

IGS - Application Management Services

**Grid Services =
Web Services ++
oder
New Skin for the Old Ceremony**

Ulf Hollberg
IBM Global Services AMS
Walldorf

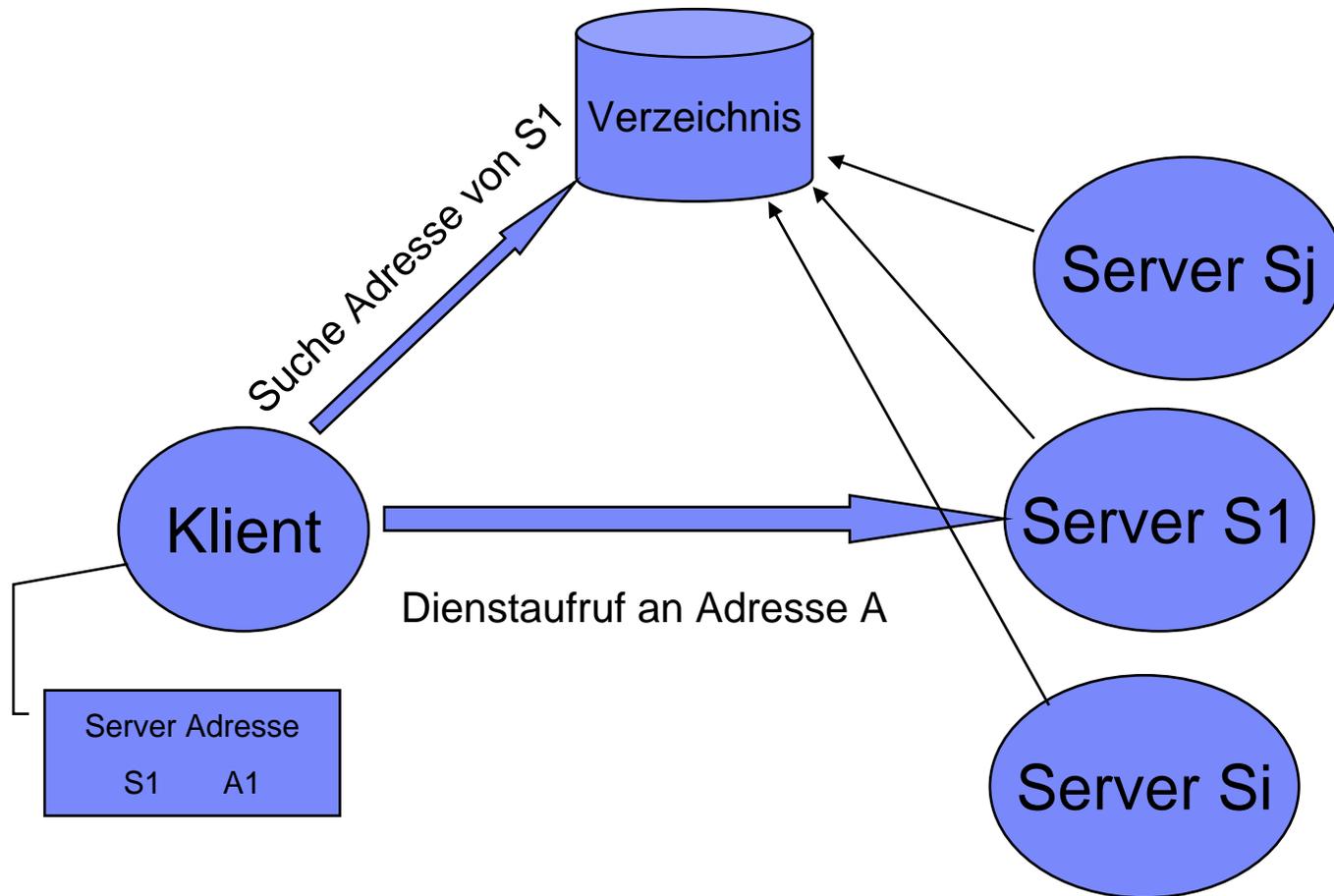
Überblick

- Zum Titel
- Old Ceremony: Verteilte Verarbeitung
 - **Client – Server Interaktionen**
 - **Web Services**
- New Skin
 - **Grid Services**
 - **Gegenüberstellung New <-> Old**
- Zusammenfassung

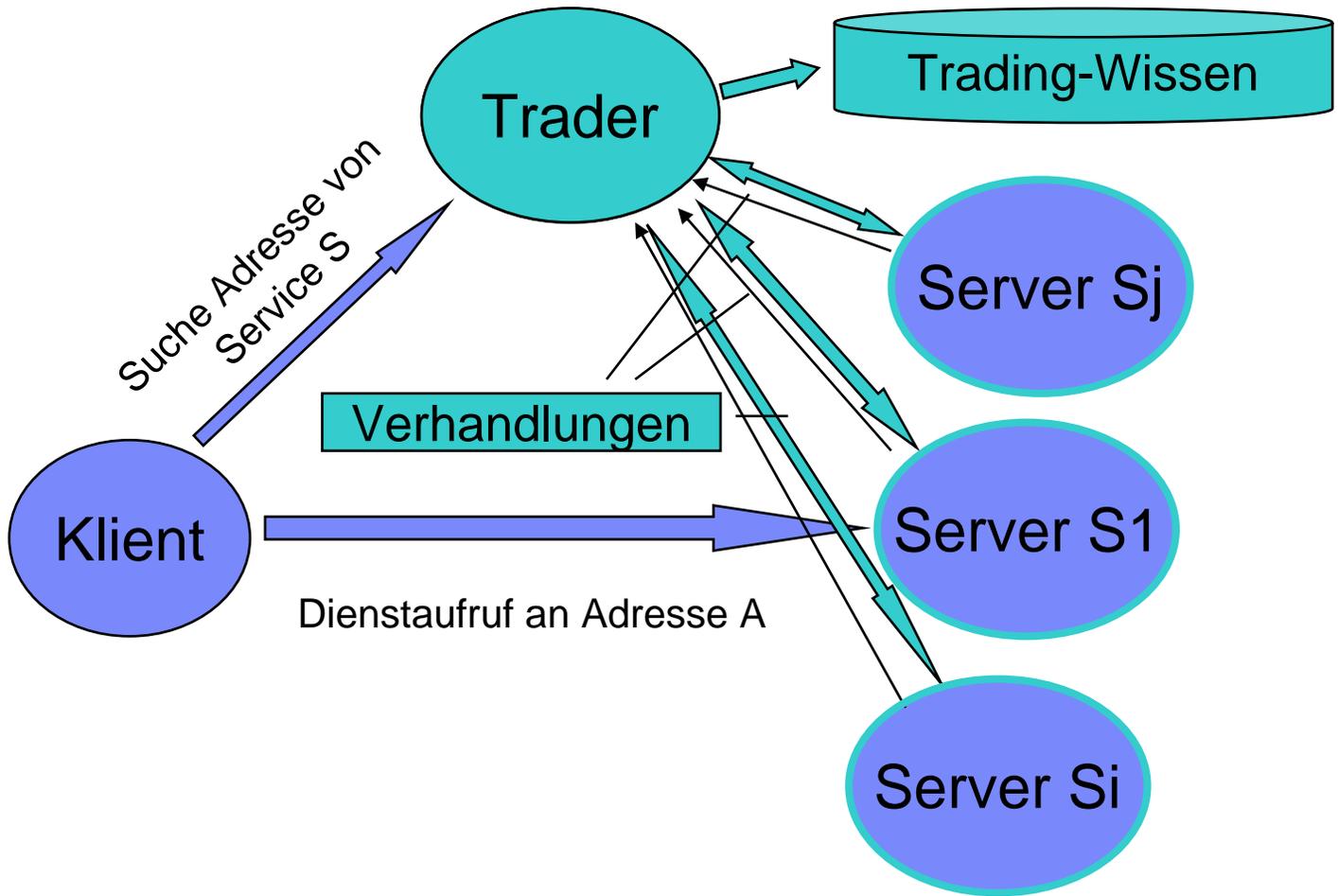
Zum Titel

- **Zur Notation**
„<X>++“ bedeutet
 „Erweiterung bzw. Nachfolger von <X>“
aus Bjarne Stroustrup:
 „The Design and Evolution of C++“ (p.64)
- **„New Skin for the Old Ceremony“**
 Platte von Leonard Cohen (1975)
- ***Old Ceremony:***
 verteilten Verarbeitung und Speicherung

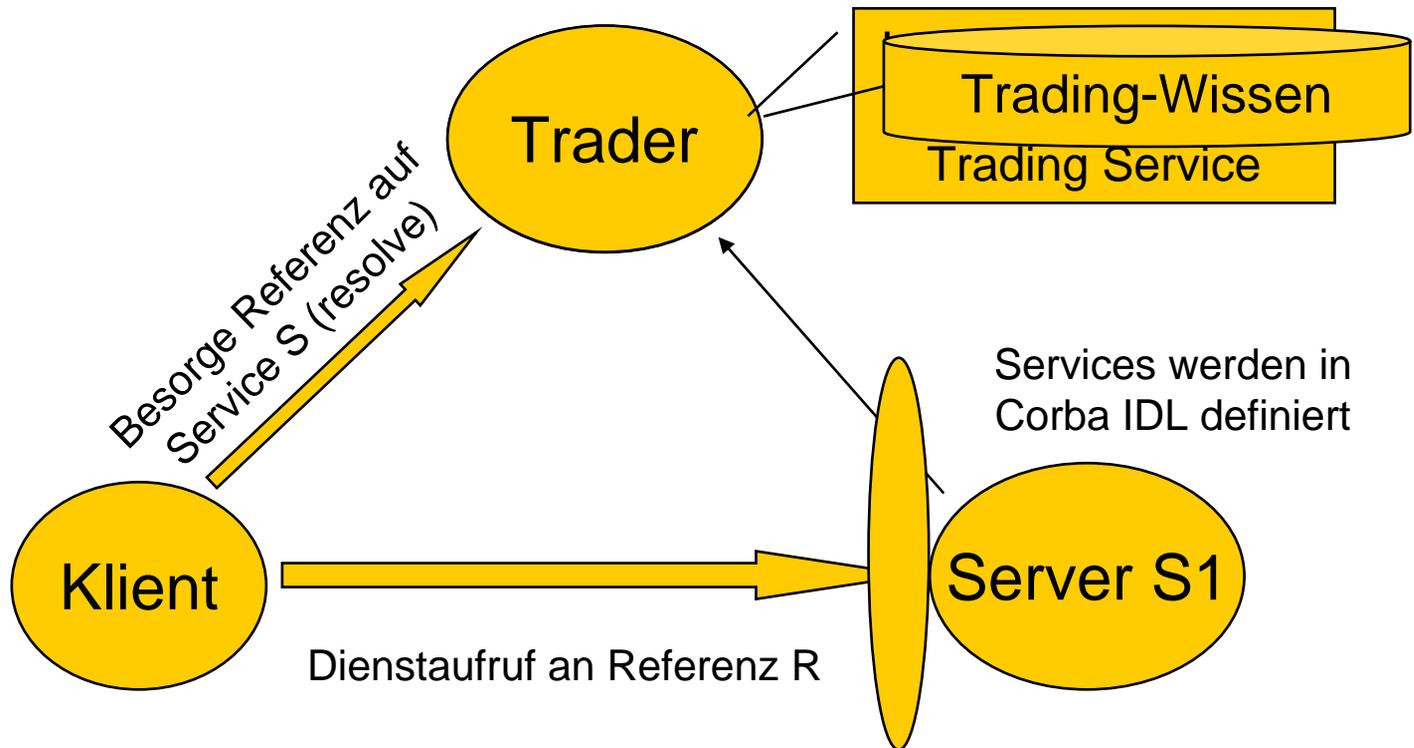
Klient - Server – Grundlegende Interaktionen



Klient - Server – Dynamischere Interaktionen



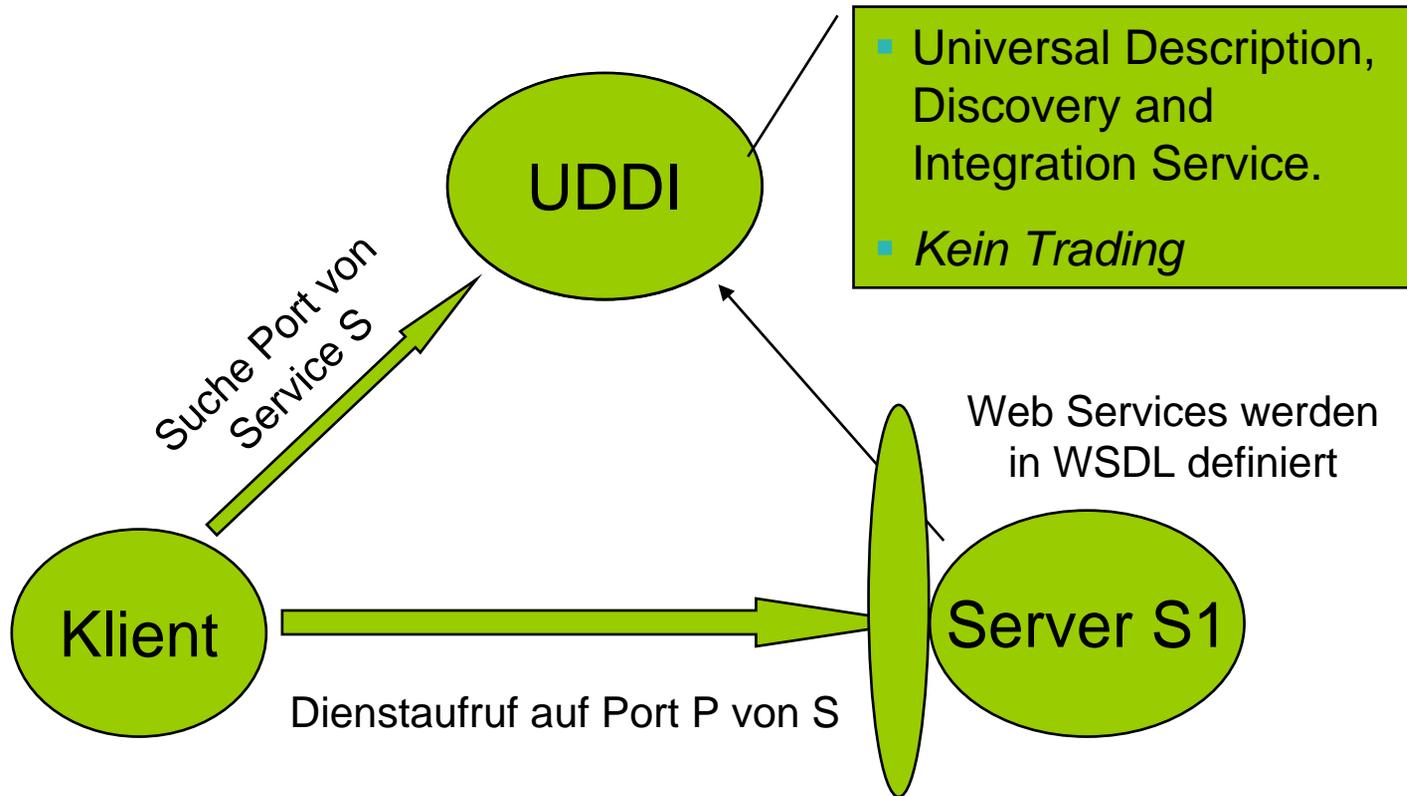
Klient - Server – Implementierung mit Corba



Corba ist seit 2000 Standard der OMG, der ISO (ISO/IEC 19500-2) und der ITU-T (ISO/IEC 10746|ITU-T Rec. X.900-904).

(<http://www.omg.org/news/releases/pr2000/2000-10-25.htm>)

Klient - Server – Implementierung mit Web Services



Web Services 1.1 ist ein Standard des World-Wide Web Consortium seit ca. 2000

WSDL 1.2 ist Draft Standard des World-Wide Web Consortium seit März 2003:

www.w3.org/TR/wsd1

UDDI 3.01 ist Standard des UDDI.org seit 2003: http://uddi.org/pubs/uddi_v3.htm

Web Services - Definitionen

- **Web Services** basieren auf **Nachrichten**, die zwischen Klient und Server ausgetauscht werden
- **Nachrichten** sind **abstrakt definiert** und werden an konkrete Netzwerkprotokolle und Nachrichtenformate gebunden
- Eine Nachricht besteht aus einer Menge von **typisierten Datenelementen**
- Der **Austausch einer Menge von Nachrichten** zwischen Klient und Server wird als **Operation** beschrieben
- Eine Menge von **Operationen** wird **Port Typ** genannt
- Ein **Service** enthält eine Menge von **Ports**, die jeweils einen **Port Typ** implementieren

Aus [WSDL 1.2]

Web Services - Yet another RPC (?)

- **Web Services nutzen WSDL als Interface Spezifikation**
 - Einfache Parametertypen
 - Komplexe Parameter sind XML-Dokumente (Strings mit XML-Tags)
 - URLs als Objektnamen
 - Keine Objektreferenzen
 - Unterstützte Protokolle
 - Weitere administrative Angaben, z.B. Standort
- **Alle Definitionen werden in XML aufgeschrieben**
- **WSDL Definitionen sind unveränderlich (nach ihrer Veröffentlichung !?)**

Web Services - Designprinzipien

- **Relativ grobkörnige Kommunikation**
bei zustandslosen („stateless“) Protokollen sind Round-Trips teuer, ihre Anzahl sollte daher minimiert werden.
- **Lose Koppelung**
Der Klient soll nur das Nötigste über den Server wissen. Die Formate der ausgetauschten XML-Nachrichten sind der Vertrag mit der Außenwelt. Diese sind in der WSDL festgelegt.
- **Asynchronität**
Es kann / darf etwas dauern, bis eine Antwort kommt. Es handelt sich also eher um eine nachrichten-orientierte Architektur.

Nach Adam Bosworth

<http://www.w3.org/2003/Talks/0818-msm-ws/slide26-0.html>

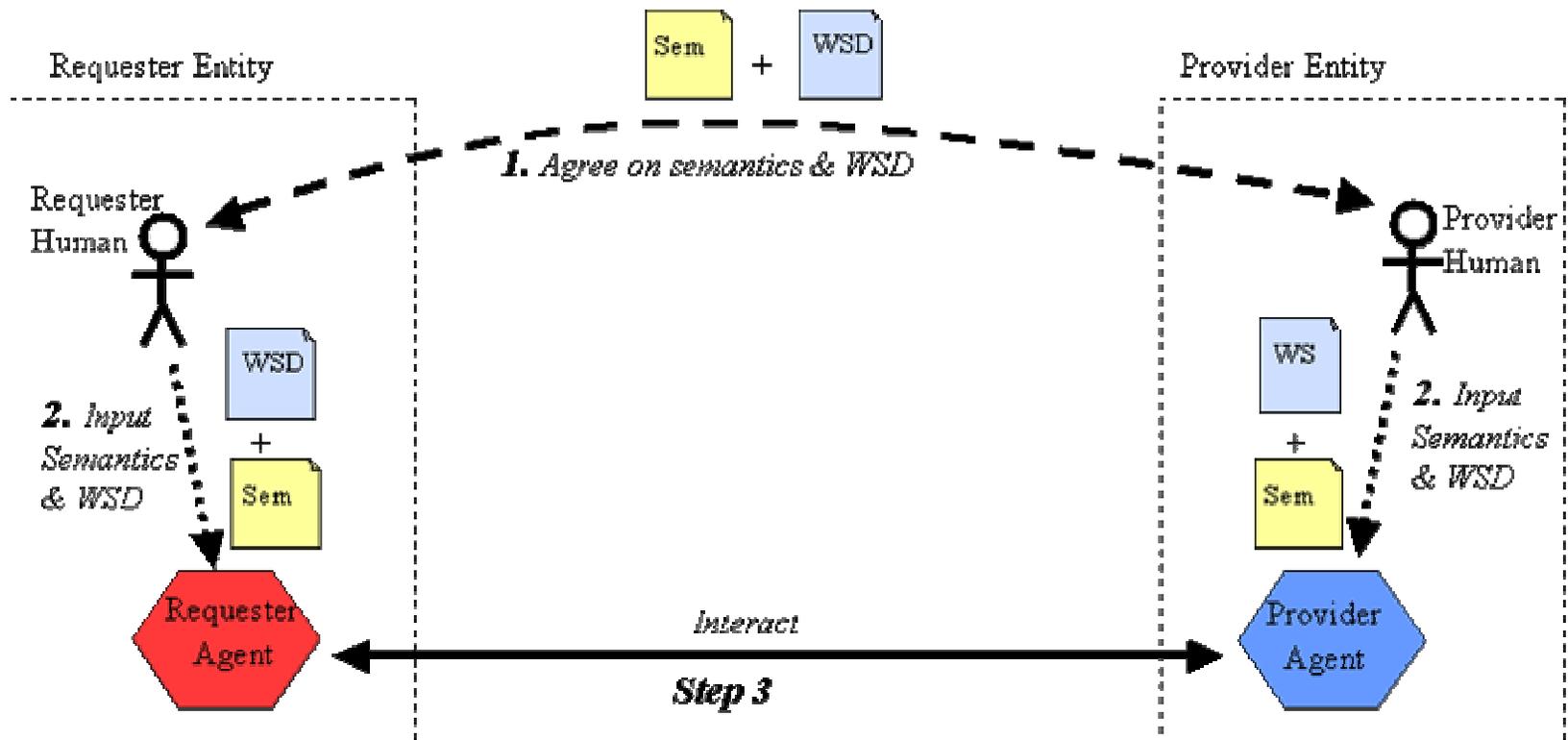
Web Services – Basisdienste

- **UDDI – Universal Description, Discovery and Integration**
 - Ein Verzeichnisdienst für Interface Definitionen in WSDL
- **Weitere Dienste**
 - **WS-Reliable Messaging** **WSRM**
 - **WS-Resource Framework** **WSRF**
 - **WS-Federation** **WSF**
 - **WS-Security** **WSS**
 - ...
 - **WS-Interoperability** **WSI**

Web Services – Einfluß auf B2B

- **Unternehmen bieten extern Dienste als WebServices an**
 - z.B. Amazon für Bestellungen und Anfrage zum Stand der Auslieferung (siehe Amazon E-Commerce Service 4.0: <http://www.amazon.com/gp/aws/sdk/102-2571238-1859323?>)
- **Öffentliche Kataloge mit UDDI-Dienste für B2B (Business to Business) Nutzung**
 - Beispiel: <http://www-306.ibm.com/software/solutions/webservices/uddi/>
 - Ähnliches gibt es von SAP, Microsoft, ..

Web Services – Automatische Bindung



From <http://www.w3.org/2003/Talks/0818-msm-ws/slide20-0.html>

Web Services - *Eine Lösung für Alles ?*

Kontroverse Diskussion in Zeitschriften und dem Internet, z.B.

- „Like it or not, Web Services are Distributed Objects“, Kenneth P. Birman, Comm. of the ACM, Dec. 2004, Vol. 47, No. 12, P. 60ff
- „Web services = or != distributed objects?“, David Orchards, http://www.pacificspirit.com/blog/2004/01/12/web_services_or_distributed_objects

There have been other distributed computing models, but **this time it's serious.**

Aus dem Buch: "Perspectives on Web Services", Springer Verlag

This is just another reinvention of the wheel, **the most pointless hype in years.**

Verteilte Systeme – einige Design-Aspekte

■ **Flexibilität**

- Konfigurierbarkeit
- Wartbarkeit
- Entkoppelung
- Erweiterbarkeit
- Elastizität

Vs.

■ **Effizienz**

- Kleiner Aufwand
- Reaktionsfähigkeit

Designprinzipien von Web Services

- Relativ grobkörnige Kommunikation
- Lose Koppelung
- Asynchronität

⇒ *Flexibilität*

Dies sind Eigenschaften der Service-orientierten Architekturen (SOA)

Service Oriented Architecture

Ein **System** hat eine **Service-orientierte Architektur**, wenn

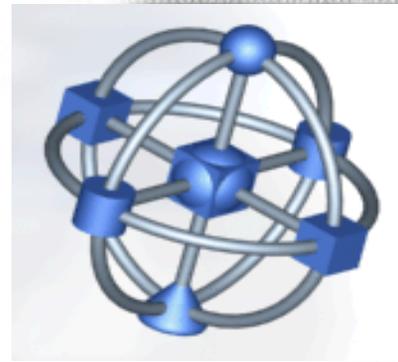
- es aus lose gekoppelten, eigenständigen Komponenten (Sub-Systemen) bestehen,
- die unabhängig voneinander ausgetauscht werden können und
- die über wohldefinierte Schnittstellen kommunizieren.
- Diese Komponenten erbringen einen **Service** für andere Komponenten.

SOAs werden häufig mit Web Services implementiert

Grid Services - New Skin -

Warum eigentlich „Grid“ ?

- *ein Gitter*
- *ein Versorgungsnetz*
- *ein geordnetes Netz*



Überblick

- **Definition**
- **Beispiel**
- **Worum geht es bei Grids?**
- **Was ist es?**
- **Wer steckt dahinter?**

Grid – Worum geht es ?

- **Definition**

Ein Grid ist

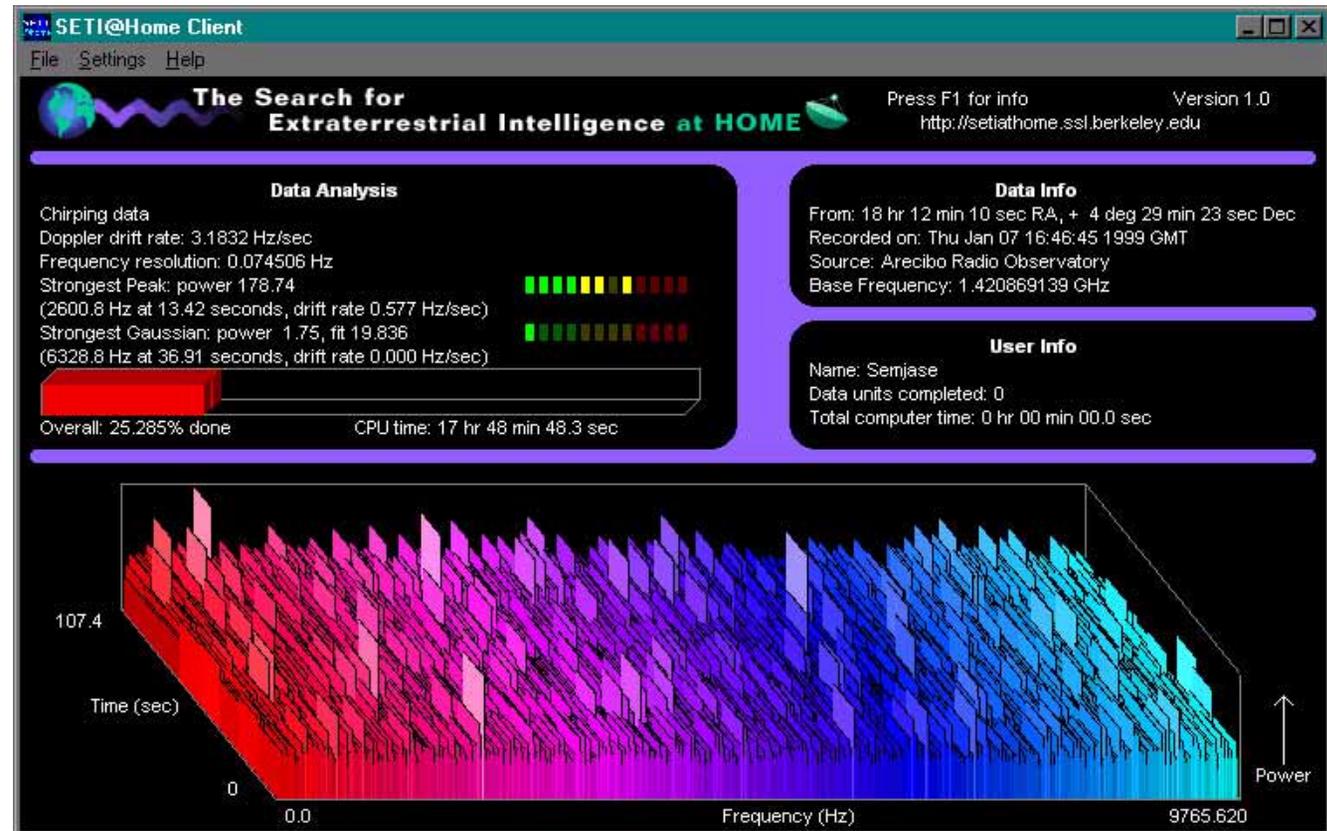
„Gemeinsame Nutzung von Betriebsmitteln und kollaboratives Problemlösen in dynamischen, Institutionen-übergreifenden virtuellen Organization.

(im Original: „Resource Sharing & Collaborative Problem Solving in a Multi-Institutional Virtual Organization“)

Grid – Beispiel SETI@HOME

■ Gemeinsame Problemlösung

- Suche nach extra-terrestrischer Intelligenz
- Etwa 4 million teilnehmende Rechner
- Wachstum ca. 2000 Rechner pro Tag
- Kids in den US nutzen es als Vergleichsmaßstab für ihre Rechner ..



Grid – Systeme – Ziele

Grid-Systeme erlauben das Etablieren virtueller Organisationen aus

- *Service Anbietern (Provider)*
- *Service Nutzern (Consumer)*
- *Service Vermittlern (Broker)*

ohne zentrale Steuerung oder Koordination

Grid – Systeme - Abgrenzung

Ein Grid ist ein System,

- die Nutzung von Betriebsmitteln koordiniert, die nicht einer zentralen Kontrolle unterliegen; Betriebsmittel und Benutzer sind in verschiedenen Domänen im Bezug auf Administration oder Technologie,

sonst wäre es ein lokales System

- das standardisierte, offene, general-purpose Protokolle und Schnittstellen nutzt,

sonst wäre es ein spezialisiertes Anwendungssystem

- das nicht-triviale Service-Qualitäten liefert, im Sinne von Antwortzeit, Durchsatz, ..

sonst wäre es von begrenztem (kommerziellen) Nutzen

Nach der Definition von Foster & Kesselman (Nov. 2002)

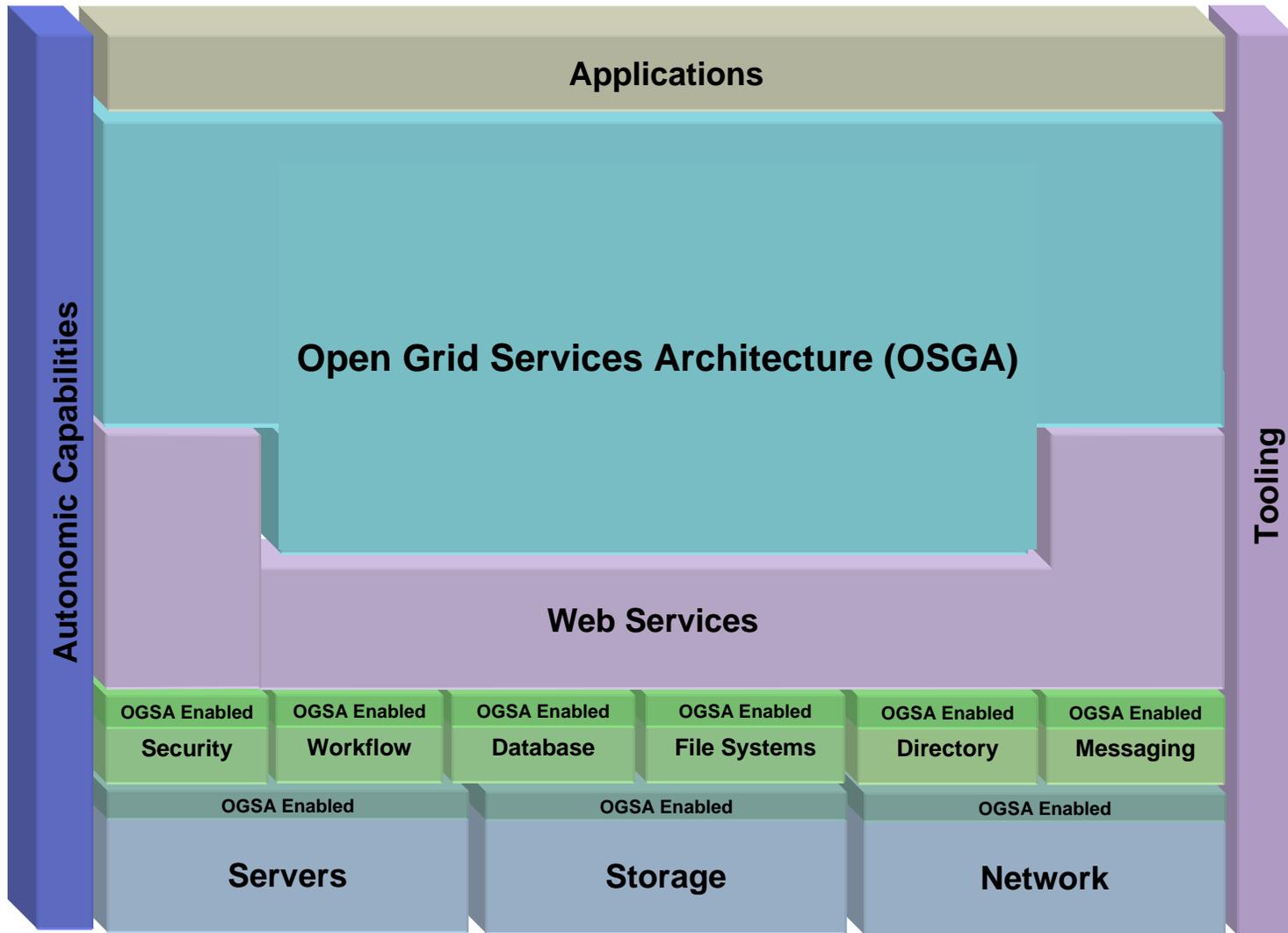
Grid – Systeme – technische Grundidee

Virtualisierung von allen Rechner-Betriebsmitteln

- Verbergen von Verteilung
- Verbergen von Heterogenität
- Verbergen von konkreten Plattformen, Sprachen

*Vereinheitlichung auf allgemeine Service-Beziehungen
zwischen den Teilnehmern*

OGSA Überblick



Grid Services der OGSA erweitern Web Services

- **Grid Services** werden in WSDL definiert, mit
- **Grid Service Descriptions** (~class) haben
 - eine Interface Syntax mit portType(s) (in WSDL)
 - Eine (informelle) Semantik
- **Port Types** (~interfaces)
 - können **Port Types** erweitern (*extends*)
 - bestehen aus **Port Type Operation Elements**
- **Port Type Operation Elements**
 - beschreiben eine **Operation** des Port Types
 - Name, Modus (in, out in-out)
 - Liste von Nachrichtentypen (~Parameter)

Grid Services der OGSA erweitern Web Services

- **Grid Service Instances** haben
 - eine Grid Service Description (~class)
 - einen Zustand (serviceData) (~Attribute)
 - einen Lebenszyklus
- **Service Data** (~Attribute)
 - besteht aus Name-Value Pairs

Grid Services der OGSA erweitern Web Services

- **Grid Service Handle**

- Sind globale Namen für Grid Service Instances
- Identifiziert global eindeutig und unveränderlich (immutable) eine Grid Service Instance
- Durch „resolve“ können GSR abgeleitet werden

- **Grid Service Reference**

- Netzwerk-Pointer auf Grid Service Instances
- Werden aus Grid Service Handles ermittelt (resolve)
- Können zum Aufruf von Grid Service Instances verwendet werden
- Können als Parameter verwendet werden

Festlegungen für Grid Services Handles in OGSA

- **Grid Service Handle**

- Syntax eines URI [RFC 2398]
- Jede Grid Service Instance hat mindestens einen Grid Services Handle

- **Grid Service Reference**

- Mehrere GSR können auf dieselbe Grid Service Instance verweisen
- Mehrere Resolutions eines GSH können verschiedenen GSR ermitteln (u.a. auch keine)

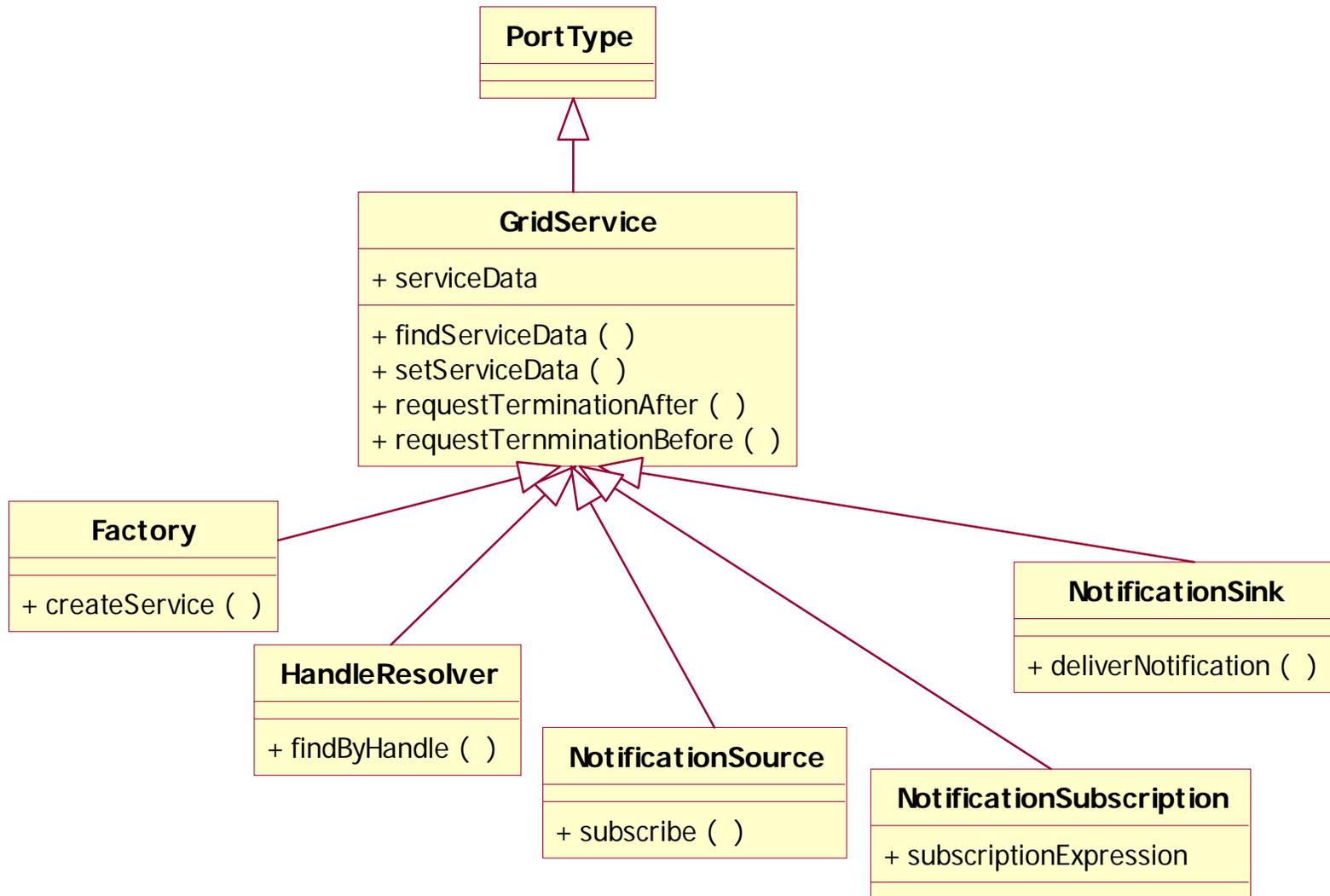
Konzepte der Grid Services der OGSA verglichen mit Corba

Grid	Corba
Grid Service	<i>Service Class</i>
Grid Service Description	IR::Interface
Grid Service Instance	<i>Service</i>
Port Type	Interface
Port Type Operation Element	Operation
Messages	Parameter
Grid Service Handle	Name
Grid Service Reference	IOR
Grid Service Attribute	Attribute

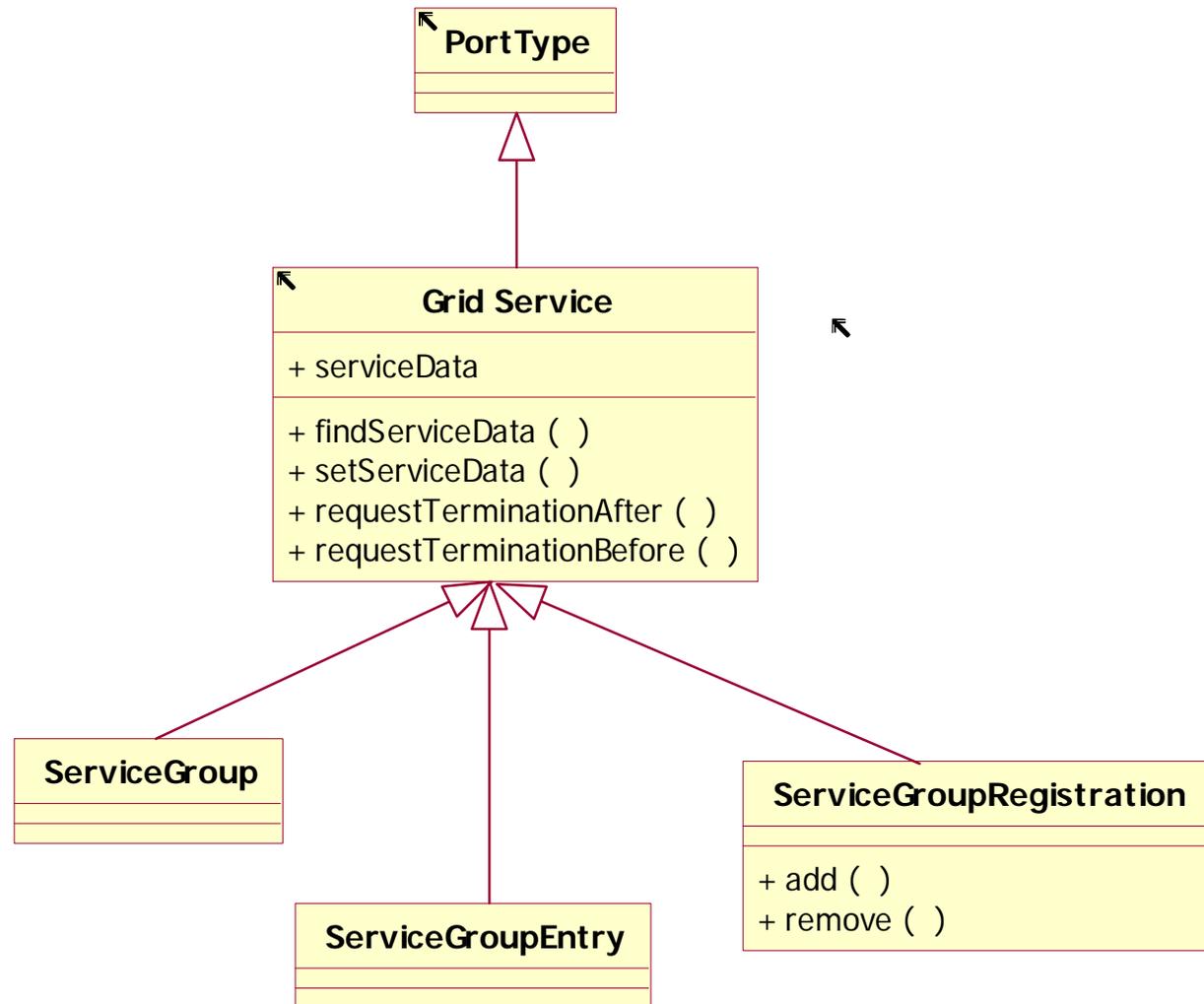
Grid Services – PortTypes of Standard Interfaces

- **GridService**
- **HandleResolver**
- **NotificationSource**
- **NotificationSink**
- **NotificationSubscription**
- **Factory**
- **ServiceGroup**
- **ServiceGroupEntry**
- **ServiceGroupRegistration**

Grid Services – PortTypes



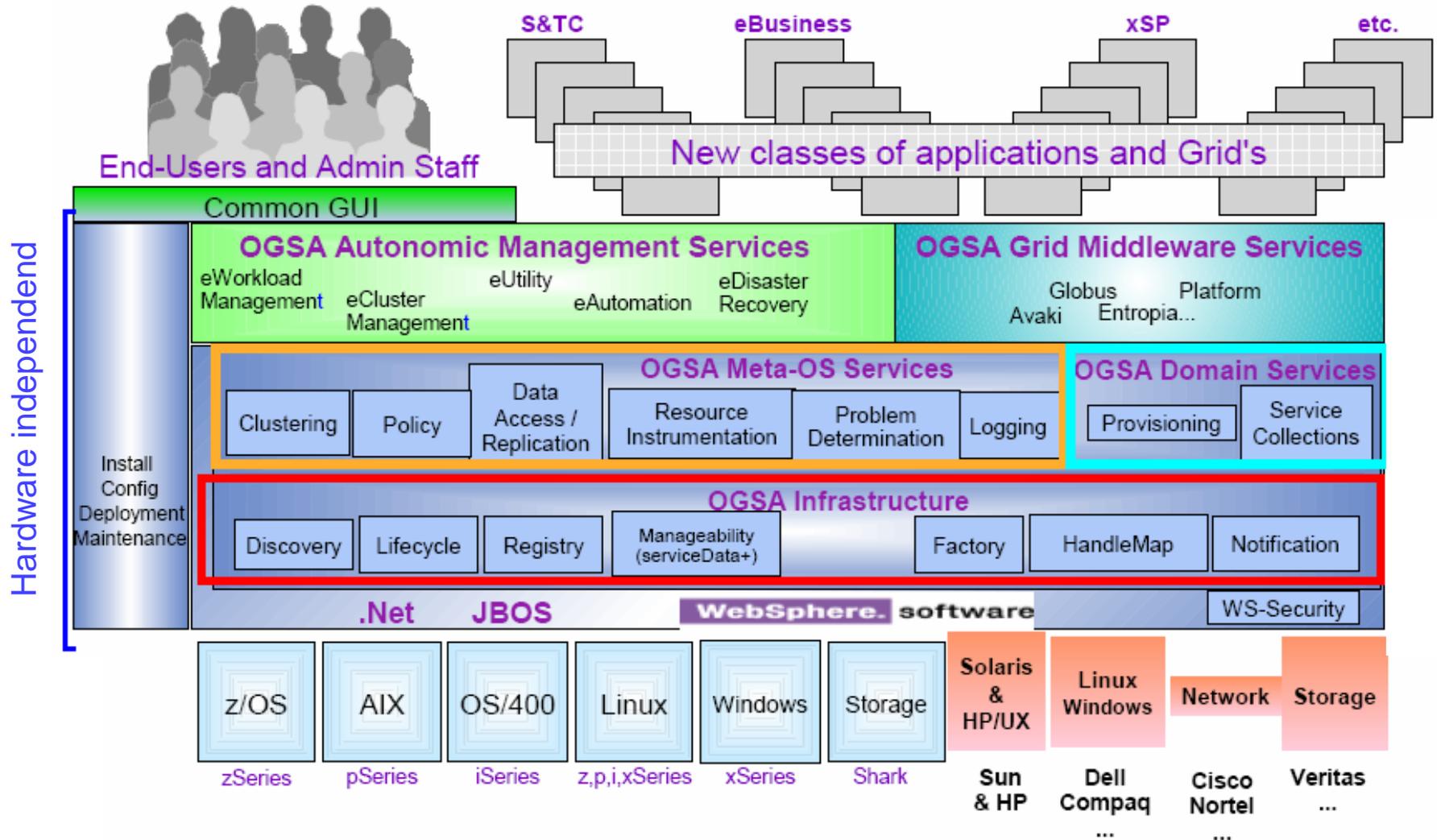
Grid Service – Port Types



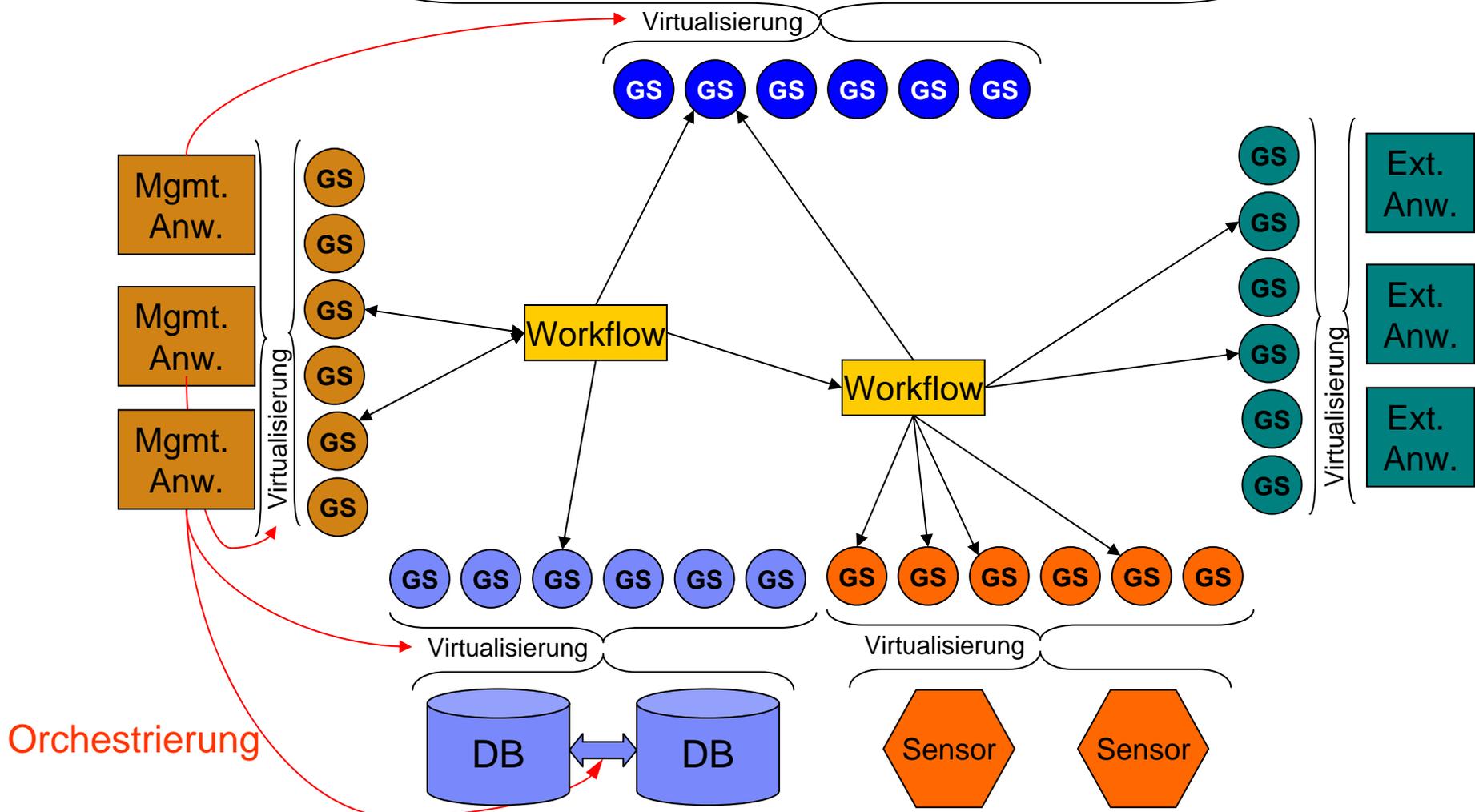
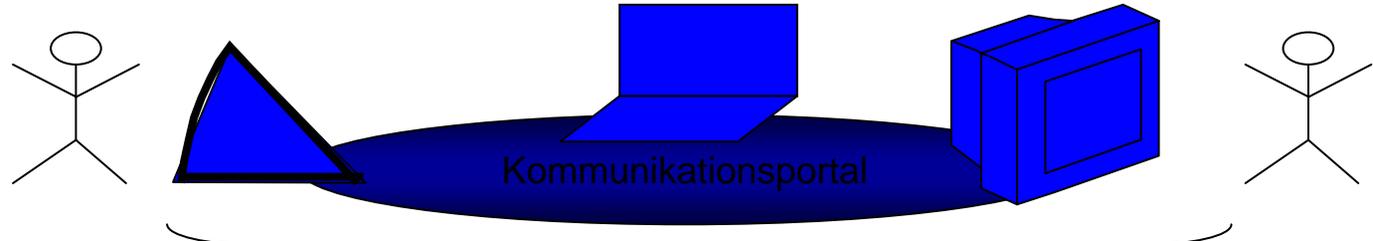
OGSA weitere Arbeitsthemen

- **Gleichheit / Identität von Grid Service Instances (,sameness‘)**
- **Service Evolution, Im Sinne von**
 - Erweiterung von Port Types
 - Erweiterung von Operationen
- **Gleichartige Behandlung von Fehlern (failures)**

Die Open Grid Service Architektur des Global Grid Forum

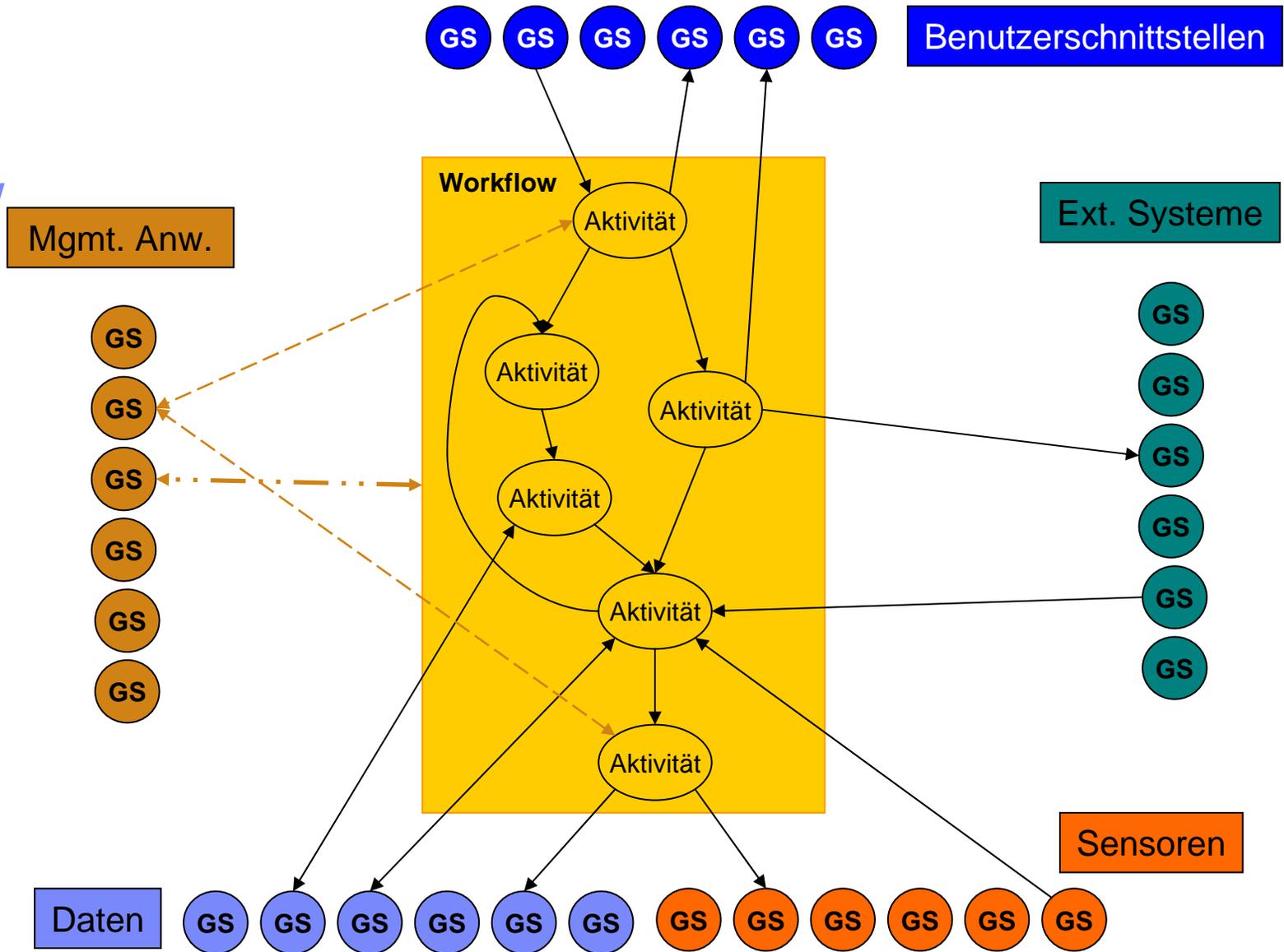


ein Grid

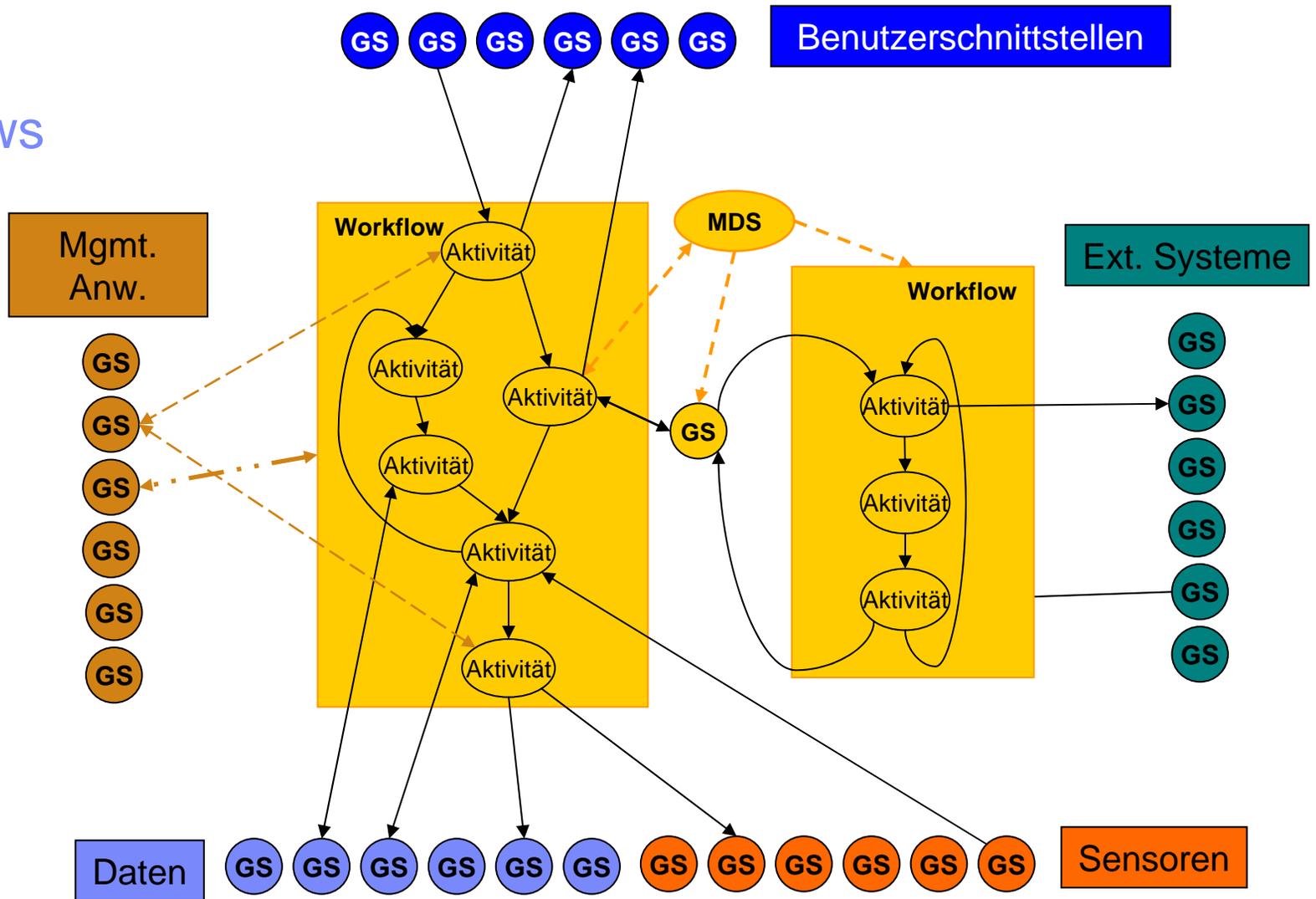


Orchestrierung

Ein Grid mit Workflow



Ein Grid mit Workflows



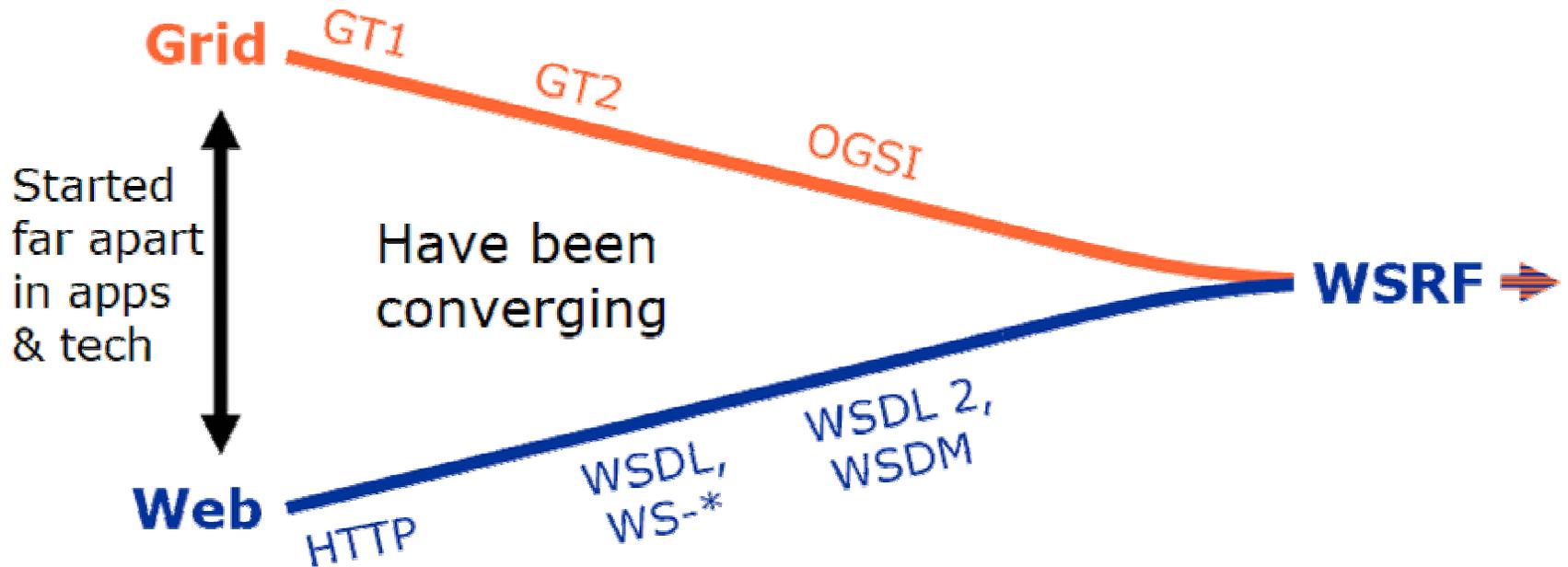
Grid Services – Wer steckt dahinter

- **Global Grid Forum**
<http://www.ggf.org/>
- **OASIS**
<http://www.oasis.org>
- **Gemeinsame Arbeiten mit Konsortien unter W3C**
<http://www.w3c.org>

Grid Services = Web Services ++

- **Grid Services bauen auf Web Services auf**
- **Grid Services fügen Konventionen und Abstraktionen aus verteilten Objekt-basierten Systemen hinzu:**
 - Objektreferenzen
 - Service Instanzen mit Zustand
- **Die Standardisierungsgremien der Grid Services führen diese Ideen in die fortlaufende Evolution der Web Services Standards ein.**

Konvergenz der Grid and Web Service Spezifikationen am Beispiel Web Service Resource Framework



Zur Diskussion

- **Gibt es Fragen ?**
- **Ist das Vorhaben der Grid-Leute sinnvoll ?**
- **Wieweit sollten Konzepte aus den verteilten, objektbasierten Systemen in eher grobkörnige Plattformen übertragen werden ?**