

# Michigan Grid Research and Infrastructure Development (MGRID)

**Abhijit Bose**

**MGRID**

**and**

**Dept. of Electrical Engineering and Computer Science**

**The University of Michigan**

**Ann Arbor, MI 48109**

**[abose@umich.edu](mailto:abose@umich.edu)**



# MGRID: Background

## ▪ Multiple Grid efforts at the UM

- Cluster Computing (ATLAS, CAC/NPACI, DZero, NCBI)
- Automated network configuration and testing, Network QoS reservation (CITI, ITCOM)
- Remote Instrument (SI – NEES Earthquake Grid)
- Collaborative tools (SI – CHEF Collaboration portal)
- Data base searches (Bioinformatics, MCBI)
- Malware threat detection grid

## ▪ Integration challenge for UM



# Collaborators

## ■ Who is MGRID?

- School of Information
- Campus Computing Sites
- Center for Information Technology Integration (CITI)
- Department of Physics
- LSA
- College of Engineering
- Center for Advanced Computing
- Duderstadt Center
- Information Technology Communications (ITCOM)
- Michigan Center for Biological Information (MCBI)
- ... and many more ...



# Why MGRID

- **Grid software (Globus etc.) is difficult to run, complex to install and manage**
  - Promote ease of use
  - More time to do science, instead of IT management
- **How to prototype the Grid to fit into UM IT environment**
  - Large (> 100,000) user base for Grid service
  - Produce a generalized Grid service
- **Leverage existing security and group services**
- **Add Fine grained policy driven access control**
  - Let the owners of resources control their resource
    - Who, what, where, when, and how
    - But make it easy for them to do so



# MGRID Funding

- Goal: build **pilot institutional grid**

## Founding Partners



## External Sponsors



NFS



NMI/NSF



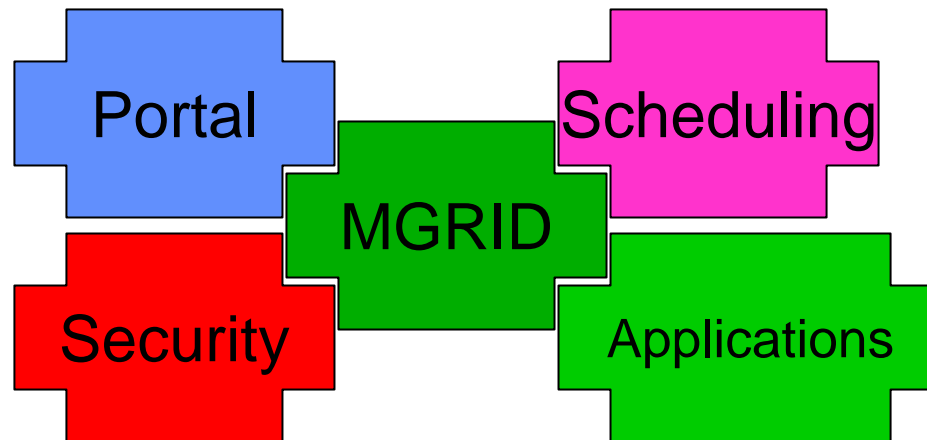
NEES



Mid-America Earthquake Center



# MGRID Overview



# MGRID Projects

## **CORE INFRASTRUCTURE**

### **Kerberos Leveraged PKI**

- kx509 Clients, KCT and mod\_KCT Apache web server modules.
- authenticates users against Globus Gatekeepers (password-less)

### **MARS**

- resource provisioning, scheduling and resource management
- fault-tolerance
- tunable resource scheduling, scheduling research (utility-driven)

### **GridNFS**

- integrates distributed file system (NFSv4) and flexible identity management to meet the needs of grid-based virtual organizations.



# MGRID Projects

## CORE INFRASTRUCTURE

### WALDEN

- eliminates the need to manage user identities on hosts that participate in a grid environment. This is accomplished by moving user authentication to the client, replacing the static mapping between X.509 identities (Distinguished Names) and local user names in the Globus grid-mapfile with a dynamic approach using secure LDAP.

### Accounting

- allows usage reports on disparate scheduler log formats, such as PBSPro and Condor. Usage logs are translated into a common, standard XML format (defined by GGF UR-WG).



# MGRID Projects

## *MGRID APPLICATIONS AT MICHIGAN*

- ATLAS
- UltraLight
- NEESGrid
- Bio-Physics
- Chemistry
- Agent-Based Simulations
- NTAP
- Secure Multipoint Video-Conferencing
- Distributed Threat Detection

## *PORTAL SOFTWARE*

- MGRID Portal
- SAKAI/MGRID



# Existing Infrastructure

- **Uniqname**
  - Unique campus wide user name to UID
- **Kerberos V5 (multiple cells)**
- **KX509**
- **Group Services**
  - AFS groups, LDAP
- **Directory services**
  - LDAP



# MGRID Portal

- **User workstation**
  - KX509 to obtain user X509 credentials
  - KX509 Certificate available to browser
- **Additions to OpenSSL (in 9.0.7), required on MGRID Portal**
  - SSL handshake recorded
- **MGRID Portal SSL configured to require user X509 credentials**

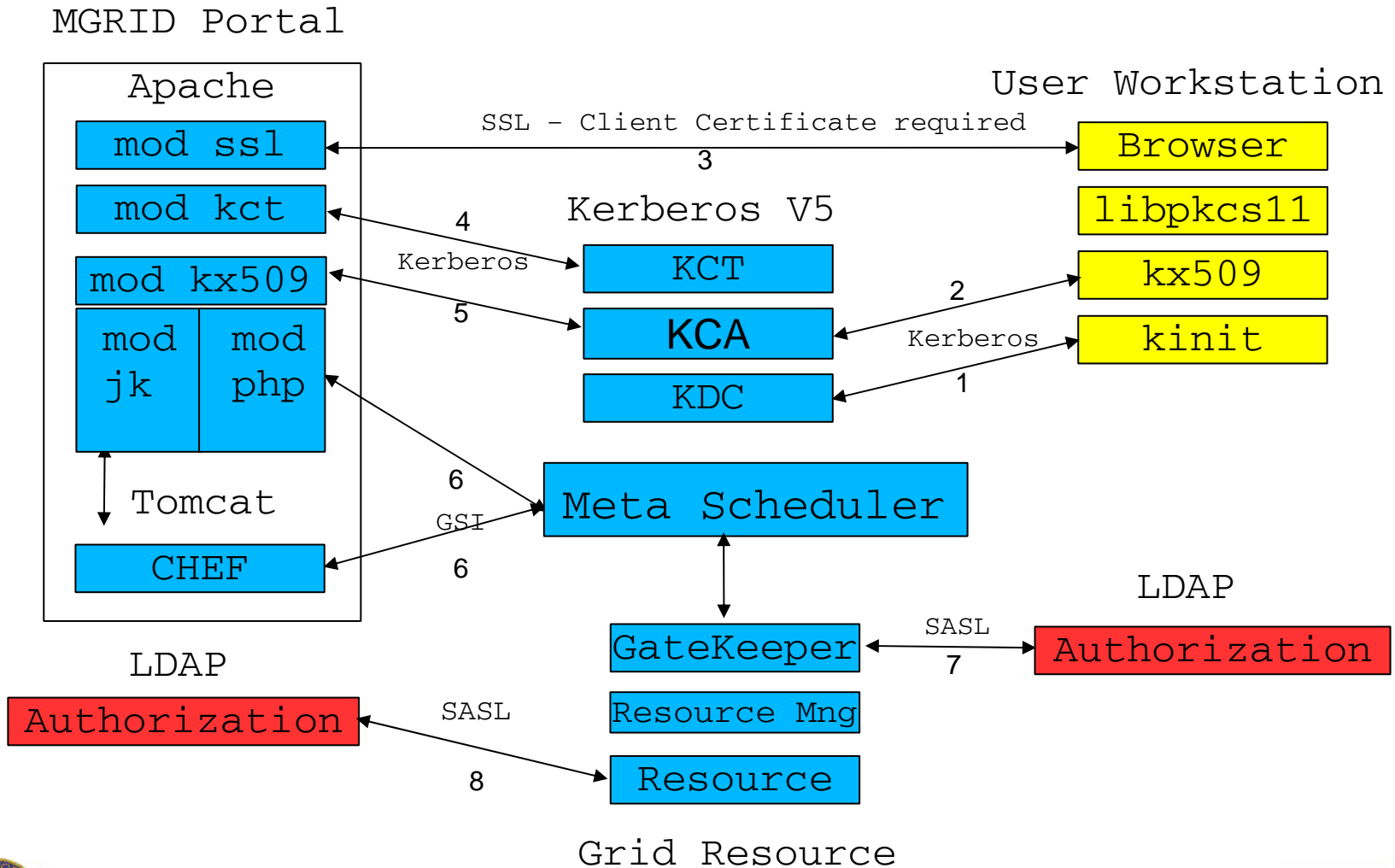


# MGRID Portal

- **Hides complexity from user**
- **Individual or Organizational presentation**
  - CHEF
- **Easily extensible**
  - Add new Grid applications
  - With generic Grid resource, can run any back-end program
- **Built on *strong security***



# MGRID Architecture



# MGRID Project Highlights

- **Multi-Resource Scheduling (MARS)**
- **Authorization (Walden)**
- **Accounting**
- **NTAP (Network Testing and Performance)**



# A Sample of MGRID Applications

- **Gaussian (Chemistry, Bio-Physics)**
- **Molecular Dynamics (user-developed)**
- **BLAST**
- **MATLAB**
- **Agent-Based Simulations (Economics)**



# The MARS Project

## Goals:

- Develop a framework for co-scheduling distributed resources and workload management
- Develop algorithms for fault-tolerant scheduling in support of extreme-scale computing

NSF Award # 0444417 (10/2004-09/2007), Ford Motor Company, MGRID, Altair



# MARS: Design Goals

- *Extensible Architecture*
  - Multiple standards for job description: JSDL, DRMAA, GRAAP
  - Remote communications with local resource managers
  - New scheduling algorithms can be easily incorporated
- *On-Demand Task Scheduling*
  - Resources on-demand (prioritized task queues and pre-emption of lower priority tasks). Example: Solar weather prediction, disaster management
- *Resource Usage Forecasting*
  - Efficient scheduling decisions across multiple resources
  - MARS currently uses low-pass filters (exponential smoothing)

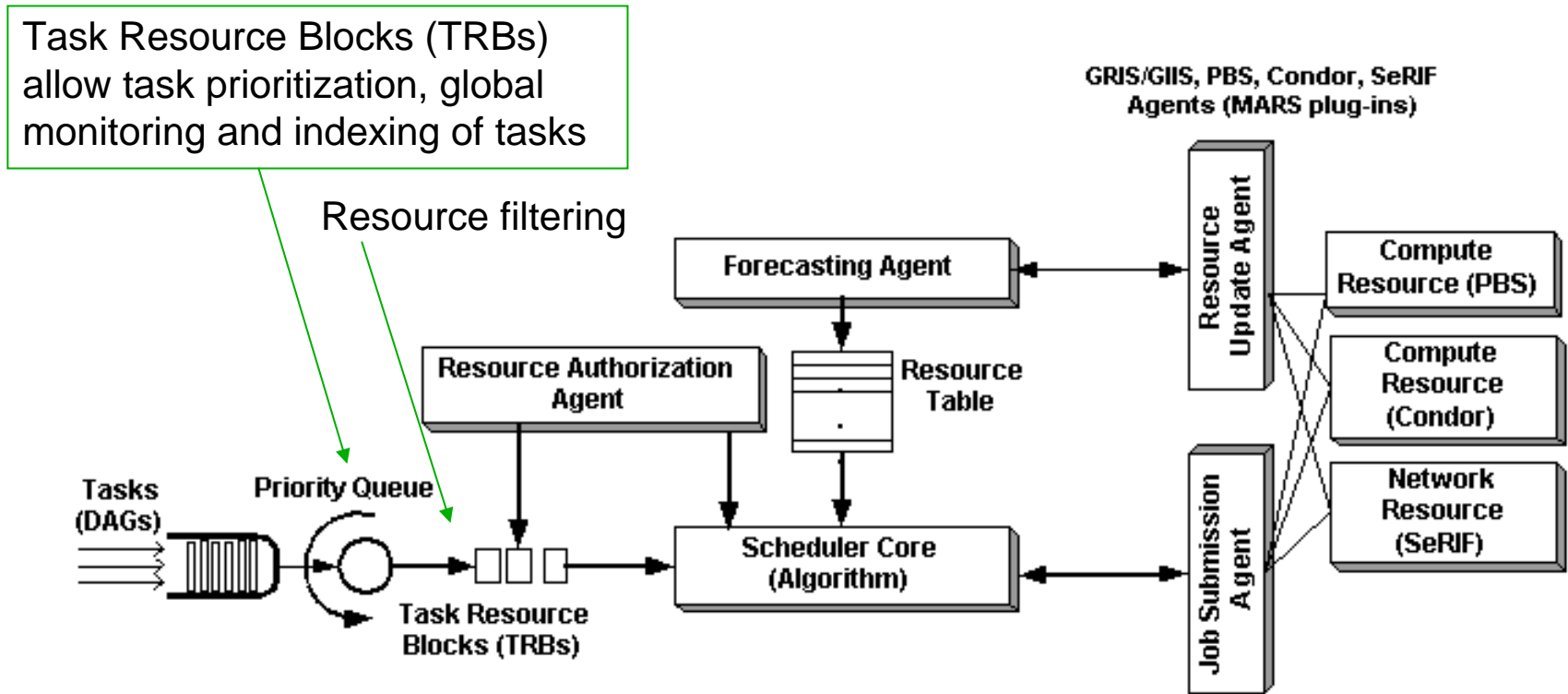


# MARS: Design Goals

- *Allow Globus and 3<sup>rd</sup> Party Deployment Paths*
  - Allow Globus MDS (Resource Discovery) and GSI (Security)
  - Allow building MARS library with and without Globus support
  - Allow kx509 proxy certificate
  - Allow 3<sup>rd</sup> party resource monitoring frameworks
  - Non-Globus path is also allowed
- *Fault-tolerance against Internal and Resource-level Faults*
  - Snapshots and incremental checkpointing of workload buffers
  - Allow transfer of running and queued workload to other resources
  - Allow internal buffers when resources are full
- *Allow 3<sup>rd</sup> party authorization and policy APIs (XML/SOAP)*



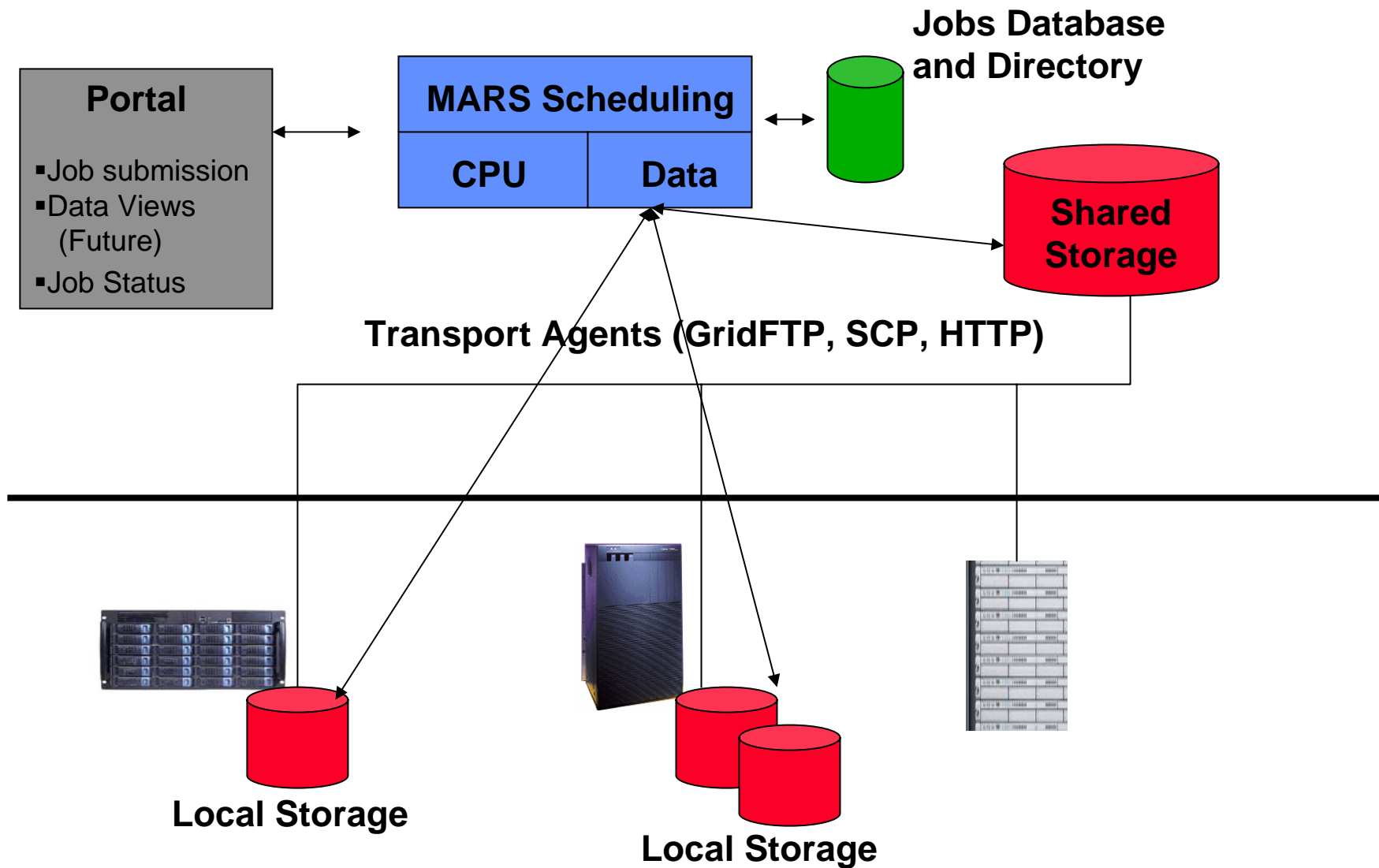
# MARS: Architecture



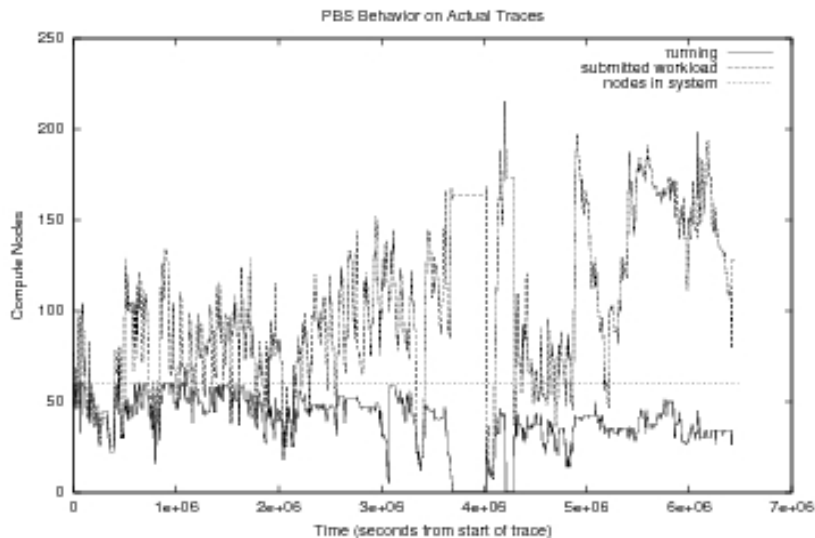
- Each task gets a TRB that includes a MARS JobID
- Individual schedulers assign their own JobIDs
- TRBs encapsulate these IDs



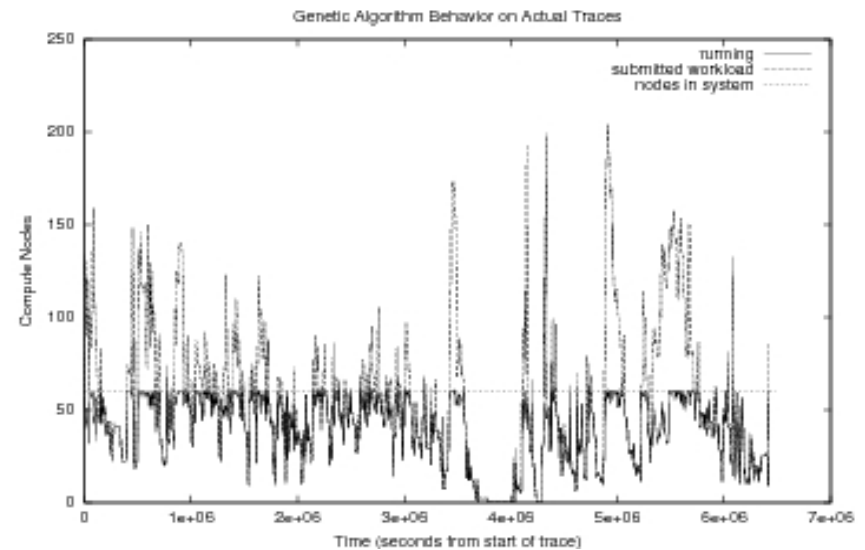
# MARS in A Typical Enterprise Data Center



# MARS: Better Workload Management



Schedule generated by PBS  
(3 month period, 120 CPUs)



Schedule generated by GA-MARS  
(3 month period, 120 CPUs)



# MGRID Authorization

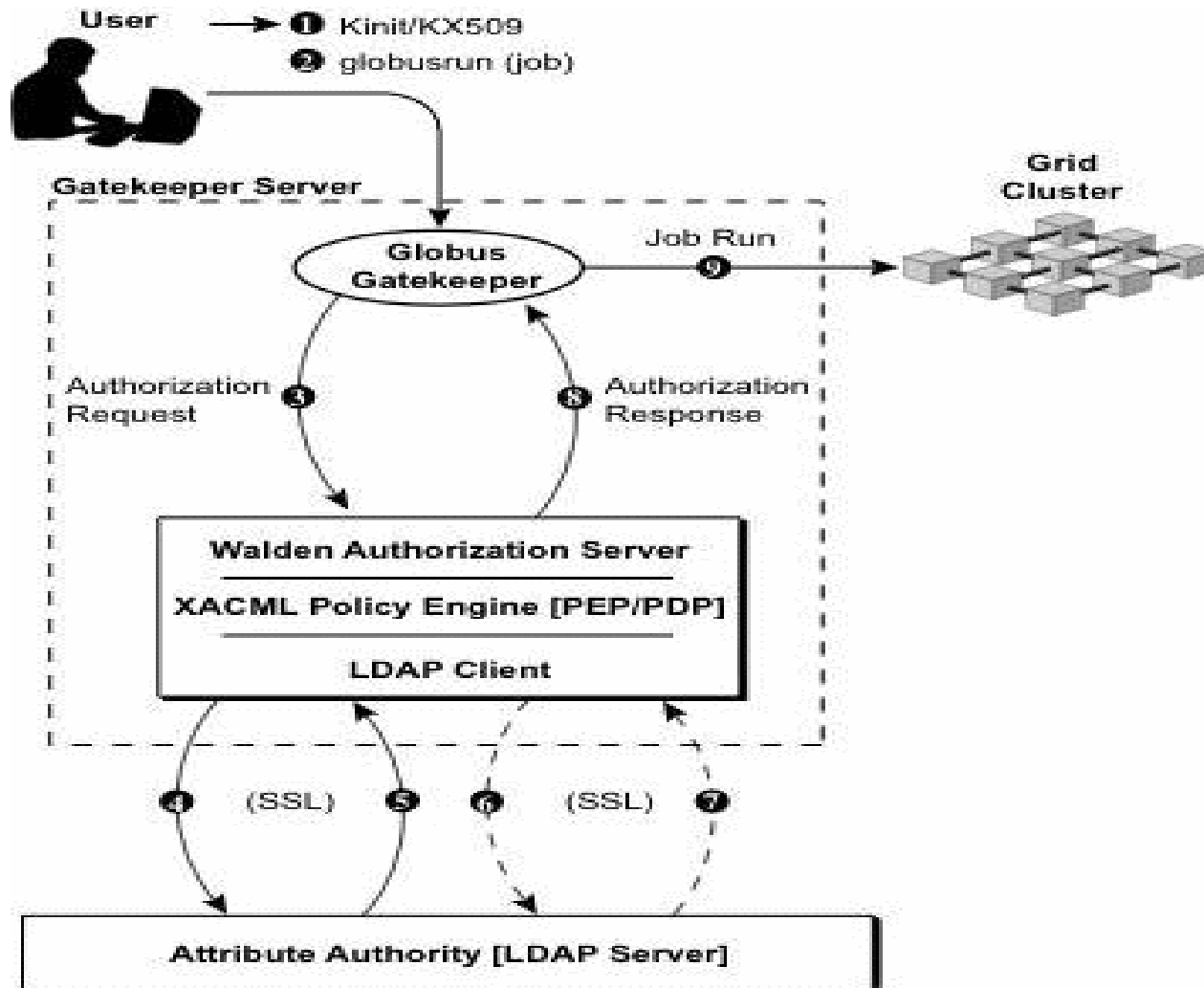
- Globus provides a static grid-mapfile for coarse-grained authorization
- Each grid-mapfile is locally maintained on each resource, mapping a user's X.509 DN to a local account
- Users either share local accounts, providing little accountability, or are granted unique local accounts, creating administrative problems
- How to provide fine-grained authorization with one-to-one user-account mapping?



# Walden Authorization

- **Fine-Grained authorization** module based on XACML standard (XACML-based policy engine)
- Cluster owners have complete administrative control over who uses their resources
- **Policy files** define rules based on group membership, time of day, resource load, etc.
- Local account management is *unnecessary*
- **Group membership** can be assigned from one or several secure LDAP servers





- ① Attribute Find (Groups)
- ② Attribute Response (Groups)
- ③ Attribute Find (Username)\*\*
- ④ Attribute Response (Username)\*\*

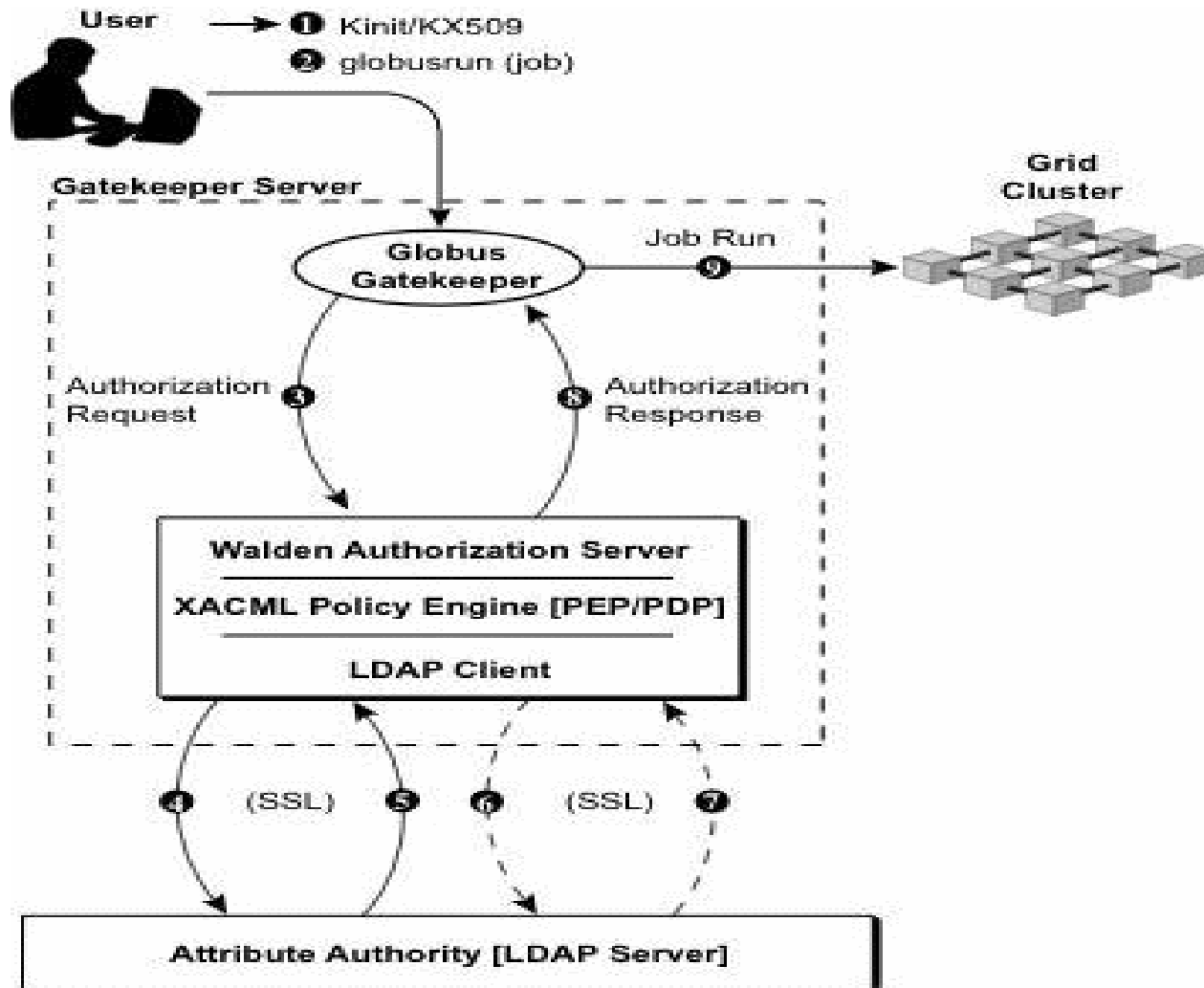
\*\*Optional



# Walden Authorization

- **Step 1:** Obtain a Kerberos V Ticket Granting Ticket (TGT), which is then used to obtain and cache a KX.509 certificate.
- **Step 2:** Submit a job request to Globus gatekeeper
- **Step 3:** Gatekeeper invokes gridmap callout function, forwarding authorization request to Walden module.
  - **Policy Enforcement Point (PEP) formats and sends request to Policy Decision Point (PDP).**
  - **PDP retrieves XACML policy (if necessary) from central policy repository**





- ① Attribute Find (Groups)
- ② Attribute Response (Groups)
- ③ Authorization Request
- ④ Authorization Response
- ⑤ (SSL)
- ⑥ Attribute Find (Username)\*\*
- ⑦ Attribute Response (Username)\*\*

\*\*Optional



# Walden Authorization

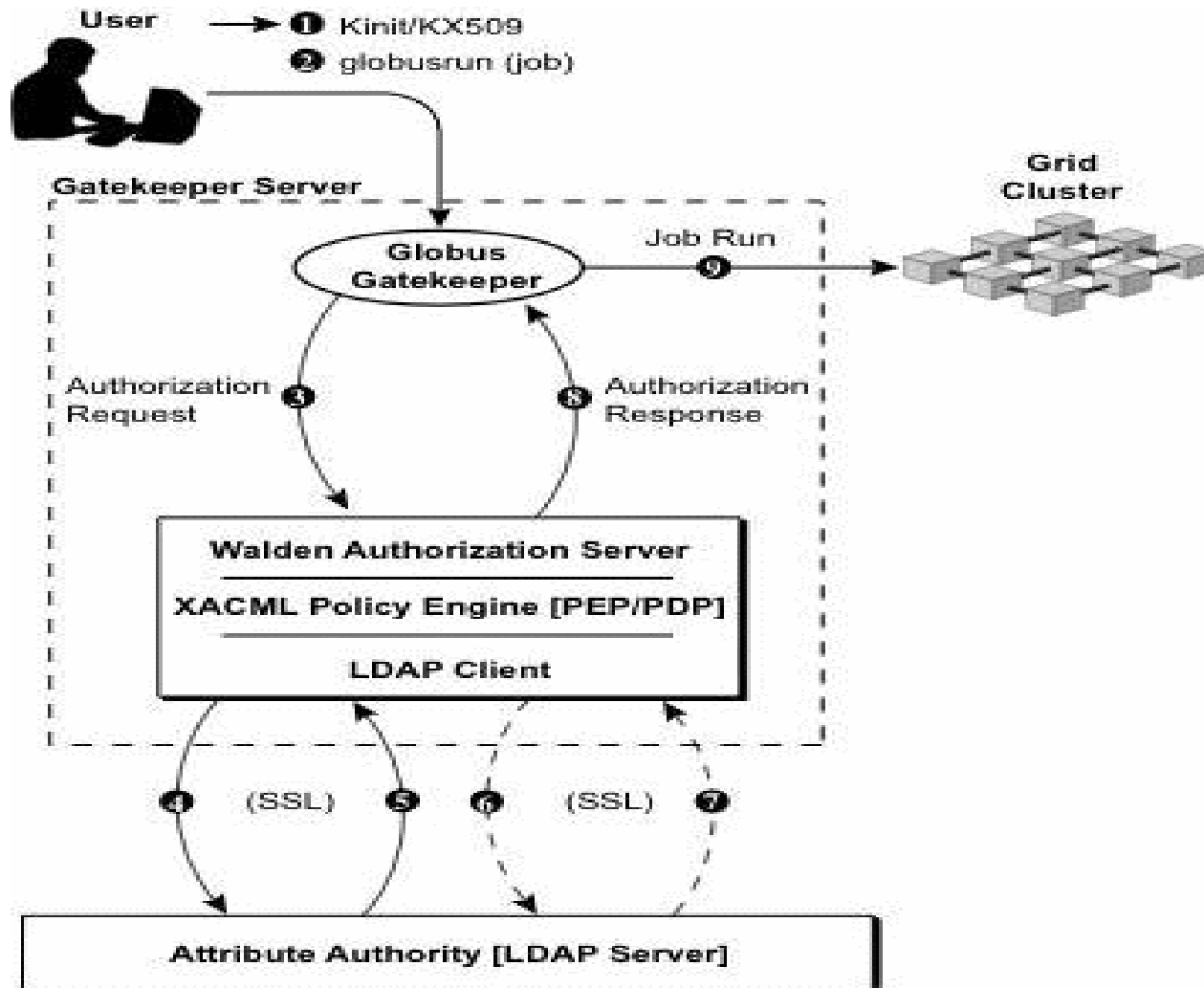
- **Step 4-5:** Policy Decision Point (PDP) retrieves a 'bag of attributes' corresponding to user from secure LDAP server, and extensible to many other sources.
  - **User attributes (e.g. Group Membership) is compared against authorization request**
  - **PDP returns a response of Permit, Deny, or indeterminate, along with any obligations.**



# Walden Authorization

- **Step 6-7:** Policy Enforcement Point (PEP) parses response and obligations.
  - **If no defined obligations, PEP binds user to (permanent) local account from secure LDAP query.**
  - **If guest user obligation defined, PEP binds user to available guest account.**





- ① Attribute Find (Groups)
- ② Attribute Response (Groups)
- ③ Attribute Find (Username)\*\*
- ④ Attribute Response (Username)\*\*

\*\*Optional



# Walden Authorization

- **Step 8:** If the user is authorized, the local account identity is returned to globus (otherwise, authorization is denied).
- **Step 9:** The globus gatekeeper submits the authorized job request to the grid cluster, using the defined permanent or guest user account.

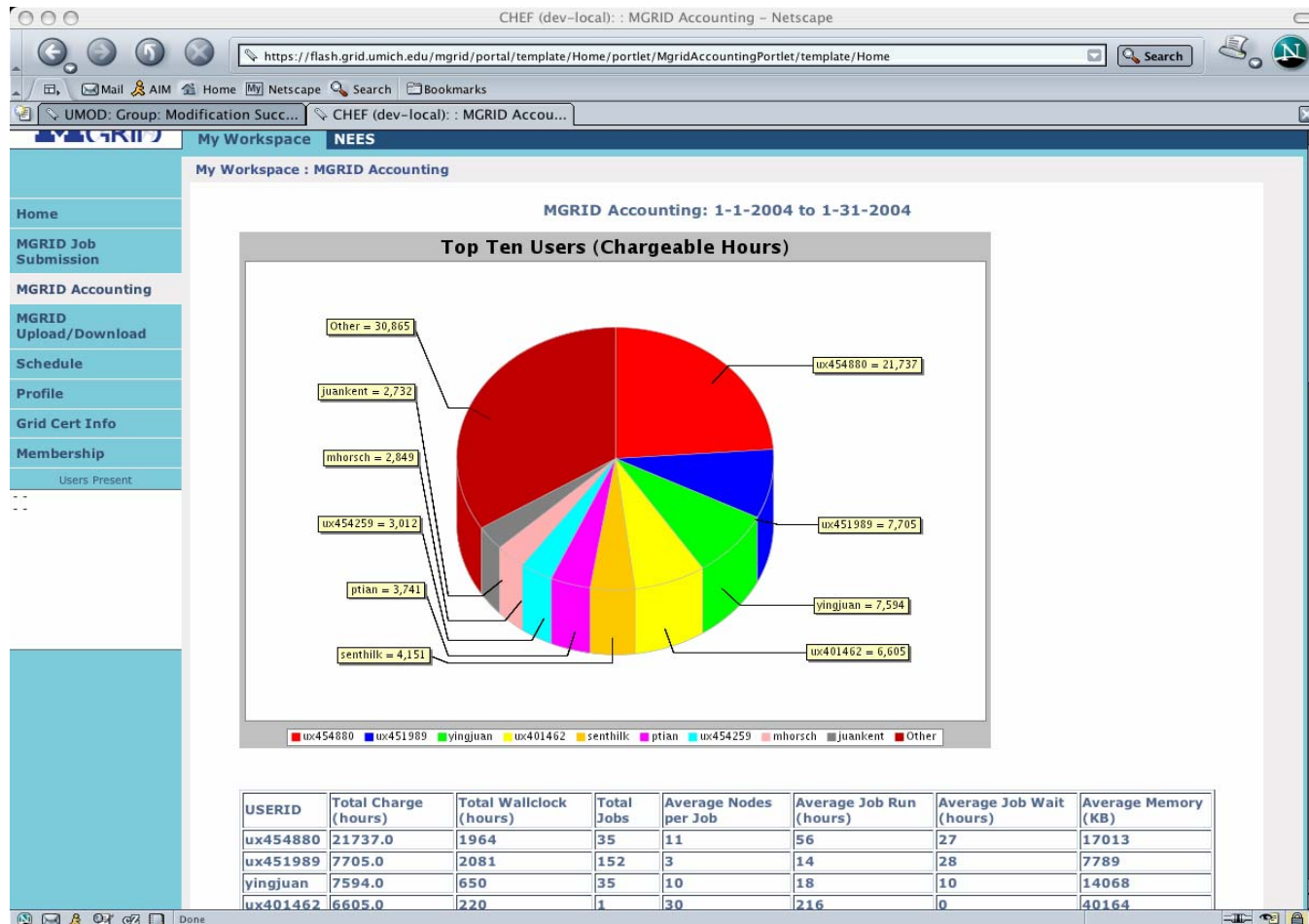


# MGRID Accounting

- Step 1: Grid scheduling software (e.g. PBSPro, Condor) generates usage log files in various formats
- Step 2: MGRID Accounting translates usage log files into common XML format
- Step 3: MGRID Accounting ingests data into MySQL database for report generation and review
- Conforms to GGF Accounting Schemas

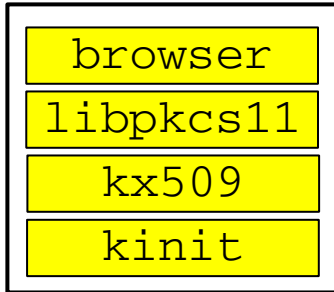


# Usage data displayed in graphical and tabular format



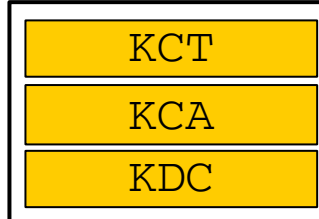
# NTAP Architecture

## User Workstation

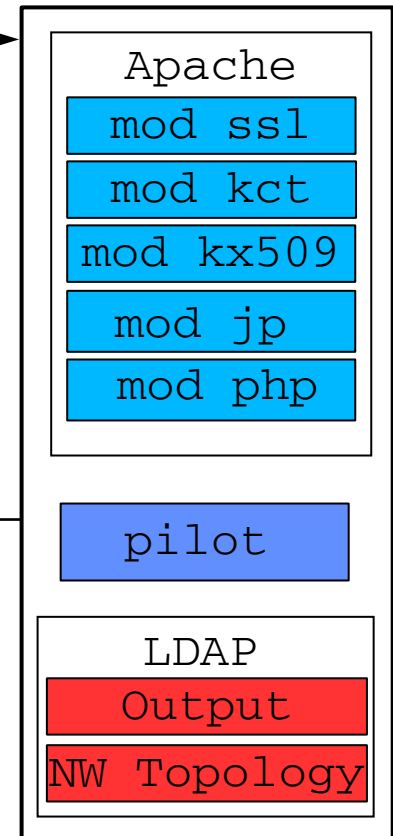


1. The user authenticates to the portal host via kx.509 and submits a network test request

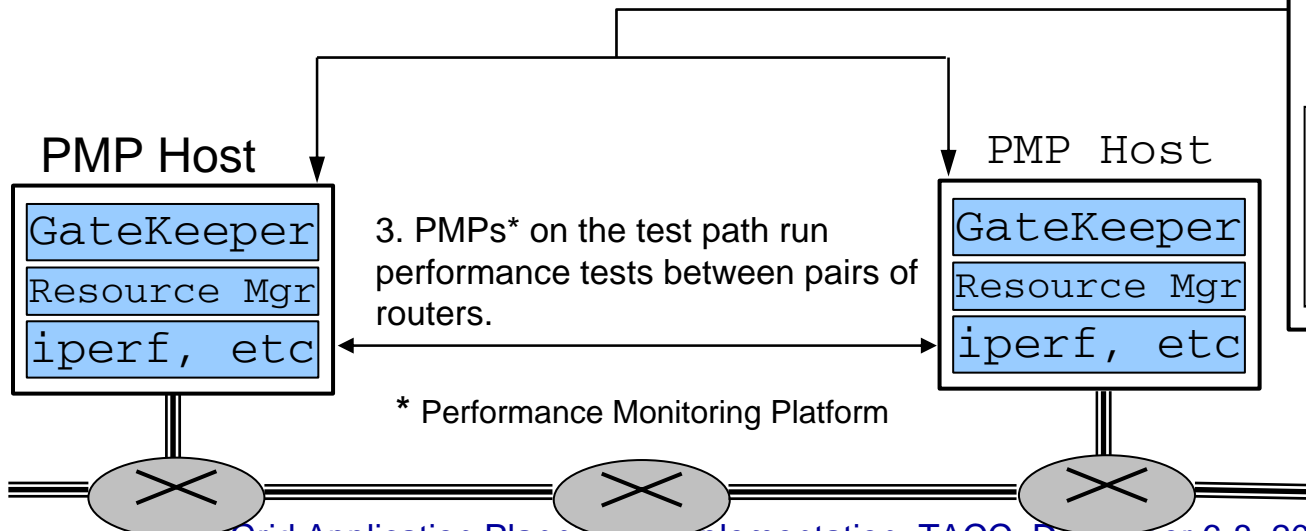
Kerberos V5



## Portal Host



2. The portal host constructs a path between specified endpoints, issues test reservations, and updates the output database.



3. PMPs\* on the test path run performance tests between pairs of routers.

\* Performance Monitoring Platform

4. The portal host displays results.



# Thank you!

