

### 3.4 Lebenszeit von Grid-Services (2)

- Keep-Alive-Nachrichten (fortges.)
  - ◆ Aufruf von `requestTerminationAfter` oder `requestTerminationBefore`
    - Parameter übergeben gewünschten frühesten oder spätesten Terminierungszeitpunkt
    - Rückgabe des vom Dienst bestimmten Terminierungszeitpunkts
- Explizites Abschalten der Dienstinstanz
  - ◆ Aufruf der Methode `destroy`



### 3.6 Service-Data-Elements

- Wertelemente
  - ◆ ein SDE kann mehrere Wertelemente besitzen
    - `minOccurs`, `maxOccurs` bestimmt Anzahl
    - Beispiel: Zähler-SDE als GWSDL-Ausschnitt

```
<serviceData name="counter" type="xs:positiveInteger"
  minOccurs="1" maxOccurs="1" mutability="mutable"
  nillable="false" modifiable="true">
  <documentation>
    ...
  </documentation>
</serviceData>
```
    - Beispiel: `Interface`-SDE aus `GridService-portType`

```
<serviceDate name="interface" type="xs:Qname"
  minOccurs="1" maxOccurs="unbound"
  mutability="mutable" modifiable="false"
  nillable="false"/>
```



### 3.5 Übertragung von Referenzen

- Übergabe von Referenzen auf Grid-Services
  - ◆ Parameterübergabe oder Rückgabewert
- Grid Service Locator
  - ◆ Container-Datenstruktur für Menge von GSHs und GSRs zum selben Grid-Service
    - Liste von GSHs
    - Liste von GSRs
    - Liste von Port-Type-Namen
  - ◆ Empfänger wählt geeignete GSR aus oder löst GSH auf



### 3.6 Service-Data-Elements (2)

- Wertelemente (fortges.)
  - ◆ Beispiel für ermittelte Wertelemente bei einem `sayHello`-Service

```
<serviceDataValues ...>
  <ogsi:interface>
    <value ...>ogsi:GridService</value>
  </ogsi:interface>
  <ogsi:interface>
    <value ...>sh:SayHelloPortType</value>
  </ogsi:interface>
  ...
</serviceDataValues>
```
  - mehrere Wertelemente enthalten die Namen der Port-Types des Service



### 3.6 Service-Data-Elements (3)

- Deklaration der SDEs
  - ◆ **mutability**: Veränderlichkeit der Wertelemente
    - **static**: Wert wird in GWSDL vorgegeben für alle Serviceinstanzen
    - **constant**: Wert wird bei Instanzerzeugung festgelegt
    - **extendable**: Wertelemente können hinzugefügt werden; bisherigen Wertelemente ändern sich nicht
    - **mutable**: beliebige Änderung möglich
  - ◆ **modifiable**: von außen änderbar
  - ◆ **nillable**: kann nill-Wert annehmen



### 3.7 Gemeinsame SDE-Definitionen

- SDEs im **GridService-portType**
  - ◆ **serviceName**: zählt alle SDE-Namen auf
  - ◆ **interface**: zählt alle Port-Types auf
  - ◆ **factoryLocator**: enthält Service-Locator für eigenen Factory-Service
  - ◆ **gridServiceHandle**: enthält GSHs für diesen Dienst
  - ◆ **gridServiceReference**: enthält GSRs für diesen Dienst (mind. eine ist in GWSDL beschrieben)
  - ◆ **findServiceDataExtensibility**: enthält Deklaration von Operationen zum Lesen von SDEs
  - ◆ **setServiceDataExtensibility**: enthält Deklaration von Operationen zum Schreiben von SDEs
  - ◆ **terminationTime**: Zeitpunkt der Dienstabstaltung



### 3.6 Service-Data-Elements (4)

- Verknüpfung eines SDE mit Lebenszeit
  - ◆ XML-Attribute geben Gültigkeit der Werte an
    - **ogsi:goodFrom**: erster Gültigkeitszeitpunkt
    - **ogsi:goodUntil**: letzter Gültigkeitszeitpunkt
    - **ogsi:availableUntil**: letzter Verfügbarkeitszeitpunkt
- Operationen zum Setzen und Lesen von Werten
  - ◆ Operationen können vom Dienst festgelegt werden
  - ◆ vorgegebenen Operationen:
    - Operation **queryByServiceDataNames** ermittelt Wertelemente gegebener SDEs
    - Operation **setByServiceDataNames** setzt Wertelemente
    - Operation **deleteByServiceDataNames** löscht SDEs



### 3.7 Gemeinsame SDE-Definitionen (2)

- Extensibility-Deklarationen
  - ◆ Angaben von XML-QNames für die gültigen Operationen
  - ◆ Beispiel:

```
<serviceDataValues ...>
  <ogsi:findServiceDataExtensibility>
    <value ...>ogsi:queryByServiceDataName</value>
  </ogsi:findServiceDataExtensibility>
  ...
</serviceDataValues>
```
  - Grid-Service kann weitere Operationen vorgeben – in einer Erweiterung des **GridService-portType**



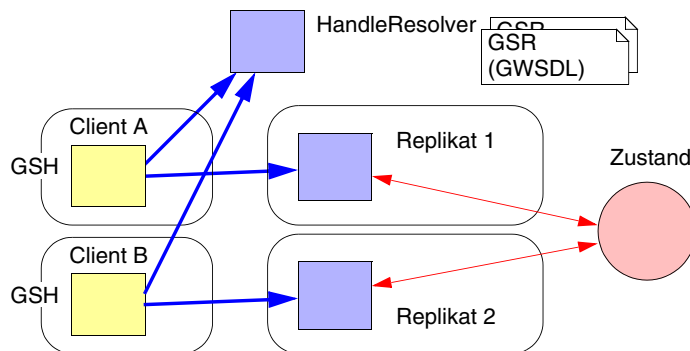
### 3.8 Benachrichtigung über SDE-Änderungen

- Voraussetzung
  - ◆ Grid-Service implementiert **NotificationSource-portType**
  - ◆ Client kennt oder implementiert **NotificationSink-portType**
- Ablauf
  - ◆ Client schreibt sein **NotificationSink-portType** beim **NotificationSource-portType** ein (**subscribe**)
  - ◆ Angabe der interessierenden SDE-Namen
  - ◆ Service ruft Sinks bei Veränderungen auf
    - Übergabe eines **serviceDataValues**-Objekt



### 4 Architekturen von Grid-Services (2)

- Beispiel: Replizierter Grid-Service
  - ◆ mehrere Replikate für eine Grid-Service-Instanz



- Zugriff bleibt transparent trotz unterschiedlicher GSRs



### 4 Architekturen von Grid-Services

- Monolithische Implementierung
  - ◆ Zugangspunkt, Protokollimplementierung und Dienstimplementierung in einem
  - ◆ z.B. Implementierung in einem Servlet
- Implementierung im Container
  - ◆ Container verwaltet Zugangspunkte und Protokolle
    - Protokolle unabhängig von Implementierung
  - ◆ z.B. .Net, (EJB)
  - ◆ Container verwaltet Zustand und Lebenszyklus
    - z.B. EJB
    - geeignete Zusammenschaltung mehrerer Container denkbar
      - z.B. für Lastbalancierung, Fehlertoleranz etc.



### 5 OGSA

- ★ Bisherige Betrachtung: im Wesentlichen OGSi
- OGSA Spezifikation: Definition von zusätzlichen Services (Capabilities)
  - ◆ Execution-Management-Services
    - steuern Ausführung von Grid-Services
  - ◆ Data-Services
    - Zugriff auf Daten
  - ◆ Ressource-Management-Services
  - ◆ Security-Services
  - ◆ Self-Management-Services
  - ◆ Information-Services
    - zeitgerechte Bereitstellung von Informationen
      - z.B. Monitoring, Anwendungsdaten



## 5.1 Execution-Management-Services (EMS)

- Aufgaben
  - ◆ Ermittlung potenzieller Ausführungsorte für einen Grid-Service
  - ◆ Ermittlung des tatsächlichen Ausführungsorts bei Instanziierung
  - ◆ Vorbereitung des Ausführungsorts
    - Installation des Codes
    - Deployment
  - ◆ Instanziierung
  - ◆ Verwaltung der Ausführung
    - Monitoring
    - Neustarts
    - Aktualisierungen
    - ...



## 5.2 Data-Services

- Grid-Services benötigen häufig große Mengen von Daten
  - ◆ OGSA definiert Data-Services zum
    - Zugriff auf Datenmengen,
    - Erzeugen von Datenmengen
  - ◆ Frameworks und Subdienste ähnlich wie bei EMS
    - Integrationsmöglichkeit mit (verteilten) Datenbanken



## 5.1 Execution-Management-Services (2)

- OGSA-Definition
  - ◆ Aufteilung in eine Menge konfigurierbarer Subdienste in verschiedenen Frameworks
    - Workload-Management-Framework
    - Workload-Optimization-Framework
    - Ressource-Management-Framework
    - Ressource-Optimization-Framework
    - Optimizing Framework
      - enthält z.B. Selektionsdienst für geeignete Orte
  - ◆ Container-ähnliche Infrastruktur für Implementierungen wird angenommen
- Einsatz der verschiedenen Subdienste stark von Einsatzgebiet des Grid-Systems abhängig



## 5.3 Resource-Management-Services

- Drei Ebenen des Managements
  - ◆ Management der Basis-Ressourcen
    - Reboot eines Rechners
    - Rekonfiguration von Routern
  - ◆ Management der Grid-Ressourcen
    - Integration mit EMS, Data-Services etc.
  - ◆ Management der OGSA-Infrastruktur
    - Verwaltung der OGSA-Dienste und -Frameworks
- Wesentliche Aufgaben der Dienstpakete
  - ◆ Reservierung von Ressourcen
  - ◆ Überwachung und Steuerung



## 5.4 Security-Service

- Dienstpaket zur Absicherung des Grid-Systems
  - ◆ Authentisierungsdienst
  - ◆ Identitätenabbildung (Basis X.509 Zertifikate)
  - ◆ Autorisierungsdienst
  - ◆ Policy-Service
    - Verwaltung und Bereitstellung der Sicherheitsanweisungen
  - ◆ Credential-Conversation-Service
    - Umwandlung und Transport von Zertifikaten und ähnlichen Credentials
  - ◆ Audit-Service
  - ◆ Profile-Service
    - Verwaltung von Benutzerpräferenzen



## 6 Globus-Toolkit

- Implementierung der Globus-Alliance
  - ◆ aktuelle Version 3.2
  - ◆ C- und Java-Komponenten
    - große Menge von Services aus der OGSA
      - z.B. Ressource-Management-Service
    - OGSI-kompatible Web-Service-Container-Implementierung



## 5.5 Self-Management-Services

- Dienstpaket zur Selbstorganisation des Grid-Systems
  - ◆ Beobachtung des Systems
    - Monitoring
  - ◆ Identifikation neuer Ressourcen
  - ◆ Integration in EMS und Data-Services
    - z.B. Neustart von Dienstinstanzen auf neuen Knoten bei Fehlern



## 7 Literatur

- Grid-Computing
  - ◆ I. Foster, C. Kesselman, J. Nick, S. Tuecke: *The physiology of the Grid, an open Grid services architecture for distributed systems integration*. 2002.  
<<http://www.globus.org/research/papers/ogsa.pdf>>
  - ◆ *Open Grid service infrastructure primer*. GGF OGS Working Group, Draft #5, March 12th, 2004.
  - ◆ *Open Grid services infrastructure (OGSI)*. Version 1.0, Draft #33, July 27th, 2003.
  - ◆ *The Open Grid service architecture (OGSA)*. Version 1.0, Draft #18, June 7th, 2004.
  - ◆ *Globus-Toolkit*: <<http://www.globus.org/toolkit>>

