



TECHNISCHE  
UNIVERSITÄT  
DRESDEN

Zentrum für Informationsdienste und Hochleistungsrechnen

# Entwicklung von Werkzeugen zum Job-Monitoring

Henrik Eichenhardt, Ralph Müller-Pfefferkorn,  
Reinhard Neumann, Thomas William

Dresden, 16.Juni 2008

---

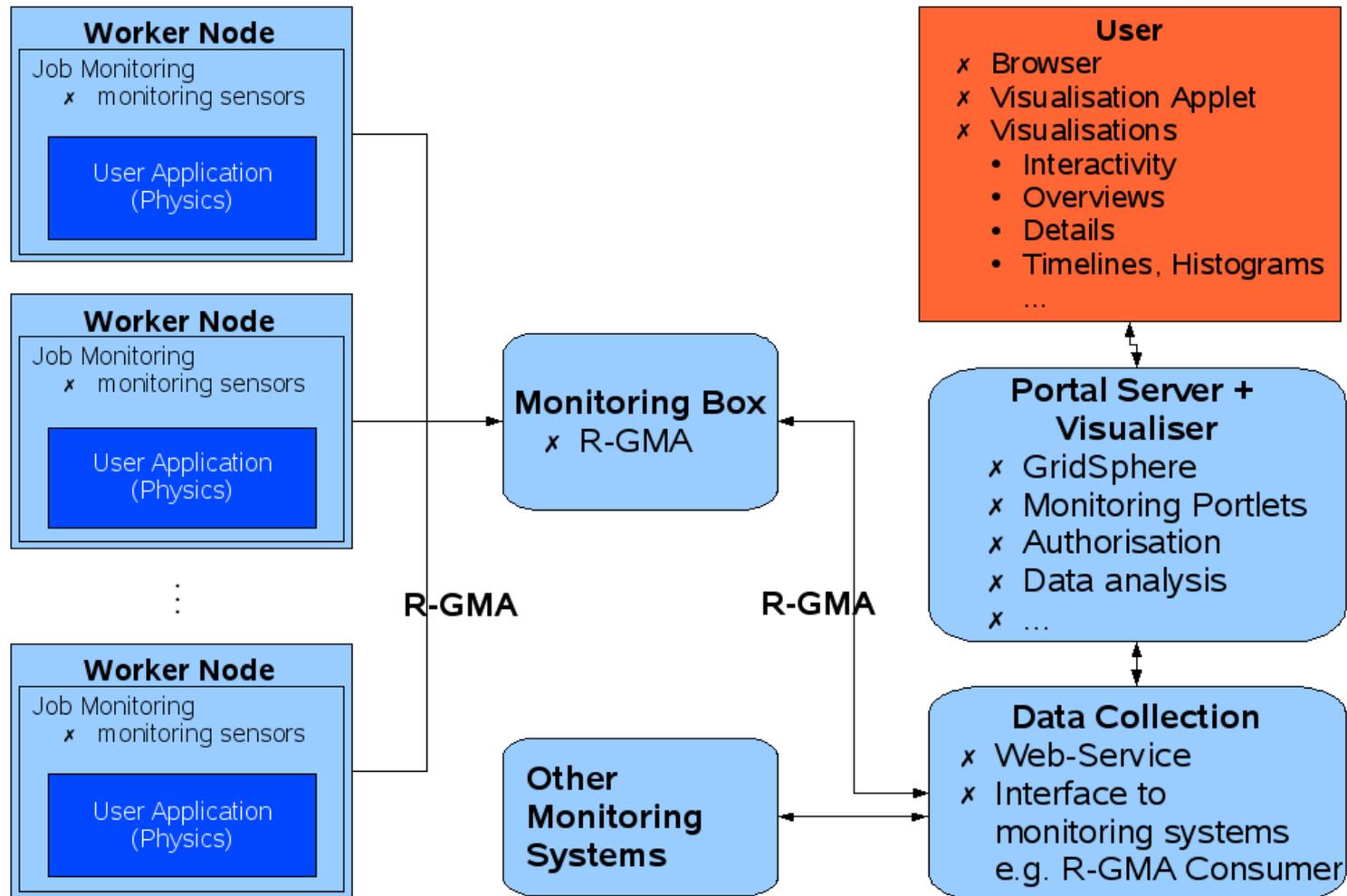
# Aufgabenstellung

- Aufbau eines Informationssystems über den Jobzustand und die Ressourcennutzung sowie deren Analyse für organisatorische und planerische Zwecke
  - ▶ Teilbereich 1: Ermittlung und Darstellung der für den Benutzer wichtigen Job Informationen
  - ▶ Teilbereich 2: Aufbau eines Informationssystems zur lokalen und LCG-weiten Ressourcennutzung (Monitoring der Ressourcennutzung) sowie deren Analyse für planerische und organisatorische Zwecke

# Auswahl der zu beobachtenden Daten

<i>Category</i>	<i>Metric</i>
<i>General Informationen</i>	job ID; user name; resource broker; computing element; worker node; local job ID on the worker node
<i>CPU</i>	WallClockTime; CPUTime; Load Averages
<i>Memory</i>	real, virtual, total and free memory; free and total SWAP;
<i>Disk</i>	free space in HOME, TEMP and the working directory; summary of the disk usage of the filesystem
<i>File I/O</i>	I/O rates for every file the application is accessing
<i>Network</i>	sent and received network traffic
<i>Output</i>	the last lines of a number of user specified output files

# Entwurf der Monitoring-Architektur



---

# Implementierung des Ressourcennutzungs-Monitoring

- Worker-Node-Monitoring
  - ▶ basierend auf lcg-mon-wn
  - ▶ bereits vorhandenes Paket sollte Integration erleichtern
  - ▶ erweitert um eine ganze Reihe von jobspezifischen und System-Informationen
  - ▶ Eigenentwicklung
    - Bibliothek zum Monitoring von I/O-Zugriffen
    - Lese-/Schreibraten einer Anwendung pro Datei
- Web Service zur Datensammlung aus R-GMA, Portal-Einbindung und Java-Applets zur Visualisierung
  - ▶ Eigenentwicklung

# Entwurf von Visualisierungskonzepten

The image displays four screenshots of the GridSphere Portal interface, illustrating various visualization concepts for job monitoring and management.

### Log-Files

This view shows the output of a benchmark test. The text includes:

```

2008-05-07 -- 18:19:00 >>>
benchmark.out:
NUMCPU=2
BenchT: Using Timer "bi_gettimeofday_improved"
BenchT: Timer granularity: 2.16840434e-10 ns
BenchT: Timer overhead: 189.5 ns
BenchT: Getting info about kernel:
kernelname=numerical.matmul.C.0.0.double, kernelstring=numerical.matmul.C.0.0.double [OK]
BenchT: Getting starting time... [OK]
BenchT: Selected kernel: "numerical.matmul.C.0.0.double"
BenchT: Initializing kernel... [OK]
BenchT: Allocating memory for results... [OK]
BenchT: Measuring...
progress scale (percent):
0-----20-----40-----60-----80-----100
  
```

### JobData (UsedCPUTime [s])

This view displays a horizontal bar chart showing the CPU time used by various jobs. The Y-axis lists job IDs (J00000 to J00150), and the X-axis represents CPU time in seconds, ranging from 954 to 3798. A color scale on the right indicates the WCTime [s] from 7 to 3798.

### JobStatus (States of Jobs)

This view shows a pie chart representing the distribution of jobs across different states. The total number of jobs is 783.

State	Count
Submitted	0
Waiting	112
Ready	2
Scheduled	0
Running	0
Done	38
Cleared	595
Aborted	35
Cancelled	1

### Job Problems (Job Filters)

This view displays a table of job problems, showing the status of various filters for each job. The filters include: NET, Swp, OverLoad, IDLe, SYS, MEM, DISK, and CalC. The table lists job IDs (J00006 to J00015) and their corresponding filter statuses (e.g., high CPU overload, high system load, low network access).

Legend: IO NET Swp OverLoad IDLe SYS MEM DISK CalC.

### Counter (Usage of Memory)

This view shows a line graph of CPU load over time. The Y-axis represents CPU load (0 to 170.0), and the X-axis represents Time (300 to 520). The graph shows a peak in CPU load around time 350. The current value is 113.33.

Time: 340.14  
Value: 113.33

October 12, 2006

---

# Änderungen während der Projektlaufzeit 1

- bereits in ersten HEPCG-Meetings
  - ▶ Forderung nach Analyse der Monitoringdaten für einzelnen Nutzer
    - nicht mehr nur reine Datensammlung und -präsentation
    - neuer Schwerpunkt:  
Analyse der Daten --> Hinweise auf mögliche Jobprobleme
  - ▶ schaut nach Problemen bei Nutzung von Ressourcen
    - CPU, Memory, Plattenplatz, Netzwerk, I/O, Rechner-Last ...

---

# Änderungen während der Projektlaufzeit 2

- deshalb „Entwicklung von Analyse-Methoden zu Grid-Performance und Resource-Brokering“ hintenangestellt
  - ▶ noch in der Entwicklung
    - z.B. Datensammlung von Resource-Broker Daten aus Logging & Bookkeeping
  - ▶ problematisch, weil Daten nur für eigentlichen Job-Submitter zugänglich
  - ▶ wird vermutlich bis Projektende nicht vollständig erreicht

---

# Integration der Entwicklungen in LCG und Bereitstellung von Paketen für D-Grid

- Worker-Node-Monitoring
  - ▶ RPM für SL3 und SL4
- Web-Service zur Datensammlung
  - ▶ basierend auf Axis – in SL3/4 inklusive, aber auch für andere Distributionen
  - ▶ am einfachsten auf Monitoring-Box installierbar
  - ▶ einzelne Server pro VO
- Portal
  - ▶ Gridsphere-Portlets
  - ▶ ein Server pro VO
- Visualisierung mit Java-Applets in Browser
  - ▶ einzige Anforderung an Nutzer: Java Policy in Browser setzen (muss Transfer von Daten vom Portal-Server an Browser erlauben)

---

# Wie geht es weiter?

- Bis Ende August
  - ▶ Arbeiten an Effizienz- und RB-Analyse
  - ▶ Integration in gLite-WMS
    - integriert in LCG-RB
    - gLite-WMS hatte Starten des Worker-Node-Monitoring nicht berücksichtigt – nach Anfrage wieder aufgenommen
  - ▶ Verbesserung der Nutzerfreundlichkeit
- Und dann?
  - ▶ AMon in DGI2
    - Erweiterung für Globus
  - ▶ HEPCG-2 ?
    - Support, Betrieb?

---

# Was hat der Nutzer nun wirklich von AMon?

- einfacher Zugang über den Browser
- einfaches Anschalten – Umgebungsvariable setzen
- grafische Darstellung der Statusinformationen inklusive Fehlermeldungen der RB/WMS
- Hinweise zu möglichen Problemen
  - ▶ pro Job Hinweise in Rot – Gelb – Grün
- Zugang zu Informationen über Ressourcen-Nutzung
  - ▶ wie auf Desktop
  - ▶ CPU, Speicher, Plattenplatz, Netz ...
- kurze Vorführung!