TECHNISCHE UNIVERSITÄT DRESDEN

Center for Information Services and High Performance Computing (ZIH)

Monitoring of jobs and their execution in the LHC Computing Grid

Ralph Müller-Pfefferkorn, Reinhard Neumannn
Technische Universität Dresden, Germany

Torsten Harenberg, Matthias Hüsken, Peter Mättig,
Markus Mechtel, David Meder-Marouelli
Bergische Universität Wuppertal, Germany

Cracow, October 17th 2006

ZIH
Center for Information Services & High Performance Computing

The Need for Job Monitoring

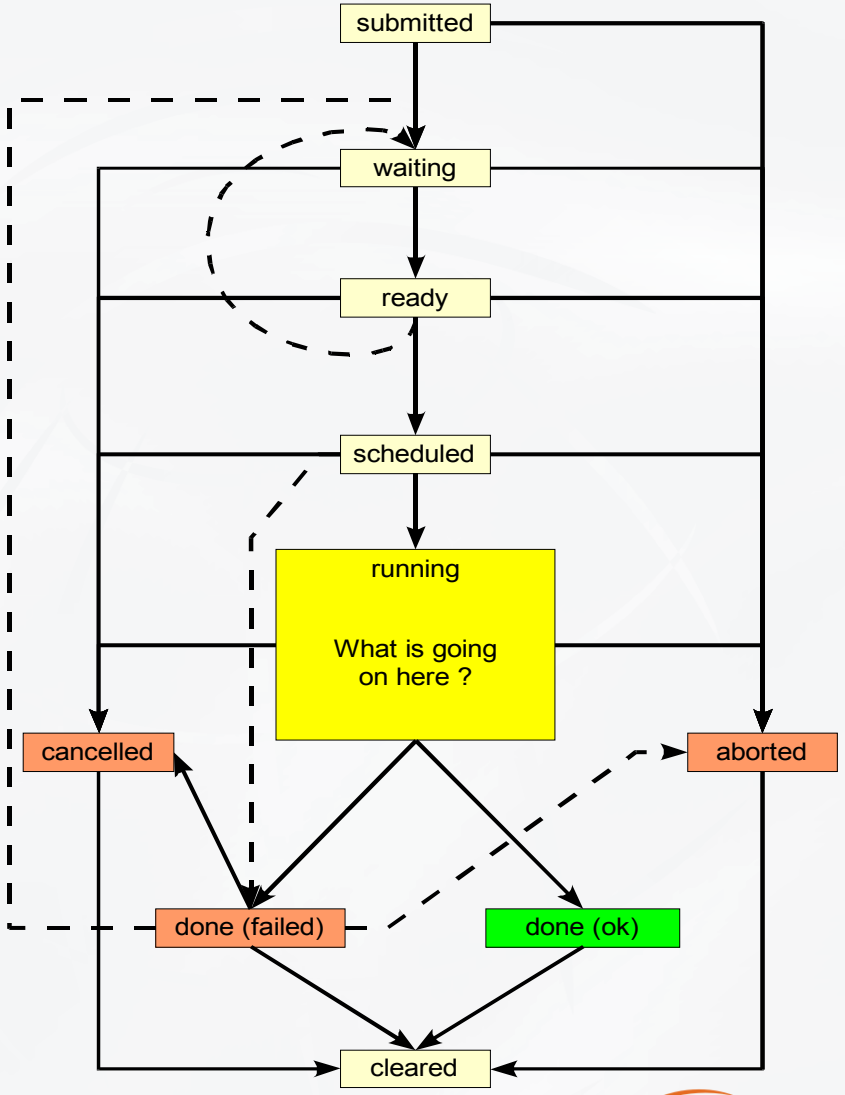
- Large Hadron Collider (LHC) at CERN will start 2007
- LHC will produce petabytes of data per year
- Huge amount of computing power and storage needed to analyse data
- Typical scenario:
  A physicist will submit hundreds of jobs for a single analysis
- And there are thousands of physicists ...
  - ➔ useful monitoring of jobs
  - ➔ identification of and recoverage from failures

High Energy Physics Community Grid (HEPCG)

- In September 2005 national German Grid initiative started
- HEPCG is one of the community Grid projects
- Create Grid tools for physicists
- Workpackage 2: Monitoring
  1. Job and Resource Usage Monitoring
  2. Job Execution Monitoring and Expert System for Failure Classification
  3. Online Steering of HEP Grid Applications (see Daniel Lorenz' talk in session C9)

1. The Job Execution Monitor

- Developed at
  Bergische Universität Wuppertal
- LCG knows only 2 final states
  - done (ok)
  - done (failed)
- No information about failure reasons !



1. The Job Execution Monitor: Goals

- Detect and identify failures
  - misconfigurations (middleware, firewall ...)
  - problems on worker node
  - hardware failures (network ...)
  - missing software
  - bugs in user application
- Possibly repair failures --> expert system

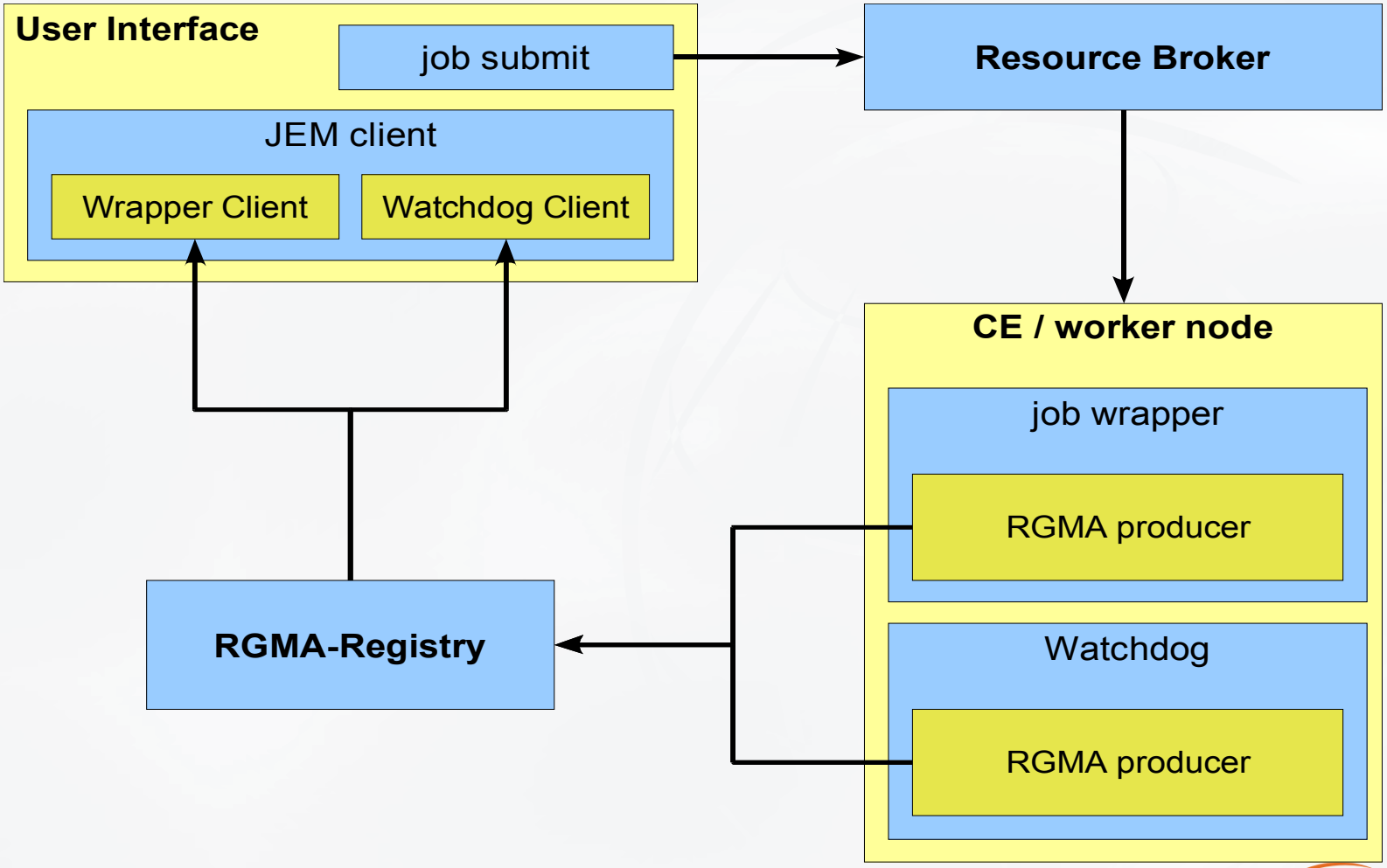
1. The Job Execution Monitor: How does it work?

Two components on the worker node:

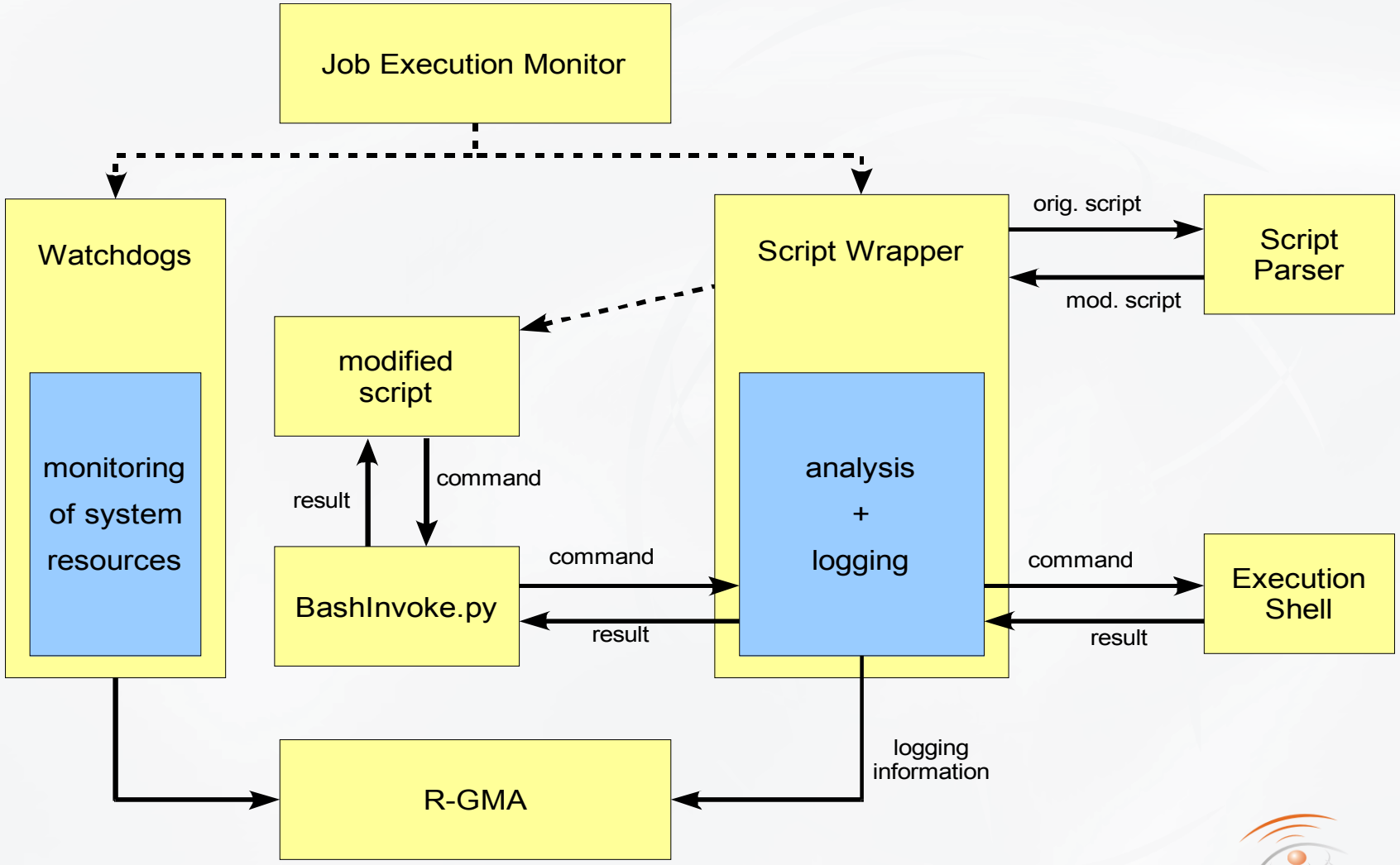
1. Step-by-step execution of the job shell script

2. Monitoring of system resources to identify possible sources of problems

1. The Job Execution Monitor: Architecture



1. The Job Execution Monitor: On the worker node



1. The Job Execution Monitor: Tests and Experiences

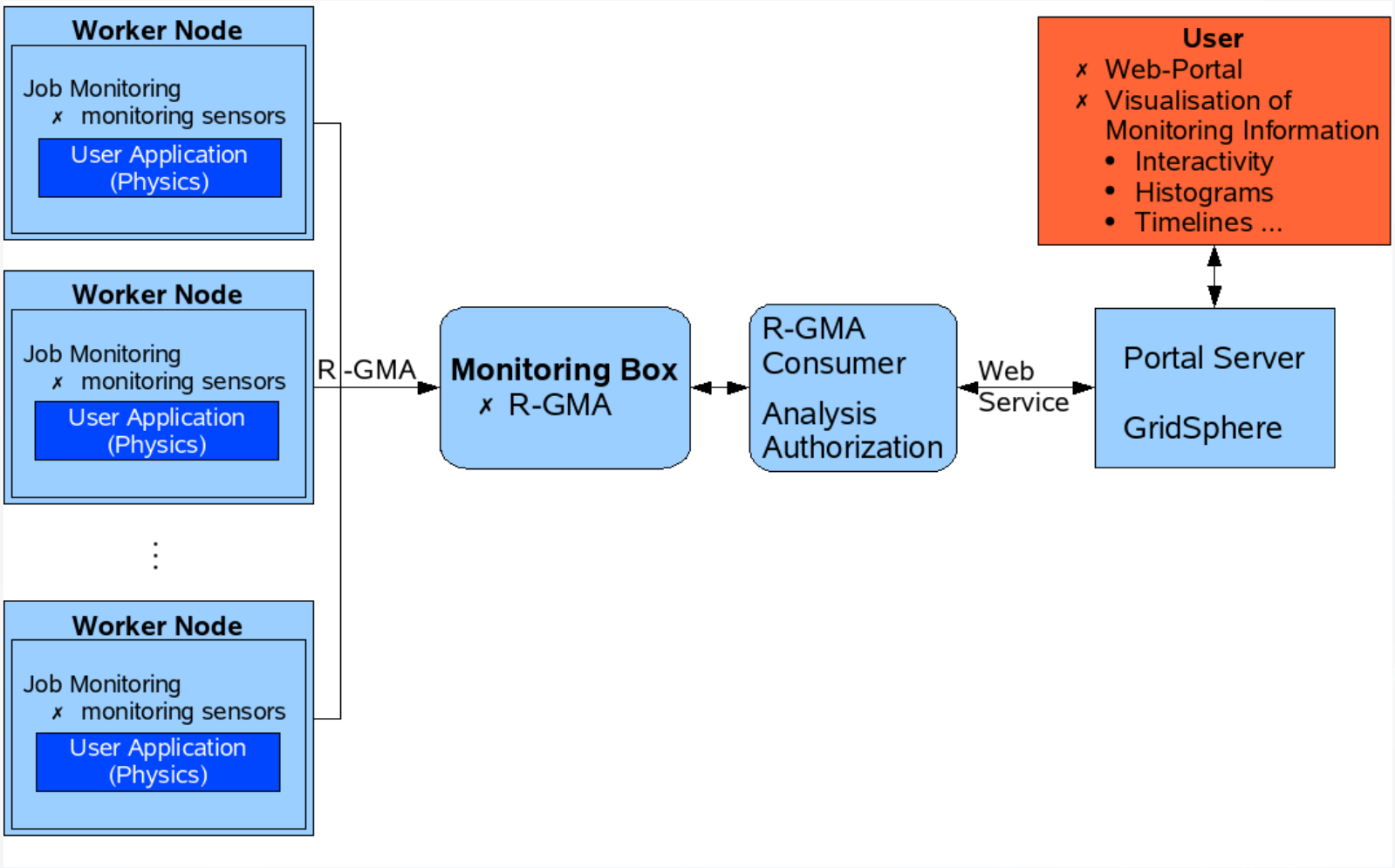
Jobs	Number	Success	failed	not started
seti @ home	110	103	7	0
distributed.net	100	97	3	0
testjobs	765	718	47	41

- Jobs not started are lost
- Identified failure reasons
  - seti@home Server in Berkeley missing
  - misconfigured batch system (jobs killed)
  - successful jobs marked as failed

2. Job and Resource Usage Monitoring: Goals

- Developed at ZIH, Technische Universität Dresden
- Usercentric monitoring
  - Job and resource usage monitoring of the hundreds or thousands of jobs of a user
  - Resource usage monitoring for resource providers
- For daily usage
  - easy access and handling
  - only limited knowledge about monitoring is needed by the user
- Support user with graphical representations of the information
- Authentication, authorization and secure data transmission
- Integrated into LCG / gLite environment

2. Job and Resource Usage Monitoring: Architecture



2. Job and Resource Usage Monitoring: Information Gathering and Storage

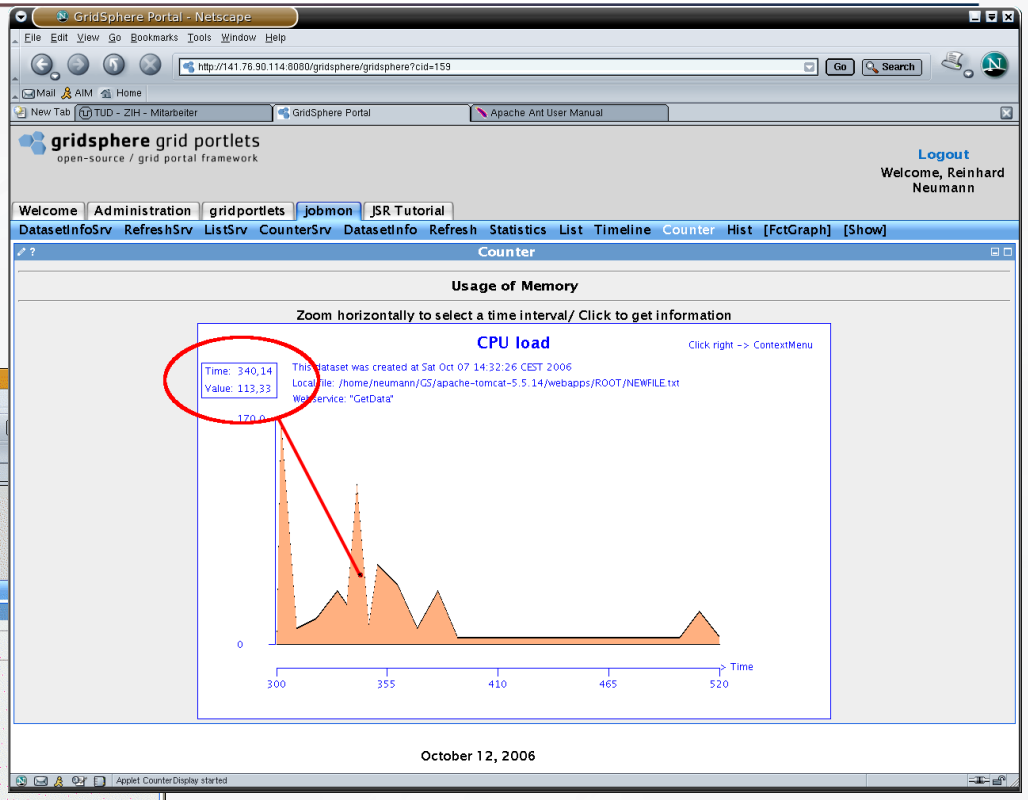
- Information Gathering
  - currently uses existing LCG Job Monitor
    - user has to set an environment variable only
    - runs parallel to job on worker node and samples information
  - currently monitors CPU usage, memory usage, loads ...
  - will be extended for more information
    e.g. I/O rates
- Data are stored in R-GMA
  - Relational-Grid Monitoring Architecture
  - a kind of distributed relational database based on OGSA - Grid Monitoring Architecture

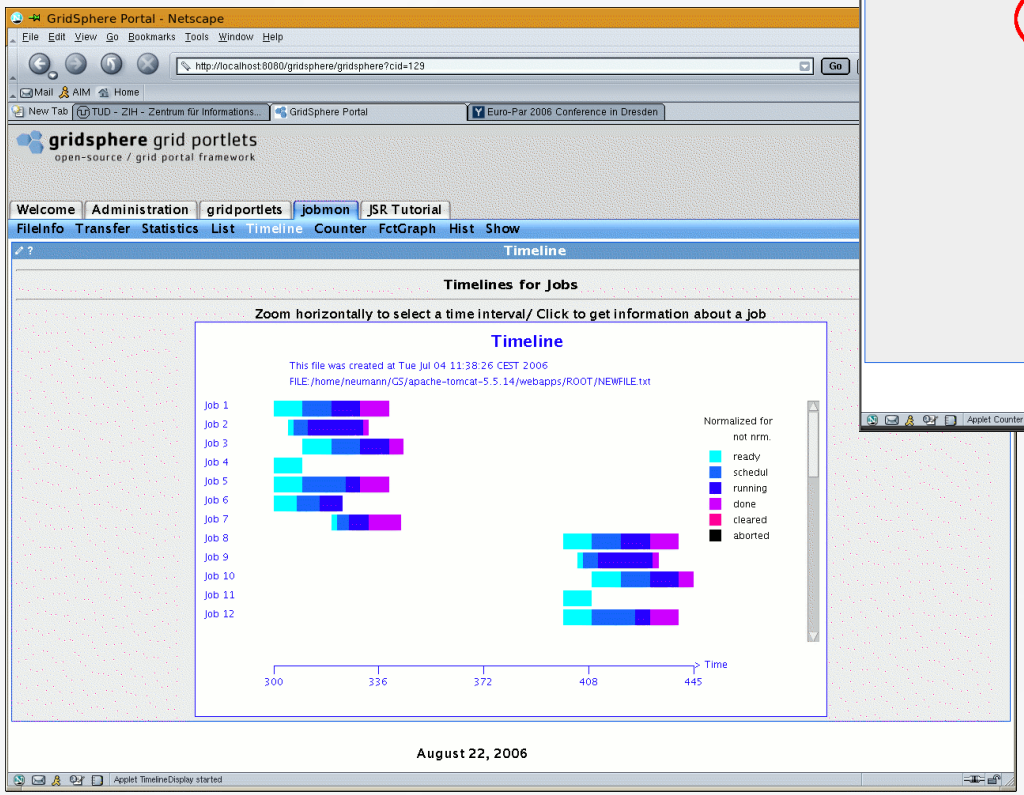
2. Job and Resource Usage Monitoring: Information Retrieval and Analysis

- R-GMA consumer reads monitoring data
- Analyses data
- Prepares data for visualization
- Is a Web Service, that is called by the visualisation

2. Job and Resource Usage Monitoring: Visualization

- Get data from Analyzer Web Service via secure transmission
- Authentification
- Make nice and useful pictures !
  - histograms, time lines, pie charts ...
- Allow interactivity
  - a click into a graphic reveals more information, e.g. details
  - zooming
- User Interface
  - browser based
  - integrated into GridSphere portal


GridSphere Portal - Netscape
gridsphere grid portlets
open-source / grid portal framework
Logout
Welcome, Reinhard Neumann
Welcome Administration gridportlets jobmon JSR Tutorial
DatasetInfoSrv RefreshSrv ListSrv CounterSrv DatasetInfo Refresh Statistics List Timeline Counter Hist [FctGraph] [Show]
Counter
Usage of Memory
Zoom horizontally to select a time interval/ Click to get information
CPU load
Click right -> ContextMenu
Time: 340,14
Value: 113,33
Time
300 355 410 465 520
October 12, 2006


GridSphere Portal - Netscape
gridsphere grid portlets
open-source / grid portal framework
Welcome Administration gridportlets jobmon JSR Tutorial
FileInfo Transfer Statistics List Timeline Counter FctGraph Hist Show
Timeline
Timelines for Jobs
Zoom horizontally to select a time interval/ Click to get information about a job
Timeline
Job 1 Job 2 Job 3 Job 4 Job 5 Job 6 Job 7 Job 8 Job 9 Job 10 Job 11 Job 12
Normalized for not nrm.
ready
schedul
running
done
cleared
aborted
Time
300 336 372 408 445
August 22, 2006

Summary and outlook

- Job execution monitoring
  - detect and identify job failures
  - first version available
  - future work:
    - automatic identification and classification
- Job and resource usage monitoring
  - collect and visualize monitoring data for hundreds of jobs of a user
  - first prototype by the end of October 06
  - future work:
    - integrate authorization
    - collect more information
    - read other existing monitoring systems