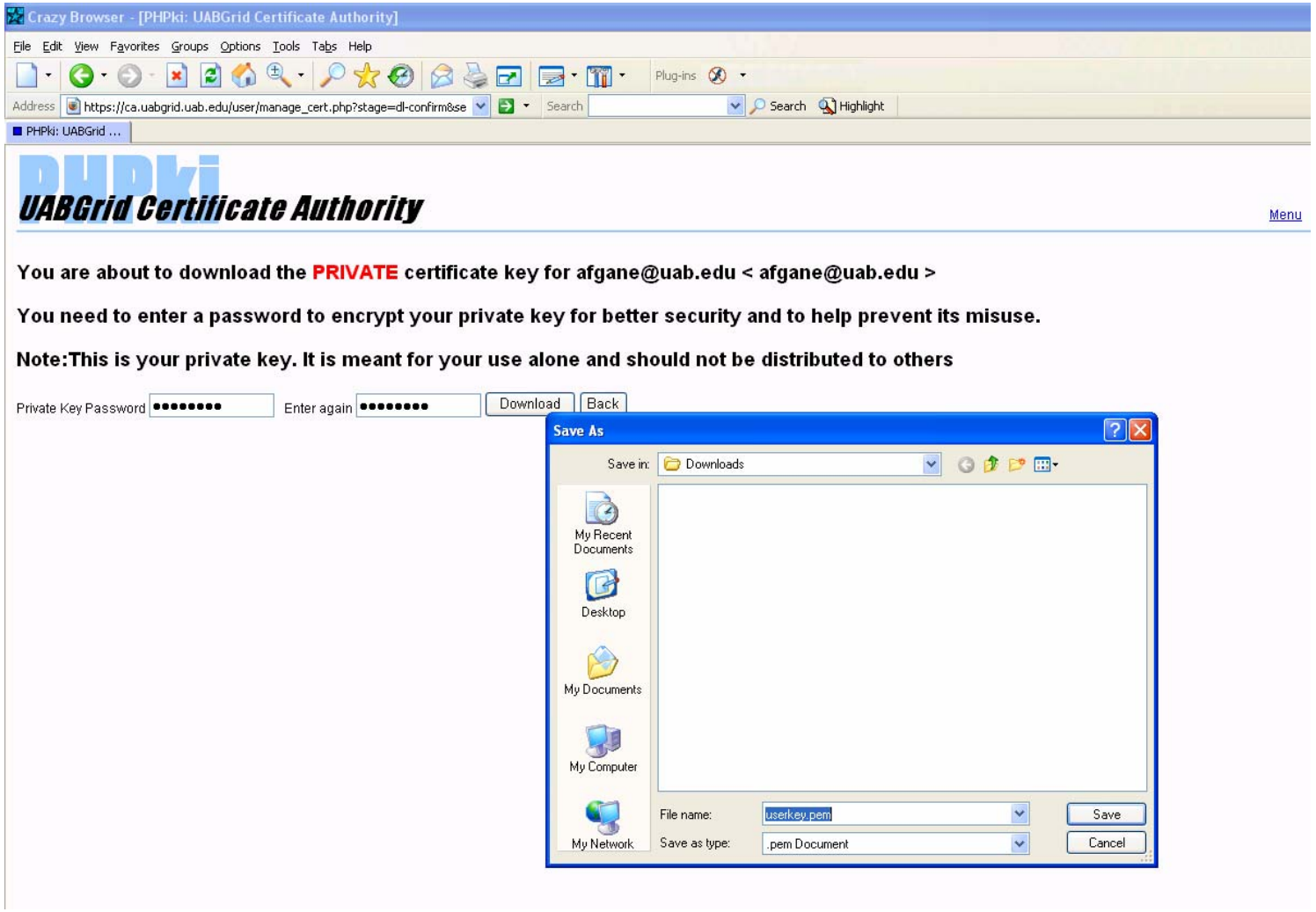
Next, request a new grid certificate based on your existing identity



Next, you should see list of your certificates. You need to download certificate by clicking on corresponding 'Download' button.



Downloading userkey.pem will require you to provide certificate password matching to your identity. Save the file to a location on your local machine.

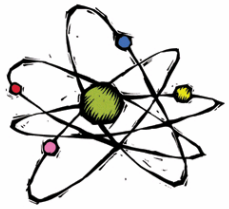


UABgrid applications

Dynamic BLAST «

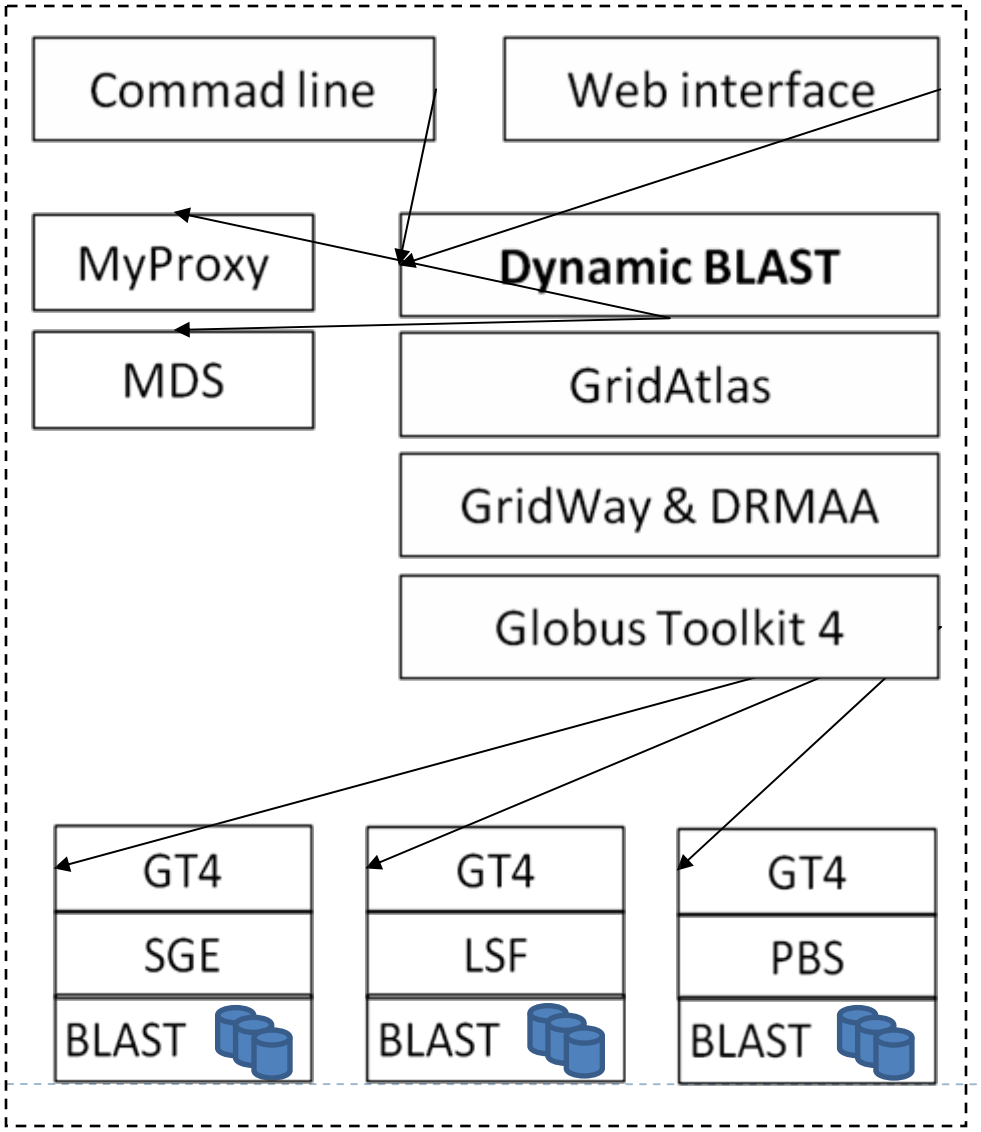
R «

Dynamic BLAST

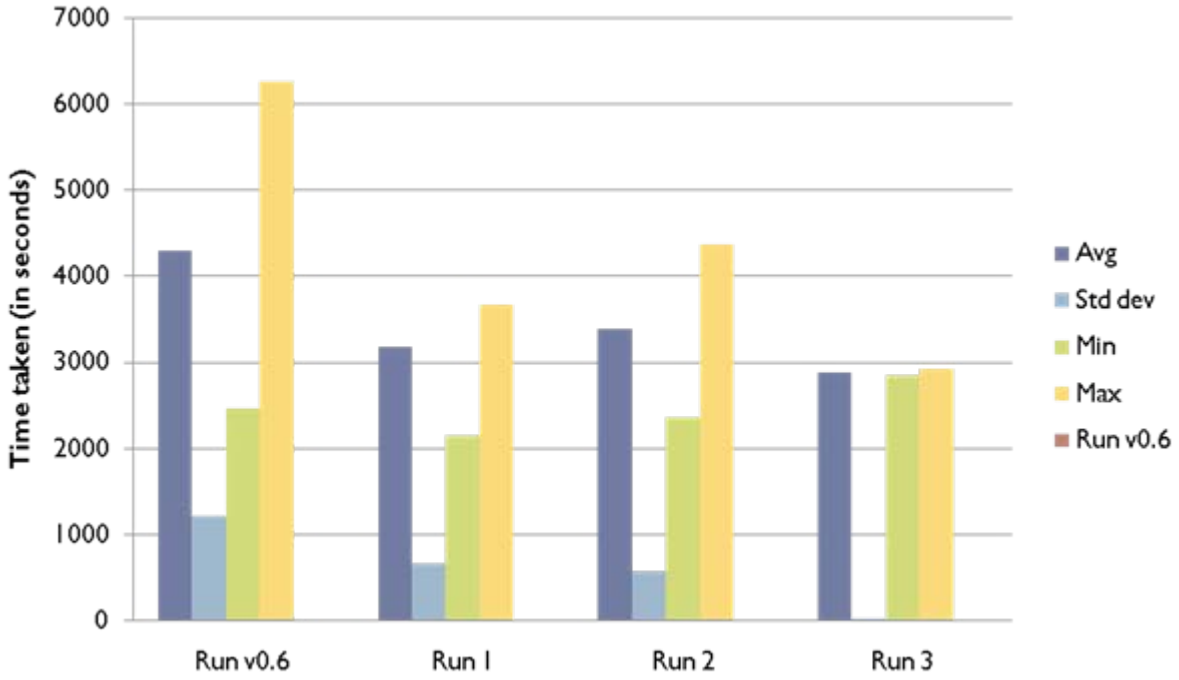


- Dynamic BLAST is a grid enabled version of BLAST
- Minimize requirement for user familiarity with grid related aspects
- Master-worker type application
- Built on grid middleware, tools, and standards to maximize application's portability, maintenance, and applicability
- Not only manage but rather take advantage of and variable resource availability and dynamic load balancing
  - Maximize resource utilization and minimize job turnaround
- Efficiently and transparently handle grid related issues such as: application availability, fault tolerance, interoperability arising because of resource heterogeneity

Dynamic BLAST & grid architecture



Job statistics for each run



- Must wait for longest fragment to complete
- Need to solve (at least) standard deviation minimization problem

Powering Statistical Genetics with Grids

- Applied Analysis
  - Real data from experimental results
  - Variables are regions of genome and may vary from 10K to 100K
  - Ex. identify region association with a disease
  - Occurs when experimental data is available
- Methodological Analysis
  - Random data generated for simulated analysis
  - Understand behavior of statistical methods used in real data analysis
  - 2k - 5k variables analyzed (maximum practical for existing, traditional 2 cluster resource pool)
  - Occurs repeatedly when methods are studied

R-A statistical package

- R heavily used in statistical genetics applied and methodological worflows
- Data analysis code written in R
- Process level and MPI granularity
- Embarrassingly (obvious) parallel (workflow)
- Workflows already benefit from cluster-scale computing (384CPUs, ~3Tflops)
- Expand to grid platform to fully power methodological workflow, consume 2 to 5 times compute power available on clusters
- Work in progress but uncovering a number of nagging issues in GT integration (SGE, MPI)

Reflects on R Effort

- Would have liked to focus on workflow migration issues, instead we had to focus on component compatibility issues
- Initial goal was intentionally simplistic but provide surprisingly challenging
- Don't be surprised when infrastructure is not ready
- Problems can be solved individually, but all will need to be solved before it will work – *this is a production application*

Issues encountered during development

- Software and technology versioning
  - ✓ Enables focused, simplified development
  - Increases dependencies
- Grid heterogeneity
- Infrastructure unreliability
- Error handling
- Documentation

Summary of suggestions for tool developers

- → Simple interfaces
- → Effective error handling
- → Good documentation
- → Adoption of standards
- → System stability
- → Consistency

Benefits of using GT4 on UABgrid

- Grid rolls with Rocks installation makes it easy to install GT
- Integration of GridWay and MDS4 with GT4
- Enables cost effective solutions (integrating UAB CIS, UAB HPC, ASA, SURAgrid)
- Integration of user management with InCommon simplifies access and provides consistent identity across web and non-web resources
- Highly visible project promises broad adoption and consistent resource interface

Challenges of using GT4 on UABgrid

- GT4 installation procedure
  - Quick start guide – quick at 50 pages?
  - Documentation (e.g., leveraging an existing CA)
- No support for SGE
  - Rocks Grid Roll seems to default to PBS
- Because GT provides a standard interface, need to specify/explain/document interface details to enable interface standardization and reliability
  - No conventions about what job types in GT mean when passed to an LRM (e.g., single, mpi, condor) -> supply best practice? (GT-bug #3348 May 2005)
- Support GRAM and LRMS separation
  - Allows site to manage multi-clusters as a unit and support secure installs (GT bug #3480 June 2005)

QUESTIONS?

UABgrid available at:

http://uabgrid.uab.edu