Seminar „Cluster- und Grid-Computing“
Universität Karlsruhe (TH)
SS 2002

European Data Grid

Christian Koch

Überblick

- Allgemeines (Finanzierung, Ziele, ...)
- Anwendungen/Anforderungen
- Aufbau/Komponenten
- offene Fragen

European Data Grid

- Projekt der EU
- Dauer: 3 Jahre (2001-2003)
- 200 Mitarbeiter
- 9,8 Millionen EURO (Europäische Union)
- Leitung: CERN (Conseil Européen pour la Recherche Nucléaire)

Ziele

- weltweites wissenschaftliches Forschen
- verteiltes Rechnen
- verteilte Datenspeicherung (mehrere PetaByte)

  (1 PB = 1.000 TB = 1.000.000 GB)
- Entwicklung von middleware und Testanwendungen
  - Skalierbarkeit !!!

„weiches“ Ziel:

Zugriff auf Daten- und Rechenressourcen für „jedermann“

Anwendungen

Hochenergie-Physik (HEP) am CERN

- Teilchenbeschleuniger mit 27km Umfang
- bis zu 11.000 Umrundungen pro Sekunde
- 2500 Mitarbeiter
- Umrüstung des Beschleunigers (bis 2005)
- ab 2005: fast 10 PetaByte Rohdaten / Jahr (=10.000 TB = 10 Mio. GB)
- benötigte Rechenkapazität: 10^8 SPECint 2000 Punkte

aktueller PC ~ 550 SPECint 2000 ⟶ 180.000 PC‘s nötig

Anwendungen

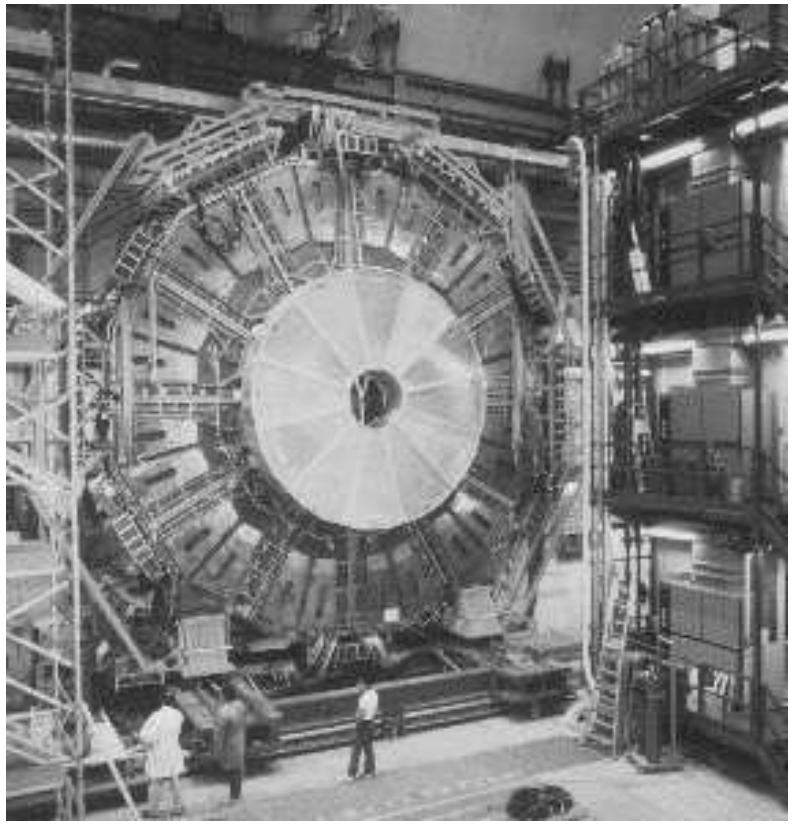
Hochenergie-Physik (HEP) am CERN

Besondere Anforderungen:

- ~ 7500 Benutzer weltweit
- teilweise hohe Parallelität in den Berechnungen
- dynamisches Verteilen der Daten

Anwendungen

Hochenergie-Physik (HEP) am CERN



Anwendungen

Erdbeobachtung/Klimaforschung

- Erdoberfläche
- Atmosphäre (Ozon,...)
- Wettervorhersage

Besondere Anforderungen:

- einheitlicher Zugriff auf große, *verteilte* Datenmengen (PetaByte-Bereich)
- Auswertung großer Menge von Rohdaten (z.B. 4,3 GB/Tag von ENVISAT)
- Wettervorhersage (alle 6 Stunden): 285 MB Eingabe, 1,7 GB Ausgabe

Anwendungen

Biologie

- Genom-Bestimmung und -Auswertung
- Speicherung von Patientendaten, -bildern

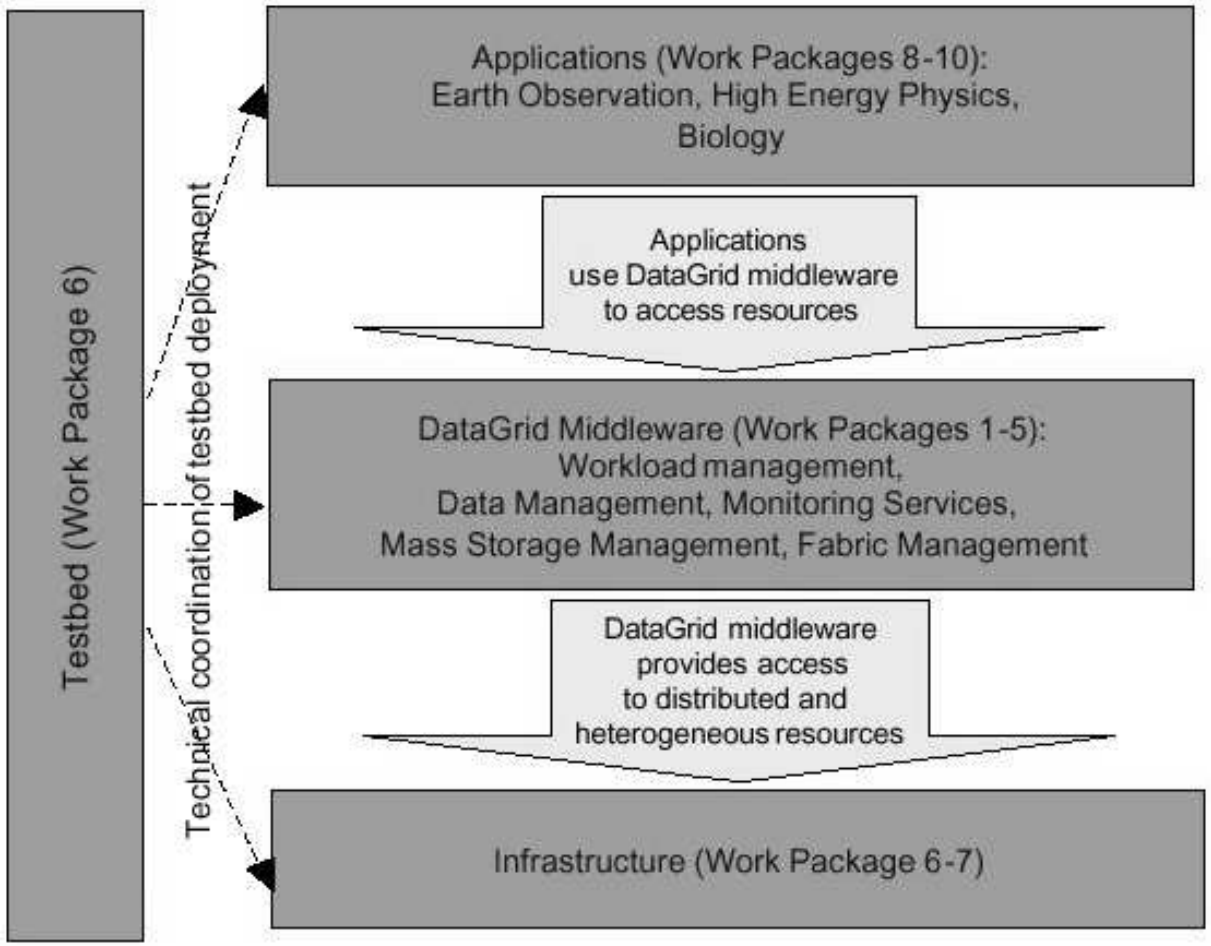
Besondere Anforderungen:

- häufiges Update der Daten (neue Abhängigkeiten bei Genombestimmung)
- Wahrung der Privatsphäre (→ Zugriffsrechte, Gruppen)
- Benutzung durch „Laien“

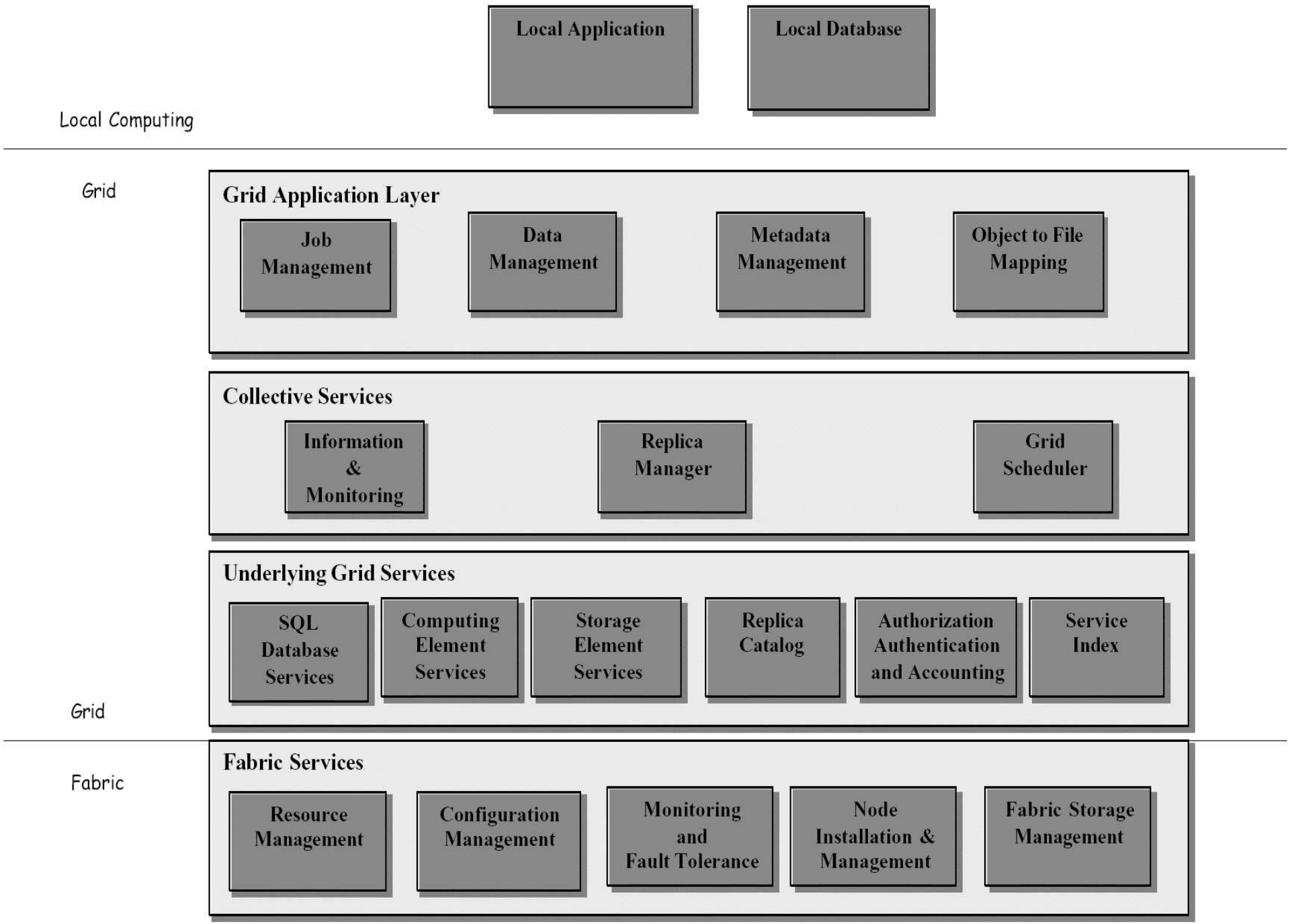
Entwicklung

- modularer Aufbau (Austauschbarkeit)
- open source
- (de-facto) Standards
- Public Key Infrastruktur (X.509)
- Einsatz vorhandener Grid-Technologien:
  - Globus
  - GriPhyN (Grid Physics Network)
  - PPDG (Particle Physics Data Grid)

12 Work Packages



Organisation of the technical work packages in the DataGrid project


Local Application
Local Database
Local Computing
Grid
Grid Application Layer
Job Management
Data Management
Metadata Management
Object to File Mapping
Collective Services
Information & Monitoring
Replica Manager
Grid Scheduler
Underlying Grid Services
SQL Database Services
Computing Element Services
Storage Element Services
Replica Catalog
Authorization Authentication and Accounting
Service Index
Grid
Fabric
Fabric Services
Resource Management
Configuration Management
Monitoring and Fault Tolerance
Node Installation & Management
Fabric Storage Management

Fabric Services

Ziel: „Bereitstellung der Programme, die zur Verwaltung eines Systems mit Clustern von mehreren Tausend Knoten nötig sind.“ *(German Cancio)*

- (Local) Resource Management:
  - Ressourcenverteilung im Teilsystem
  - Grid Scheduler „im Kleinen“

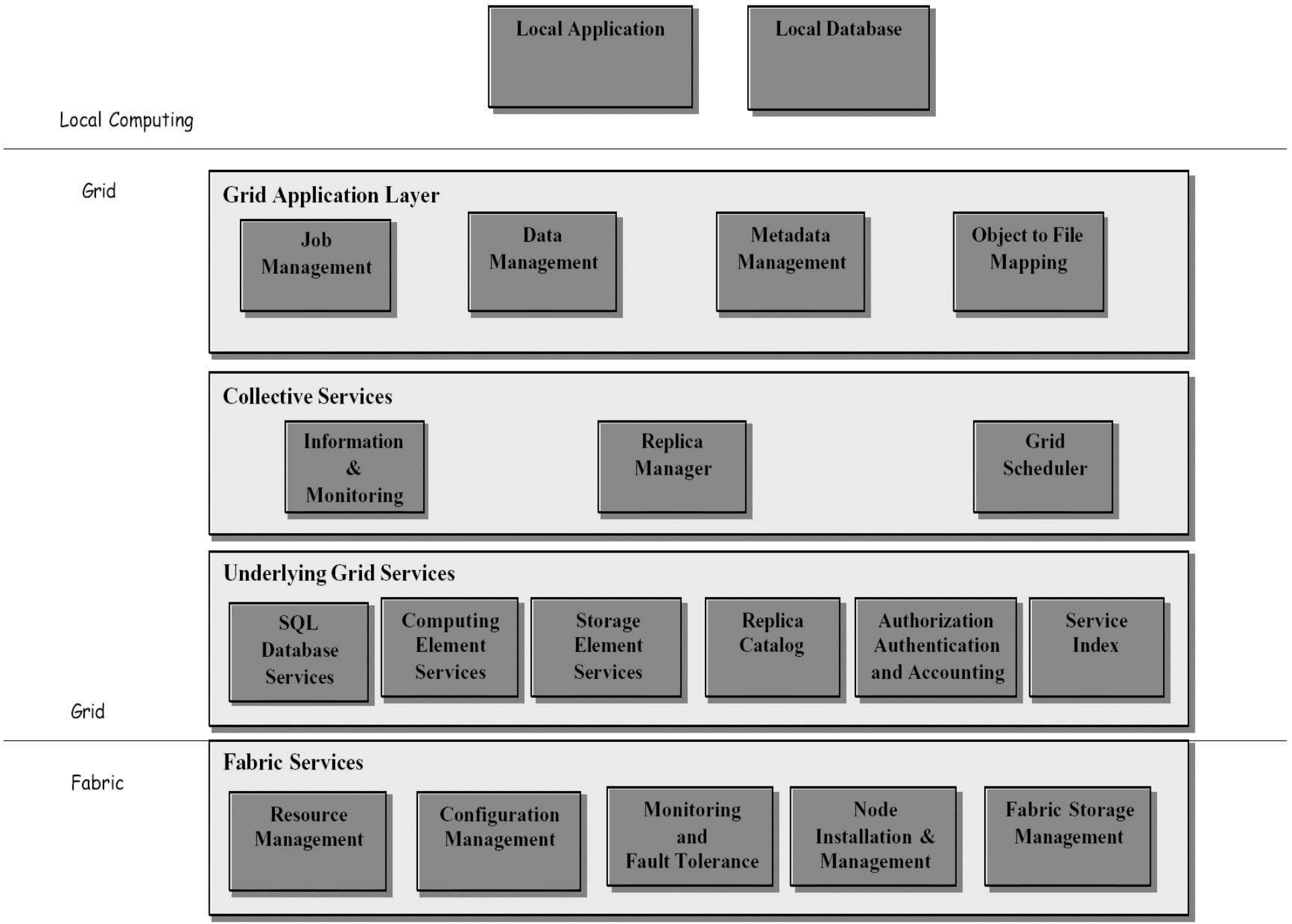
Fabric Services

- Configuration Management:
  - Speichern und Bereitstellen der Struktur eines Teilsystems
    - Hardwareausstattung
    - Konfiguration
    - Dienste
- Monitoring and Fault Tolerance:
  - Leistungsüberwachung und -aufzeichnung
  - Fehlererkennung und -behebung

Fabric Services

- Installation and Management:
  - Installation der Software
  - Konfiguration
  - Update

- Fabric Storage Management:
  - interne Datenspeicherung
  - ist Schnittstelle zu Grid Services


Local Application
Local Database
Local Computing
Grid
Grid Application Layer
Job Management
Data Management
Metadata Management
Object to File Mapping
Collective Services
Information & Monitoring
Replica Manager
Grid Scheduler
Underlying Grid Services
SQL Database Services
Computing Element Services
Storage Element Services
Replica Catalog
Authorization Authentication and Accounting
Service Index
Grid
Fabric
Fabric Services
Resource Management
Configuration Management
Monitoring and Fault Tolerance
Node Installation & Management
Fabric Storage Management

Underlying Grid Services

- Bereitstellen einer **Schnittstelle** zwischen Grid & Rechenzentren...
- ... und grundlegende Grid-Funktionen

- SQL Database Service:
  - Metadaten
  - Schnittstelle zu lokalen oder entfernten Datenbanken
    - insert(), delete(), update(), query()

Underlying Grid Services

- Computing Element Service:
  - Schnittstelle zwischen Grid und Teilsystem
  - Verwaltung, Entgegennahme der Arbeitsaufträge

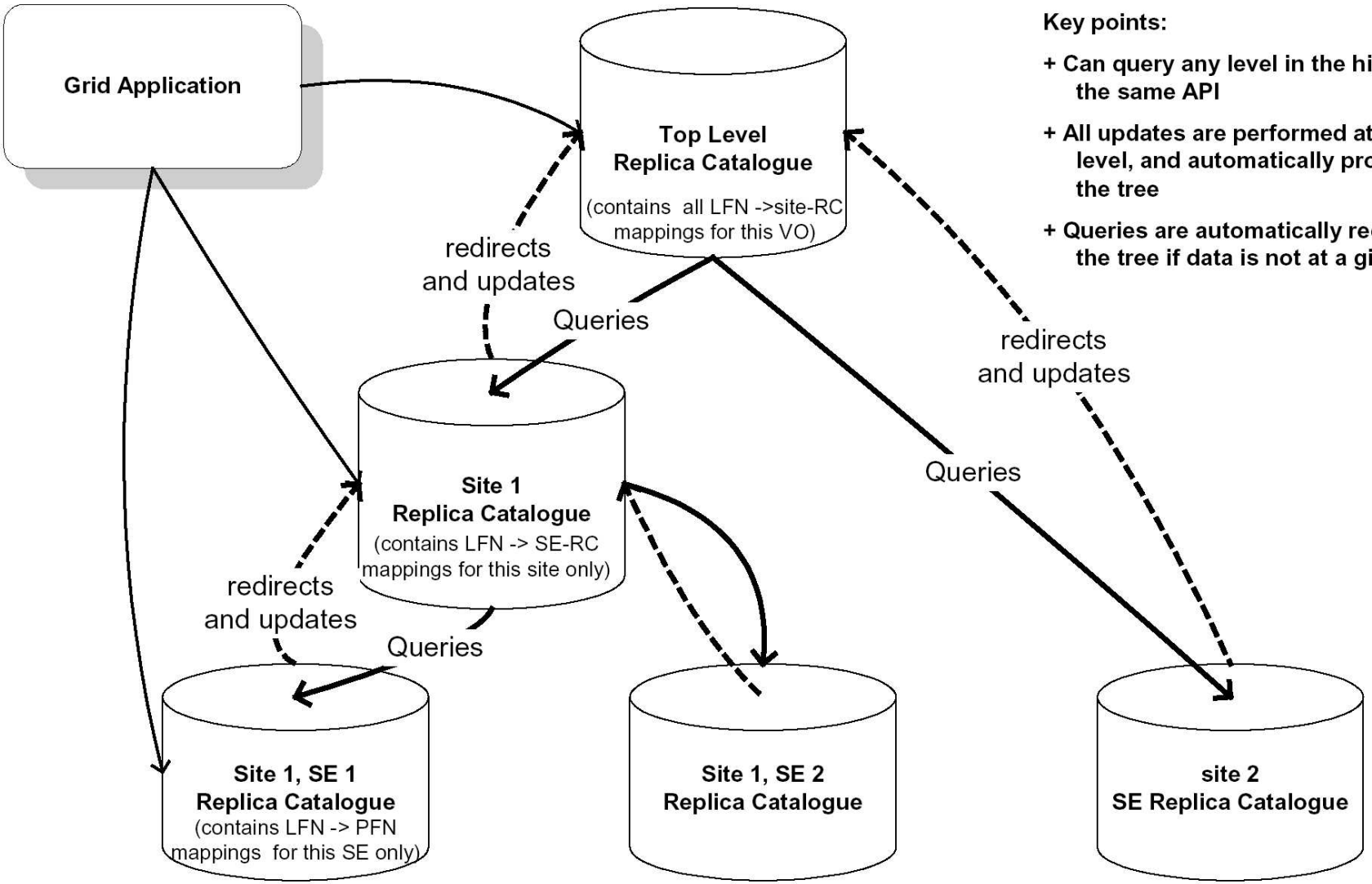
- Storage Element Service:
  - ähnlich dem Computing Element Service
  - Verwaltet Speicherplatz
  - Schnittstelle zum Dateizugriff, z.B. per GridFTP

Underlying Grid Services

- Replica Catalog:
  - Zuordnung logischer Dateiname → physischer Dateiname

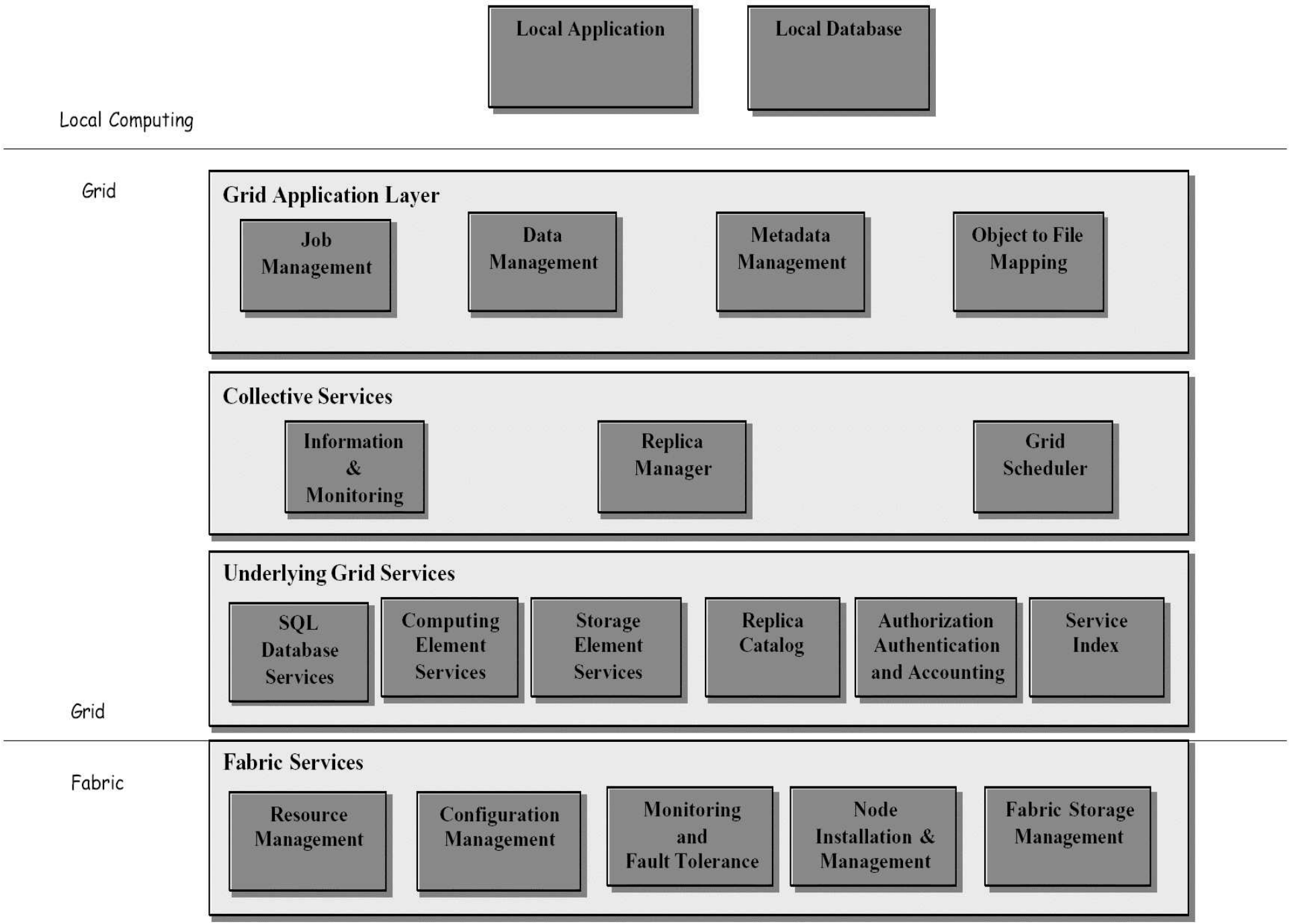
    1 : n
  - Informationen über Dateien (Größe, Datum, ...)
  - Zentrales System:

    + keine Synchronisation nötig - nicht skalierbar
  - daher: Schichtenmodell


Grid Application
Top Level
Replica Catalogue
(contains all LFN ->site-RC
mappings for this VO)
redirects
and updates
Queries
redirects
and updates
Queries
Site 1
Replica Catalogue
(contains LFN -> SE-RC
mappings for this site only)
redirects
and updates
Queries
Site 1, SE 1
Replica Catalogue
(contains LFN -> PFN
mappings for this SE only)
Site 1, SE 2
Replica Catalogue
site 2
SE Replica Catalogue
Key points:
+ Can query any level in the hierarchy with the same API
+ All updates are performed at the SE-RC level, and automatically propagated up the tree
+ Queries are automatically redirected up the tree if data is not at a given level

Underlying Grid Services

- Authorisation, Authentication, Accounting:
  - Public Key Infrastruktur
  - single sign-on
  - Bezahlung?
- Service Index:
  - Informationen über die Funktionen des Grids
    Wo gibt es welchen Service?
  - benutzt seinerseits GRID-Funktionen (SQL) zur Speicherung


Local Application
Local Database
Local Computing
Grid
Grid Application Layer
Job Management
Data Management
Metadata Management
Object to File Mapping
Collective Services
Information & Monitoring
Replica Manager
Grid Scheduler
Underlying Grid Services
SQL Database Services
Computing Element Services
Storage Element Services
Replica Catalog
Authorization Authentication and Accounting
Service Index
Grid
Fabric
Fabric Services
Resource Management
Configuration Management
Monitoring and Fault Tolerance
Node Installation & Management
Fabric Storage Management

Collective Services

- Information and Monitoring:
  - allgemeines Verwalten von Informationen (durch ein "Informationsregister")

  dadurch:
  - Auslastung, freie Ressourcen, ...
  - Wo läuft Job x ?

- Replica Manager:
  - Entscheidet **wo** eine Datei physisch gespeichert wird (Duplikate, Effizienz)
  - Kommunikation mit dem Replica Catalog

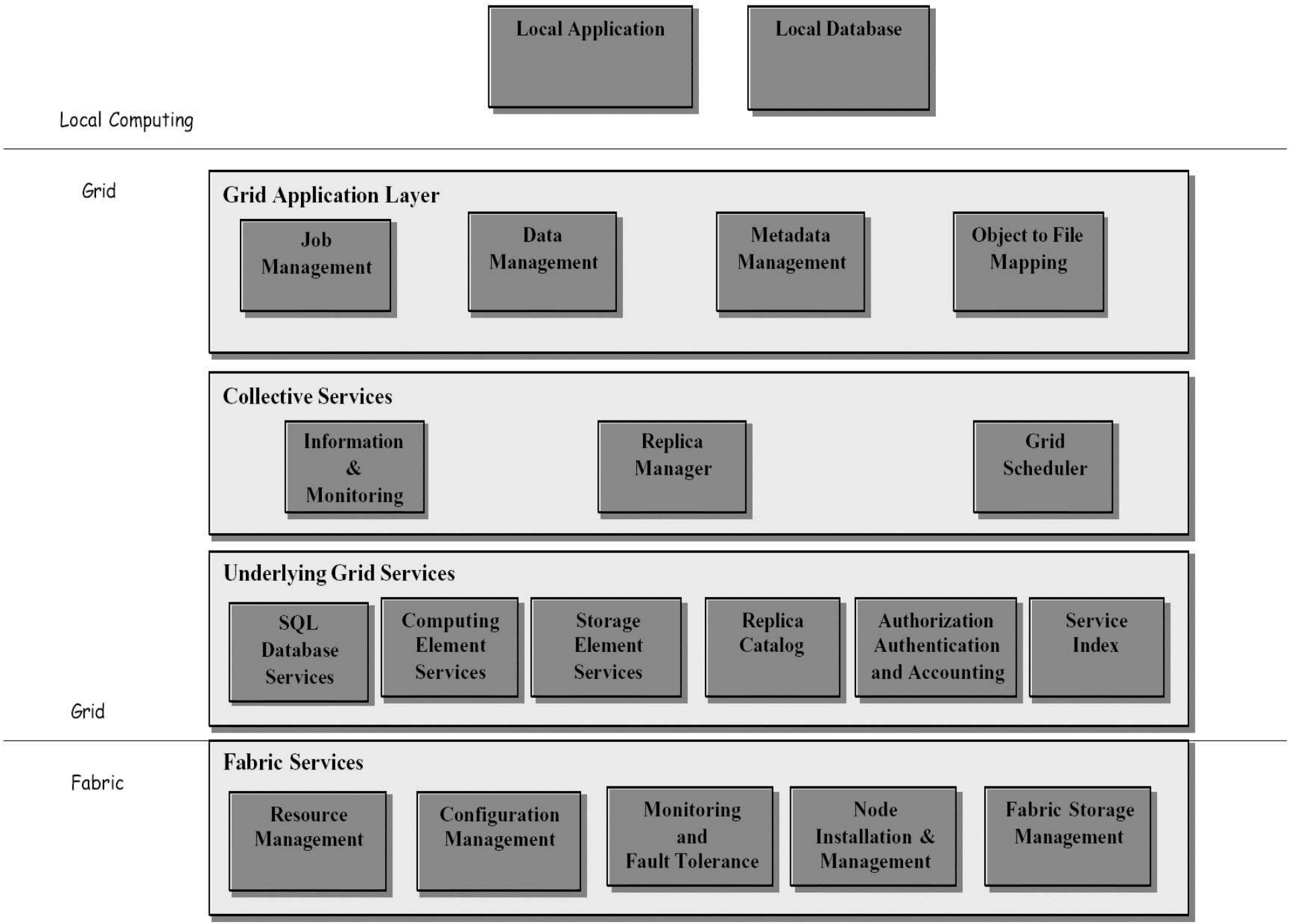
Collective Services

- Replica Manager:
  - Schnittstelle zum Replica Catalog
  - Funktionen:
    - add/deleteLogicalFileName(LogicalFileName)
    - add/deletePhysicalFileName(LogicalFileName, PhysicalFileName)
    - getPhysicalFileName(LogicalFileName)
    - ...
  - Attribute setzen / löschen

Collective Services

- Grid Scheduler:
  - Hauptkomponente des European Data Grid
  - nimmt Jobs entgegen
  - Verteilung der Rechenjobs auf passende Ressourcen mit
    - ausreichende Leistung (CPU, Speicher, Netzwerk, ...)
    - passenden Benutzerrechten
    - Berücksichtigung des Speicherorts von Ein- und Ausgabedaten
  - Überwachung der Jobs

Später mehr dazu ...


Local Application
Local Database
Local Computing
Grid
Grid Application Layer
Job Management
Data Management
Metadata Management
Object to File Mapping
Collective Services
Information & Monitoring
Replica Manager
Grid Scheduler
Underlying Grid Services
SQL Database Services
Computing Element Services
Storage Element Services
Replica Catalog
Authorization Authentication and Accounting
Service Index
Grid
Fabric
Fabric Services
Resource Management
Configuration Management
Monitoring and Fault Tolerance
Node Installation & Management
Fabric Storage Management

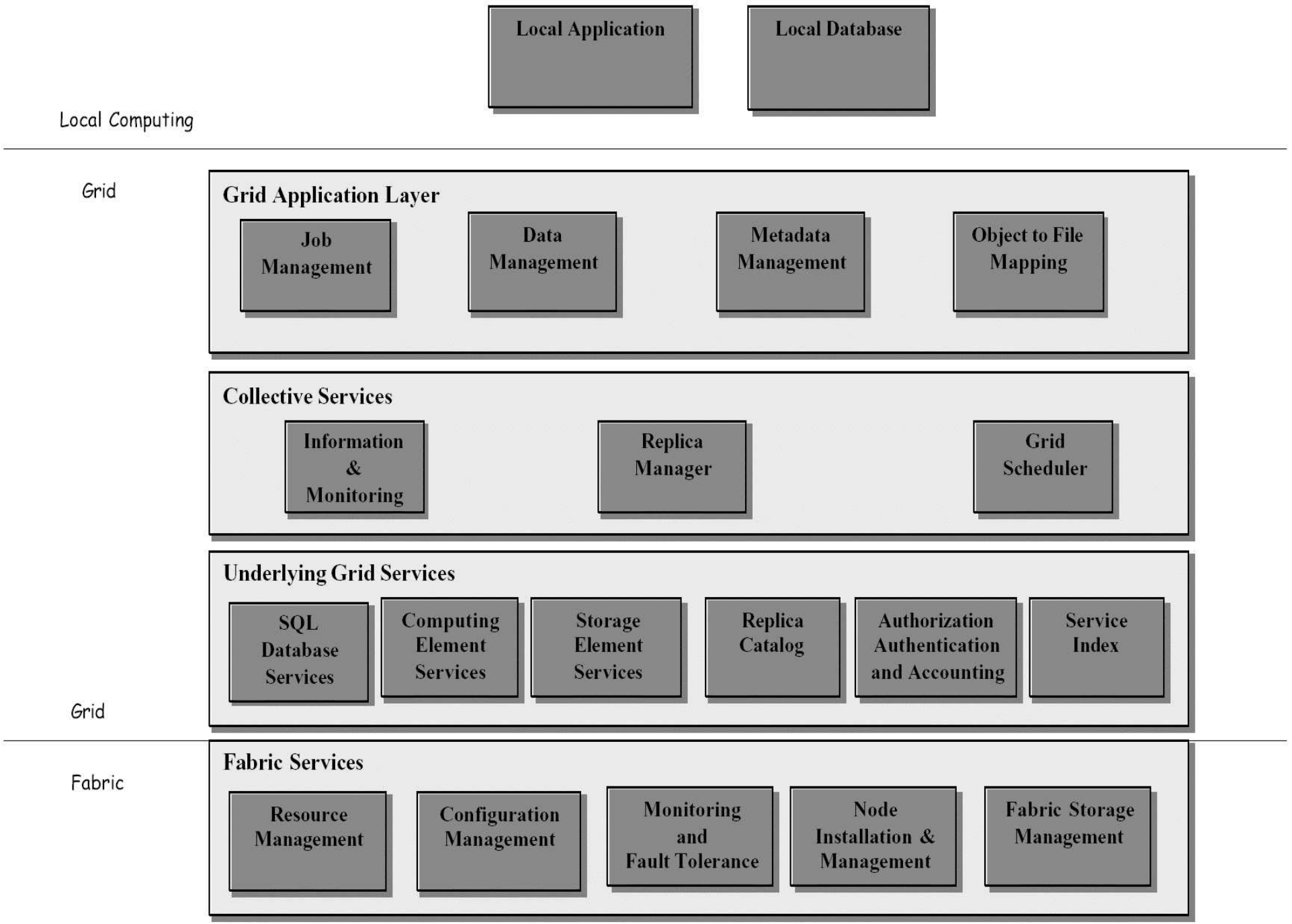
Grid Application Layer

Schnittstelle zwischen Grid und Applikationen

- Job Management:
  - Entscheidung welche Jobs ans Grid übergeben werden
- Data Management:
  - Dateien ins Grid schreiben / aus dem Grid lesen

Grid Application Layer

- Metadata Management:
  - Speicherung von Informationen über im Grid gespeicherte Daten (Applikations-spezifisch)
  - Grid hat keine Metainformationen
- Object to File Mapping:
  - Grid kann nur Dateien speichern, daher:
    - Auffinden spezieller Objekte in Dateien
  - Form von Metainformation


Local Application
Local Database
Local Computing
Grid
Grid Application Layer
Job Management
Data Management
Metadata Management
Object to File Mapping
Collective Services
Information & Monitoring
Replica Manager
Grid Scheduler
Underlying Grid Services
SQL Database Services
Computing Element Services
Storage Element Services
Replica Catalog
Authorization Authentication and Accounting
Service Index
Grid
Fabric
Fabric Services
Resource Management
Configuration Management
Monitoring and Fault Tolerance
Node Installation & Management
Fabric Storage Management

Grid Scheduler (Nachtrag)

Scheduling-Kriterien:

- freie Ressourcen
- Benutzerrechte
- Speicherort Ein-/Ausgabedaten
- Effizienz der aktuellen Strategie
  - Wartezeiten
  - Abbrüche (Frequenz, abs. Häufigkeit)

    → erneute Jobausführung

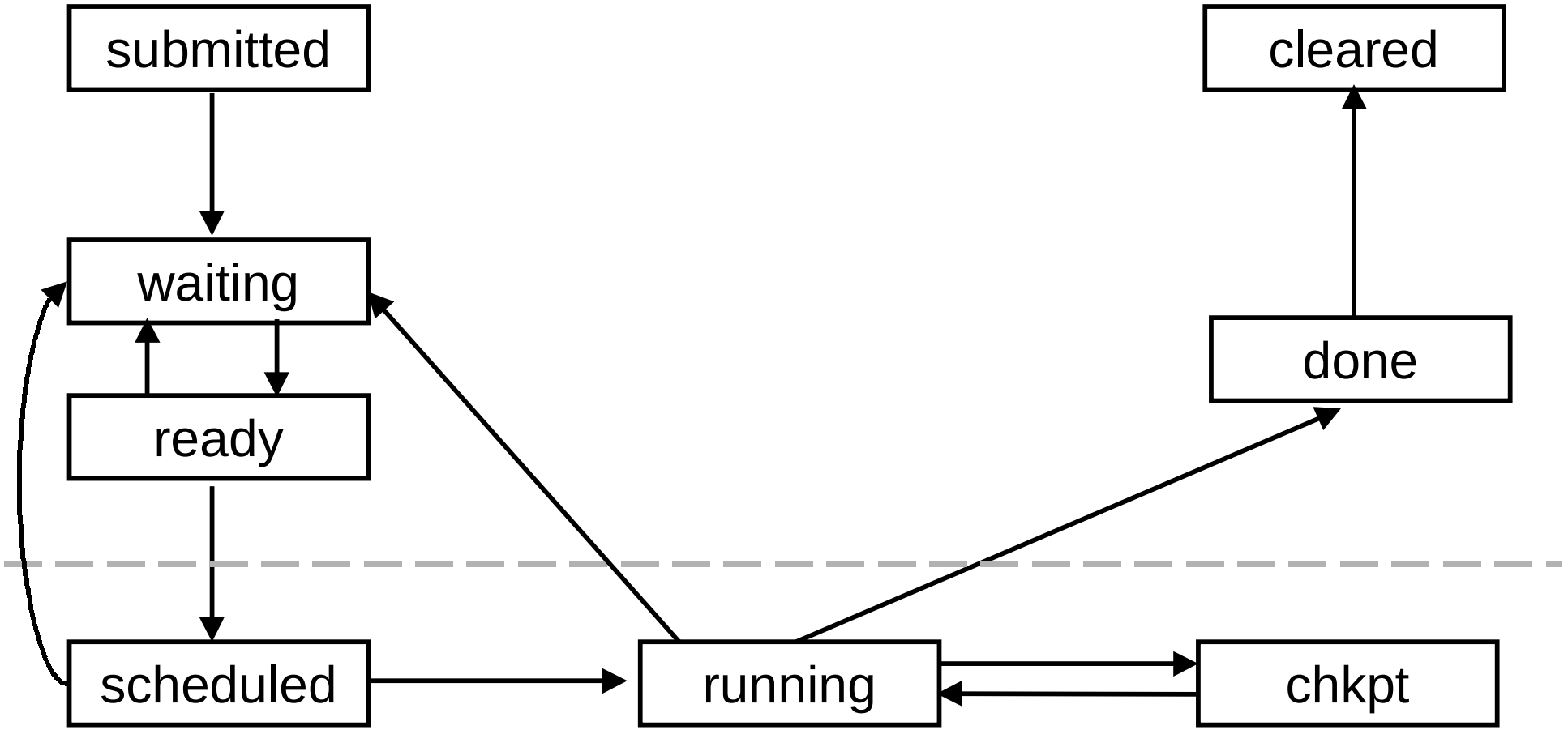
Grid Scheduler (Nachtrag)

Fehlererkennung/-behebung:

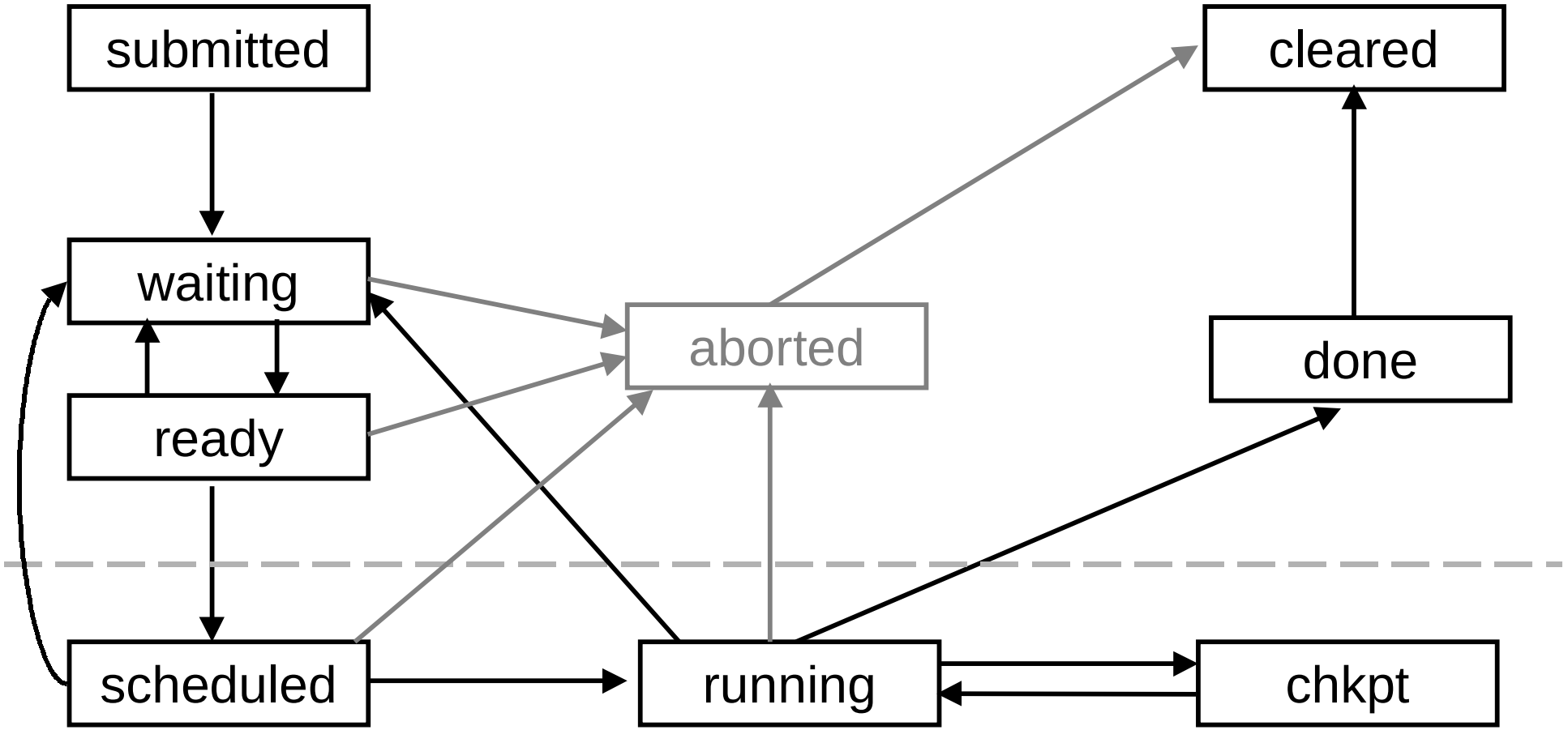
- nur Fehler des Grids berücksichtigen
- Zusammenarbeit der Gridkomponenten

  (ausgewählte Komponenten nicht verfügbar)
- periodische Überprüfung
  - Fehlermeldungen können auf Grund der Ausfälle (Netzwerk) verloren gehen

Grid Scheduler (Nachtrag)



Grid Scheduler (Nachtrag)



Grid Scheduler (Nachtrag)

Job Description Language (Beispiel):

```
[
    Executable              = "simula";
    Arguments               = "1 2 3";
    StdInput                = "simula.config";
    StdOutput               = "simula.out";
    StdError                = "simula.err";
    InputSandbox            = {"/home/joe/simula.config",
                               "/usr/local/bin/simula"};
    OutputSandbox           = {"simula.out", "simula.err",
                               "core"};
    InputData               = "LF:test367-2";
```

Grid Scheduler (Nachtrag)

Job Description Language (Beispiel):

```
    Replica Catalog          = "ldap://pcrc.cern.ch:2010/
                               rc=Replica Catalog, dc=pcrc,
                               dc=cern, dc=ch"
    DataAccessProtocol       = {"file", "gridftp"};
    OutputSE                 = "lxde01.pd.infn.it";
    Requirements             = other.Architecture == "INTEL" &&
                               other.OpSys == "LINUX";
    Rank                     = other.AverageSI00;
]
```

Offene Fragen

- Prioritäten – interaktive Programme
- Programmausführung:
  - check-points
  - Migration auf andere Ressource
- lokale Programmausführung <-> Programme vom Grid
- Objekte innerhalb von Dateien effizient finden

Literatur

- http://eu-datagrid.web.cern.ch/eu-datagrid/
- http://www.cern.ch/

Fragen???