

# Air Quality Forecasting in Campus Grid Environment

Yonghong Yan, Barbara M. Chapman, Babu Sundaram  
University of Houston

GGF14  
June 27<sup>th</sup> 2005

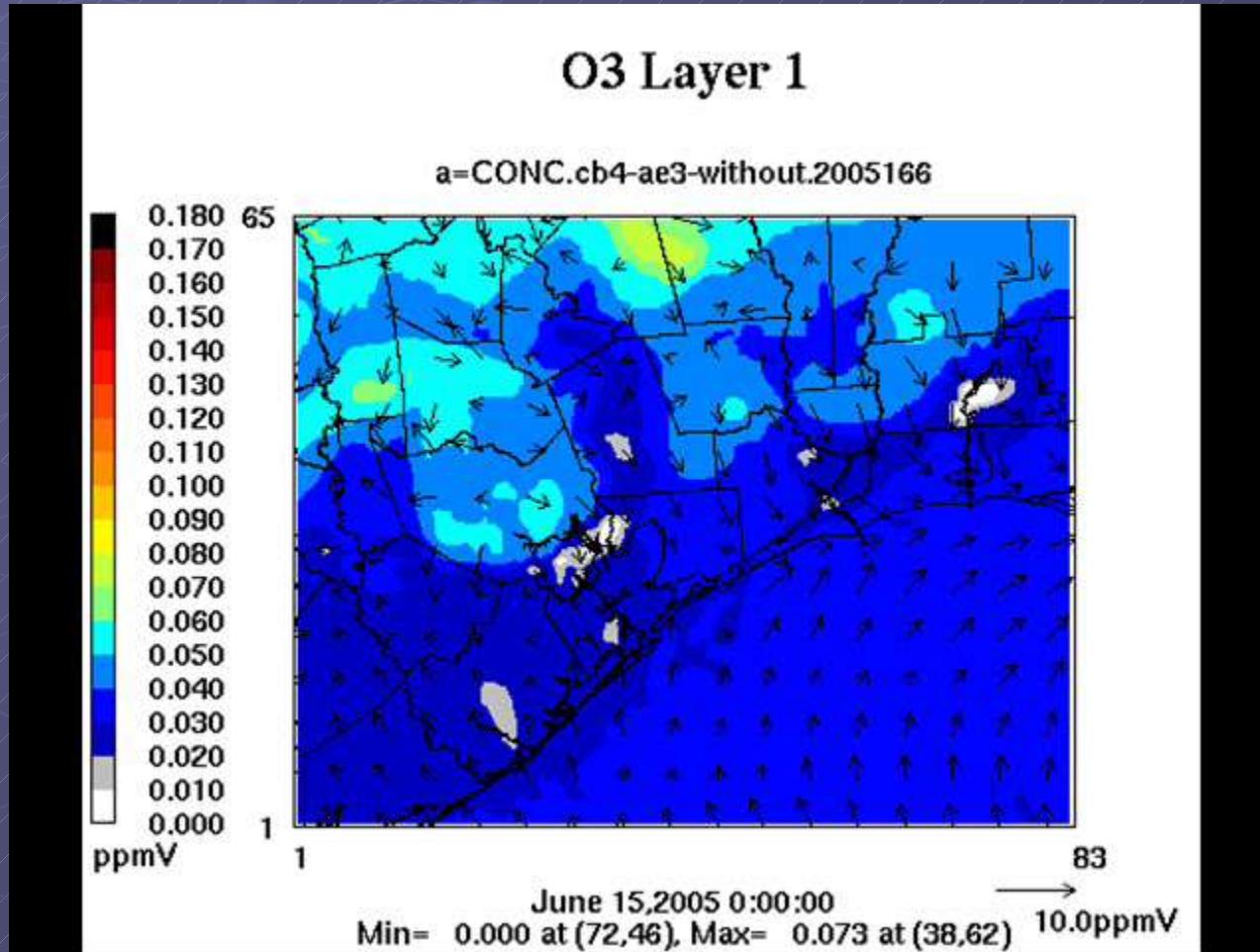
# Contents

- Introduction of AQF application
- AQF on UH campus grid
- Gracce for AQF-like applications
  - GAMDL
  - Metascheduler architecture
- Experience sharing

# Air Quality Forecasting (AQF)

- Attempts to provide a basis for improving air quality by reliably predicting atmospheric pollution, especially high levels of ozone
- Our work in UH addresses important air pollution concerns of Texas

# A Regional 2-day Forecasting



# AQF Execution

- Complex Workflow

- MM5-MCIP/SMOKE-CMAQ
- Increased resolution (36km-12km-4km) with decreasing geographical boundaries (US-TX-HGA)

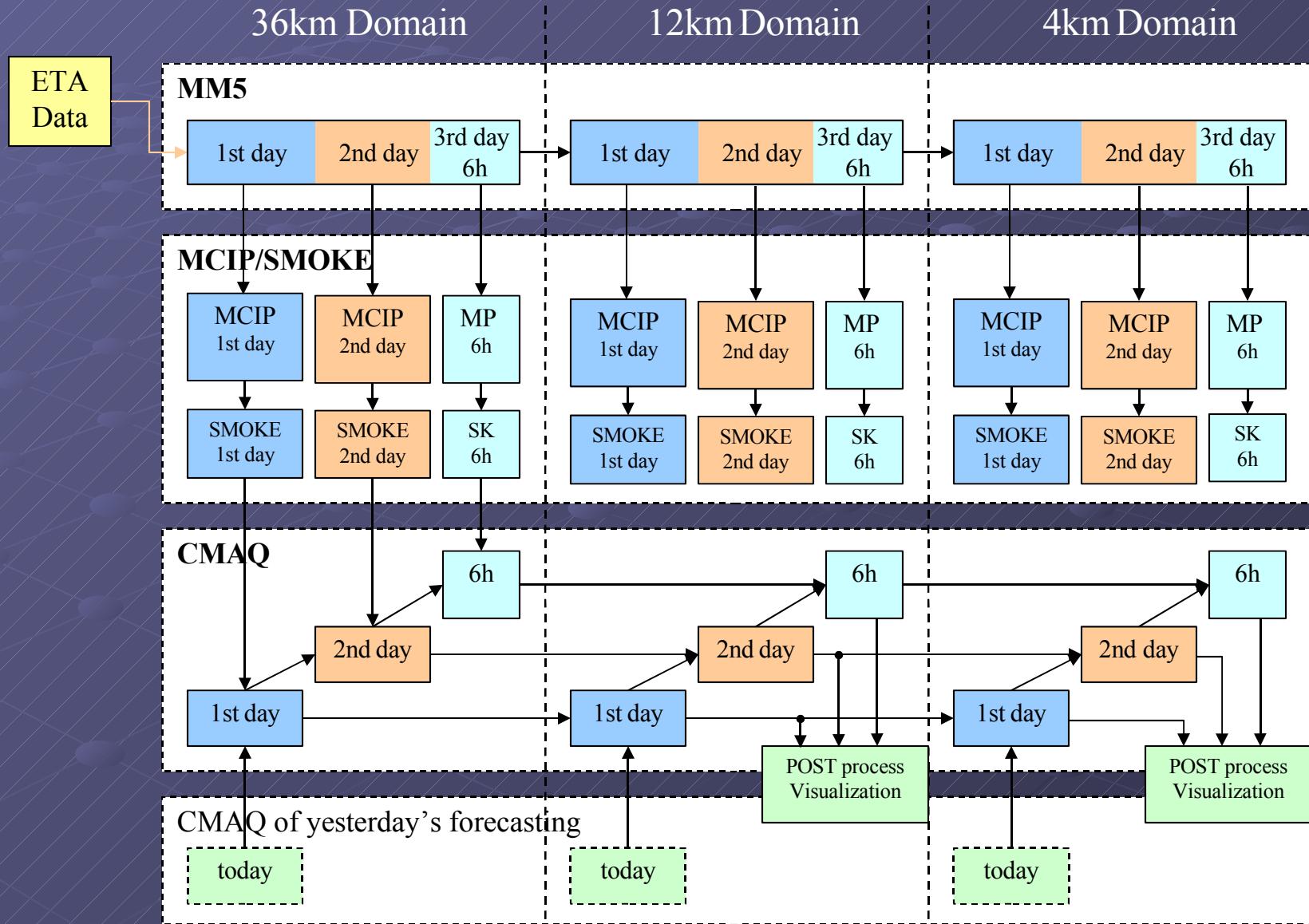
- Computational and data intensive

- Runs ~3h for parallel modules on 32-128 CPUs
- 30GB data daily

- Strict deadline

- 15:30PM – 6:00AM

# Workflow of AQF Run



# AQF on Campus Grid

- Resources

- Sun Fire Clusters, Beowulf, NFS and SAN, Internet2 connection to ETA

- SGE, LSF, Globus 2.x/4.x, SRB

- UH HPCC CA

- EZ-Grid portal for users to interact with grid resources

- Light-weight solution for AQF
  - Dynamic queue info interface with SGE

# AQF Workflow

- AQF modules are installed on resources
  - They are thoroughly profiled
- An XML file describes module dependencies
  - Minimum DTD for AQF
- A master Perl script controls AQF workflow
  - Job submission, file transfer
- Other scripts control the internal workflow of each module
  - MM5

# Current AQF Workflow Execution: Rigid Interaction Pattern

- Resource allocation
  - Pre-allocated by system admin and specified in scripts/RSL
  - AQF modules have high priority in local schedulers
- Remote execution using rsh, ssh, or Globus
- Files are shared via NFS, or transferred by gridFTP, scp
- Manually initiated, human monitoring, user intervention if exceptions arise
- Flexible and adaptive AQF execution needs support in:
  - Workflow-coordinated grid scheduling
  - Modeling applications for grid
  - A grid environment for domain-specific applications

# Gracce

- Grid Application Coordination, Collaboration and Execution

- Coordination: internal workflow, tight interaction
- Collaboration: external interaction, loose interaction
- Execution: scheduling, runtime and monitor

- To provide a deployment platform for AQF-like applications on grid environments

- To fill the gaps between domain users and grid middleware and infrastructure

- Approaches

- Users describe the structure of domain applications
- Gracce sets up an application-specific environment, from application and data management, to automatic workflow execution and user interfaces

# Current Gracce Solutions

- Gracce Application Modeling and Description Language (GAMDL)
- Gracce Metascheduler architecture
- Others such as ad-hoc VO management, RM adaptor and portal plug-in

# GAMDL

- Data-flow description of application logics
  - Executable, dataset, module, dependency, job
- Powerful
  - Control flow can be described using conditioned properties and conditioned pipes
  - Multiple-value property for describing similar entities
- Easy-to-use, does not require knowledge of grid
- Clear document structure by using entity Uid
- Provides rich information about the job for grid scheduling purpose
- GAMDL specification and AQF example: [1, 4]

# Gracce Metascheduling Architecture

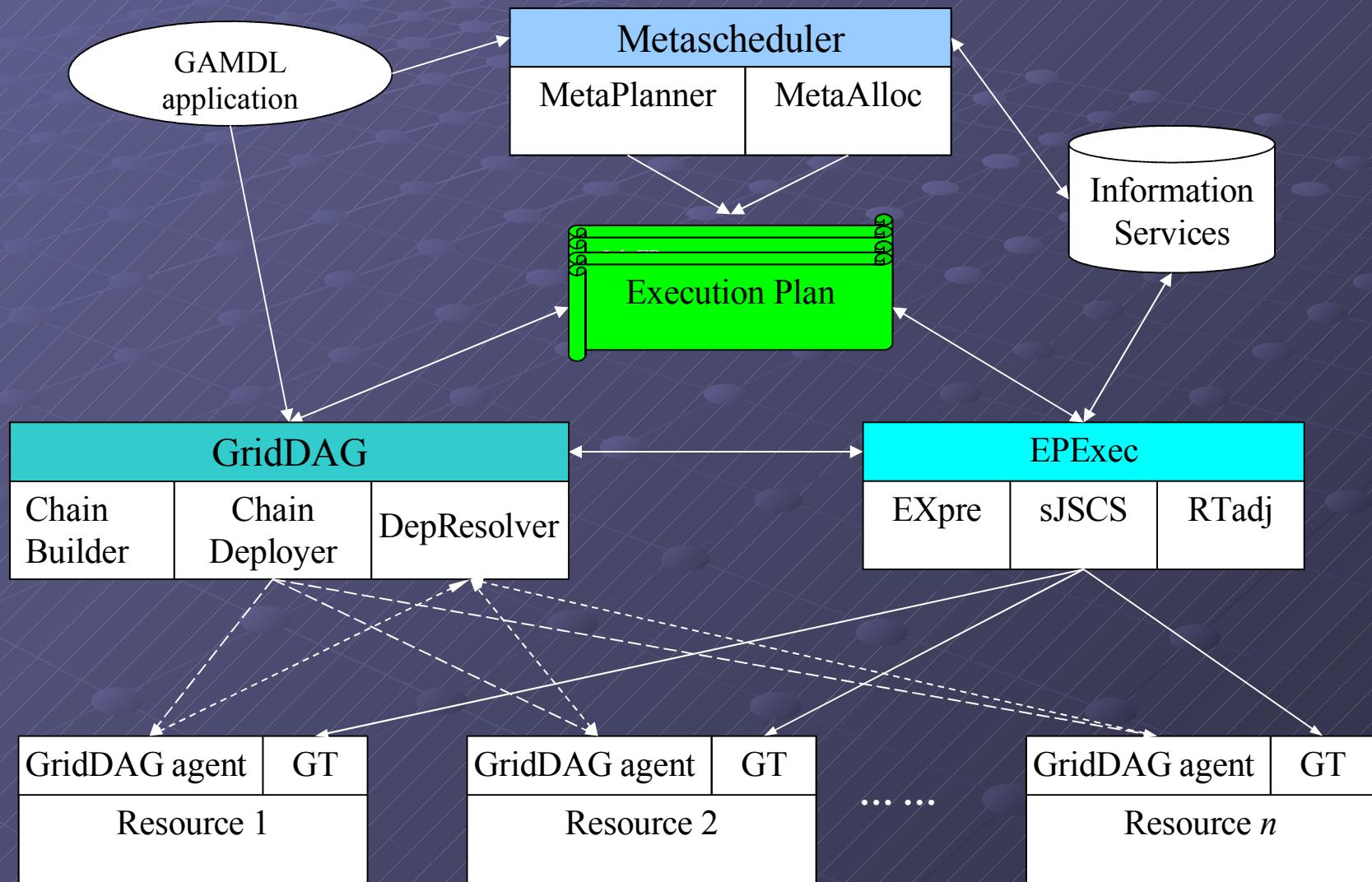
## ● Metascheduler

- Discovers, evaluates, and co-allocates resources for application coordination and collaboration,
- Coordinates activities between multiple heterogeneous schedulers that operate at the local or cluster level.

## ● Gracce Metascheduling

- Workflow orchestration in scheduling and execution
- Planning and resource negotiation/reservation
- Scheduling isolated from runtime management
- Execution Plan

# Metascheduler and Execution



# Gracce Metascheduling

- Metascheduler

- Plan workflow execution and allocate/reserve grid resources
- Scheduling decision is written as execution plan (EP)

- GridDAG event-driven workflow engine

- Coordinate metascheduling process with application workflow
- Coordinate workflow execution, basically handling and resolving dependencies

# Grid Metascheduling

- EPExec runtime
  - Prepare workflow execution
  - Execute workflow according to application EP
  - Handling runtime exceptions and mismatches between plan and runtime

# Experiences

## ● Deploying legacy applications

- Started with rigid interoperation and interaction pattern between AQF and campus grid
- Reduced operational complexity and dynamics as much as possible at first stage
- Application-centric in middleware integration

## ● Domain users expect an “environment”

- not just tools, technology and standard
- an integrated platform so that users can grid-enable applications via an intuitive and simple GUI

# Experiences

- Our expectations from grid middleware
  - High-level solutions for enabling applications, e.g. metascheduler, brokerage, and portal
  - Workflow and application co-co support from OGSA/Globus or other grid middleware
  - Modeling or packaging complex legacy applications for grid

# Reference

- GAMDL: Application Modeling and Description for Grid Integration, Y. Yan, B. M. Chapman, IEEE/ACM Workshop on Grid Computing 2005, submitted
- Air Quality Prediction in a Production Quality Grid Environment, Engineering the Grid: status and perspective, B. M. Chapman, P Raghunath, B. Sundaram, Y. Yan, edited by J. Dongarra, H. Zima, A. Hoisie, L. Yang and B. D. Martino, Spring 2005
- Grid Environment With Web-Based Portal Access For Air Quality Modeling, B. M. Chapman, H. Donepudi, Y. Li, P. Raghunath, Y. Yan And B. Sundaram and J. He, Parallel And Distributed Scientific And Engineering Computing: - 2004 Practice And Experience Advances In Computation: Theory And Practice, Volume 15
- Gracce: <http://www.cs.uh.edu/~hpctools/gracce>

