



TECHNISCHE
UNIVERSITÄT
DRESDEN

Zentrum für Informationsdienste und Hochleistungsrechnen

Sammlung und Analyse von jobspezifischen Monitoring-Daten

Ralph Müller-Pfefferkorn

Dresden, 27.5.2008

Übersicht

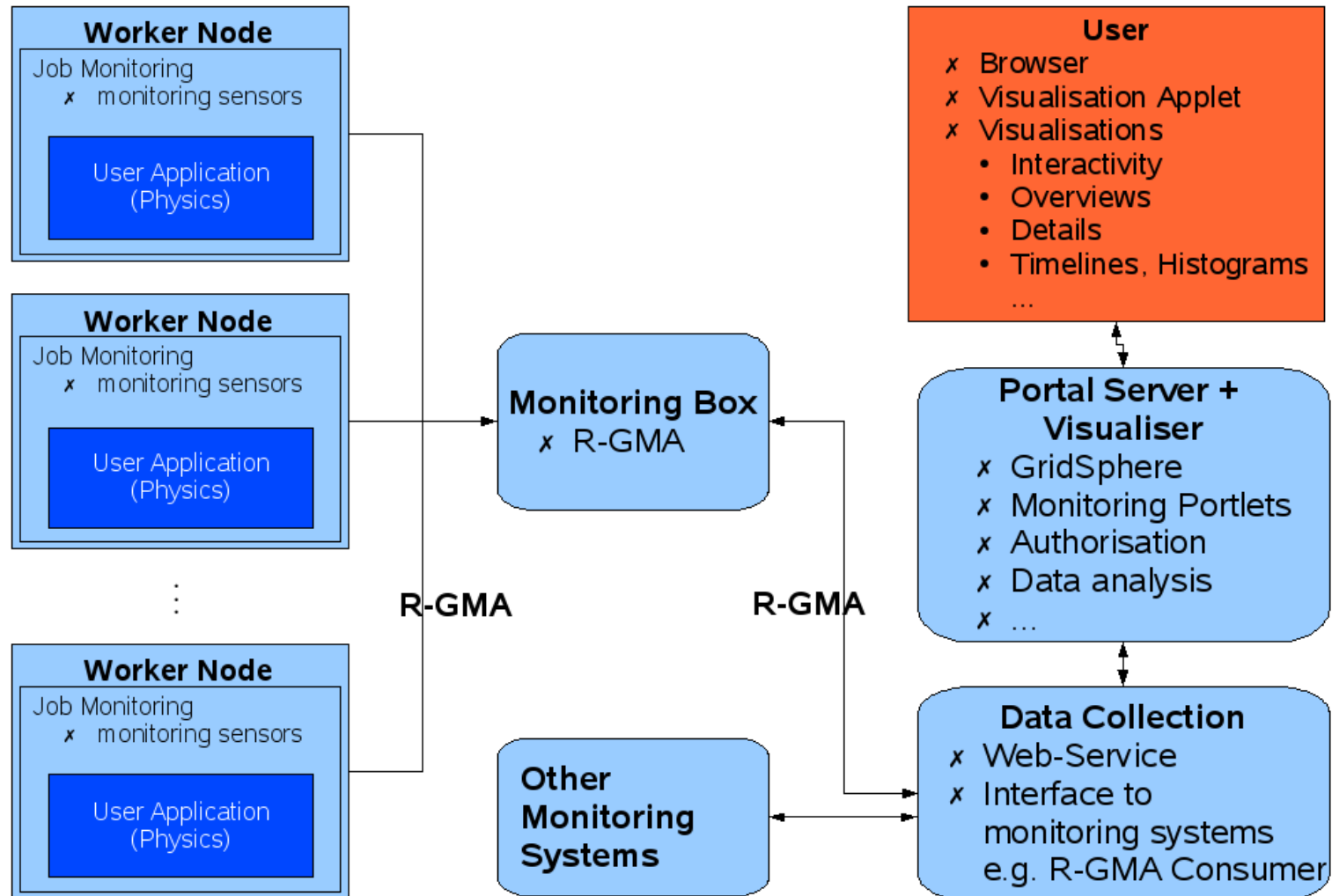
- Was meint jobspezifisches Monitoring?
- AMon - Kurze Vorstellung
- Nutzerunterstützung durch Analyse
- AMon in DGI2

Was meint jobspezifisches Monitoring?

- Nutzer ist Informationen über Jobablauf gewöhnt
 - ▶ Auf Desktop-Rechner unter Unix/Linux: top, ps ...
 - ▶ Im Batch-System: qstat, ...
 - ▶ Im Grid
 - Wenige jobspezifische Informationen
 - Status, welcher Host, vielleicht hostspezifische Ganglia-Informationen
- Jobspezifische Informationen
 - ▶ Informationen über jeden einzelnen Job
 - In Realzeit
 - Für jeden Job
 - Äquivalent zu Informationen auf Desktop-Rechner
 - ▶ Analyse der Daten zur Nutzerunterstützung

AMon – Analysis and Monitoring

- In HEPCG auf Basis von gLite und R-GMA entwickelt
- R-GMA – Relational Grid Monitoring Architecture
 - ▶ Eine Art verteilte relationale Datenbank
 - ▶ Basierend auf GMA-“Standard“ des OGF
 - ▶ Consumer-Producer-Registry
 - ▶ Zeitlich definierbares Vorhalten der Monitoring-Daten



Skalierung

- durch Aufgabenverteilung
 - ▶ Worker Nodes
 - Lokale Datensammlung
 - ▶ Web Service
 - sammelt verteilte Daten ein
 - ▶ Portal
 - Nutzerzugriff und Datenabfrage, Voranalyse
 - ▶ Desktop-Rechner
 - Visualisierung und Interaktion

Daten

<i>Category</i>	<i>Metric</i>
<i>General Information</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

Nutzerschnittstelle und Visualisierung

- Browser basierend
- Integriert in Portal – Gridsphere
- Interaktive Visualisierung mittels Java Applets
- Autorisierung durch Anbindung an Autorisierungssysteme
 - ▶ VOMS
 - ▶ Hole Grid-Proxy von MyProxy-Server
 - ▶ Anfrage an VOMS-Server
 - ▶ Nutzer bekommt nur seine Daten zu sehen

Lokale persistente Datenspeicherung

- Möglichkeit für Ressourcen-Provider
- Speichere Monitoring-Daten der Site persistent in Datenbank
- Spätere Analyse der Daten
 - ▶ z.B. nach Problemen

Nutzerunterstützung – Charakterisierung von Job-Zuständen

- HEP: Ein Nutzer schickt Hunderte oder tausende Jobs ab
- Wie soll er die Übersicht behalten?
 - ▶ Grafische Darstellung der Monitoring-Daten
 - ▶ Analyse der Monitoring-Daten
 - Alle „guten“ Jobs interessieren im Wesentlichen nicht
 - Möchte Hinweise auf „böse“ Jobs (hängend, crashed ...)
 - Charakterisierung von Job-Zuständen auf Basis der gesammelten Daten sowohl zum augenblicklichen Zeitpunkt als auch in der Vergangenheit
- Charakterisierung
 - ▶ „Schlecht“, „Mittel“, „Gut“ = Rot, Gelb, Grün

Einzeljob-Analyse

- Verschiedene Filter schauen auf die Daten jedes einzelnen Jobs
- Jeder Filter sucht nach bestimmten Problemen
 - ▶ Last, Speicherauslagerung (Swap), Plattennutzung, I/O, CPU-Nutzung, Speichernutzung, Leerlauf, Netzwerk, System-Aktivitäten
- Mischung aus Job- und Hostspezifischen Daten
- Beispiel: Swap-Filter
 - ▶ Performance-Einbußen durch Swapping (Auslagern von Hauptspeicher auf Festplatte)
 - ▶ Filter prüft Last des Rechners durch I/O, gesamte und job/prozess-spezifische Belegung des Hauptspeichers, Größe des Swap-Bereiches ...

Vergleich von Jobzuständen

- Analyse der in der Einzeljob-Analyse gefundenen Jobzustände
- Einfachste Aussage
 - ▶ Wenn ein bestimmter Anteil an „schlechten“ Zuständen in einem Job überschritten wird, ist der Job problematisch.
- Gleichartige Jobs
 - ▶ z.B. eine Analyse in der Teilchenphysik, bei der jeder Job nur verschiedene Datenpakete liest
 - ▶ Zustandsverteilung ist gleich
 - ▶ Seltener Zustand = ein Problem?
 - Ein Zustand ist selten, wenn er nur in weniger als 5% aller Zustände aller Jobs auftritt.
 - ▶ Problem, wenn Abweichung vom mittleren Rechenzeitverbrauch
 - Verhältnis genutzte CPU-Zeit zu Laufzeit
 - Abweichung von mehr als 2 Standardabweichung der Verteilung über alle Jobs

Augenblickliche und weitere Arbeiten

- Analyse der Monitoring-Daten
 - ▶ Studien zu einer Clusteranalyse
 - finde gleichartige Jobtypen (z.B. Jobs einer Analyse eines Nutzers)
 - Monitoring-Daten als N-dimensionale Vektoren bilden Cluster bei gleichartigen Jobs
 - problematische Jobs sind „Ausreißer“
- Vergleich Anforderungen des Nutzers an die Ressource und Verbrauch
 - ▶ Nutzer kann beim Submittieren Anforderungen spezifizieren (z.B. mindestens 2 Gbyte Hauptspeicher)
 - ▶ Vergleich mit Verbrauch / Hostdaten

Jobspezifisches Monitoring in DGI2

- Seit 1.5.2008 Weiterentwicklung von AMon in DGI2
- Anbindung an Globus4
 - ▶ Integration des Monitoring-Dämons zum Sammeln der lokalen Daten
 - ▶ Speicherung der Daten in MDS
 - ▶ Auslese aus MDS
- Anbindung an andere Autorisierungssysteme
- Anpassung der Visualisierung
- Support der Communities bei Integration, Installation und Einsatz des Monitoring

Kurze Demo