The Virtual Grid Application Development Software (VGrADS) Project

"VGrADS: Enabling e-Science Workflows on Grids and Clouds with Fault Tolerance"

http://vgrads.rice.edu/

VGrADS Goal: Distributed Problem Solving

- **Where We Want To Be**
  - o **Transparent computing**
    - – **In an increasingly distributed space**
    - – **Applications to HPC**
    - – **Applications to cloud computing**

- **Where We Were (circa 2003)**
  - o **Low-level hand programming**
  - o **Programmer had to manage:**
    - – **Heterogeneous resources**
    - – **Scheduling of computation and data movement**
    - – **Fault tolerance and performance adaptation**

- **What Progress Have We Made?**
  - o **Separate application development from resource management**
    - – **VGrADS provides a uniform "virtual grid" abstraction atop widely differing resources**
  - o **Provide tools to bridge the gap**
    - – **Scheduling, resource management, distributed launch, simple programming models, fault tolerance, grid economies**

VGrADS
Virtual Grid Application Development Software Project

Overview of SC'xx Activities for VGrADS

- **Built on previous SC 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

- **Current status**
  - 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!

VGrADS
Virtual Grid Application Development Software Project

VGrADS Components

o **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

o **Eucalyptus – *Developed for VGrADS (now Eucalyptus Inc.)***
  - Implements cloud computing on Xen-enabled clusters
  - Open-source software infrastructure that is compatible with Amazon EC2
  - → vgES "thinks" a Eucalyptus cloud is EC2

o **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

Executing LEAD Workflow Sets

o **Demonstrate planning and execution of LEAD workflow sets execution atop virtualized cloud and Grid resources.**

o **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)**

o **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

Workflow Orchestration

o **Workflow Orchestration**

　o **Phase 1:** *Minimal Scheduling* – **Schedule minimum fraction of workflows using simple probabilistic DAG scheduler**

　o **Phase 2:** *Fault Tolerance Tradeoff* – **Compare scheduling additional workflows with increasing fault-tolerance of one or more tasks of the scheduled workflows**

　o **Phase 3:** *Additional Scheduling* – **Use available slots for other scheduling**

　o **Phase 4:** *EC2 Scheduling* - **For tasks below certain threshold, schedule on EC2**

o **Execution Manager**

　– **A prototype for ordered execution of tasks on the slot based on the schedule determined by the orchestration.**

Comparison of the LEAD-VGrADS collaboration system with cyberinfrastructure production deployments
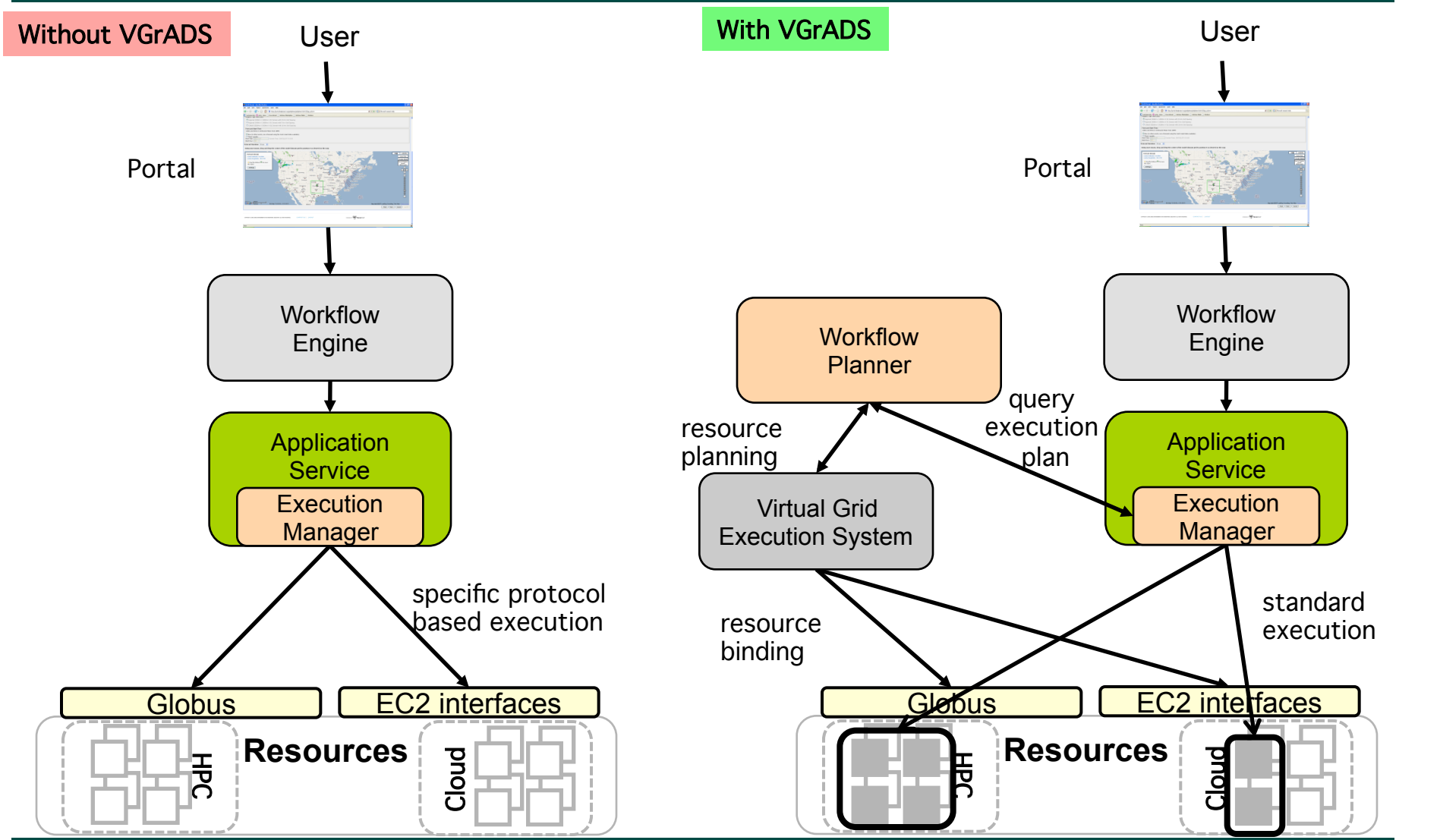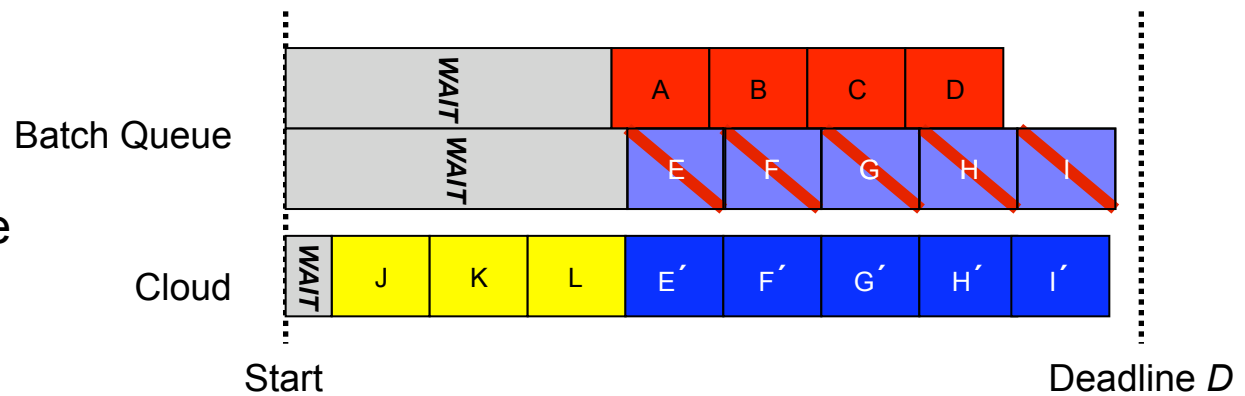
Example Scheduling of Workflows
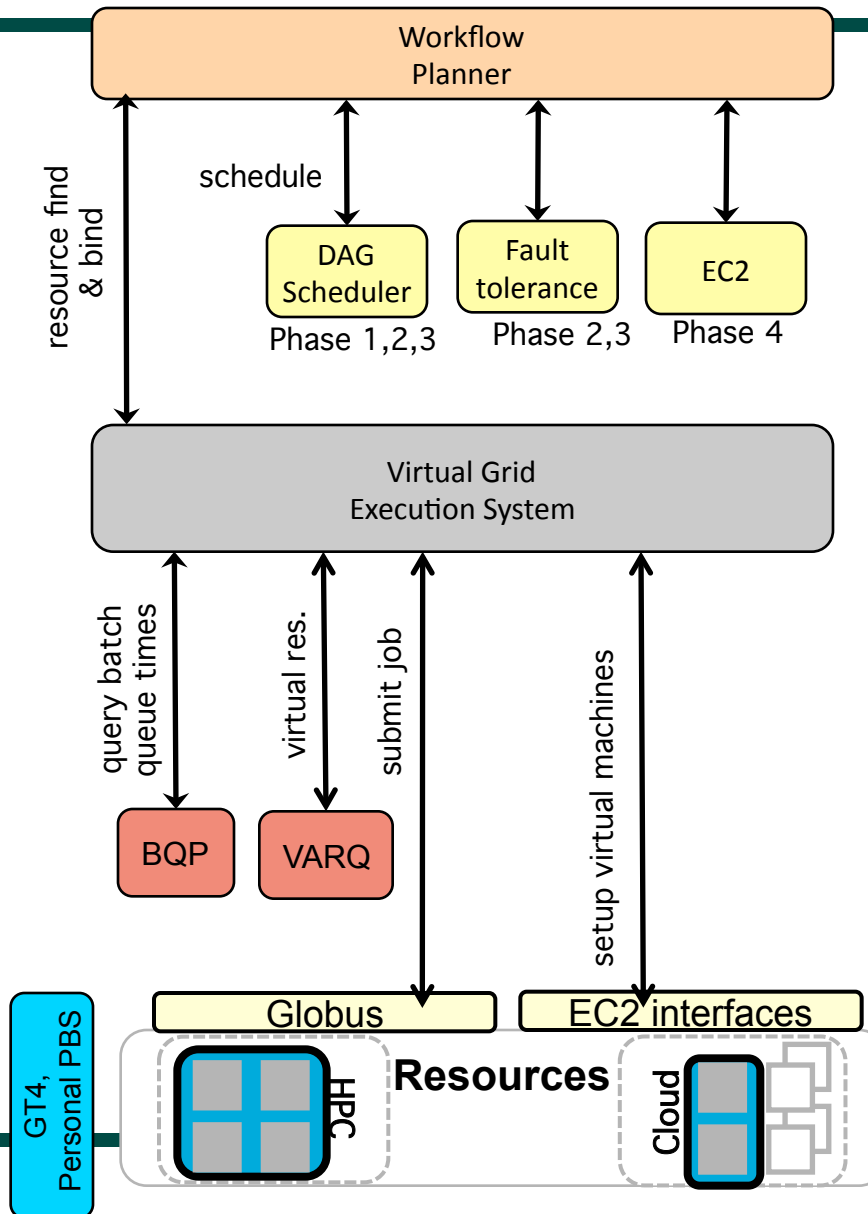

(a)
Example Workflow Set
need *F*=2/3
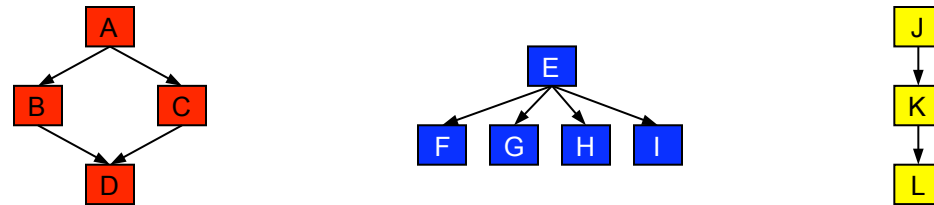
(b)
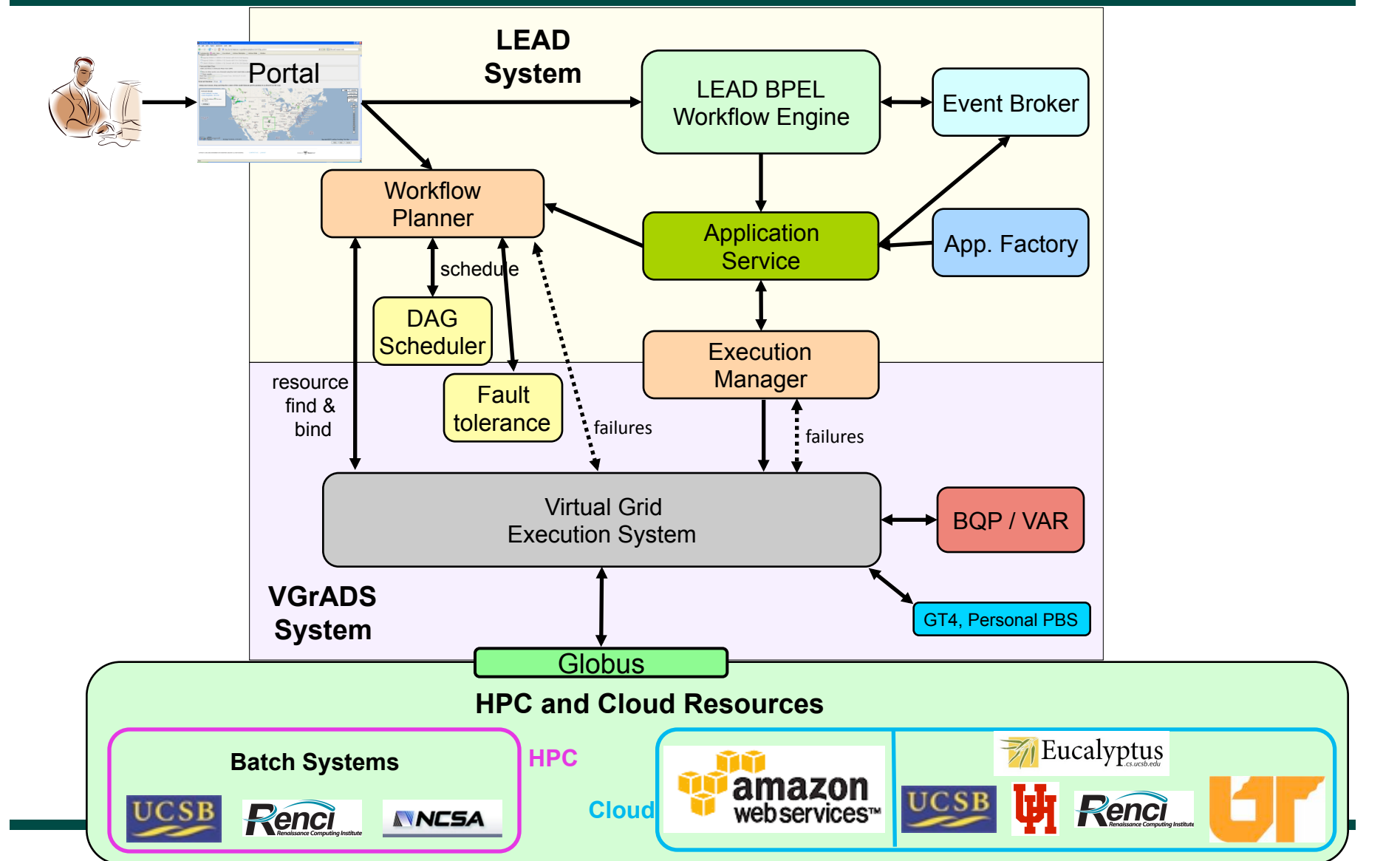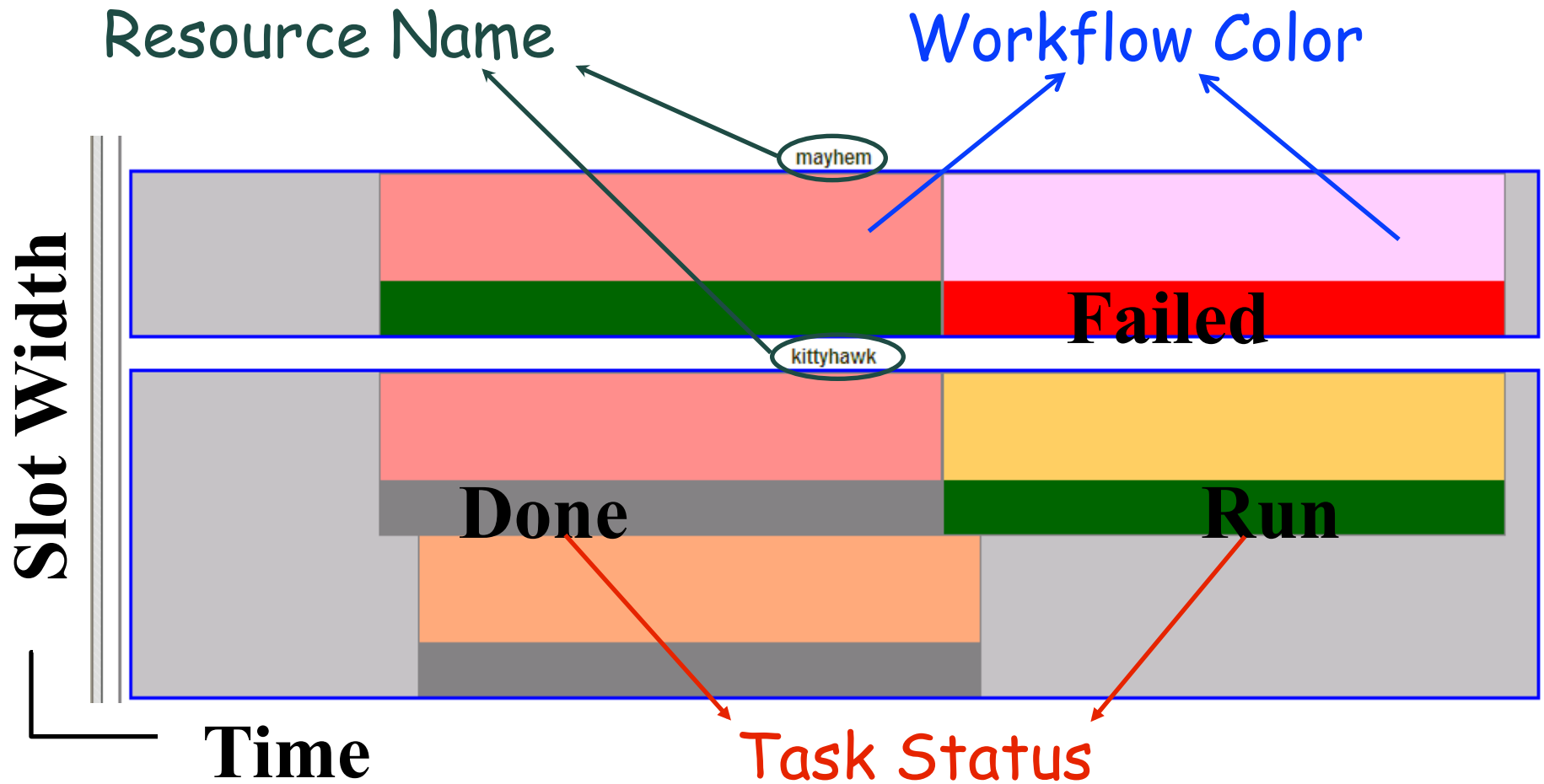Uncoordinated Schedule
Without VGrADS
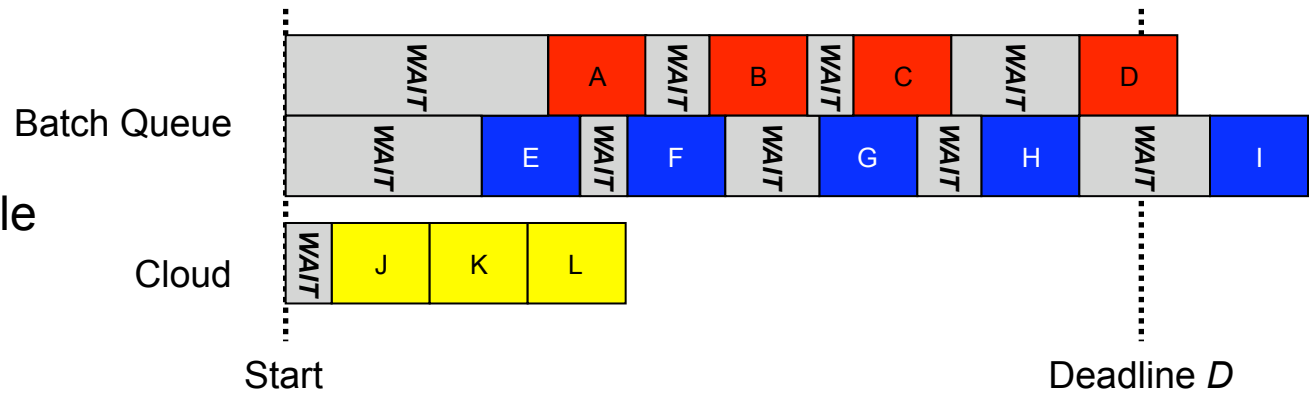
(c)
Coordinated Schedule
With VGrADS

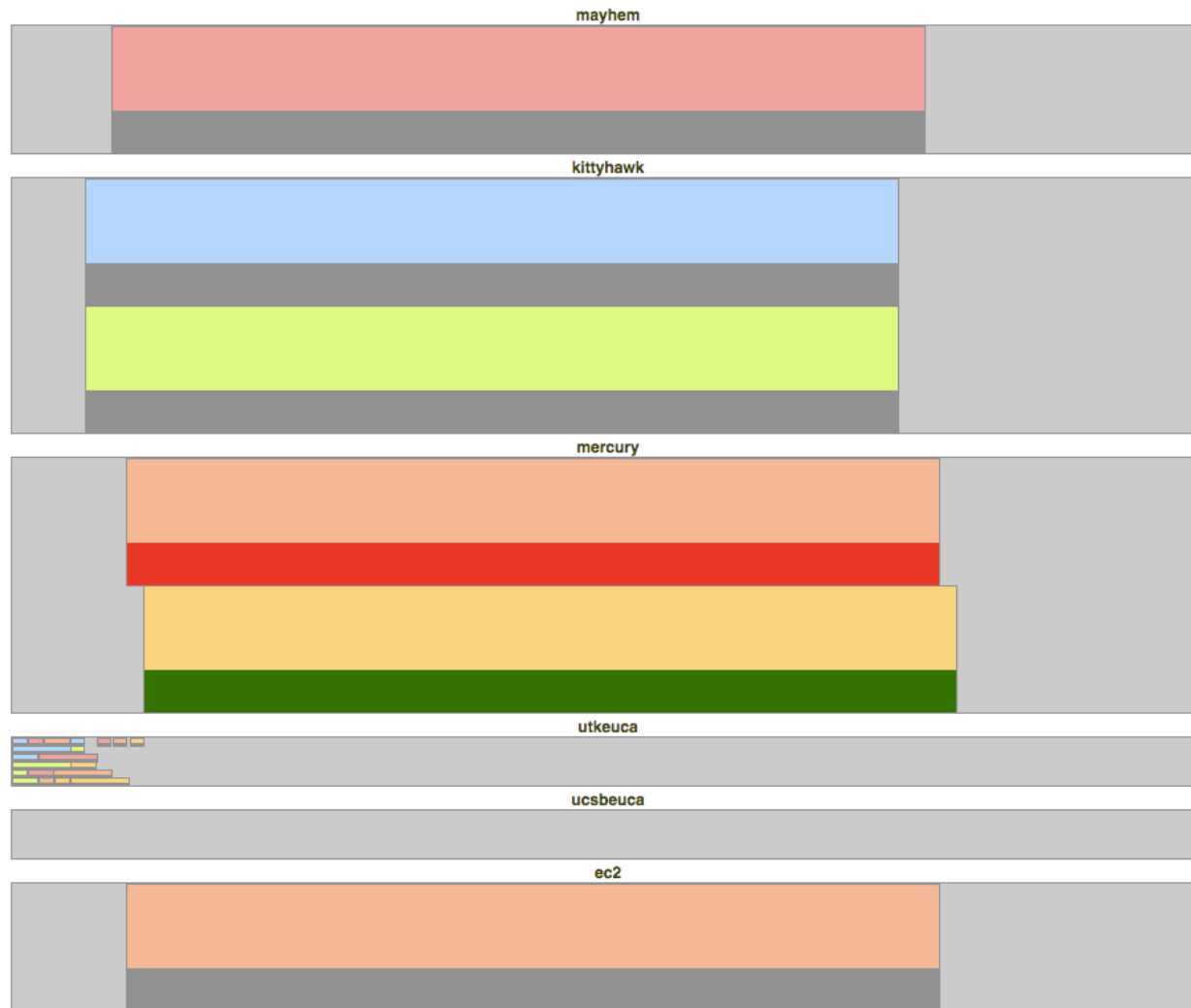# Interaction of system components for resource procurement and planning

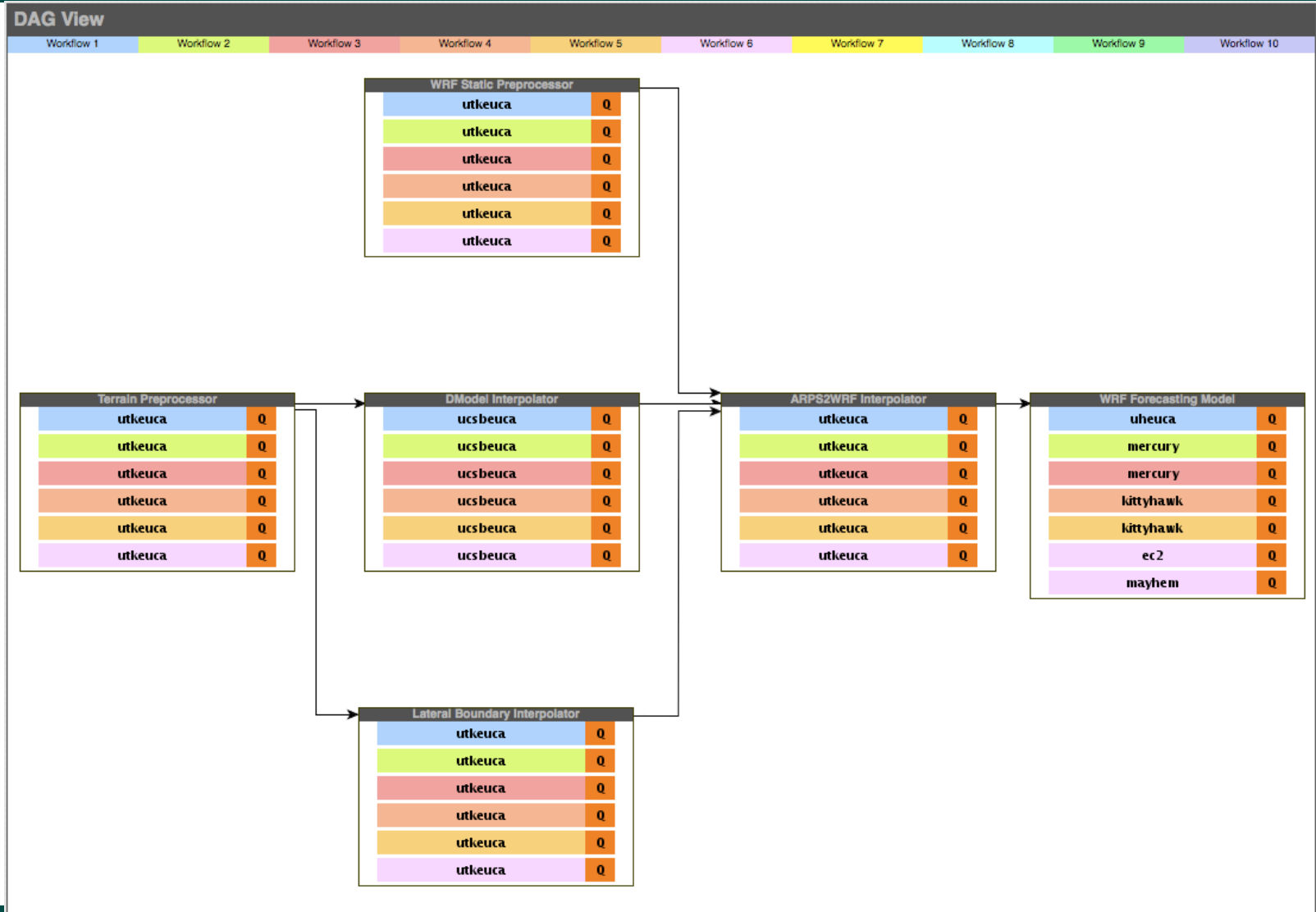# LEAD / VGrADS Architecture : Putting It All Together

LEAD System

Portal

LEAD BPEL Workflow Engine

Event Broker

Workflow Planner

Application Service

App. Factory

schedule

DAG Scheduler

Execution Manager

Fault tolerance

resource find & bind

failures

failures

VGrADS System

Virtual Grid Execution System

BQP / VAR

GT4, Personal PBS

Globus

HPC and Cloud Resources

Batch Systems

UCSB   Renci Renaissance Computing Institute   NCSA

HPC

Cloud

amazon web services™

Eucalyptus cs.ucsb.edu

UCSB   UH   Renci Renaissance Computing Institute   UT

Virtual Grid Application Development Software Project
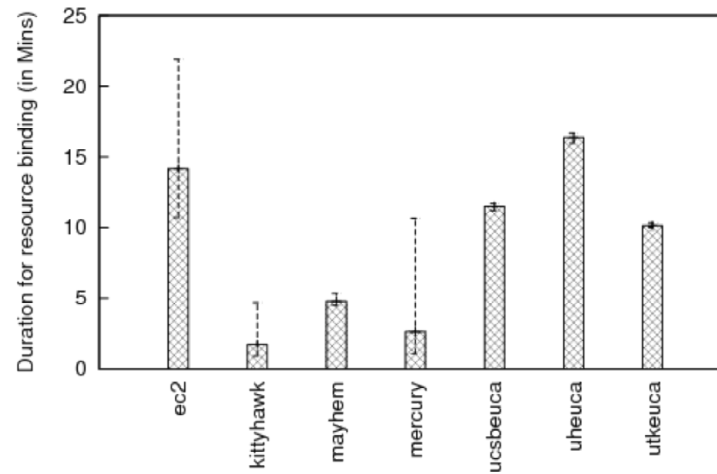
Visualization Key

Snapshot of Execution of 6 Workflows

6 Workflows on 7 Clusters

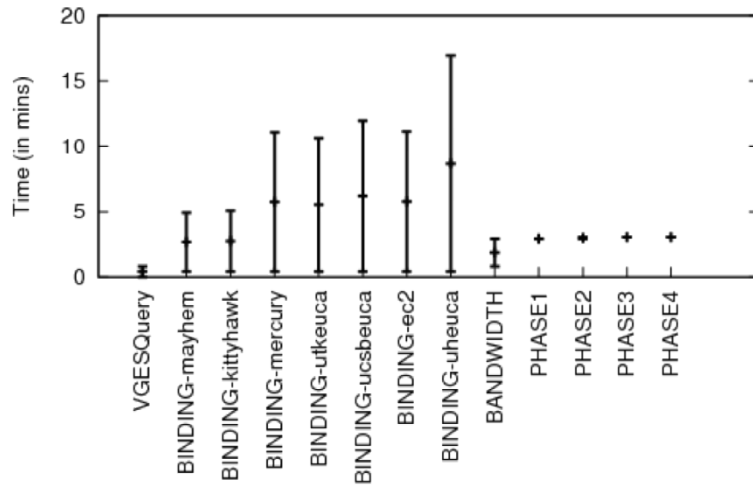# Infrastructure Timing Metrics

Binding time

Planning time line
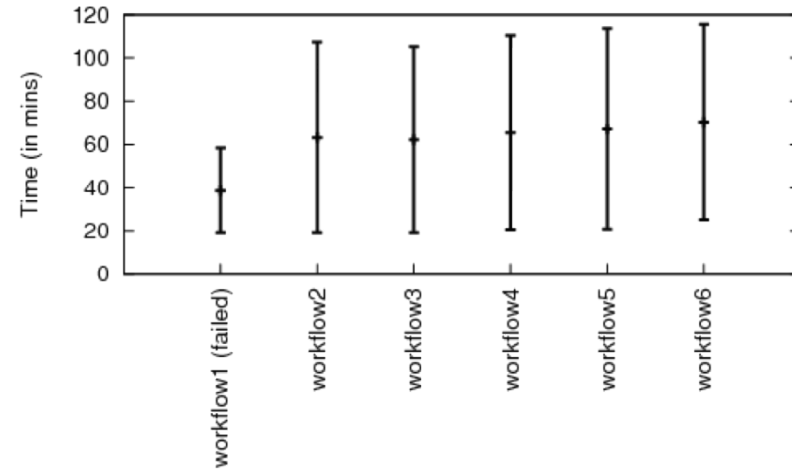
Workflow execution time line

Fault Tolerance Exploration
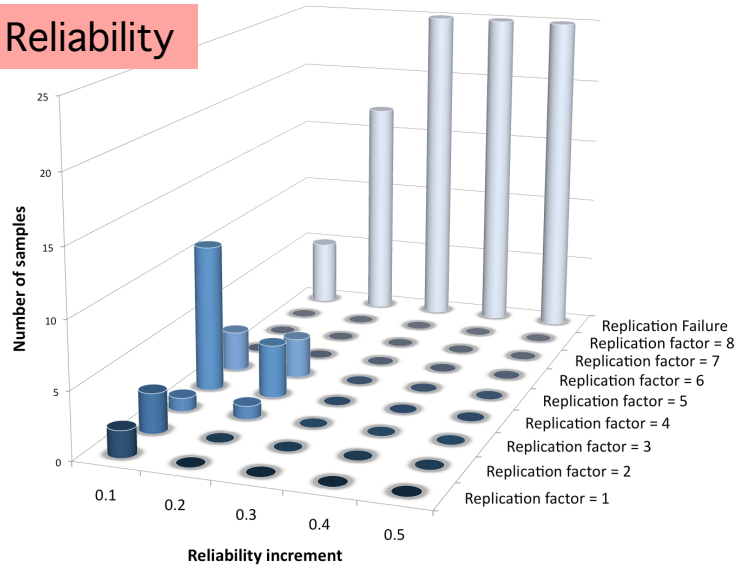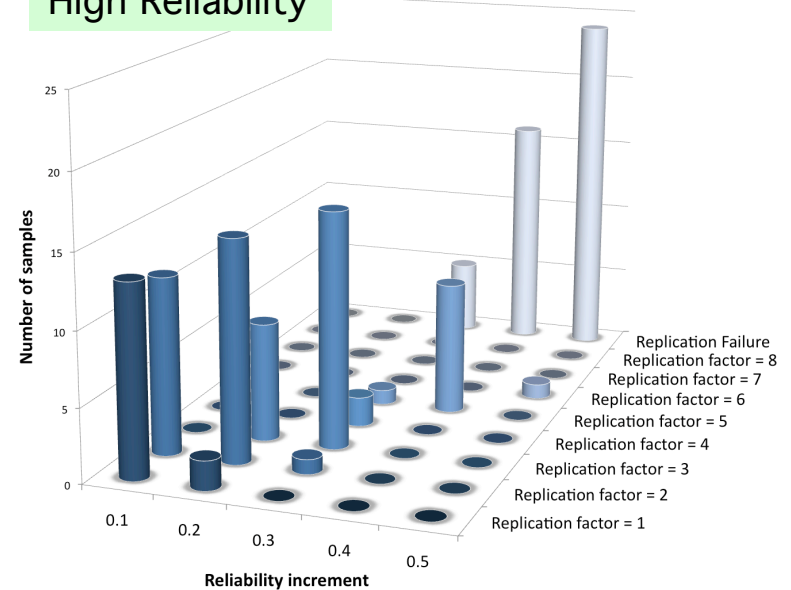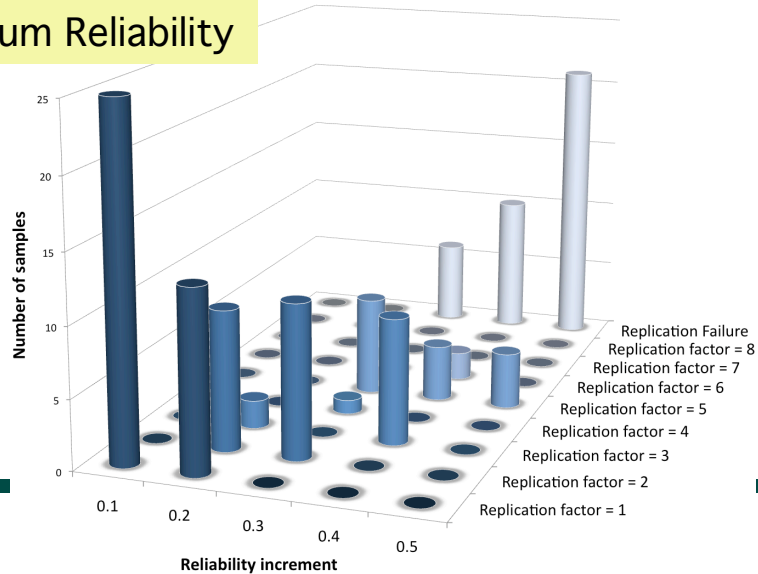
Conclusions

- VGrADS' virtual grid abstraction simplifies
  - Programming grid and cloud systems for e-Science workflows
  - Managing QoS (performance and reliability)
- The VGrADS system unifies workflow execution over
  - batch queue systems (with and without advanced reservations), and
  - cloud computing sites (including Amazon EC2 and Eucalyptus)
- The system provides an enabling technology for executing deadline-driven, fault-tolerant workflows
- The integrated cyber-infrastructure from the LEAD and VGrADS system components provides a strong foundation for next-generation dynamic and adaptive environments for scientific workflows

Thank You

- "VGrADS: Enabling e-Science Workflows on Grids and Clouds with Fault Tolerance", L. Ramakrishnan, D. Nurmi, A. Mandal, C. Koelbel, D. Gannon, T. M. Huang, Y. S. Kee, G. Obertelli, K. Thyagaraja, R. Wolski, A. Yarkhan and D. Zagorodnov

- Paper to be presented at 3:30 p.m., Thursday, Nov. 19 in Room E145 - 146

Thank you..