

Job Execution Monitor

Ahmad Hammad Torsten Harenberg Peter Mättig Markus Mechtel David Meder-Marouelli

HEPCG Workshop, April 27th, 2006





Bundesministerium für Bildung und Forschung



Markus Mechtel

Job Execution Monitor



- Motivation
- Goals
- Constraints
- Job Execution Monitor
 - Watchdog
 - Script wrapper
- Experiences
- Summary
- Outlook









Motivation

submitted LCG jobs either finish **ok** or waiting fail • ready user doesn't get any information about the reason of job failures sceduled running 30% of real LCG jobs fail What is going on here ? cancelled aborted LCG-software evaluates function of grid-infrastructure only, not the results of jobs done (failed) done (ok) cleared



Job Execution Monitor

D-GRID sources of failures

possible sources of failures

- grid middleware configuration
- workernode configuration
- workernode problems
 - disk full
 - lost network connection
 - firewall misconfiguration
- missing software (e.g. required libraries)
- problems in user software





- detect errors within running jobs
- identify sources of failures
- correct errors automatically (expert system)

user benefits

- user knows, which command leads to failures
- much information about jobs without additional work

site administrator benefits

- current status of hardand software
- information about site problems





- job monitoring on the workernode
 - stepwise execution of shell-scripts
 - process monitoring
- realtime information
 - user knows current state of his jobs
 - stdout/stderr access even in case of failures (in LCG only available on successful jobs)
- graphical user interface for easy access
 → Dresden



Constraints

- stability
 - on internal errors \rightarrow run job without monitoring
 - Python
 - installed on every LCG machine
 - platform-independant (programs run on 64bit CPUs without modification)
- information exchange via RGMA only
 - preserves security context
 - no firewall problems





- client/server modell
- 2 components
 - watchdog (workernode monitoring)
 - job wrapper (stepwise execution of shell-scripts)
- monitoring system is automatically added to the user's job submission, no additional work for user







- monitors system resources:
 - free memory
 - free disk space
 - network I/O
 - processor load
- provides graphical views using rrdtool (round robin database)
- data published regularly via RGMA
- missing RGMA \rightarrow fallback to RPC







Job Execution Monitor



System watchdogs - 3 hour view

1 Hour View 13 Hour View 110 Hour View 11 Day View 11 Week View



Network Interfaces





Markus Mechtel

Job Execution Monitor

Script-wrapper and -parser



characteristics:

- lexical and syntactical analysis of shell-scripts
- critical actions are made more robust, e.g. file transfers
- stepwise execution
 - \rightarrow traceback of script execution in case of errors



Script-wrapper and -parser

operating principle:

- parser identifies commands in shell-script
- wrapper starts isolated shell
- wrapper starts modified shellscript
- modified script sends single commands to subshell
- subshell executes commands separately
- wrapper monitors and logs results of commands







results of some test jobs:

job	number of jobs	successful	aborted	not started
Seti@home	110	103	7	0
Distributed.net	100	97	3	0
test jobs	765	718	47	41

- jobs, which are not started, are lost
- all other sources of failures could be identified (not possible without monitoring)
 - Seti@home server in Berkeley not available
 - misconfigured batch systems: job preemptively aborted by PBS
 - successful jobs mistakenly marked as failed because LCG had problems getting the output





reached goals:

- stepwise execution of shell-scripts
- traceback of failures possible
- monitoring of system resources
- much more information about job execution

known restrictions:

parser knows bash- and sh-syntax





work in progress:

- identification of job failures
- classification of errors
- expert system to automatically recover most error conditions

timetable:

- job wrapper could enter LCG/gLite release soon
- other components later (when sufficiently mature)
- first full release planned for Q4/2006

