The Virtual Grid Application Development Software (VGrADS) Project

The VGrADS Team

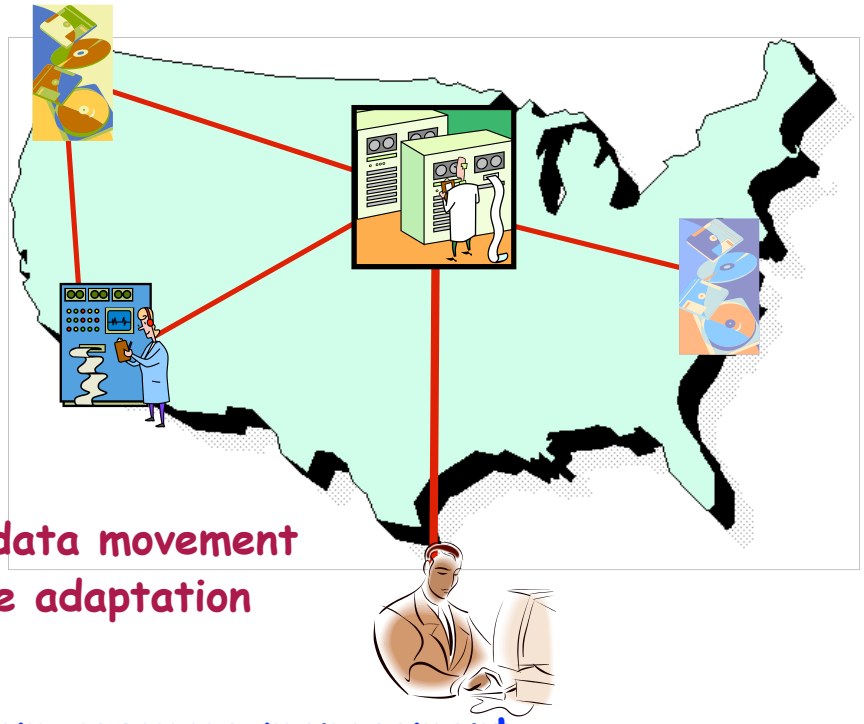
THE UNIVERSITY of NORTH CAROLINA at CHAPEL HILL

http://vgrads.rice.edu/

VGrADS
Virtual Grid Application Development Software Project

VGrADS Goal: Distributed Problem Solving

- Where We Want To Be
  - Transparent computing
    - In an increasingly distributed space
    - Applications to HPC
    - Applications to cloud computing
- Where We Were (circa 2003)
  - Low-level hand programming
  - Programmer had to manage:
    - Heterogeneous resources
    - Scheduling of computation and data movement
    - Fault tolerance and performance adaptation
- What Progress Have We Made?
  - Separate application development from resource management
    - VGrADS provides a uniform "virtual grid" abstraction atop widely differing resources
  - Provide tools to bridge the gap
    - Scheduling, resource management, distributed launch, simple programming models, fault tolerance, grid economies



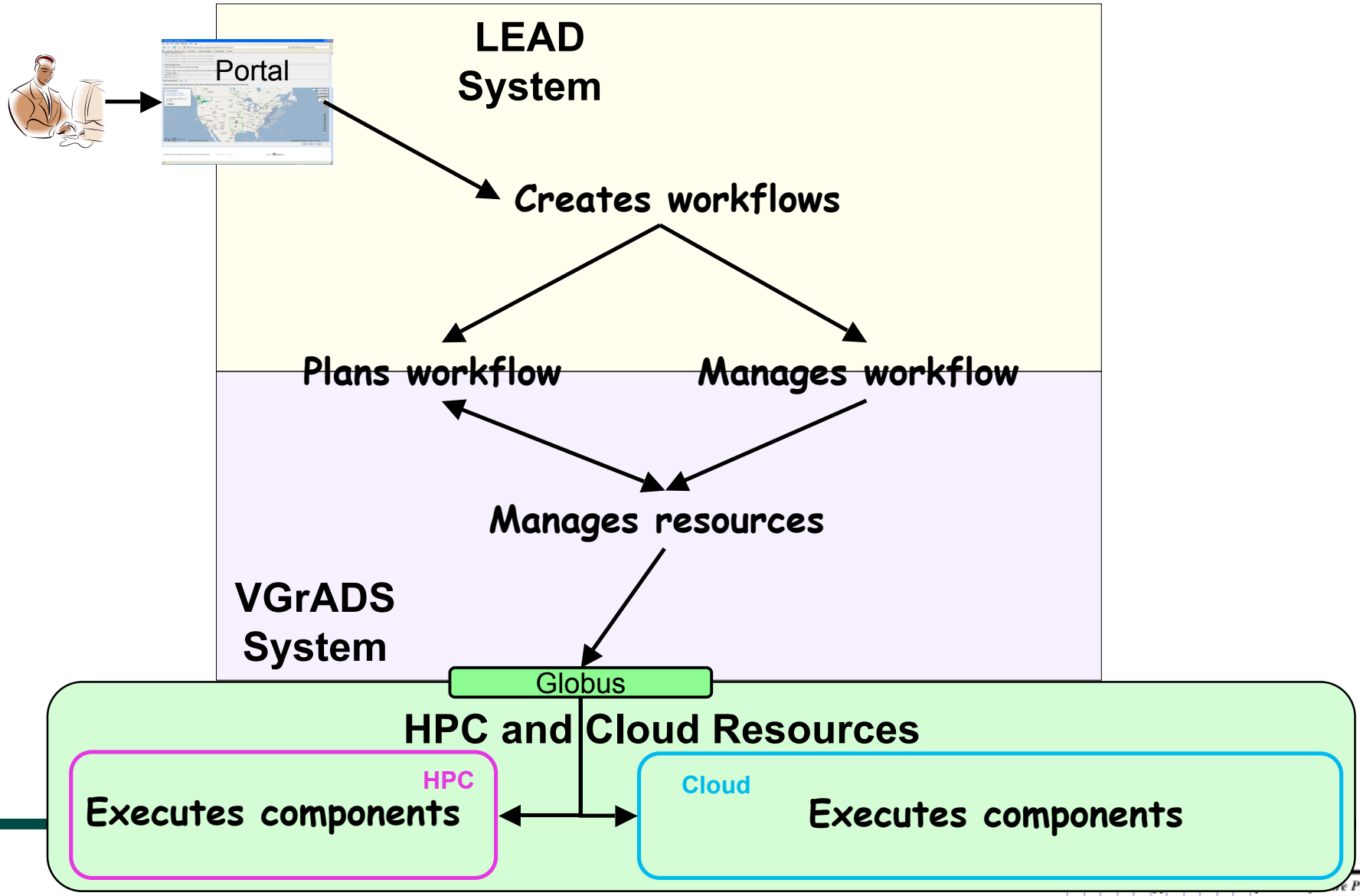
VGrADS Goal: Distributed Problem Solving

- Where We Want To Be
  - Transparent computing
    - In an increasingly distributed space
    - Applications to HPC
    - Applications to cloud computing
- Where We Were (circa 2003)
  - Low-level hand programming
  - Programmer had to manage:
    - Heterogeneous resources
    - Scheduling of computation and data movement
    - Fault tolerance and performance adaptation
- What Progress Have We Made?
  - Separate application development from resource management
    - VGrADS provides a uniform "virtual grid" abstraction atop widely differing resources
  - Provide tools to bridge the gap
    - Scheduling, resource management, distributed launch, simple programming models, fault tolerance, grid economies

Overview of SC08 Demo

- Building on SC04 - SC07 demonstrations
  - Gradually built up system to handle LEAD workflow
  - Previous years focused improved performance estimates, scheduling methods, fault tolerance
  - Use LEAD as an application driver
- New for SC08
  - VGrADS integrates HPC and cloud resources
    - Using TeraGrid (HPC), Amazon EC2 (cloud), Eucalyptus (cloud) resources
    - Using reservations, batch queues, and on-demand clouds
  - Scheduling for balancing deadlines, reliability, and cost
    - vgES supports search for best set of resources
    - Application-specific trade-offs of reliability, time, cost
  - Abstractions really do work!

Demo Architecture



Executing LEAD Workflow Sets

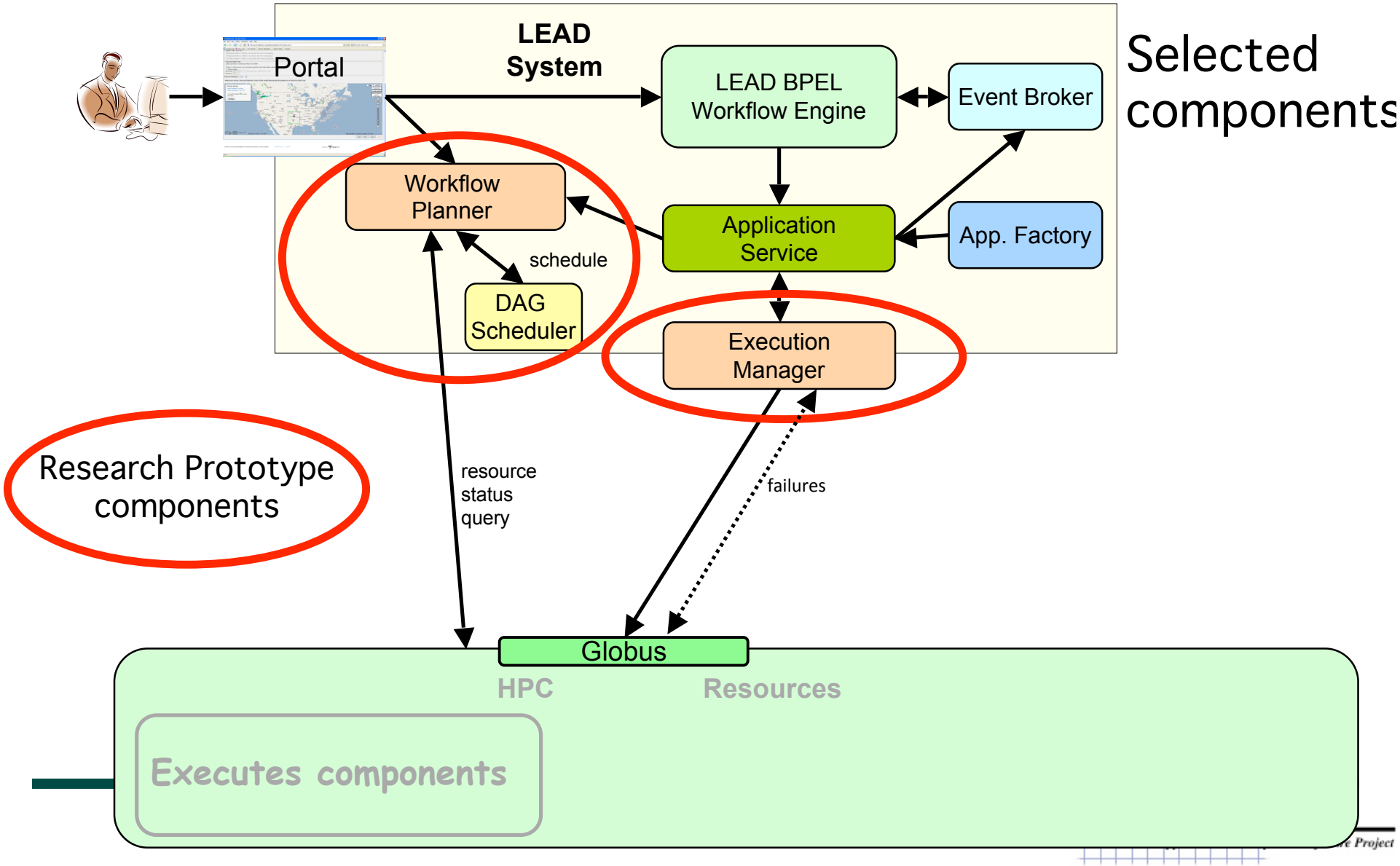
www.leadproject.org

- Demonstrate planning and execution of LEAD workflow sets execution atop virtualized cloud and Grid resources.
- LEAD Workflow Orchestration schedules a set of independent workflows with characteristics
  - a deadline D (e.g. 2 hours)
  - fraction F such that at least F of the workflows finish by the deadline (e.g. 3/8)
- Virtual Grid Execution System (vgES) provides an abstraction over batch and cloud systems including Amazon EC2 and Eucalyptus cloud sites.

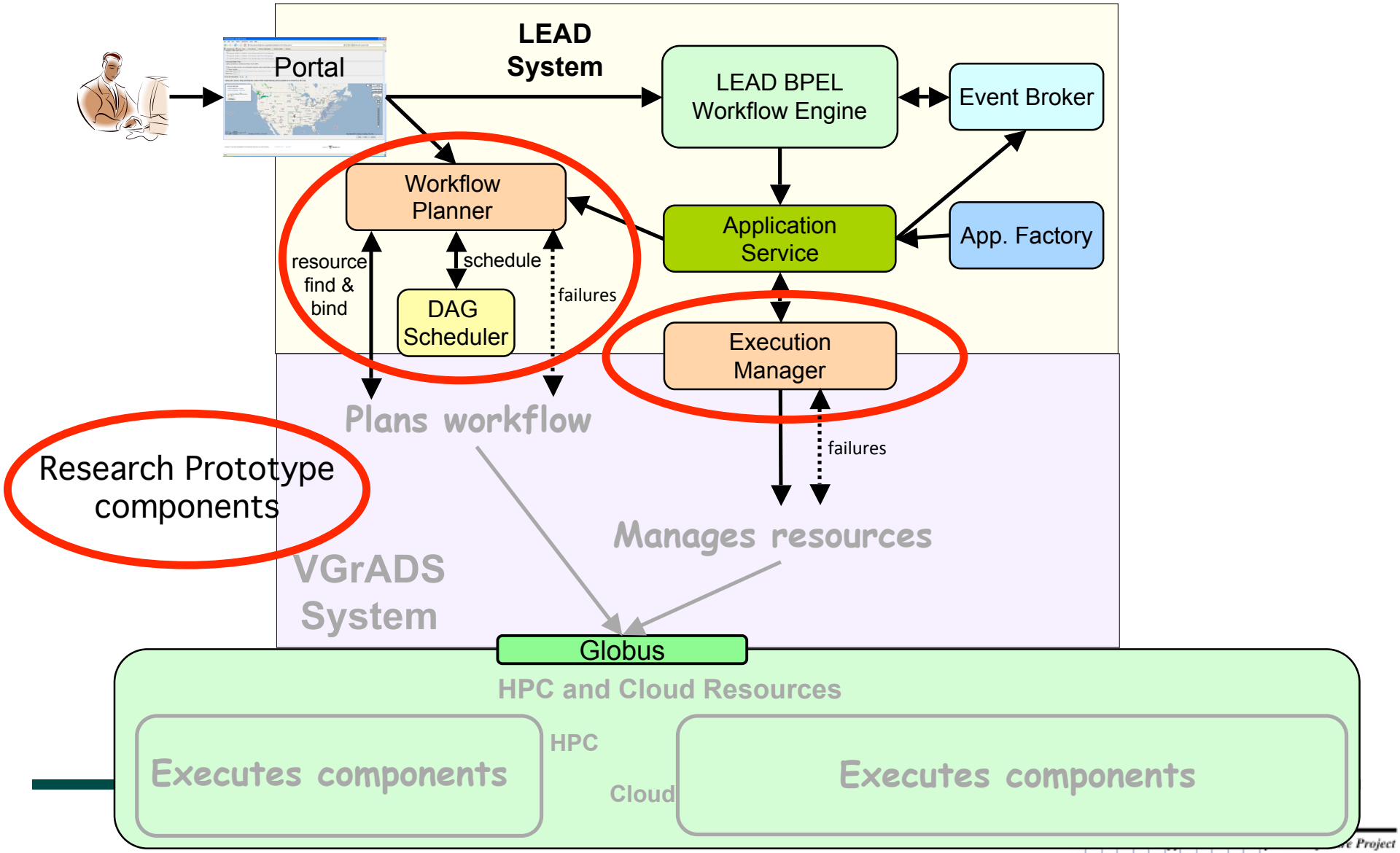
VGrADS
Virtual Grid Application Development Software Project

LEAD System (without VGrADS)

www.leadproject.org



LEAD System (with VGrADS)

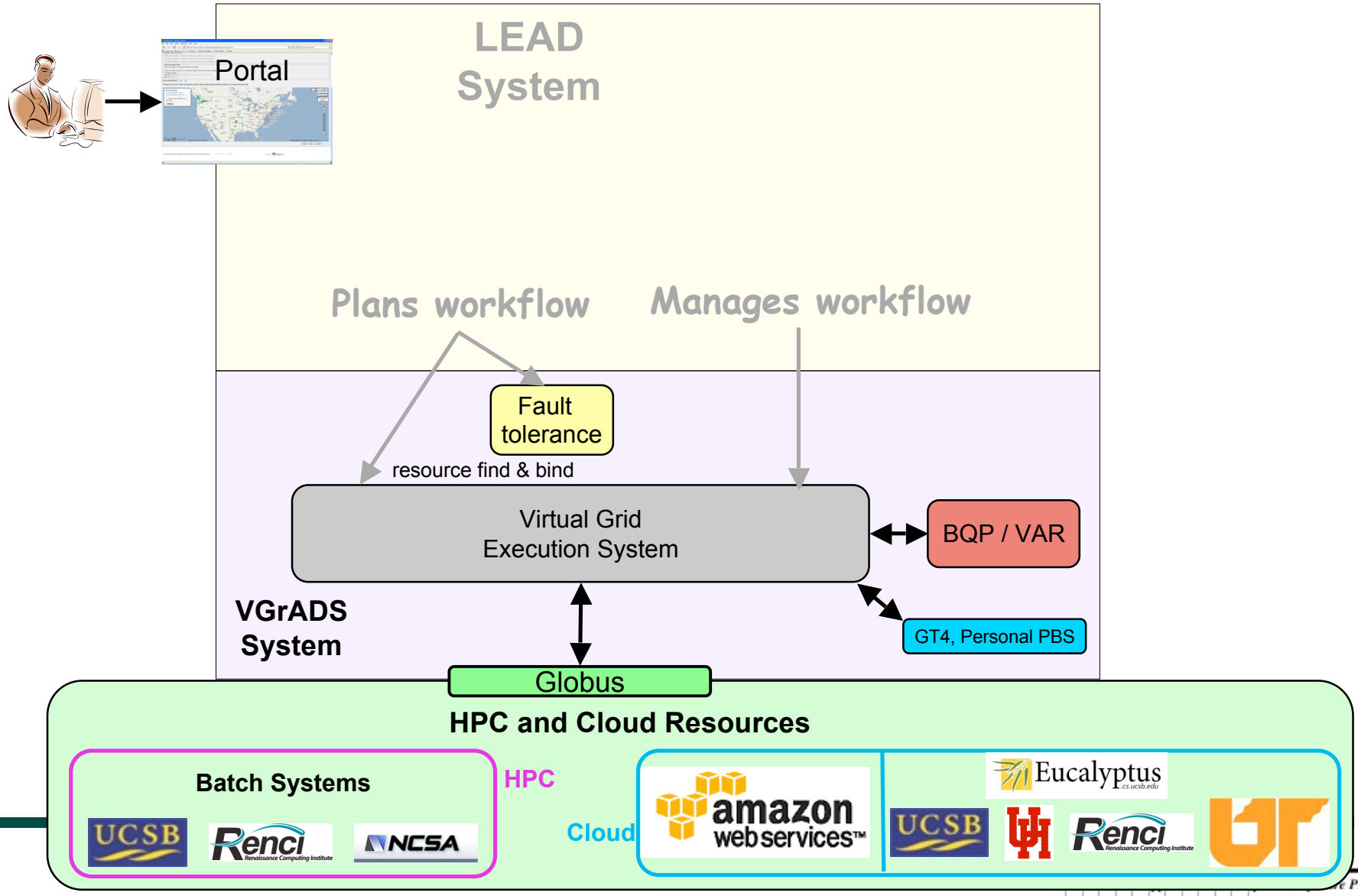


VGrADS Components at SC08

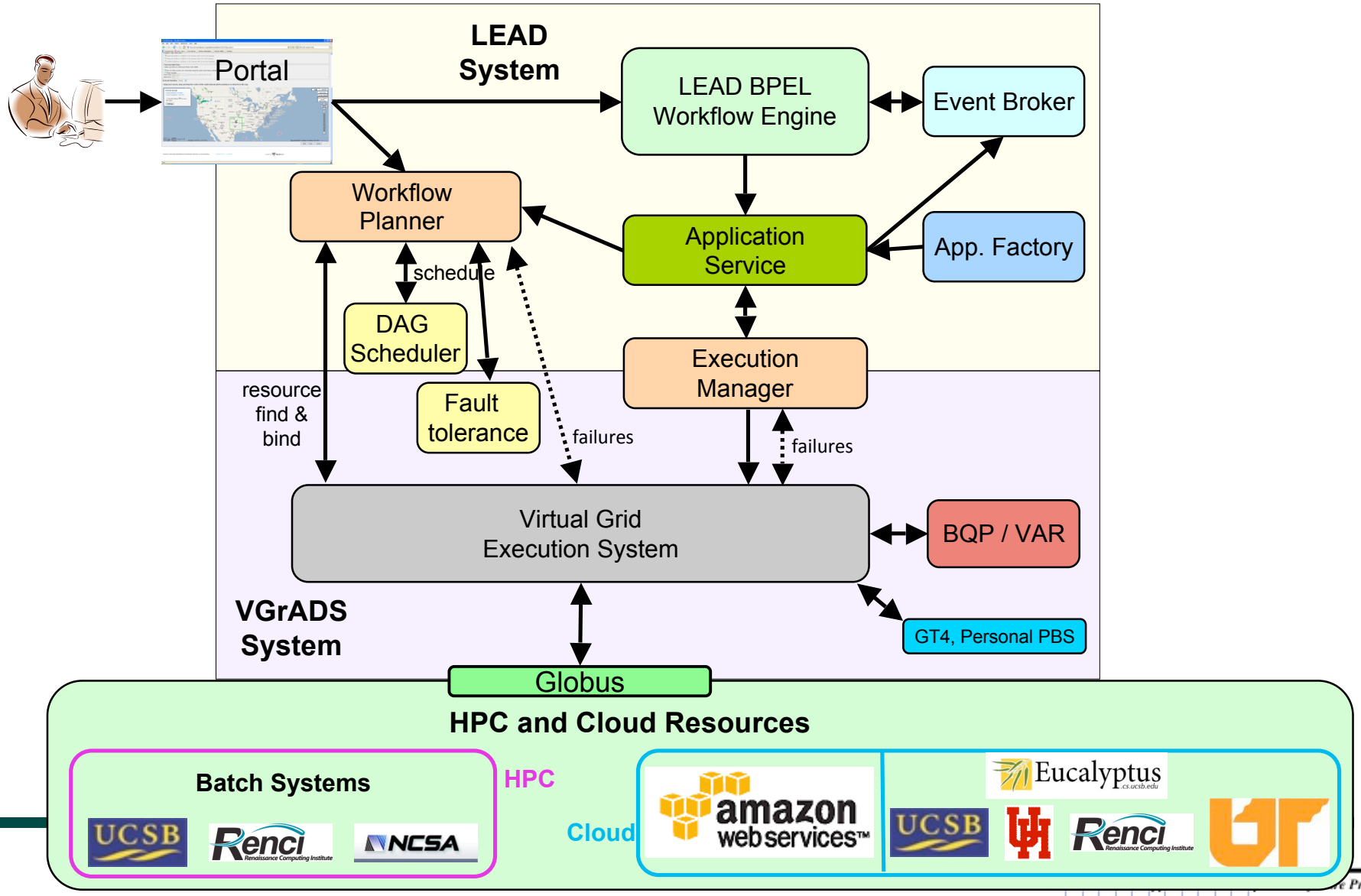
- Virtual Grid Execution System (vgES)
  - Uses Amazon EC2 tools to interact with cloud resources
  - Uses QBETS and Globus to provision batch resources
  - Uses Personal PBS to control execution on batch resources
  - Provides a "resource gantt chart" view of resources to aid higher level workflow orchestration tool
- Eucalyptus – *Developed for VGrADS*
  - Implements cloud computing on Xen-enabled clusters
  - Open-source software infrastructure that is compatible with Amazon EC2

  → vgES "thinks" a Eucalyptus cloud is EC2
- Fault Tolerance (FTR)
  - Schedules a task to increase the probability of successful execution of a task up to a desired level, constrained by resource availability and application deadlines

VGrADS System



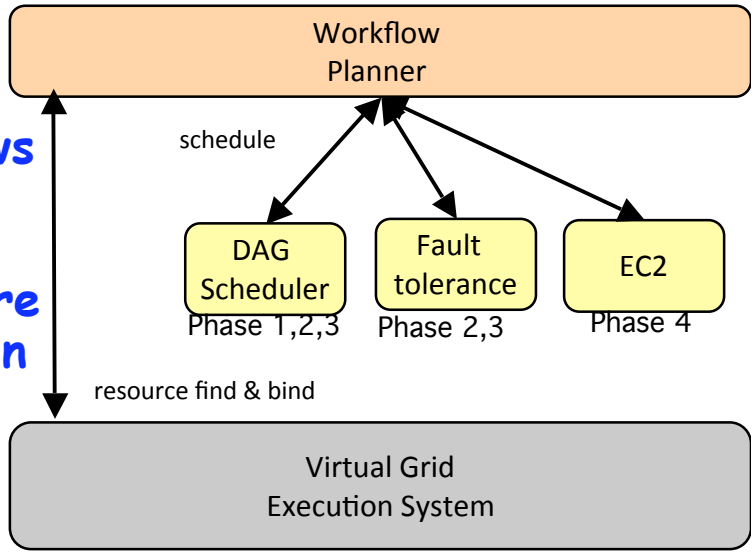
LEAD / VGrADS Architecture



And Now, On With the Demo

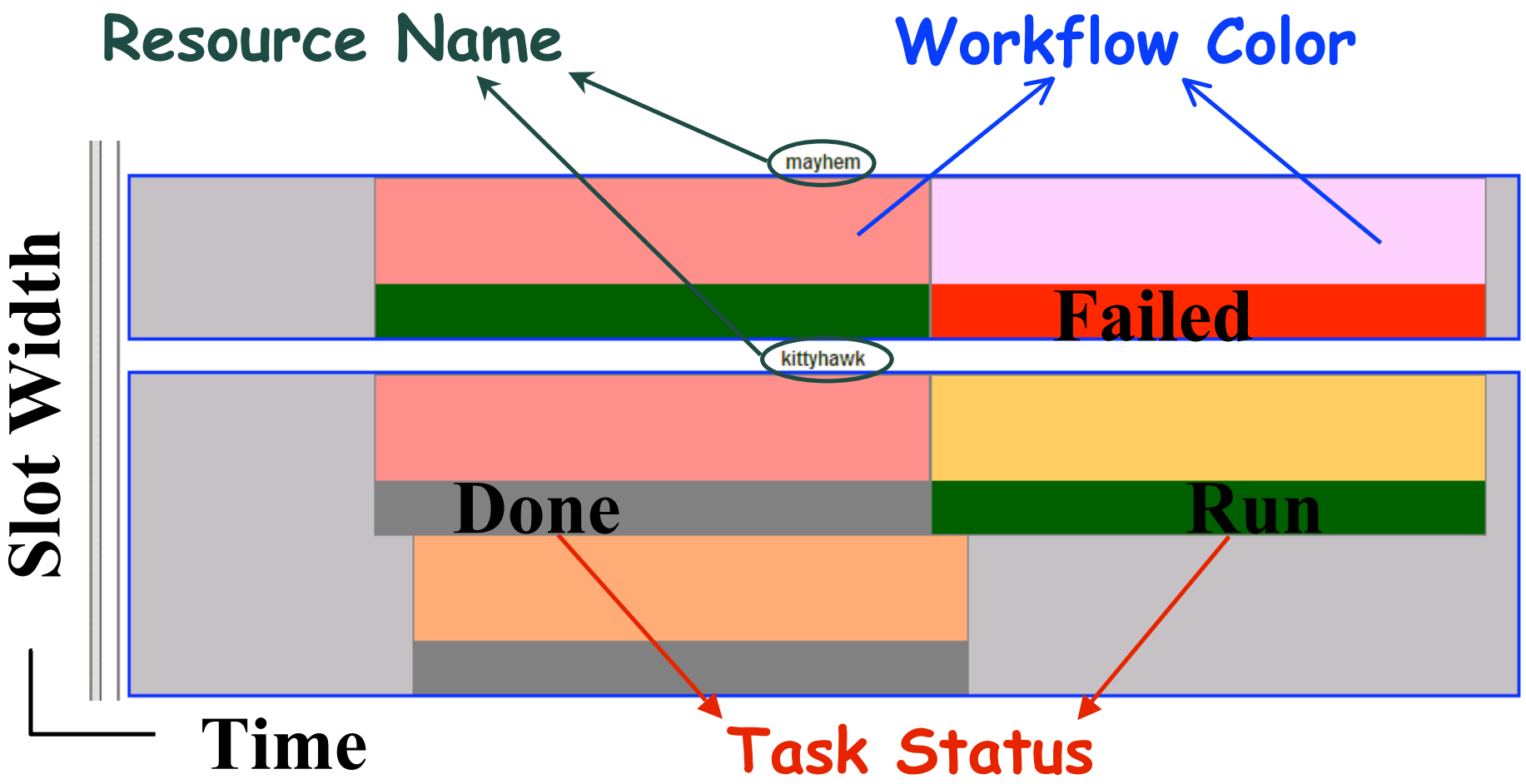
LEAD System Components

- Workflow Orchestration
  - Query vgES for resources
  - Phase 1: Schedule fraction F of workflows
  - Phase 2: Sort the scheduled tasks of fraction F by success probability. Compare scheduling from FTR and DAGscheduler on each DAG to make next orch. Decision
  - Phase 3: Try remaining FTR or remaining DAGs
  - Phase 4: For tasks below certain threshold, schedule on EC2



- Execution Manager
  - Early prototype for ordered execution of tasks on the slot based on the schedule determined by the orchestration.

Demo Vizualization Key



VGrADS at SC08

- Booth Talks and Demos
  - When and where
    - Tuesday, 11:30am, RENCI booth (2633)
    - Wednesday, noon, GCAS booth (285)
    - Wednesday, 2:00Pm, SDSC booth (568)
    - Wednesday, 4:00pm, RENCI booth (2633)
- Other talks
  - Paper: "Efficient Auction-based Grid Reservations using Dynamic Programming" by Andrew Mutz and Rich Wolski
    - Tuesday, 2:30-3:00pm, Ballroom G
  - Paper: "Analysis of Application Heartbeats: Learning Structural and Temporal Features in Time Series Data for Identification of Performance Problems" by Emma Buneci and Dan Reed
    - Thursday, 2:30-3:00pm, Ballroom E
  - Doctoral Research Showcase: "Paravirtualization Performance and Programming Support for Next Generation HPC System" by Lamia Youseff and Rich Wolski
    - Thursday, 11:30-11:45am, Room 17A/B

The VGrADS Team

THE UNIVERSITY of NORTH CAROLINA at CHAPEL HILL

Dan Reed

Anirban Mandal

Keith Cooper

Ken Kennedy

Charles Koelbel

Richard Tapia

Linda Torczon

Jack Dongarra

Carl Kesselman

Fran Berman

Andrew Chien

Henri Casanova

Rich Wolski

Lennart Johnsson

- **Plus many graduate students, postdocs, and technical staff!**

Thanks to the National Science Foundation

VGrADS

Virtual Grid Application Development Software Project

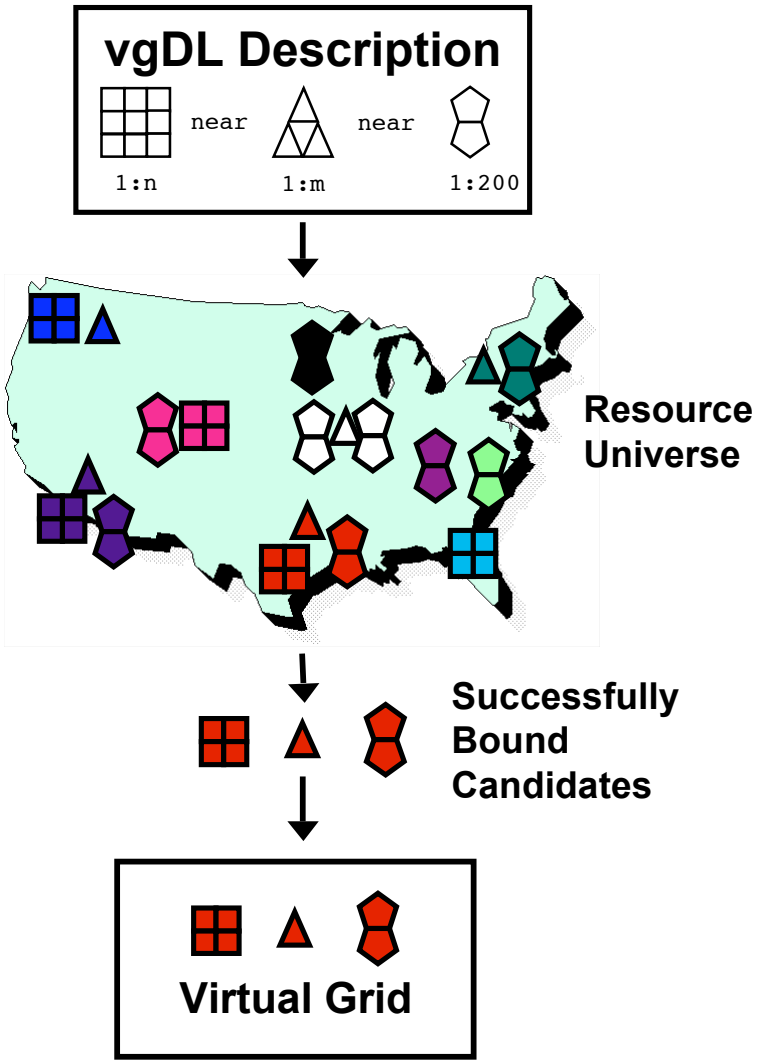
Backup slides

VGrADS Big Ideas

- Virtualization of Resources
  - Application specifies required resources in Virtual Grid Definition language (vgDL)
    - Give me a tight bag of as many Opterons as possible
  - Virtual Grid Execution System (vgES) produces specific virtual grid matching specification
    - May involve adding layer of capabilities
    - vgES also reports properties of the virtual grid to application
- Generic In-Advance Scheduling of Application Workflows
  - Application includes performance models for all workflow nodes
    - Performance models automatically constructed
  - Software schedules applications onto virtual Grid, minimizing total makespan
    - Including both computation and data movement times

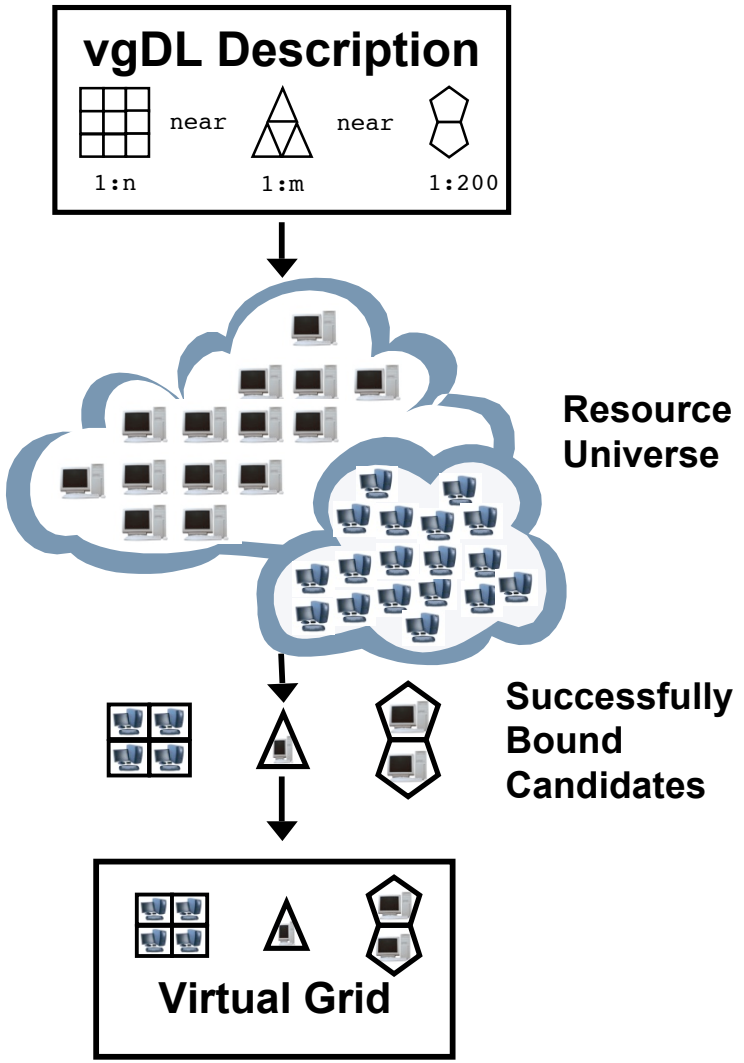
Virtual Grids (VGs)

- A Virtual Grid (VG) takes
  - Shared heterogeneous resources
  - Scalable information service
- and provides
  - A hierarchy of application-defined aggregations with constraints and rankings
  - E.g. Cluster of Opteron nodes
- Virtual Grid Execution System (vgES) implements VG
  - Brings order out of chaos
- Can be implemented on multiple platforms
  - Grids
  - Compute clouds
  - HPC
  - Manycore chips



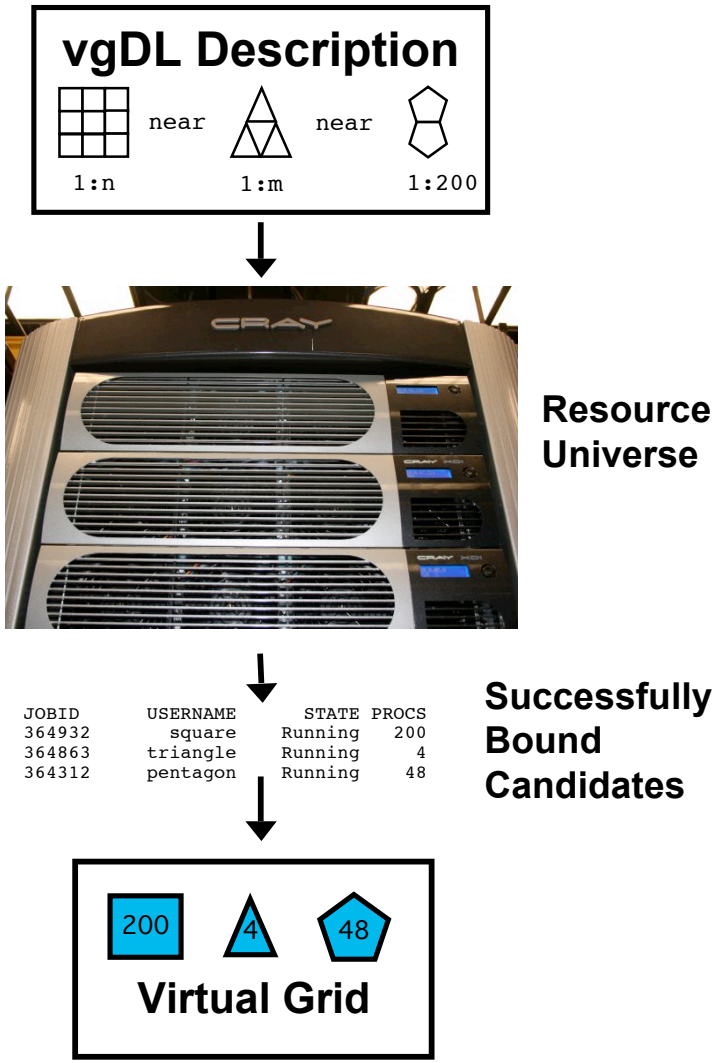
Virtual Grids (VGs)

- A Virtual Grid (VG) takes
  - Shared heterogeneous resources
  - Scalable information service
- and provides
  - A hierarchy of application-defined aggregations with constraints and rankings
  - E.g. Cluster of Opteron nodes
- Virtual Grid Execution System (vgES) implements VG
  - Brings order out of chaos
- Can be implemented on multiple platforms
  - Grids
  - Compute clouds
  - HPC
  - Manycore chips



Virtual Grids (VGs)

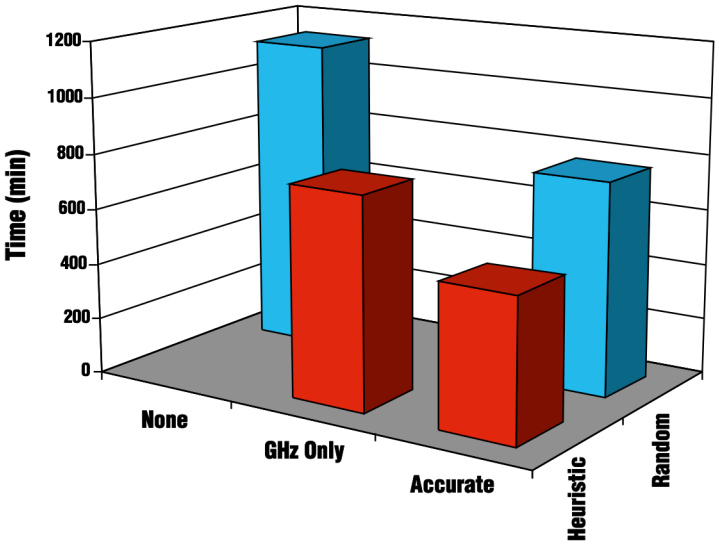
- A Virtual Grid (VG) takes
  - Shared heterogeneous resources
  - Scalable information service
- and provides
  - A hierarchy of application-defined aggregations with constraints and rankings
  - E.g. Cluster of Opteron nodes
- Virtual Grid Execution System (vgES) implements VG
  - Brings order out of chaos
- Can be implemented on multiple platforms
  - Grids
  - Compute clouds
  - HPC
  - Manycore chips



VGrADS
Virtual Grid Application Development Software Project

VGrADS Tool Research

- **Scheduling of workflow computations**
  - Off-line look-ahead scheduling dramatically improves total time
  - Accurate performance models significantly affect quality of scheduling
  - Batch queue behavior can be predicted accurately enough for scheduling decisions
- **Fault tolerance**
  - Diskless checkpointing for linear algebra computations (application-specific)
  - Qualitative performance analysis for fault detection
  - Optimal checkpoint frequency for iterative applications
  - Availability prediction



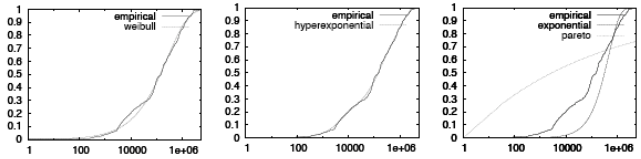


Fig. 1. CSIL data with Weibull fit

Fig. 2. CSIL data with hyperexponential fit

Fig. 3. CSIL data with exponential and Pareto fits

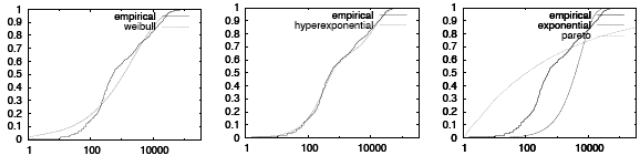


Fig. 4. Condor data with Wiebull fit

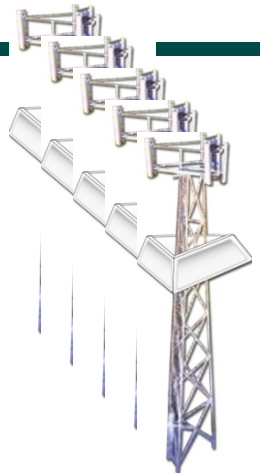
Fig. 5. Condor data with hyperexponential fit

Fig. 6. Condor data with exponential and Pareto fits

VGrADS: Application-driven Research

- SC04
  - Executing the EMAN application
    - Incorporated performance models
- SC05
  - Executing the EMAN application
    - Used batch queue predictions (and performance models)
- SC06
  - Executing the LEAD application
    - Scheduled to vgES virtualized resource reservation slots (leveraging queue predictions and performance models)
- SC07: *Fault tolerance through virtual grid abstractions*
  - Executing the LEAD application
    - Incorporates reliability information (and above features)
    - Combine replication and migration of workflow nodes

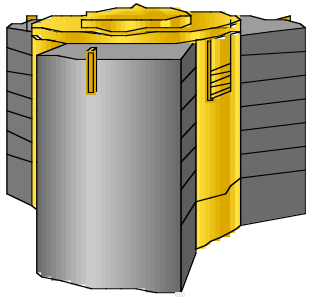
The LEAD Vision: A Paradigm Shift



Dynamic Observations

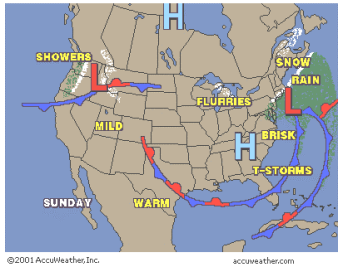
Analysis/Assimilation

Quality Control
Retrieval of Unobserved Quantities
Creation of Gridded Fields



Prediction/Detection

PCs to Teraflop Systems



Product Generation, Display, Dissemination

Models and Algorithms Driving Sensors

End Users

NWS
Private Companies
Students

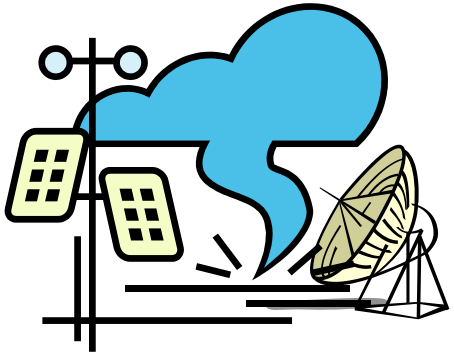
The CS challenge: Build cyberinfrastructure services that provide adaptability, scalability, availability, useability, and real-time response.

VGrADS
Virtual Grid Application Development Software Project

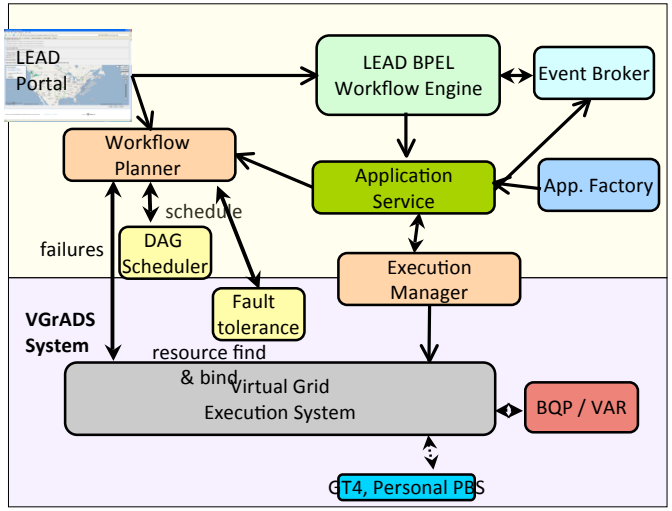
Dynamic Observations

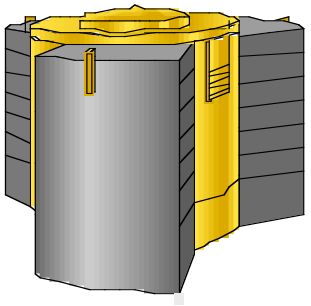
Provide data to

Data Analysis & Simulations



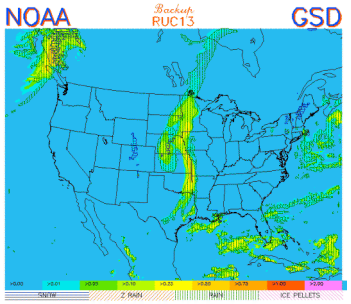
Coordinated by VGrADS





Steer collection of

Generate sets of



Predictions

6 Workflows on 7 Clusters