Global Grid Forum 14, Grid Applications Workshop:
From Early Adopters to Mainstream Users

Experiences from Simulating the Global Carbon Cycle in a Grid Computing Environment

Presenter: Jason Cope
PhD Student
University of Colorado, Boulder
Jason.Cope@colorado.edu

NCAR

Motivation: NCAR as an Integrator

- It is our position that NCAR must provide integrated solutions to the community.
- Scientific workflows are becoming too complicated for manual (or semi-manual) implementation.
- Not reasonable to expect a scientist to:
  - Design simulation solutions by chaining together application software packages
  - Manage the data lifecycle (check out, analysis, publishing, and check in)
  - Do this in an evolving computational and information environment
- NCAR must provide the software infrastructure to allow scientist to seamlessly (and painlessly) implement their workflows, thereby allowing them to concentrate on what they're good at: SCIENCE!
- Goal is increased scientific productivity and requires an unprecedented level of integration of **both systems and software**.
- Long term investment: return to the organization won't show up in the bottom line immediately.

Motivation: Robust Modeling Environments

- Our goal is to develop a simple, production quality modeling environment for NCAR and the geoscience community that insulates scientists from the technical details of the execution environment
  - Cyberinfrastructure
  - System and software integration
  - Data archiving
- Grid-BGC is an example of such an environment and is the first of these environments developed for NCAR
  - Learning as we develop and deploy
  - Tasked by the geoscience community, but developed services are applicable to other collaborative research projects

Outline

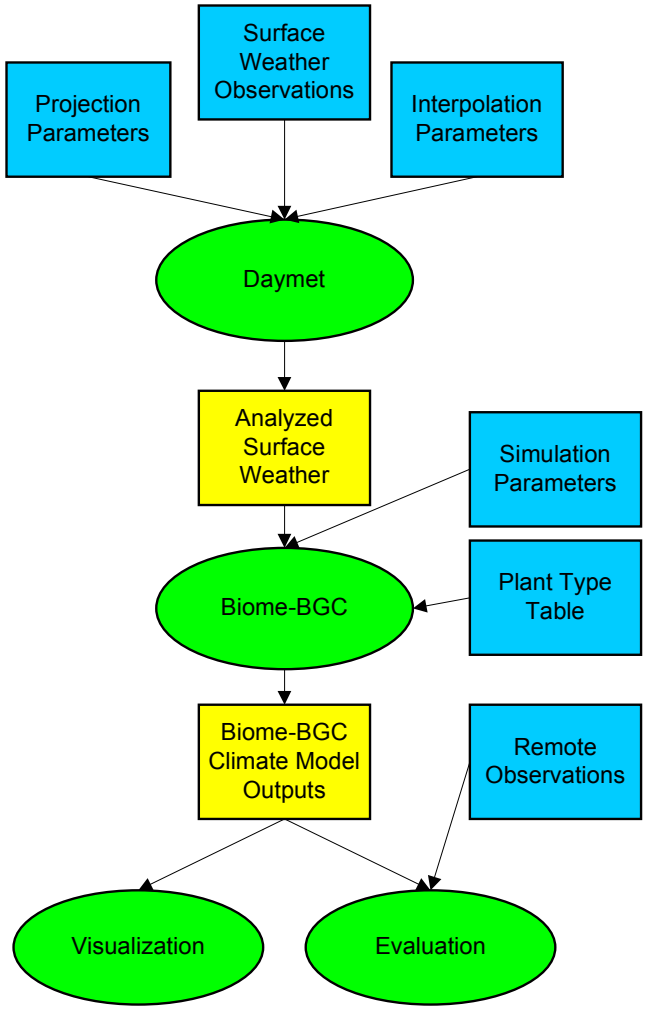
- Introduction
- Carbon Cycle Modeling
- Grid-BGC Prototype Architecture
- Experiences with the Grid
- Grid-BGC Production Architecture
- Future Work

Introduction: Participants

- This is a collaborative project between the National Center for Atmospheric Research (NCAR) and the University of Colorado at Boulder (CU)
- NASA has provided funding for three years via the Advanced Information Systems Technology (AIST) program
- Researchers:
  - Peter Thornton (PI), NCAR
  - Henry Tufo (co-PI), CU
  - Luca Cinquini, NCAR
  - Jason Cope, CU
  - Craig Hartsough, NCAR
  - Rich Loft, NCAR
  - Sean McCreary, CU
  - Don Middleton, NCAR
  - Nate Wilhelmi, NCAR
  - Matthew Woitaszek, CU

Carbon Cycle Modeling: Workflow

- Scientific models
  - Daymet
  - Biome-BGC
- Daymet interpolates a high resolution grid of weather observations for a region
- Biome BGC calculates carbon cycle parameters at the individual grid points for each region
- Models originally intended for analysis of small geographic regions.
- Analysis of larger regions is accomplished by simulating its composite regions



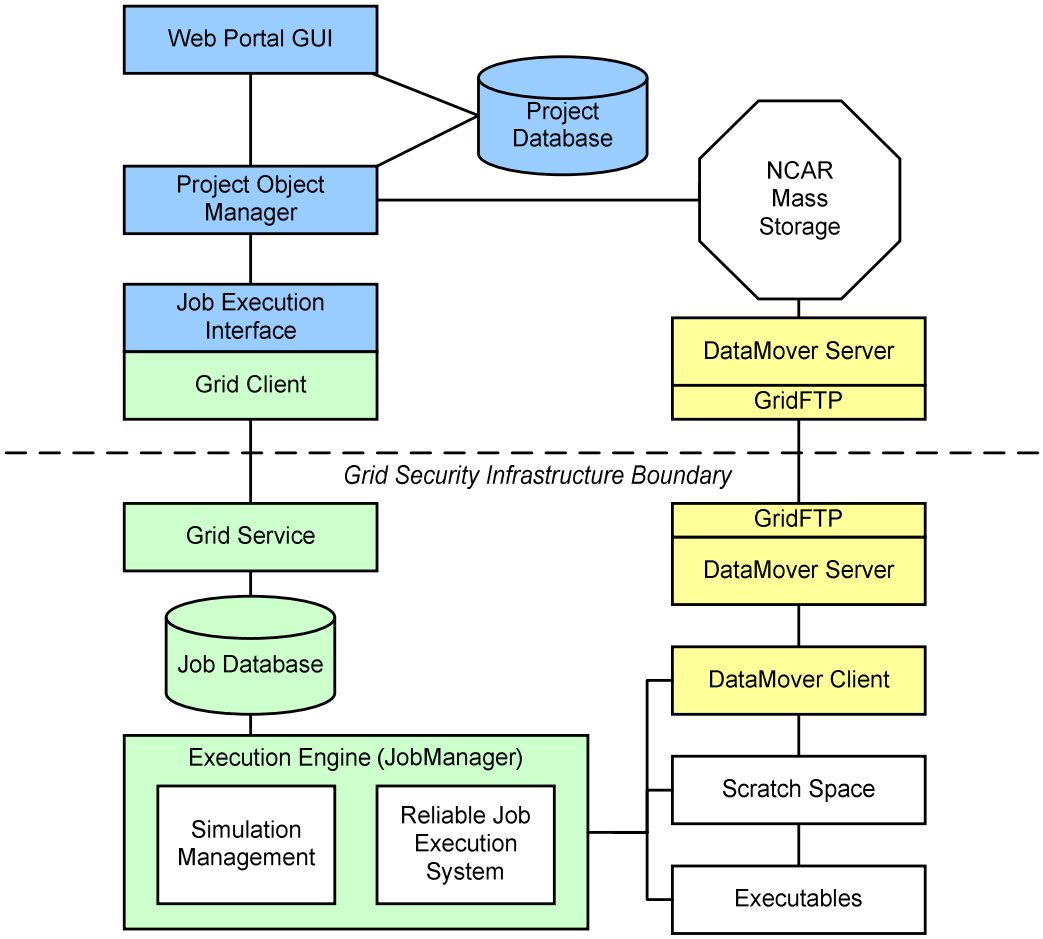
Carbon Cycle Modeling: Grid-BGC Motivation

Goal: Create an easy to use computational environment for scientists running large scale carbon cycle simulations.

- Requires the management of multiple simultaneously executing workflows
  - Execution management
  - Data management
  - Task automation
- Distributed resources across multiple organizations
  - Data archive and front-end portal are located at NCAR
  - Execution resources are located at CU

Grid-BGC Prototype Architecture (June, 2004): Overview

- Main components
  - Web portal GUI
  - JobManager
  - DataMover
- Prototype goals
  - Create a computational grid from GT 3.2
  - Create a reliable execution environment
  - Simplify usage and management



Experiences

- Security
  - Security breach during Spring 2004 impacted Grid-BGC's design
  - Continually evolving requirement and component
    - One time password authentication
    - Short lived certificates
  - Unique requirements may make implementation difficult
- Reliability and Fault Tolerance
  - Grid-BGC is self healing
  - Detect and correct failures without scientist intervention
  - Hold uncorrectable errors for administrative action

Experiences: Grid Development

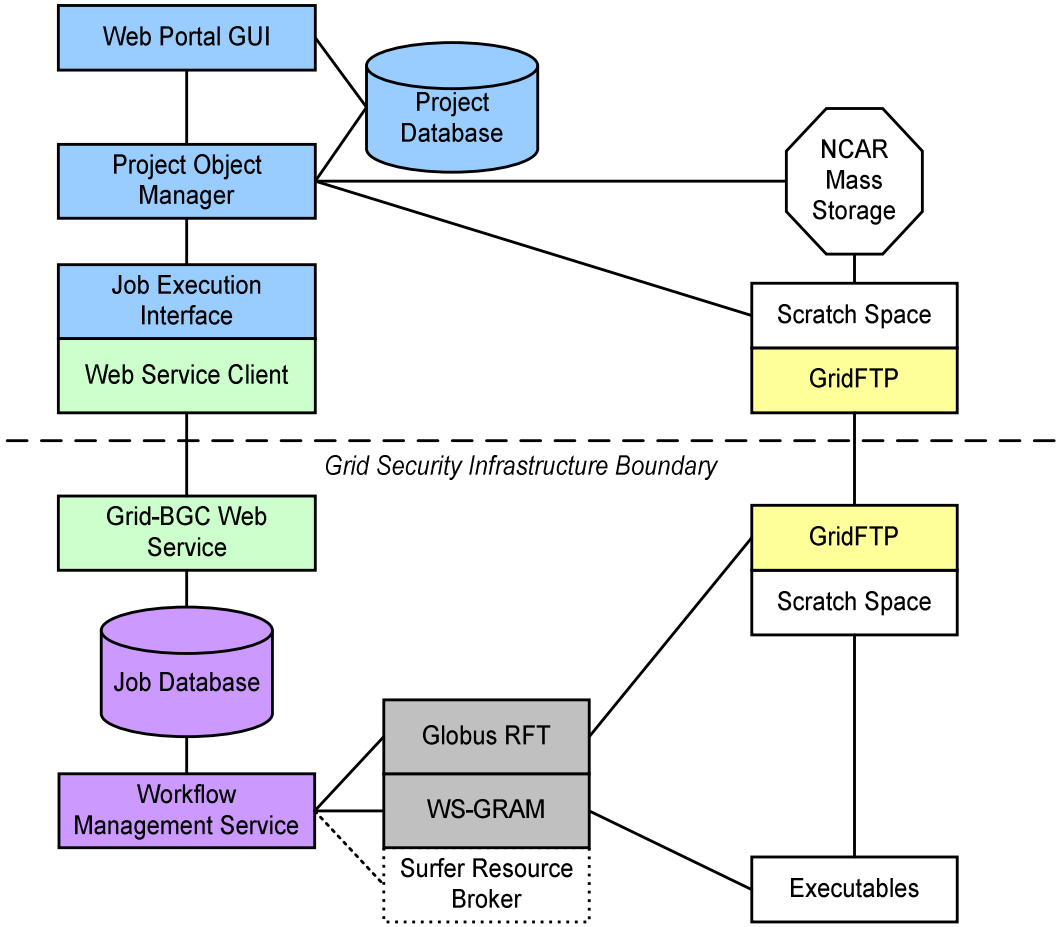
- Grid-BGC is the first production quality computational grid developed by NCAR and CU.
- Prototype development began January 2004 and ended September 2004
- More difficult than we anticipated
  - Pleasant persistence in the face of frustration is an essential quality for developers of a grid computing environment
    - We found that the middleware and development tools for GT 3.2 were not "production grade" for our environment, but were rapidly improving
    - Distributed, heterogeneous, and asynchronous system development and debugging are difficult tasks
  - Developers' mileage with grid middleware will vary
    - Globus components may or may not be useful
    - While critical components are available, more specific components will certainly need development

Experiences: Analysis of the Prototype

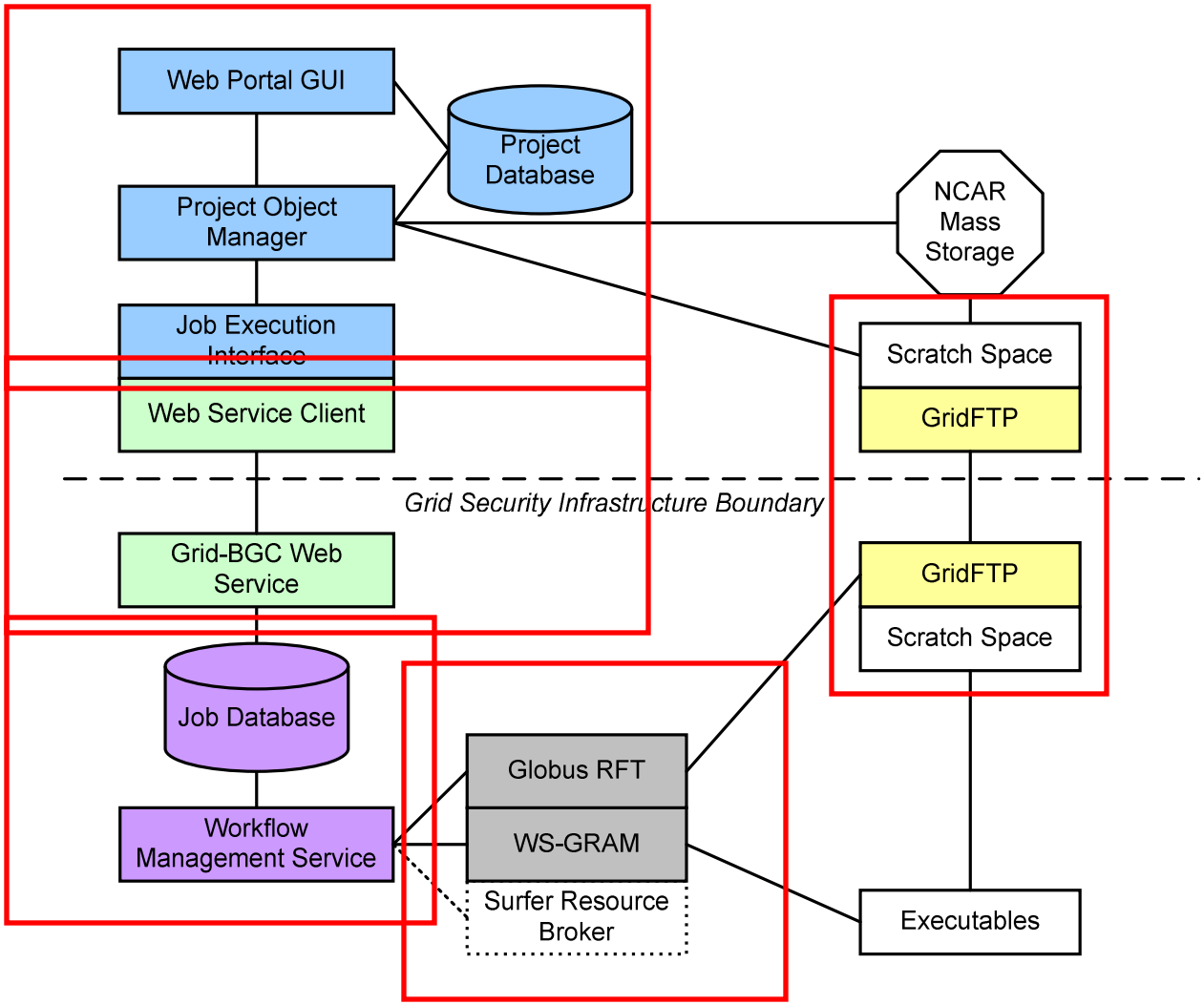
- What we did well
  - Proof of concept grid environment was a success
  - Portal, grid services, and automation tools create a simplified user environment
  - Fault tolerant
- What we needed to improve
  - Modularize the monolithic architecture
    - Break out functionality
    - Move towards a service oriented architecture
  - Re-evaluate data management policies and tools
    - GridFTP / Replica Location Service (RLS) vs. DataMover
    - Misuse of NCAR MSS as temporary storage platform
  - Use the appropriate Globus compliant tools
    - Many Globus components can be used in place **third-party** or **in-house** tools
    - More recent release of GT have improved the quality and usability of the components and documentation

Grid-BGC Production Architecture (June, 2005): Overview

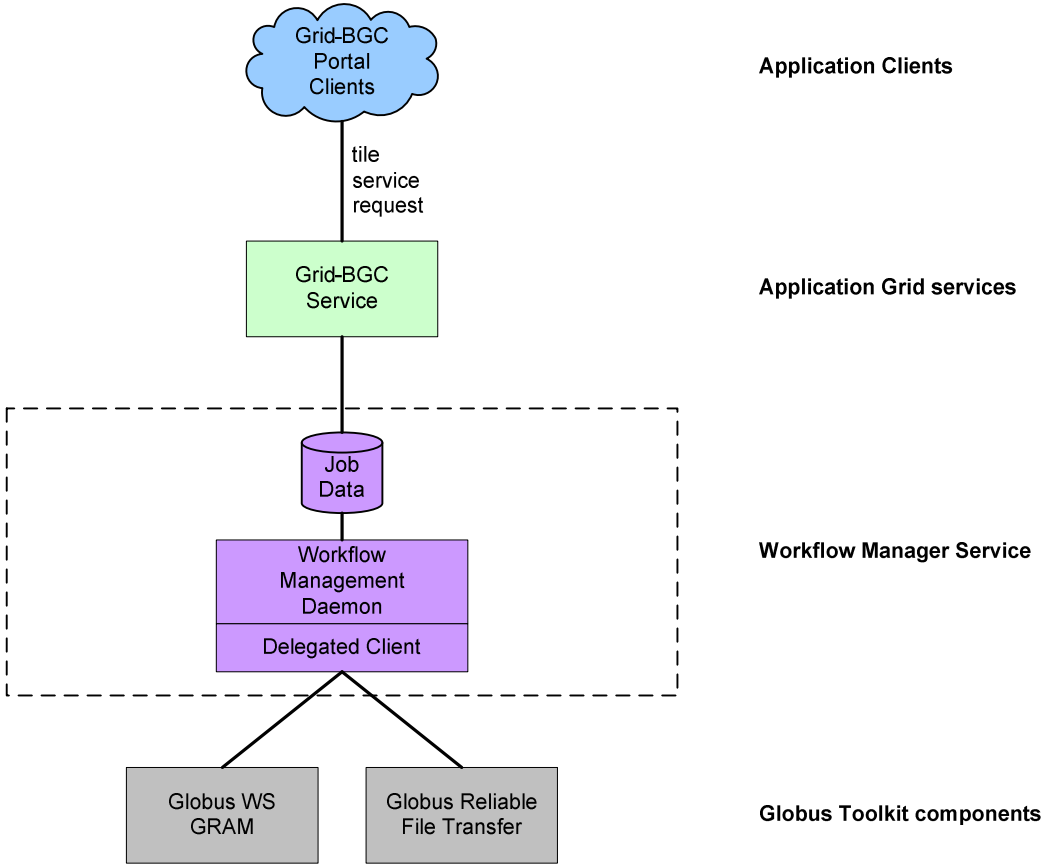
- Production architecture address prototypes goals
  - Ease of use
  - Efficient and productive
- Additionally, production architecture addresses
  - Modular design
  - GT4 Compliance
  - Restructuring execution and data management



Grid-BGC Production Architecture (June, 2005): Overview



Grid-BGC Production Architecture



Grid-BGC Production Architecture: Improvements

- Usage of more Globus Toolkit components and services
  - WS-GRAM
  - GridFTP and Reliable File Transfer (RFT) service
- Restructuring data management
  - Remove of DataMover, replace with RFT
  - Use Replica Location Service (RLS) as needed
- Restructuring execution management
  - Removal of JobManager, replace with WS GRAM and a custom client
  - Develop a simple workflow manager
- Modularize components
  - Break out individual components from the monolithic kernel
  - Create a service oriented architecture
  - Separate functional components
  - Generic and reusable

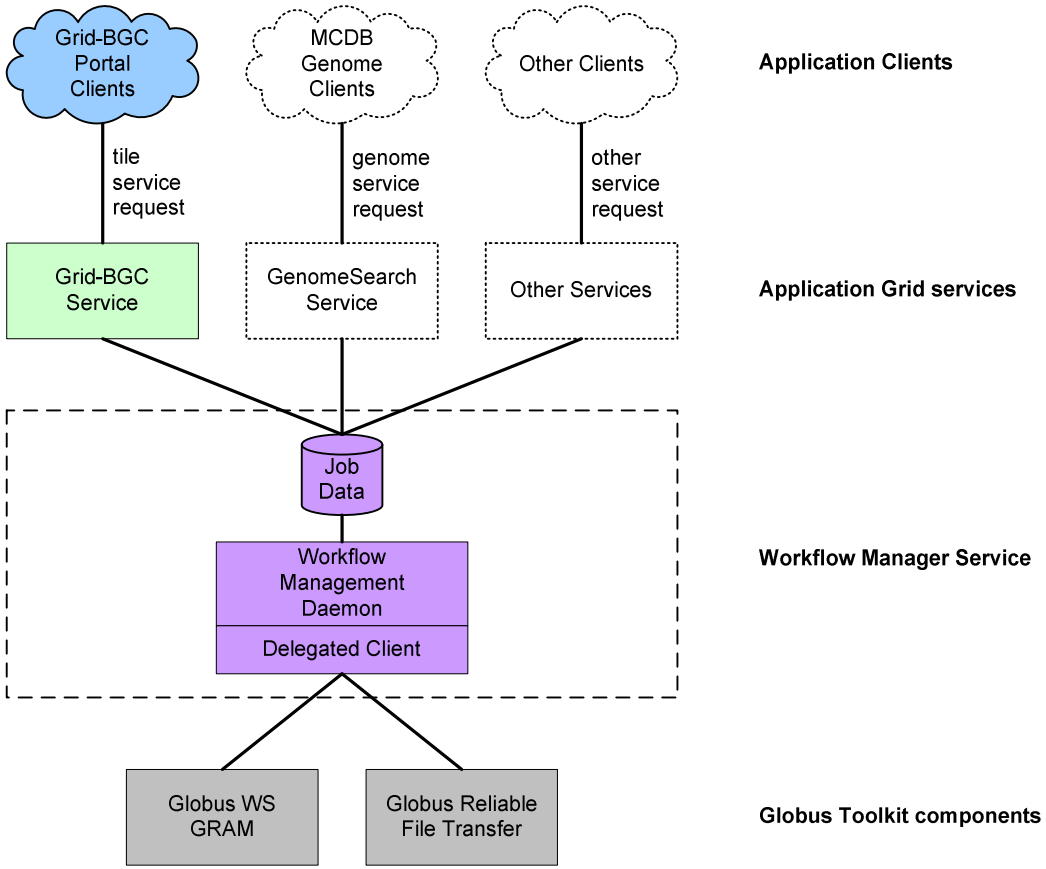
Grid-BGC Production Architecture: GT4 Experience

- Improvements in Grid middleware immediately useful in our environment
  - MyProxy officially supported
  - WS GRAM meets our expectations
  - RFT and GridFTP
- Porting Grid-BGC prototype was easier than expected
  - Modular design limited the amount of changes needed
  - Improved documentation of GT components

Future Work: Expansion of the Grid-BGC Environment

- Integrate NASA's Columbia Supercomputer into the Grid-BGC environment
  - Late Summer 2005
  - Deploy the computational framework and components
  - End-to-end testing between distributed Grid-BGC components
- Integrate resources provided by the system's users

Future Work: CU / NCAR HTWM



Experiences from Simulating the Global Carbon Cycle in a Grid Computing Environment

This research was supported in part by the National Aeronautics and Space Administration (NASA) under AIST Grant AIST-02-0036, the National Science Foundation (NSF) under ARI Grant #CDA-9601817, and NSF sponsorship of the National Center for Atmospheric Research.

Questions?
Ideas? Comments?
Suggestions?
http://www.gridbgc.ucar.edu

Presenter's email:
Jason.Cope@colorado.edu

NCAR