

Grid and Cloud Computing

Pertemuan 3 : Arsitektur Grid Computing

Alauddin Maulana Hirzan

Fakultas Teknologi Informasi dan Komunikasi
Universitas Semarang

Outline

Grid and
Cloud
Computing

Alauddin
Maulana
Hirzan

Arsitektur
Grid
Computing

Conclusion

1 Arsitektur Grid Computing

2 Conclusion

Komponen Grid Computing

Grid and
Cloud
Computing

Alauddin
Maulana
Hirzan

Arsitektur
Grid
Computing

Conclusion

Untuk membentuk Grid Computing setidaknya diperlukan dua (2) komputer yang masing-masing memiliki role atau peran sebagai **Master/Control Node** dan **Grid Node**

- 1 **Control Node** memiliki tugas untuk mengelola seluruh jaringan dan menyimpan sumber daya dalam jaringan.
- 2 **Grid Node** memiliki tugas untuk menyumbangkan sumber dayanya ke kumpulan sumber daya jaringan.
- 3 **Pengguna** mengacu pada komputer yang menggunakan sumber daya di jaringan untuk menyelesaikan tugas.

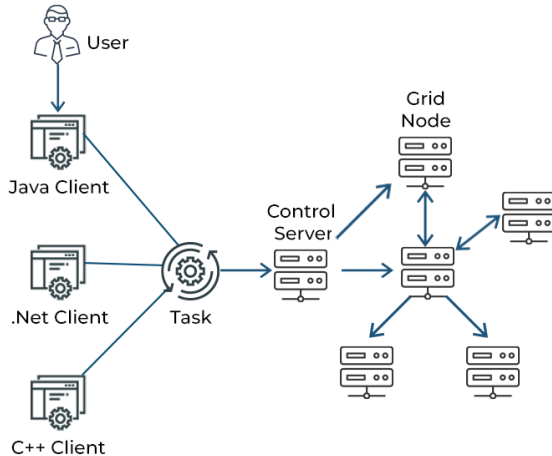
Ilustrasi

Grid and
Cloud
Computing

Alauddin
Maulana
Hirzan

Arsitektur
Grid
Computing

Conclusion



Komponen Grid Computing

Grid and
Cloud
Computing

Alauddin
Maulana
Hirzan

Arsitektur
Grid
Computing

Conclusion

Untuk membentuk lingkungan Grid Computing diperlukan beberapa komponen internal seperti:

- 1 Antarmuka (User Interface)
- 2 Keamanan
- 3 Penjadwalan
- 4 Manajemen Data
- 5 Manajemen Beban Kerja / Sumber Daya

Komponen Antarmuka

Grid and
Cloud
Computing

Alauddin
Maulana
Hirzan

Arsitektur
Grid
Computing

Conclusion

Antarmukanya memiliki gaya portal untuk membantu pengguna melakukan kueri dan menjalankan berbagai fungsi di grid secara efektif. Pengguna grid melihat satu komputer virtual besar yang menawarkan sumber daya komputasi, mirip dengan pengguna internet yang melihat konten terpadu di web.

Platform yang biasa digunakan seperti

- 1 Web Console
- 2 Terminal Console
- 3 Desktop App

Contoh Antarmuka Web

Grid and
Cloud
Computing

Alauddin
Maulana
Hirzan

Arsitektur
Grid
Computing

Conclusion

The screenshot displays the Portainer.io Business Edition dashboard. The interface is dark-themed with a sidebar on the left containing navigation links: Home, DockerGentoo, Dashboard (selected), App Templates, Stacks, Containers, Images, Networks, Volumes, Events, and Host. The main content area is titled 'Environment summary' and 'Dashboard'. It features an 'Environment info' section with details: Environment (DockerGentoo), URL (/var/run/docker.sock), GPU (none), and Tags (-). Below this, there are four summary cards: 0 Stacks, 1 Container (with 1 running, 0 healthy, 0 stopped, and 0 unhealthy status), 3 Images (674.4 MB), and 1 Volume. The bottom left corner shows the copyright notice: © Portainer Business Edition 2.19.1.

Environment summary

Dashboard

Home

DockerGentoo

Dashboard

App Templates

Stacks

Containers

Images

Networks

Volumes

Events

Host

Environment info

Environment	DockerGentoo 8 33 GB - Standalone 24.0.5
URL	/var/run/docker.sock
GPU	none
Tags	-

0 Stacks

1 Container

1 running 0 healthy 0 stopped 0 unhealthy

3 Images 674.4 MB

1 Volume

© Portainer Business Edition 2.19.1

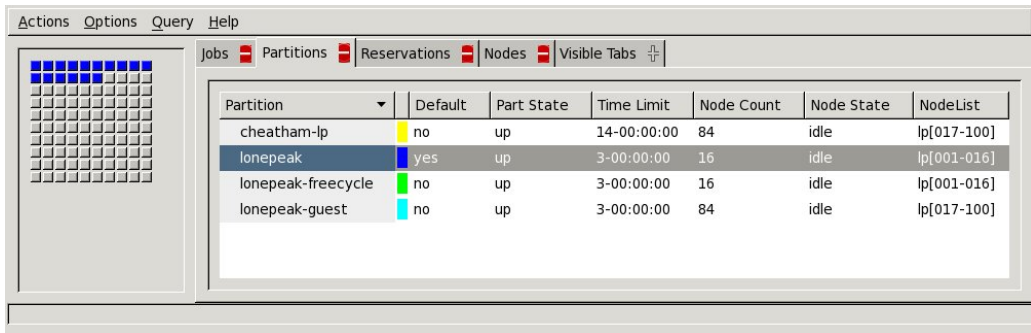
Contoh Antarmuka Desktop

Grid and
Cloud
Computing

Alauddin
Maulana
Hirzan

Arsitektur
Grid
Computing

Conclusion



The screenshot shows a desktop application window with a menu bar containing 'Actions', 'Options', 'Query', and 'Help'. Below the menu bar is a tabbed interface with tabs for 'Jobs', 'Partitions', 'Reservations', 'Nodes', and 'Visible Tabs'. The 'Partitions' tab is currently selected. On the left side of the window is a sidebar containing a grid of small square icons, some of which are blue. The main area of the window displays a table with the following data:

Partition	Default	Part State	Time Limit	Node Count	Node State	NodeList
cheatham-lp	no	up	14-00:00:00	84	idle	lp[017-100]
lonepeak	yes	up	3-00:00:00	16	idle	lp[001-016]
lonepeak-freecycle	no	up	3-00:00:00	16	idle	lp[001-016]
lonepeak-guest	no	up	3-00:00:00	84	idle	lp[017-100]

Contoh Antarmuka Console

Grid and
Cloud
Computing

Alauddin
Maulana
Hirzan

Arsitektur
Grid
Computing

Conclusion

```
[admin@mgmt ~]$ squeue
      JOBID PARTITION    NAME    USER ST       TIME  NODES NODELIST(REASON)
      999      cpu MS_3HCSE    dkn  R 24-19:08:43      1 node06.chess
     1000      cpu MS_3HCUC    dkn  R 24-19:07:33      1 node37.chess
     1001      cpu MS_3HIV9    dkn  R 24-19:07:00      1 node38.chess
     1743      cpu MS_8ERNH    dkn  R 2-17:05:06      1 node45.chess
     1744      cpu MS_8EVRG    dkn  R 2-17:02:48      1 node32.chess
     1745      cpu MS_8EITQ    dkn  R 2-17:01:22      1 node33.chess
     1817      cpu MS_CENBC    dkn  R   13:21:59      1 node43.chess
     1825      gpu label-re    tens R       21:11      1 node02.chess
[admin@mgmt ~]$
```

Komponen Keamanan

Grid and
Cloud
Computing

Alauddin
Maulana
Hirzan

Arsitektur
Grid
Computing

Conclusion

Keamanan adalah salah satu perhatian utama dalam lingkungan komputasi grid. Mekanisme keamanan dapat mencakup otentikasi, otorisasi, enkripsi data, dan lain-lain. Infrastruktur keamanan jaringan (GSI) merupakan unsur penting di sini. Ini menguraikan spesifikasi yang membangun komunikasi rahasia dan anti-rusak antara entitas perangkat lunak yang beroperasi di jaringan grid.

Teknologi Keamanan Grid seperti:

- 1 Otentifikasi
- 2 Delegasi

Komponen Keamanan

Grid and
Cloud
Computing

Alauddin
Maulana
Hirzan

Arsitektur
Grid
Computing

Conclusion

- Otentifikasi
 - Merupakan prosedur standar yang digunakan untuk Grid Computing mengamankan komputernya dari pihak yang tidak bertanggung jawab.
 - User-Password
 - Smart Card
 - Fingerprint
- Memberikan hak akses kepada seseorang sehingga tidak perlu melakukan registrasi secara penuh ke sistem
 - Smart Card

Contoh Keamanan

Grid and
Cloud
Computing

Alauddin
Maulana
Hirzan

Arsitektur
Grid
Computing

Conclusion



Komponen Penjadwalan

Penjadwal mungkin tidak diperlukan jika tugas mandiri ingin dijalankan dan tidak menunjukkan saling ketergantungan. Namun, jika pengguna ingin menjalankan tugas tertentu secara bersamaan yang memerlukan komunikasi antar-proses, penjadwal pekerjaan cukup mengoordinasikan pelaksanaan subtugas yang berbeda.

Standar algoritma:

- ➊ First Come First Serve (FCFS)
- ➋ Priority Scheduling
- ➌ Shortest-Job-First (SJF) Scheduling
- ➍ Round Robin Scheduling
- ➎ Shortest Remaining Time
- ➏ Multilevel Queue Scheduling

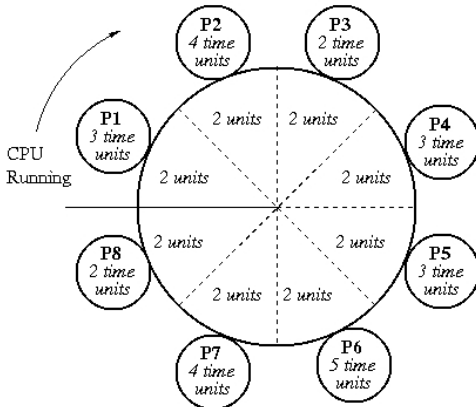
Contoh Round Robin

Grid and
Cloud
Computing

Alauddin
Maulana
Hirzan

Arsitektur
Grid
Computing

Conclusion



Komponen Manajemen Data

Grid and
Cloud
Computing

Alauddin
Maulana
Hirzan

Arsitektur
Grid
Computing

Conclusion

Manajemen data sangat penting untuk lingkungan jaringan. Diperlukan mekanisme yang aman dan andal untuk memindahkan atau membuat data atau modul aplikasi apa pun dapat diakses oleh berbagai node dalam jaringan.

Komponen ini menawarkan komponen manajemen data yang disebut akses jaringan ke penyimpanan sekunder (GASS). Ini mencakup GridFTP yang dibangun pada protokol FTP standar dan menggunakan GSI untuk otentikasi dan otorisasi pengguna. Setelah autentikasi, pengguna dapat memindahkan file menggunakan fasilitas GridFTP tanpa melalui proses login di setiap node.

Komponen Manajemen Data

Grid and
Cloud
Computing

Alauddin
Maulana
Hirzan

Arsitektur
Grid
Computing

Conclusion

- **GridFTP**
 - Grid File Transfer Protocol, adalah perpanjangan dari protokol FTP tradisional yang disesuaikan untuk lingkungan komputasi grid.
- **HTTP dan HTTPS**
 - Hypertext Transfer Protocol (HTTP) dan versi amannya, HTTPS, banyak digunakan untuk berbagi data berbasis web di lingkungan komputasi grid.
- **SRB/iRODS**
 - Storage Resource Broker (SRB) dan Integrated Rule-Oriented Data System (iRODS) adalah sistem manajemen data yang digunakan dalam lingkungan komputasi grid.

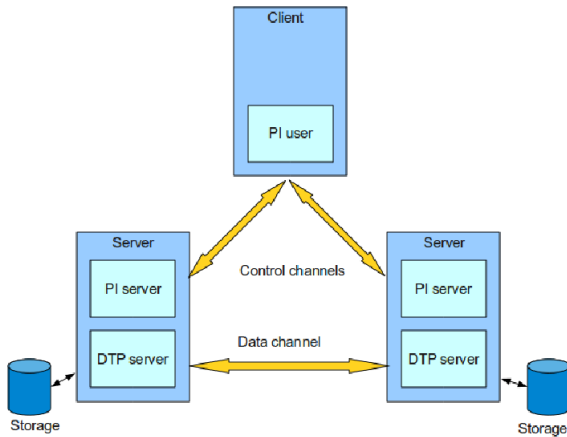
Contoh Manajemen Data

Grid and
Cloud
Computing

Alauddin
Maulana
Hirzan

Arsitektur
Grid
Computing

Conclusion



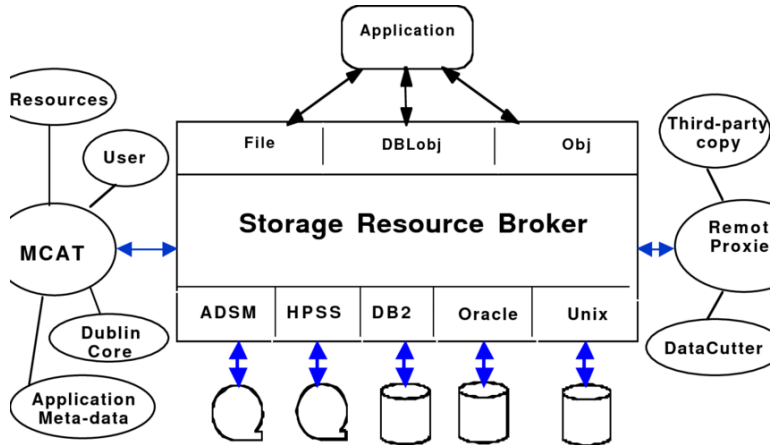
Contoh Manajemen Data

Grid and
Cloud
Computing

Alauddin
Maulana
Hirzan

Arsitektur
Grid
Computing

Conclusion



Komponen Manajemen Beban dan Sumber Daya

Grid and
Cloud
Computing

Alauddin
Maulana
Hirzan

Arsitektur
Grid
Computing

Conclusion

Komponen beban kerja & sumber daya memungkinkan peluncuran sebenarnya suatu pekerjaan pada sumber daya tertentu, memeriksa statusnya, dan mengambil hasilnya ketika pekerjaan selesai. Katakanlah pengguna ingin menjalankan aplikasi di grid. Dalam hal ini, aplikasi harus menyadari sumber daya yang tersedia di jaringan untuk mengambil beban kerja.

Dalam lingkungan komputasi grid, manajemen beban kerja sangat penting untuk mendistribusikan dan mengelola tugas komputasi secara efisien di seluruh jaringan sumber daya yang terdistribusi.

Komponen Manajemen Beban dan Sumber Daya

Grid and
Cloud
Computing

Alauddin
Maulana
Hirzan

Arsitektur
Grid
Computing

Conclusion

- Condor
 - Condor dirancang untuk mengelola dan menjadwalkan pekerjaan pada beragam sumber daya komputasi, termasuk cluster, grid, dan stasiun kerja khusus.
- Sun Grid Engine (SGE)
 - Manajer beban kerja sumber terbuka yang menyediakan alokasi sumber daya dan penjadwalan pekerjaan di lingkungan grid dan cluster.
- Slurm
 - Slurm telah diadaptasi untuk lingkungan komputasi grid dan menawarkan penjadwalan pekerjaan dan manajemen sumber daya.
- Torque
 - Manajer beban kerja sumber terbuka lainnya yang dapat digunakan dalam komputasi grid untuk penjadwalan pekerjaan dan manajemen sumber daya.

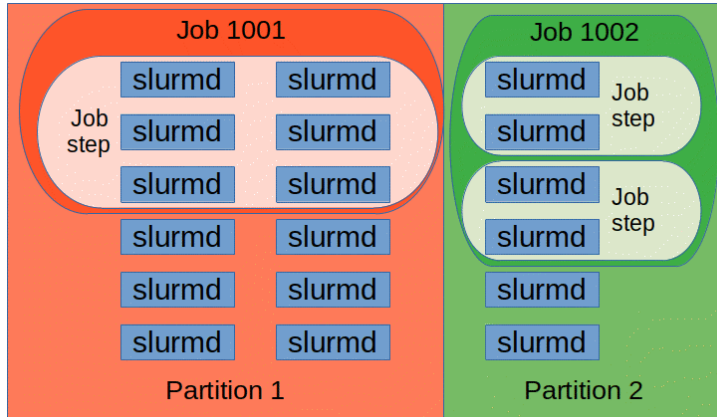
Contoh Manajemen Beban dan Sumber Daya

Grid and
Cloud
Computing

Alauddin
Maulana
Hirzan

Arsitektur
Grid
Computing

Conclusion



Conclusion

Grid and
Cloud
Computing

Alauddin
Maulana
Hirzan

Arsitektur
Grid
Computing

Conclusion

Terima Kasih