

# gLite

eine Grid Distribution der EGEE

Fakultät für Physik

GRID Seminar am 08.11.2007

## 1 Geschichte

- Der Opa, Onkel, Bruder und Gleichgesinnte
- Warum eine Distribution?

## 2 Ein Weg durch gLite

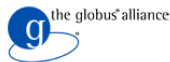
- Authentifizierung
- Verschicken von Jobs
- Monitoring und Abrechnung
- Transfer von Daten

## 3 Wie weiter?

- Probleme
- Was bleibt für die folgenden Seminarwochen?

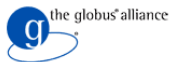
- Globus

- 1994 I-Way (Illinois)
  - 1998 Version 1.0 Toolkit mit zahlreichen Modulen
  - 2006 4.0 auf Basis von open-standard Grid & Web Services
  - Ermöglicht verschiedenste Formen eines Grids
- LHC Computing Grid (LCG) Projekt (CERN)
  - AliEn (ALICE Kollaboration, CERN)
  - EGEE als europäische Initiative



- Globus

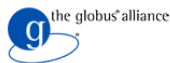
- 1994 I-Way (Illinois)
  - 1998 Version 1.0 Toolkit mit zahlreichen Modulen
  - 2006 4.0 auf Basis von open-standard Grid & Web Services
  - Ermöglicht verschiedenste Formen eines Grids
- LHC Computing Grid (LCG) Projekt (CERN)
- AliEn (ALICE Kollaboration, CERN)
- EGEE als europäische Initiative



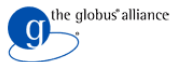
- Globus

- 1994 I-Way (Illinois)
- 1998 Version 1.0 Toolkit mit zahlreichen Modulen
- 2006 4.0 auf Basis von open-standard Grid & Web Services
- Ermöglicht verschiedenste Formen eines Grids

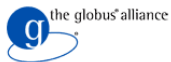
- LHC Computing Grid (LCG) Projekt (CERN)
- AliEn (ALICE Kollaboration, CERN)
- EGEE als europäische Initiative



- Globus
  - 1994 I-Way (Illinois)
  - 1998 Version 1.0 Toolkit mit zahlreichen Modulen
  - 2006 4.0 auf Basis von open-standard Grid & Web Services
    - Ermöglicht verschiedenste Formen eines Grids
- LHC Computing Grid (LCG) Projekt (CERN)
- AliEn (ALICE Kollaboration, CERN)
- EGEE als europäische Initiative



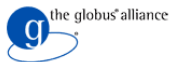
- Globus
  - 1994 I-Way (Illinois)
  - 1998 Version 1.0 Toolkit mit zahlreichen Modulen
  - 2006 4.0 auf Basis von open-standard Grid & Web Services
  - Ermöglicht verschiedenste Formen eines Grids
- LHC Computing Grid (LCG) Projekt (CERN)
- AliEn (ALICE Kollaboration, CERN)
- EGEE als europäische Initiative



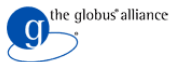
AliEn<sup>2</sup>  
@GRID

EGEE  
Enabling Grids  
for E-science

- Globus
  - 1994 I-Way (Illinois)
  - 1998 Version 1.0 Toolkit mit zahlreichen Modulen
  - 2006 4.0 auf Basis von open-standard Grid & Web Services
  - Ermöglicht verschiedenste Formen eines Grids
- LHC Computing Grid (LCG) Projekt (CERN)
- AliEn (ALICE Kollaboration, CERN)
- EGEE als europäische Initiative



- Globus
  - 1994 I-Way (Illinois)
  - 1998 Version 1.0 Toolkit mit zahlreichen Modulen
  - 2006 4.0 auf Basis von open-standard Grid & Web Services
  - Ermöglicht verschiedenste Formen eines Grids
- LHC Computing Grid (LCG) Projekt (CERN)
- AliEn (ALICE Kollaboration, CERN)
- EGEE als europäische Initiative



- Globus
  - 1994 I-Way (Illinois)
  - 1998 Version 1.0 Toolkit mit zahlreichen Modulen
  - 2006 4.0 auf Basis von open-standard Grid & Web Services
  - Ermöglicht verschiedenste Formen eines Grids
- LHC Computing Grid (LCG) Projekt (CERN)
- AliEn (ALICE Kollaboration, CERN)
- EGEE als europäische Initiative

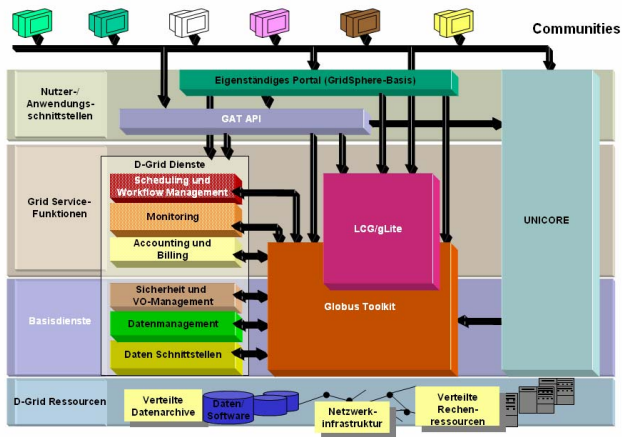


- Grid-Architektur modular
- Zahlreiche, genormte Schnittstellen
- Große Auswahl bei einzelnen Modulen
- Standardplattform innerhalb eines Projekts erstrebenswert

- Grid-Architektur modular
- Zahlreiche, genormte Schnittstellen
- Große Auswahl bei einzelnen Modulen
- Standardplattform innerhalb eines Projekts erstrebenswert

- Grid-Architektur modular
- Zahlreiche, genormte Schnittstellen
- Große Auswahl bei einzelnen Modulen
- Standardplattform innerhalb eines Projekts erstrebenswert

- Grid-Architektur modular
- Zahlreiche, genormte Schnittstellen
- Große Auswahl bei einzelnen Modulen
- Standardplattform innerhalb eines Projekts erstrebenswert



# gLite

- implementiert auf den meisten Ebenen OGF Standards (OGSA - Open Grid Service Architecture)
- basiert auf einer Service orientierten Architektur (SOA)
- besteht im wesentlichen aus 4 größeren Teilen
  - Benutzerauthentifizierung
  - Jobsubmission/ Workload Management
  - Informationsservices (Abrechnung, Logging, Fehlererkennung)
  - Datenhaltung- und übermittlung

# gLite

- implementiert auf den meisten Ebenen OGF Standards (OGSA - Open Grid Service Architecture)
- basiert auf einer Service orientierten Architektur (SOA)
- besteht im wesentlichen aus 4 größeren Teilen
  - Benutzerauthentifizierung
  - Jobsubmission/ Workload Management
  - Informationsservices (Abrechnung, Logging, Fehlererkennung)
  - Datenhaltung- und übermittlung

- implementiert auf den meisten Ebenen OGF Standards (OGSA - Open Grid Service Architecture)
- basiert auf einer Service orientierten Architektur (SOA)
- besteht im wesentlichen aus 4 größeren Teilen
  - Benutzerauthentifizierung
  - Jobsubmission/ Workload Management
  - Informationsservices (Abrechnung, Logging, Fehlererkennung)
  - Datenhaltung- und übermittlung

- implementiert auf den meisten Ebenen OGF Standards (OGSA - Open Grid Service Architecture)
- basiert auf einer Service orientierten Architektur (SOA)
- besteht im wesentlichen aus 4 größeren Teilen
  - Benutzerauthentifizierung
  - Jobsubmission/ Workload Management
  - Informationsservices (Abrechnung, Logging, Fehlererkennung)
  - Datenhaltung- und übermittlung

- implementiert auf den meisten Ebenen OGF Standards (OGSA - Open Grid Service Architecture)
- basiert auf einer Service orientierten Architektur (SOA)
- besteht im wesentlichen aus 4 größeren Teilen
  - Benutzerauthentifizierung
  - **Jobsubmission/ Workload Management**
  - Informationsservices (Abrechnung, Logging, Fehlererkennung)
  - Datenhaltung- und übermittlung

- implementiert auf den meisten Ebenen OGF Standards (OGSA - Open Grid Service Architecture)
- basiert auf einer Service orientierten Architektur (SOA)
- besteht im wesentlichen aus 4 größeren Teilen
  - Benutzerauthentifizierung
  - Jobsubmission/ Workload Management
  - Informationsservices (Abrechnung, Logging, Fehlererkennung)
  - Datenhaltung- und übermittlung

- implementiert auf den meisten Ebenen OGF Standards (OGSA - Open Grid Service Architecture)
- basiert auf einer Service orientierten Architektur (SOA)
- besteht im wesentlichen aus 4 größeren Teilen
  - Benutzerauthentifizierung
  - Jobsubmission/ Workload Management
  - Informationsservices (Abrechnung, Logging, Fehlererkennung)
  - Datenhaltung- und übermittlung

# Was sind eigentlich... Webservices?

- Buzzword des w3c
- Bezeichnet das Anbieten von Computerdiensten über das Internet
- Beispiele:
  - Google Search API
  - Flickr API
  - Reuters Finanzdaten
- z.B. als Entfernter Funktionsaufruf (RPC, Remote Procedure Call)
- auf der Basis von genormten Protokollen (auf XML Basis)
- meist via http (SOAP, WS-I), aber auch smtp, xmpp (jabber)

# Was sind eigentlich... Webservices?

- Buzzword des w3c
- Bezeichnet das Anbieten von Computerdiensten über das Internet
- Beispiele:
  - Google Search API
  - Flickr API
  - Reuters Finanzdaten
- z.B. als Entfernter Funktionsaufruf (RPC, Remote Procedure Call)
- auf der Basis von genormten Protokollen (auf XML Basis)
- meist via http (SOAP, WS-I), aber auch smtp, xmpp (jabber)

# Was sind eigentlich... Webservicees?

- Buzzword des w3c
- Bezeichnet das Anbieten von Computerdiensten über das Internet
- Beispiele:
  - Google Search API
  - Flickr API
  - Reuters Finanzdaten
- z.B. als Entfernter Funktionsaufruf (RPC, Remote Procedure Call)
- auf der Basis von genormten Protokollen (auf XML Basis)
- meist via http (SOAP, WS-I), aber auch smtp, xmpp (jabber)

# Was sind eigentlich... Webservicees?

- Buzzword des w3c
- Bezeichnet das Anbieten von Computerdiensten über das Internet
- Beispiele:
  - Google Search API
  - Flickr API
  - Reuters Finanzdaten
- z.B. als Entfernter Funktionsaufruf (RPC, Remote Procedure Call)
- auf der Basis von genormten Protokollen (auf XML Basis)
- meist via http (SOAP, WS-I), aber auch smtp, xmpp (jabber)

# Was sind eigentlich... Webservice?

- Buzzword des w3c
- Bezeichnet das Anbieten von Computerdiensten über das Internet
- Beispiele:
  - Google Search API
  - Flickr API
  - Reuters Finanzdaten
- z.B. als Entfernter Funktionsaufruf (RPC, Remote Procedure Call)
- auf der Basis von genormten Protokollen (auf XML Basis)
- meist via http (SOAP, WS-I), aber auch smtp, xmpp (jabber)

# Was sind eigentlich... Webservice?

- Buzzword des w3c
- Bezeichnet das Anbieten von Computerdiensten über das Internet
- Beispiele:
  - Google Search API
  - Flickr API
  - Reuters Finanzdaten
- z.B. als Entfernter Funktionsaufruf (RPC, Remote Procedure Call)
- auf der Basis von genormten Protokollen (auf XML Basis)
- meist via http (SOAP, WS-I), aber auch smtp, xmpp (jabber)

# Was sind eigentlich... Webservicees?

- Buzzword des w3c
- Bezeichnet das Anbieten von Computerdiensten über das Internet
- Beispiele:
  - Google Search API
  - Flickr API
  - Reuters Finanzdaten
- z.B. als Entfernter Funktionsaufruf (RPC, Remote Procedure Call)
- auf der Basis von genormten Protokollen (auf XML Basis)
- meist via http (SOAP, WS-I), aber auch smtp, xmpp (jabber)

# Was sind eigentlich... Webservices?

- Buzzword des w3c
- Bezeichnet das Anbieten von Computerdiensten über das Internet
- Beispiele:
  - Google Search API
  - Flickr API
  - Reuters Finanzdaten
- z.B. als Entfernter Funktionsaufruf (RPC, Remote Procedure Call)
- auf der Basis von genormten Protokollen (auf XML Basis)
- meist via http (SOAP, WS-I), aber auch smtp, xmpp (jabber)

# Was sind eigentlich... Webservice?

- Buzzword des w3c
- Bezeichnet das Anbieten von Computerdiensten über das Internet
- Beispiele:
  - Google Search API
  - Flickr API
  - Reuters Finanzdaten
- z.B. als Entfernter Funktionsaufruf (RPC, Remote Procedure Call)
- auf der Basis von genormten Protokollen (auf XML Basis)
- meist via http (SOAP, WS-I), aber auch smtp, xmpp (jabber)

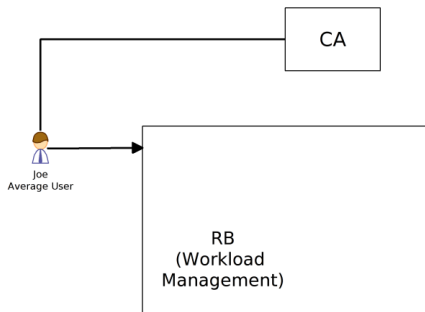
# Was sind eigentlich... Webservices?

```
1 <s:Envelope
2 xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
3 xmlns:xsi="http://www.w3.org/1999/XMLSchema-
4 instance"
5 xmlns:xsd="http://www.w3.org/1999/XMLSchema">
6 <s:Body>
7     <m:sayHello xmlns:m='urn:Example1'>
8         <name xsi:type='xsd:string'>Coyote</name>
9     </m:sayHello>
10 </s:Body>
11 </s:Envelope>
12
```

Beispiel aus: *James Snell, Doug Tidwell & Pavel Kulchenko: Programming Web Services with SOAP; Oreilly 2001*

# Was sind eigentlich... Webservice?

```
1 <s:Envelope
2 xmlns:s="http://www.w3.org/2001/06/soap-envelope"
3 xmlns:xsi="http://www.w3.org/1999/XMLSchema-
4 instance"
5 xmlns:xsd="http://www.w3.org/1999/XMLSchema">
6 <s:Body>
7   <n:sayHelloResponse xmlns:n="urn:Example1">
8     <return xsi:type="xsd:string">
9       Hello Coyote
10    </return>
11   </n:sayHelloResponse>
12 </s:Body>
13 </s:Envelope>
14
```



- Jeder Benutzer hat einen persönlichen Schlüssel
- Ausgestellt von einer Autorität (CA)
- Auf Basis von OpenSSL
- Bei jeder Anmeldung ein temporäres ( $\approx 12\text{h}$ ) Ticket (*Proxy*)
- Benutzer weist sich an versch. Stellen mit dem Ticket aus

```
~> ls -la ~/.globus
-rw-r--r--    1 moh moh  523 2007-11-03 16:42 usercert.pem
-rw-----    1 moh moh   42 2007-11-03 16:43 userkey.pem
~> voms-proxy-init -voms acme
```

- Jeder Benutzer hat einen persönlichen Schlüssel
- Ausgestellt von einer Autorität (CA)
- Auf Basis von OpenSSL
- Bei jeder Anmeldung ein temporäres ( $\approx 12\text{h}$ ) Ticket (*Proxy*)
- Benutzer weist sich an versch. Stellen mit dem Ticket aus

```
~> ls -la ~/.globus
-rw-r--r--    1 moh moh  523 2007-11-03 16:42 usercert.pem
-rw-----    1 moh moh   42 2007-11-03 16:43 userkey.pem
~> voms-proxy-init -voms acme
```

# Das WorkloadManagementSystem WMS

- Absenden des Auftrags (*jobs*) an den zentralen Ressource Broker (RB)
- Beschreibung von Aufträgen via Job Description Language (JDL)
- Möglichkeit von Statusabfrage, Killen, Resubmission...
- Userschnittstelle als CLI, GUI, Browser oder API (C(++), Java, Python)
- Schnittstelle im Hintergrund (neu): SOAP+WSDL (mit Apache als Backend)

# Das WorkloadManagementSystem WMS

- Absenden des Auftrags (*jobs*) an den zentralen Ressource Broker (RB)
- Beschreibung von Aufträgen via Job Description Language (JDL)
- Möglichkeit von Statusabfrage, Killen, Resubmission...
- Userschnittstelle als CLI, GUI, Browser oder API (C(++), Java, Python)
- Schnittstelle im Hintergrund (neu): SOAP+WSDL (mit Apache als Backend)

# Das WorkloadManagementSystem WMS

- Absenden des Auftrags (*jobs*) an den zentralen Ressource Broker (RB)
- Beschreibung von Aufträgen via Job Description Language (JDL)
- Möglichkeit von Statusabfrage, Killen, Resubmission...
- Userschnittstelle als CLI, GUI, Browser oder API (C(++), Java, Python)
- Schnittstelle im Hintergrund (neu): SOAP+WSDL (mit Apache als Backend)

# Das WorkloadManagementSystem WMS

- Absenden des Auftrags (*jobs*) an den zentralen Ressource Broker (RB)
- Beschreibung von Aufträgen via Job Description Language (JDL)
- Möglichkeit von Statusabfrage, Killen, Resubmission...
- Userschnittstelle als CLI, GUI, Browser oder API (C(++), Java, Python)
- Schnittstelle im Hintergrund (neu): SOAP+WSDL (mit Apache als Backend)

# Das WorkloadManagementSystem WMS

- Absenden des Auftrags (*jobs*) an den zentralen Ressource Broker (RB)
- Beschreibung von Aufträgen via Job Description Language (JDL)
- Möglichkeit von Statusabfrage, Killen, Resubmission...
- Userschnittstelle als CLI, GUI, Browser oder API (C(++), Java, Python)
- Schnittstelle im Hintergrund (neu): SOAP+WSDL (mit Apache als Backend)

```
1 VirtualOrganisation = "acme";
2 Executable = "mandelbrot.sh";
3 Comment = "My first submitted job";
4
5 StdOutput = "mbrot.out";
6 StdError = "mbrot.err";
7
8 InputSandbox = {"mandelbrot.sh","inputdata.txt","worker.py"};
9 OutputSandbox = {"mbrot.out", "mbrot.err"};
10
11 RetryCount = 2;
12 ShallowRetryCount = 10;
13
14 Requirements = Member( "VO-acme-production-0.0.1",
15 other.GlueHostApplicationSoftwareRunTimeEnvironment)
16 && other.GlueCEPolicyMaxCPUtime > 1000
17 && other.GlueCEStateWaitingJobs < 20 ;
```

```
~> glite-wms-job-submit -a mandelbrot.jdl
```

```
1 VirtualOrganisation = "acme";
2 Executable = "mandelbrot.sh";
3 Comment = "My first submitted job";
4
5 StdOutput = "mbrot.out";
6 StdError = "mbrot.err";
7
8 InputSandbox = {"mandelbrot.sh","inputdata.txt","worker.py"};
9 OutputSandbox = {"mbrot.out", "mbrot.err"};
10
11 RetryCount = 2;
12 ShallowRetryCount = 10;
13
14 Requirements = Member( "VO-acme-production-0.0.1",
15 other.GlueHostApplicationSoftwareRunTimeEnvironment)
16 && other.GlueCEPolicyMaxCPUtime > 1000
17 && other.GlueCEStateWaitingJobs < 20 ;
```

```
~> glite-wms-job-submit -a mandelbrot.jdl
```

GANGA

Job Folders View Help

Scriptor Log Job Builder

New... Save... Kill Remove Extras Submit

Job 66

Job attributes

- application (Athena)
- backend (LCG)
  - CE
  - locache
  - jobtype
  - middleware
- requirements (LCGRequirements)
- info (JobInfo)
- inputdata (None)
- inputsandbox
- merger (None)
- name
- outputdata (None)
- outputsandbox

Job application: Athena

Jobs

id	status	name	application	exe filename	backend	backend id
12	completed		Executable	File	LCG	https://b110.cern.ch:9000/aPSeA
13	completed	Ex_3_2_1	Athena	"N/A"	LCG	https://b110.cern.ch:9000/3iQUfN
16	completed	Ex_3_2_1	Athena	"N/A"	LCG	https://b110.cern.ch:9000/uMULA
17	completed		Executable	gangatest.sh	LCG	https://b103.cern.ch:9000/1BsofE
24	failed	Own	Athena	"N/A"	LCG	https://b103.cern.ch:9000/mb_wB
36	new	Own	Athena	"N/A"	LCG	None
37	failed	Own	Athena	"N/A"	LCG	https://b103.cern.ch:9000/73TU_L
38	failed	Own	Athena	"N/A"	LCG	https://b103.cern.ch:9000/mV3KUI
47	failed	Own	Athena	"N/A"	LCG	https://b103.cern.ch:9000/MPFUTe
48	failed	Own	Athena	"N/A"	LCG	https://b103.cern.ch:9000/3APDm
50	failed	Own	Athena	"N/A"	LCG	https://b103.cern.ch:9000/03Mj6k
51	failed	Own	Athena	"N/A"	LCG	https://b103.cern.ch:9000/cgg3ve
52	failed	Own	Athena	"N/A"	LCG	https://b103.cern.ch:9000/CHvvg
53	failed	Own	Athena	"N/A"	LCG	https://b103.cern.ch:9000/WNvGE
54	failed	Own	Athena	"N/A"	LCG	https://b103.cern.ch:9000/n_XUa
55	new	Own	Athena	"N/A"	LCG	None
56	new	Own	Athena	"N/A"	Local	None

Job Details

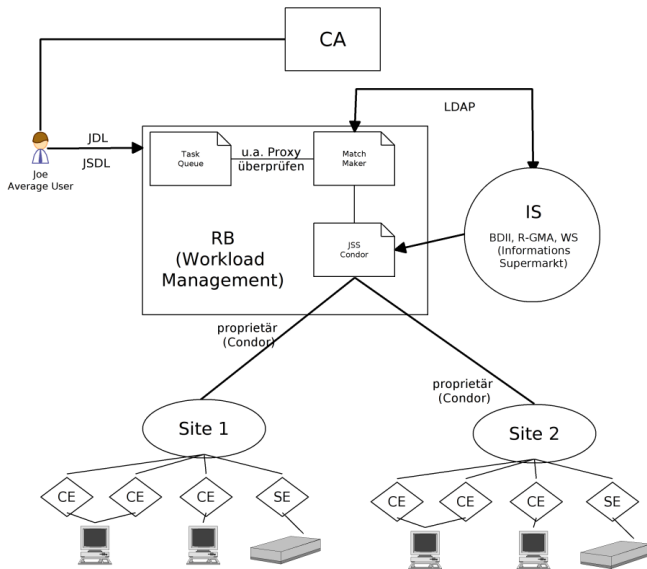
```

subjobs = []
backend = LCG (
  status = 'Done (Success)',
  actualCE = 'dgc-grid-44.brunel.ac.uk:2119/jobmanager-lcgpbs-atlas',
  locache = '',
  middleware = 'GLITE',
  CE = None,
  reason = 'Job terminated successfully',
  id = 'https://b110.cern.ch:9000/uMULAAeRi-wPwJ0Ddp_Vw',
  jobtype = 'Normal',
  exitcode = '0',
  requirements = LCGRequirements (
    nodenumber = 1,
    ipconnectivity = False,
    cputime = None,
    other = [],
    memory = None,
    software = [],
    walltime = None
  )
)
  
```

Job 62 status changed to "failed"

operator [] is deprecated, use {} instead, e.g. jobs(id)  
operator [] is deprecated, use {} instead, e.g. jobs(id)  
operator [] is deprecated, use {} instead, e.g. jobs(id)  
operator [] is deprecated, use {} instead, e.g. jobs(id)  
operator [] is deprecated, use {} instead, e.g. jobs(id)

VERSITÄT  
TTINGEN



# Information Supermarket

- Zentraler Informationsservice
- Kommunikation über BDII (ldap, in use), R-GMA (sql), SOAP (XML/ WSRF)
- Push-Prinzip - Die Computing Elements (CE) informieren den IS
- Informationsbeschreibung nach dem GLUE-Schema

# Information Supermarket

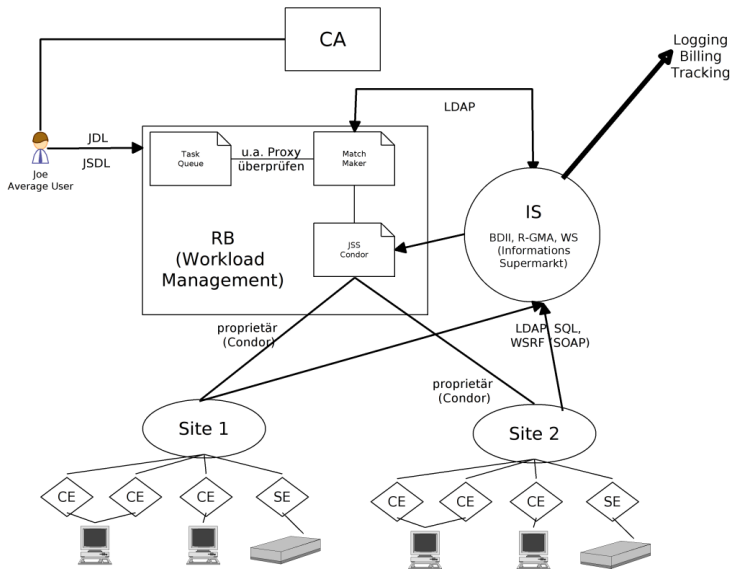
- Zentraler Informationsservice
- Kommunikation über BDII (ldap, in use), R-GMA (sql), SOAP (XML/ WSRF)
- Push-Prinzip - Die Computing Elements (CE) informieren den IS
- Informationsbeschreibung nach dem GLUE-Schema

# Information Supermarket

- Zentraler Informationsservice
- Kommunikation über BDII (ldap, in use), R-GMA (sql), SOAP (XML/ WSRF)
- Push-Prinzip - Die Computing Elements (CE) informieren den IS
- Informationsbeschreibung nach dem GLUE-Schema

# Information Supermarket

- Zentraler Informationsservice
- Kommunikation über BDII (ldap, in use), R-GMA (sql), SOAP (XML/ WSRF)
- Push-Prinzip - Die Computing Elements (CE) informieren den IS
- Informationsbeschreibung nach dem GLUE-Schema

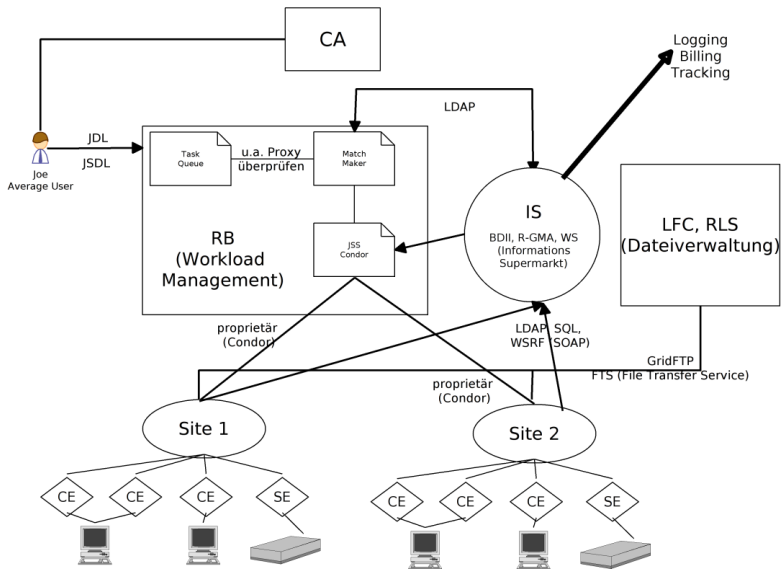


```
_____ mandelbrot.sh _____  
1 export LFC_HOST='lcg-infosites --vo acme lfc'  
2 python worker.py inputdata.txt  
3 lcg-cr --vo acme -v -n 16 file://$PWD/result.txt \  
4 -l lfn:/grid/acme/users/rrunner/'date +%Y-%m-%d' /result.txt
```

- Jede Datei hat eine GUID (global unique id)
- und beliebig viele LFN (logical file name)
- Ein zentraler Server (*LFC - Logical File Catalog*) kennt die Zuordnung von LFN zu GUID und von GUID zur *Site URL* (physikalische Adresse(n))
- Datenübertragung via GridFTP

```
_____ mandelbrot.sh _____  
1 export LFC_HOST='lcg-infosites --vo acme lfc'  
2 python worker.py inputdata.txt  
3 lcg-cr --vo acme -v -n 16 file://$PWD/result.txt \  
4 -l lfn:/grid/acme/users/rrunner/'date +%Y-%m-%d' /result.txt
```

- Jede Datei hat eine GUID (global unique id)
- und beliebig viele LFN (logical file name)
- Ein zentraler Server (*LFC - Logical File Catalog*) kennt die Zuordnung von LFN zu GUID und von GUID zur *Site URL* (physikalische Adresse(n))
- Datenübertragung via GridFTP



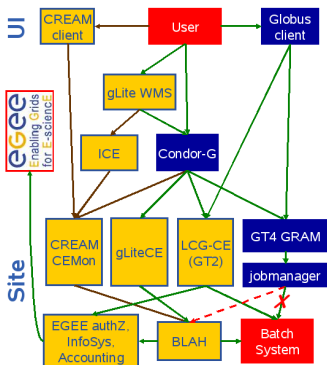
- gLite ist speziell auf ein System optimiert (Scientific Linux)
- Selbst mit Optimierungen scheint es nicht ohne großen Aufwand auf nicht-Unix Systeme portierbar
- Die Unübersichtlichkeit wächst auch bei gLite an

- gLite ist speziell auf ein System optimiert (Scientific Linux)
- Selbst mit Optimierungen scheint es nicht ohne großen Aufwand auf nicht-Unix Systeme portierbar
- Die Unübersichtlichkeit wächst auch bei gLite an

- gLite ist speziell auf ein System optimiert (Scientific Linux)
- Selbst mit Optimierungen scheint es nicht ohne großen Aufwand auf nicht-Unix Systeme portierbar
- Die Unübersichtlichkeit wächst auch bei gLite an

- LCG-CE (GT2 GRAM)
  - Not ported to GT4. To be dismissed
- gLite-CE (Condor-C+GSI)
  - Deployed (GT2 version) but still needs tuning
- CREAM (WS-I)
  - Prototype. OGF-BES (see demo at SC'06)
- Possible developments:
  - GT4 → BLAH submissions?

Choose your preferred path to the Batch System!



# Fragen

?

# Was bleibt für die folgenden Seminarwochen?

- modulare Grid-Architektur
- offene, dokumentierte Schnittstellen
- austauschbare Module

# Was bleibt für die folgenden Seminarwochen?

- modulare Grid-Architektur
- offene, dokumentierte Schnittstellen
- austauschbare Module

# Was bleibt für die folgenden Seminarwochen?

- modulare Grid-Architektur
- offene, dokumentierte Schnittstellen
- austauschbare Module