

# Services – Prinzipien und Taxonomie

München, 23.01.2007

Hartmut Wilms

[hartmut.wilms@innoq.com](mailto:hartmut.wilms@innoq.com)

OOP 2007 / München

- Senior Software-Architect
- innoQ Deutschland GmbH
- Verteilte Systeme
- SOA, Web Services
- Windows Communication Foundation
- Autor von Fachartikeln
- Sprecher auf Konferenzen
- <http://www.innoq.com/blog/hw>



# Services & SOA

Lassen Sie uns rekapitulieren ...

## Unternehmer/Manager-Sicht

- Unternehmensweites Konzept zur Integration und zur Veröffentlichung von Software
- Systeme und Dienste, die leicht anpassbar sind und selbst in nicht voraussehbaren Szenarien eingesetzt werden können
- Hilft der IT mit den ständig wechselnden Anforderungen des Geschäfts mitzuhalten

### Entwickler-Sicht

- Service als Schlüsselkonzept
- Lose Kopplung
- Nachrichten, Dokumente
- Metadaten
- Fokus auf
  - Service, Kommunikationsmuster
  - Contract & Policy
  - Standards & Konventionen

- Service
  - Schlüsselkonzept
- Orientiert
  - Philosophie (Paradigma)
  - Prinzipien und Grundsätze
- Architektur
  - Umsetzung des Konzepts ...
  - ... innerhalb der Grenzen der (externen) Anforderungen

1. Boundaries are explicit
2. Services are Autonomous
3. Services share Schema and Contract, not Class
4. Compatibility is based upon Policy

Don Box: “A Guide to Developing and Running Connected Systems with Indigo”

- Kopplung ist der Grad der (nicht-funktionalen) Abhängigkeit zwischen zwei Artefakten
- Lose Kopplung bedeutet Autonomie
- Kopplung hat mehrere Dimensionen
- Nicht alle Dimensionen können/sollten immer gleichzeitig lose gekoppelt sein

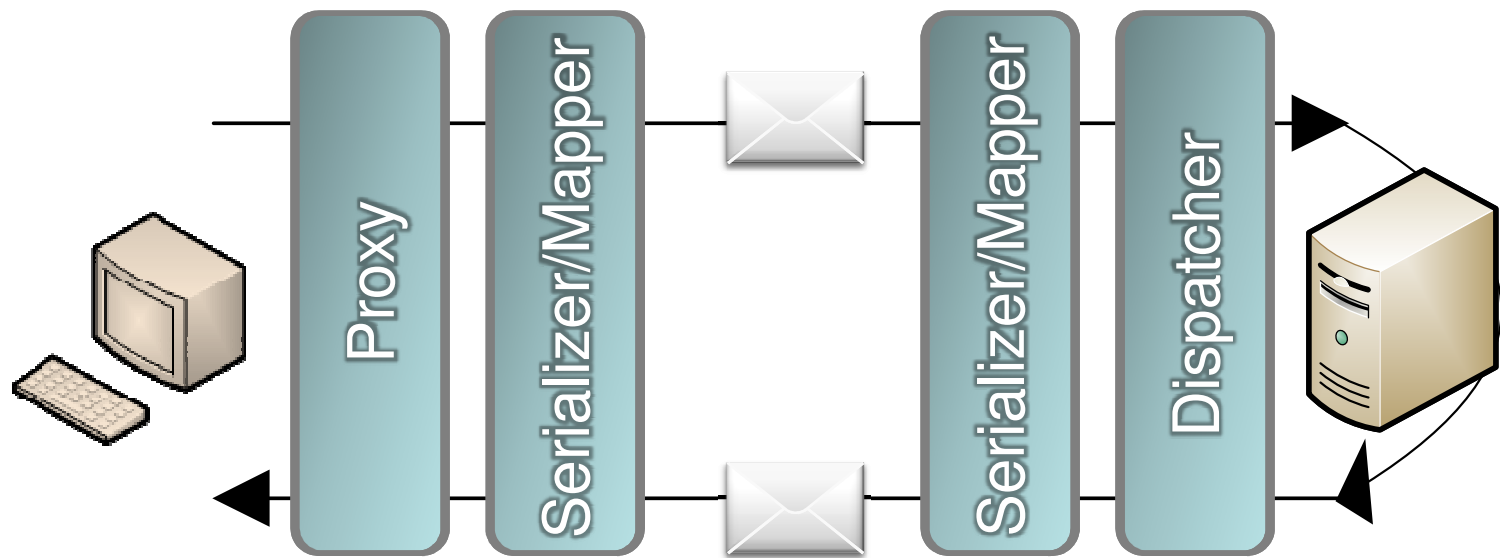


Dimension	Enge Kopplung	Lose Kopplung
Schnittstelle	Klassen und Methoden	Schema, Vertrag
Kommunikation	RPC	Nachricht, Dokument
Bindung	Fix, abgestimmt	Variabel, unabhängig
Zeit	Synchron	Asynchron
Interaktion	Direkt	Vermittelt
Verträge	Nicht-technische Absprache, implizit	Selbstbeschreibend, explizit
Verhalten	Geplant	Reaktiv

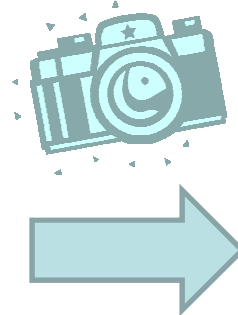
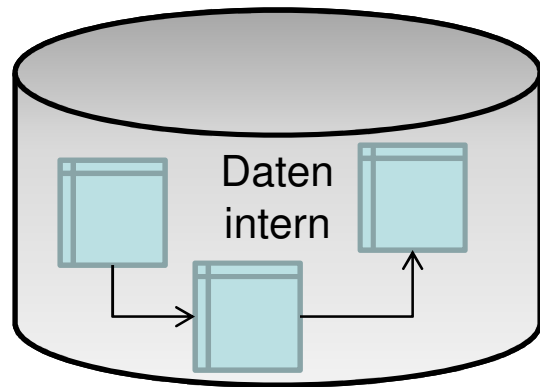
(vgl. [Carloz Perez: <http://www.manageability.org/blog/stuff/loosely-coupled-dimensions>])

# Vom Prinzip ...

- Nachrichten-Metapher
  - XML Messaging, SOAP
- Dokumenten-basiert
  - XML, XML Schema
- Metadaten
  - Contract (WSDL & XSD)
  - Policy (WS-Policy)
- Separation of Concerns



- Interne vs. externe Daten

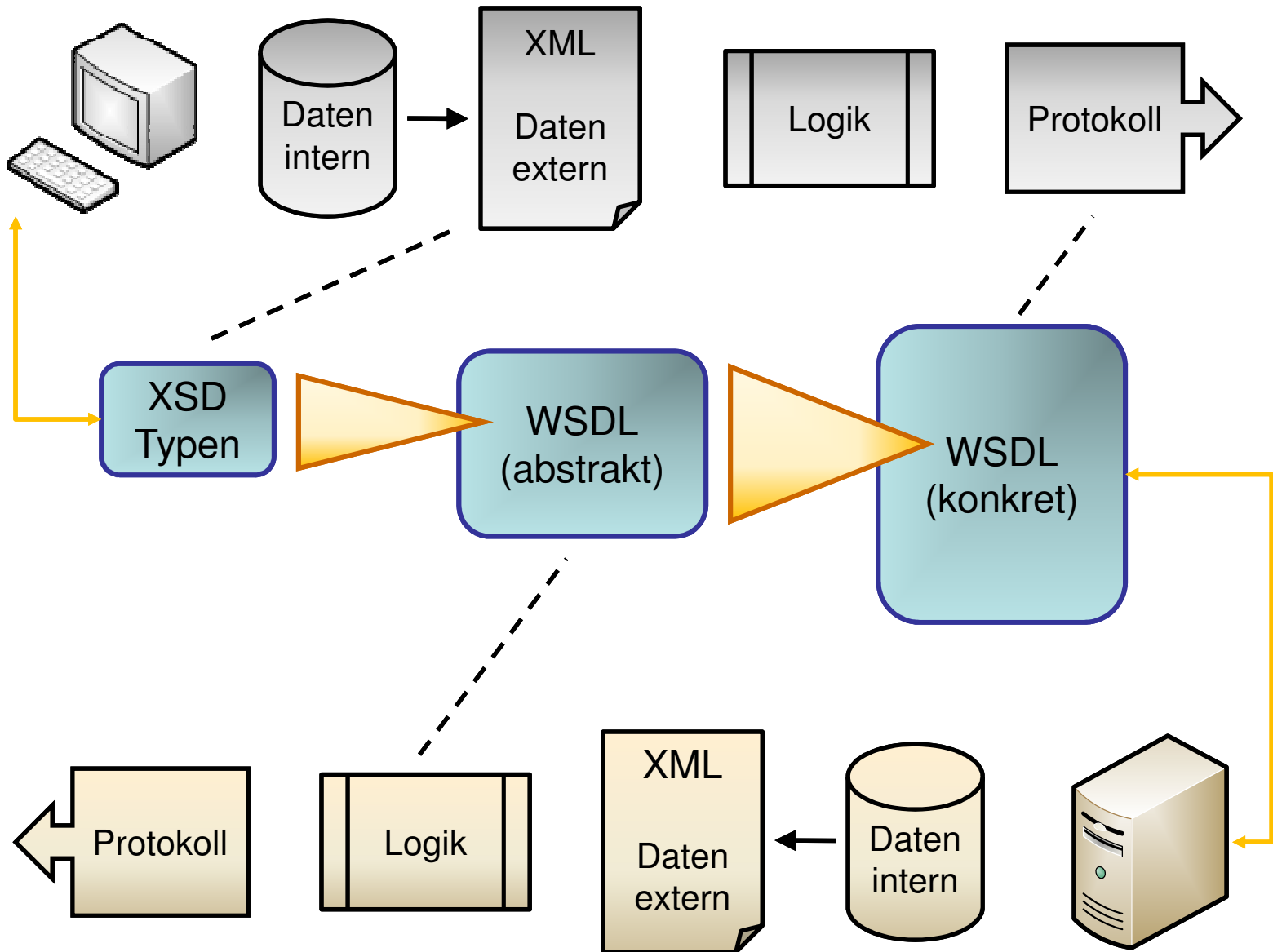


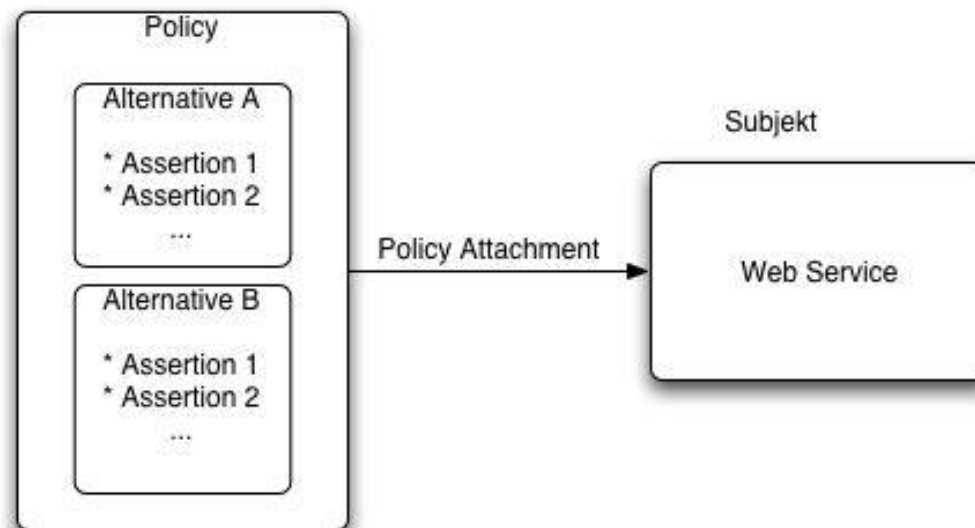
```
<myEntitySnapshot>
  <aggregatedData>
    <a>
    </a>
    <b minOccurs=0>
    </b>
    <aggregatedData>
  </myEntitySnapshot>
```

- XML, XML Schema
- Unabhängigkeit zwischen Contract, Schema, Dokument & Verarbeitung
  - ❑ Optionale Elemente
  - ❑ Teilbäume
- Explizite Constraints

## Beispiele

- Service-Beschreibung
- Service-Schnittstelle
- Organisatorische Einheiten
- Versionen
- Wiederverwendbare Schemata
- Adressen, Verträge, Policys

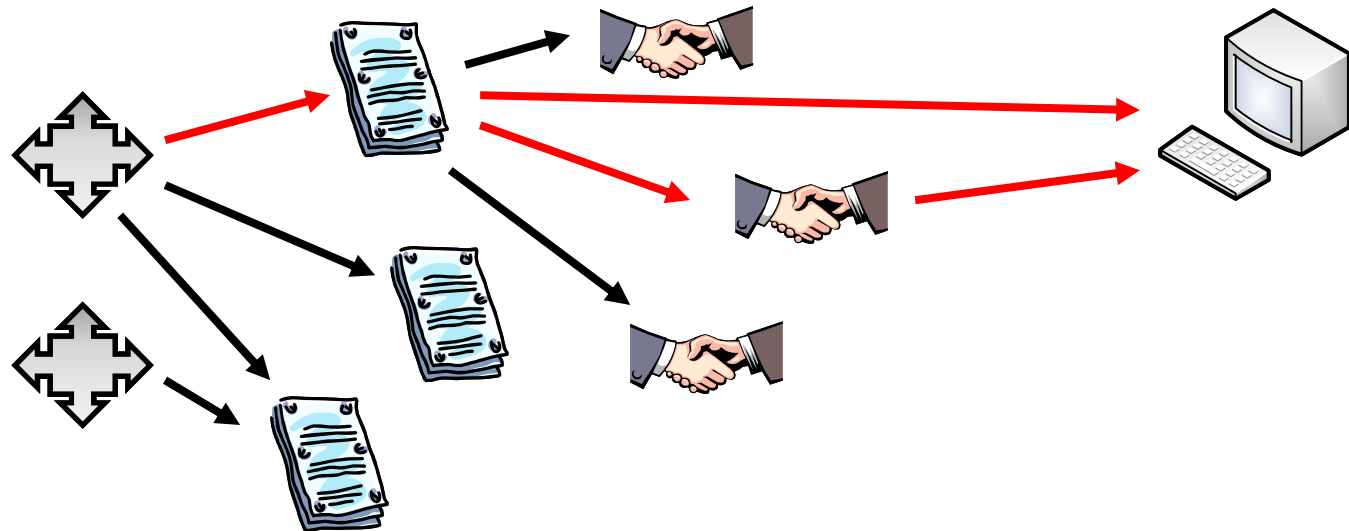






Beispiel: Trennung nicht-funktionaler Aspekte von der Funktionalität

- Trennung von Implementation & Contract
- Trennung von Contract & Policy



Service	Objekt
Heterogene Umgebung	Homogene Laufzeitumgebung
Schema & Contract	Spezifischer Typ & Interface
Asymmetrisch, Verzögerungen	Symmetrisch, schnelle Interaktion
Laufzeit-Bindung	Linker & Loader
(Indirekte) Adressierung	(Direkte) Objektreferenzen
Modell der Fachlichkeit	Implementation des Service

## ➤ Makrokosmos

- ❑ Repräsentiert komplexe, übergreifende Funktionen
- ❑ Hohe Abstraktion
  - Funktionen & Ergebnisse, Akteure
  - Was, Wer, Wo

## ➤ Mikrokosmos

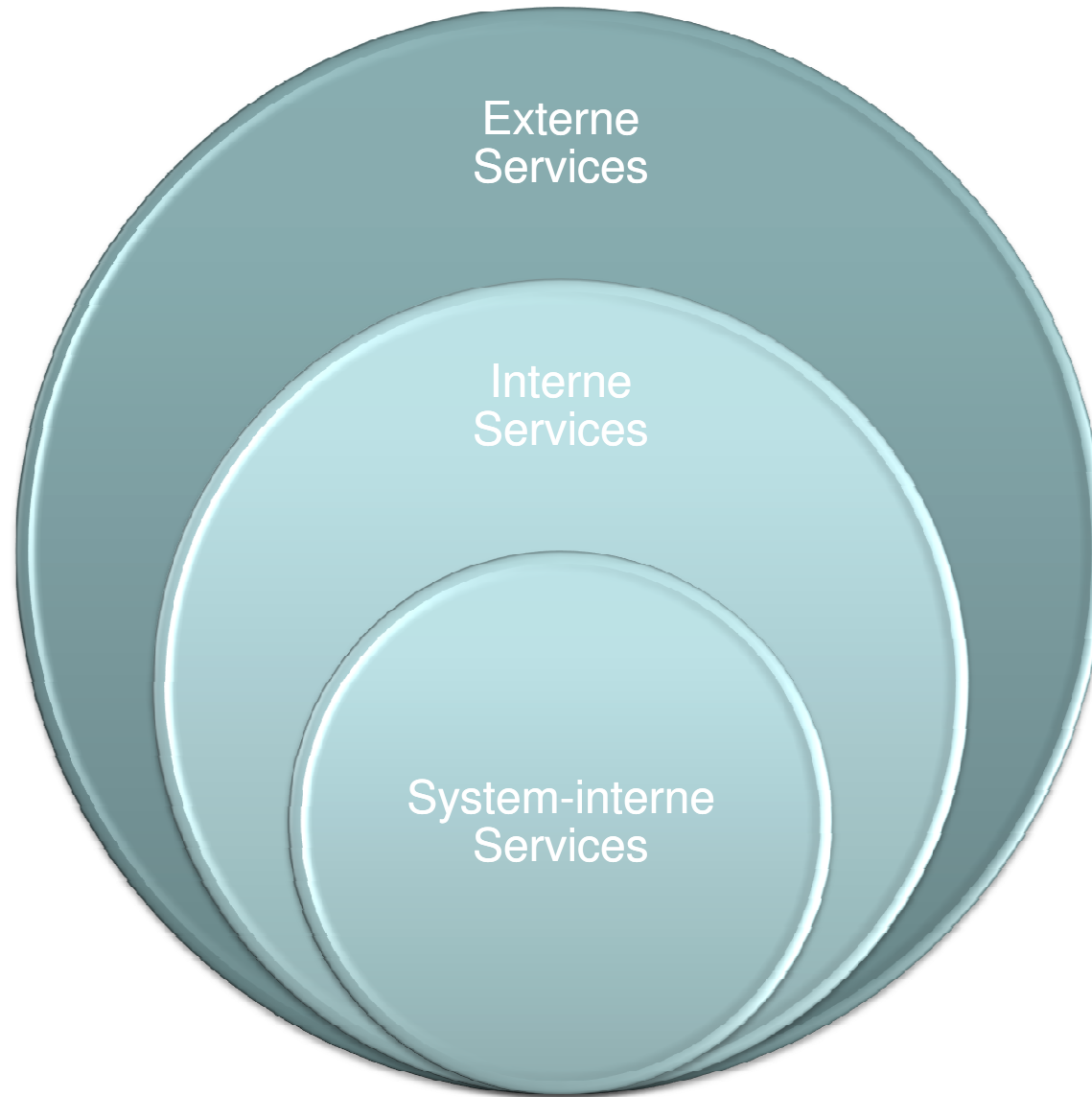
- ❑ Repräsentiert „Dinge“
- ❑ Konkretes Verhalten
  - Wie

# ... über die Taxonomie ...

(griech. taxis „Ordnung“, -nomia „Verwaltung“)

- Klassifizierung, Kategorisierung von Services
  - Entwicklung
  - Einsatz
  
- (Business-)Service-Registry
  - Yellow Pages
  - Einordnung nach fachlichen Aspekten

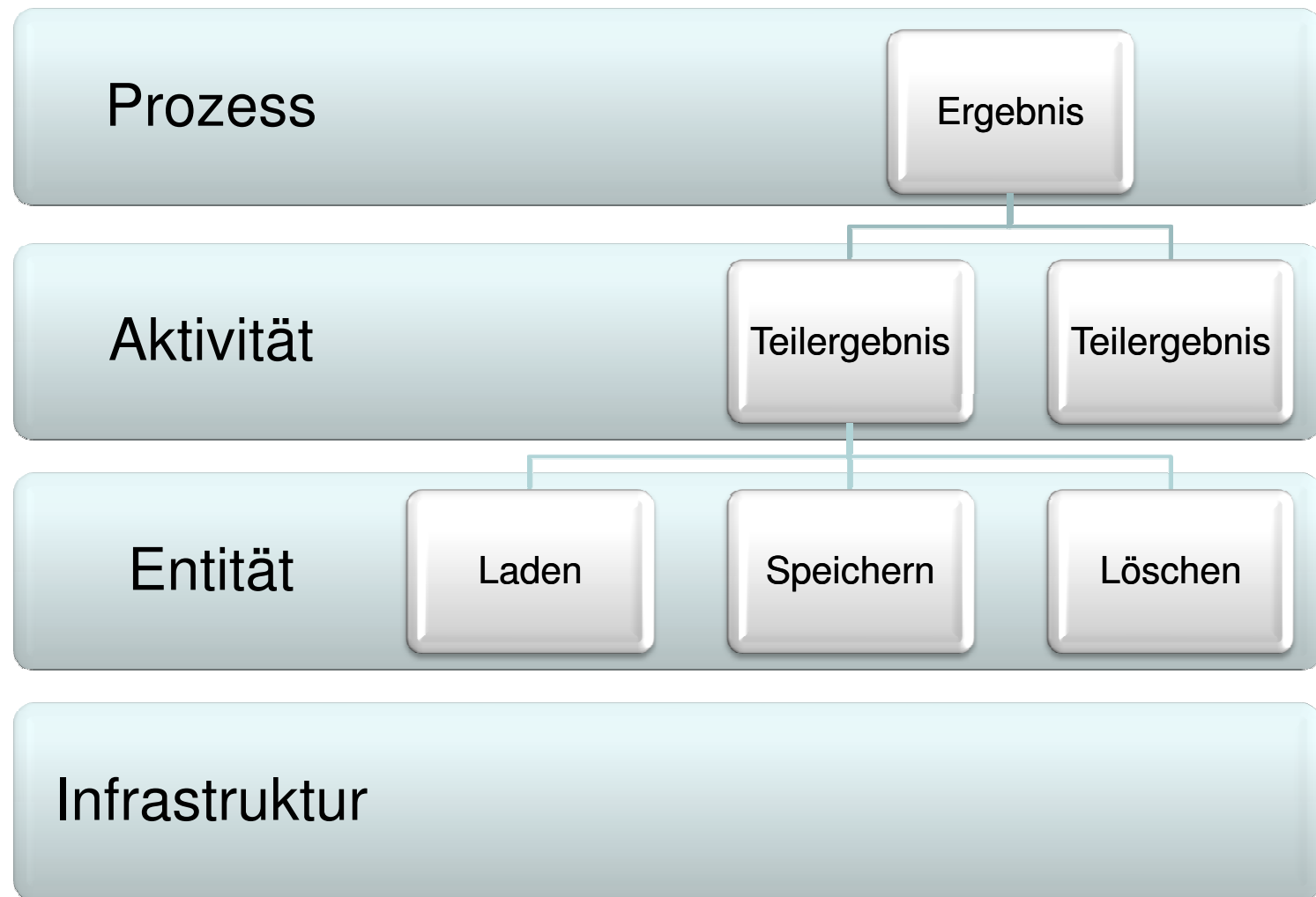
- Umgebung
- Anwendungsfall
- Qualität und Quantität von Änderungen
- Funktionalität und Granularität



- Service-oriented Enterprise
- Integration
- Einfach gute Software



- Änderungshäufigkeit
- Qualität der Änderung
  - Daten
  - Algorithmen
  - Prozesse

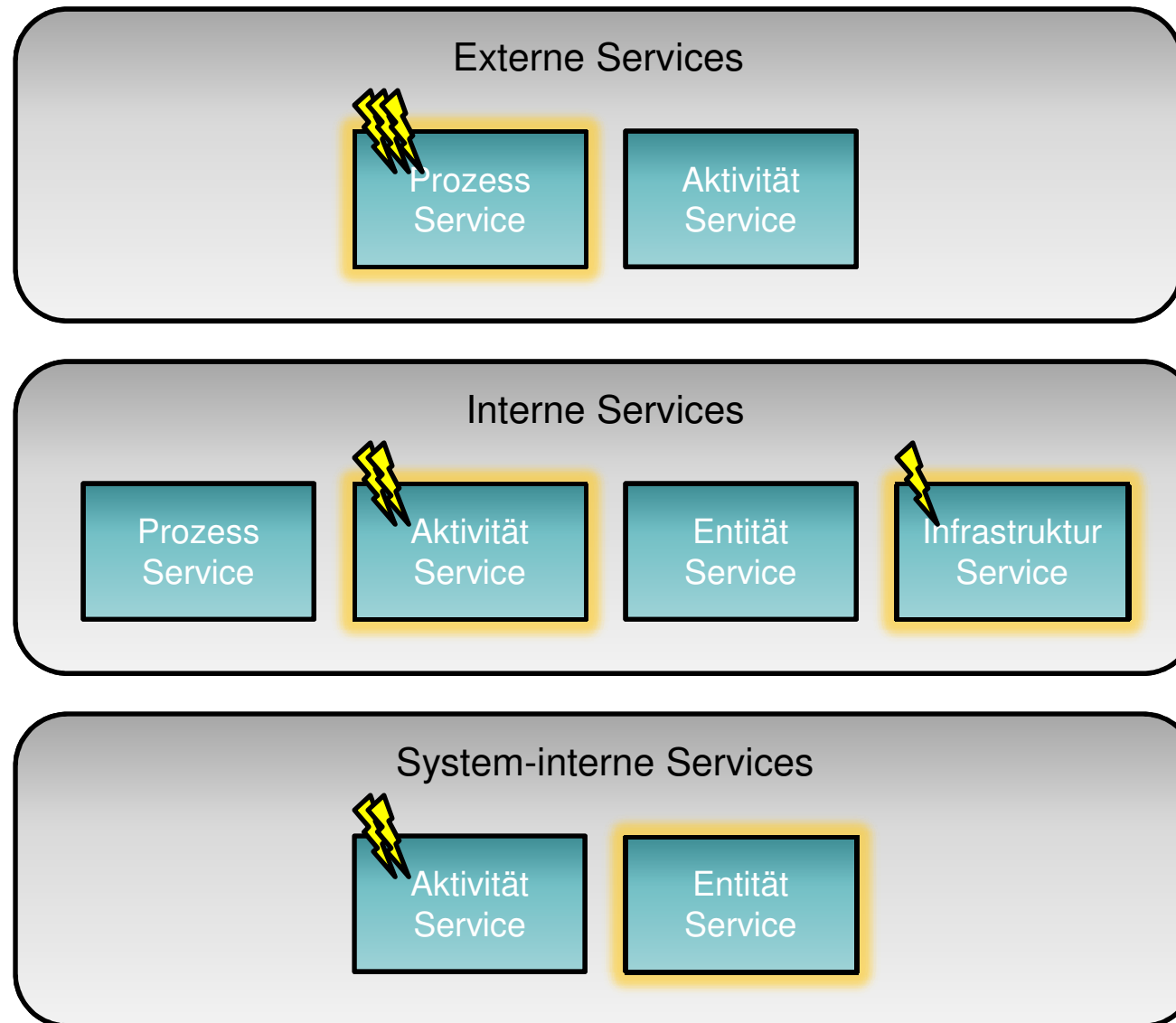


- Geschäftsprozess
- Komplex, langläufig
- Zusammengesetzte Services können wieder zusammengesetzt werden
- Orchestration autonomer Aktivitäten
- Hauptziel für Optimierungen und Änderungen

- Autonome Geschäftsaktivität
- Jede Aktivität repräsentiert eine logische „Unit of Work“ mit einem bewertbarem Ergebnis
- Teil eines Prozesses oder eigenständig
- relativ stabil

- Verwaltet und repräsentiert Entitäten und Assoziationen
- Sorgt für Datenkonsistenz innerhalb der Autonomiegrenzen
- Schema
- Stabilität wird durch Schema bestimmt

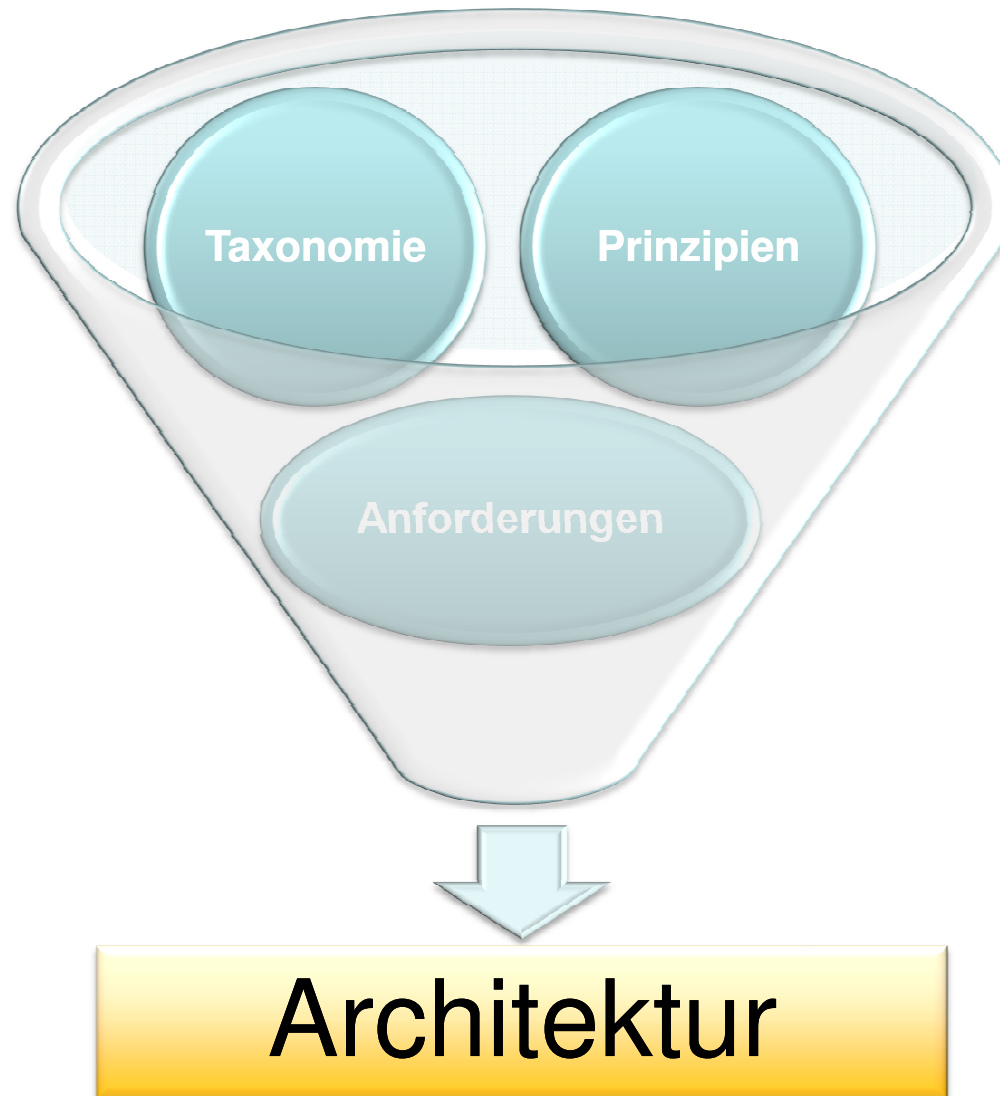
- Isolierte technische Dienste, die keine anderen Services aufrufen
- Wird von anderen Services genutzt



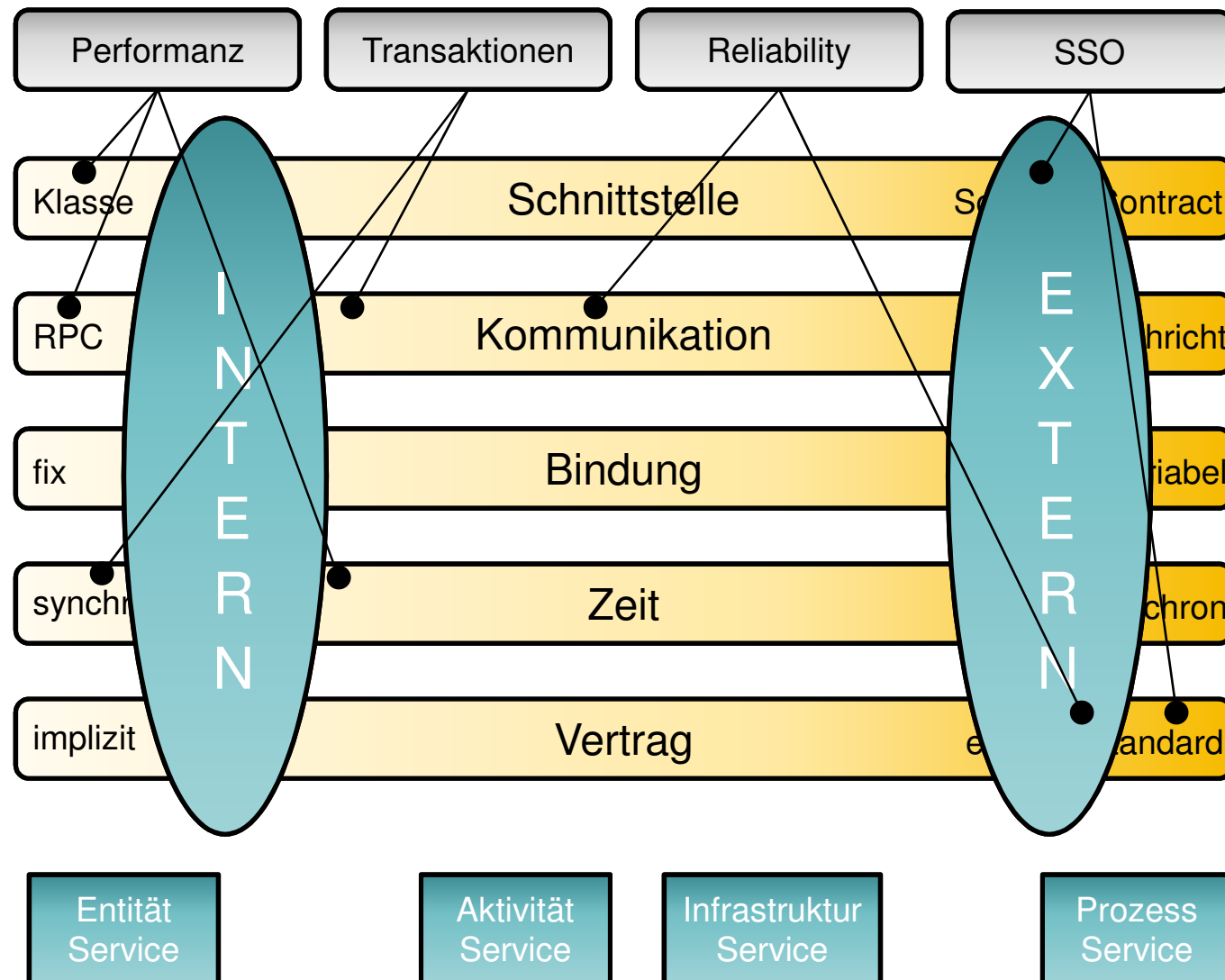
- (Nicht-funktionale) Anforderungen und Service-Kategorien schließen sich teilweise aus
  - ❑ Performanz, Reliability, Transaktionen
  - ❑ Integration von existierenden Komponenten
  - ❑ Know-how
  - ❑ Politik
- Einige Kopplungsdimensionen können nicht erfüllt werden
- Dimensionen ohne Einfluss auf die Anforderungen sollten lose gekoppelt bleiben



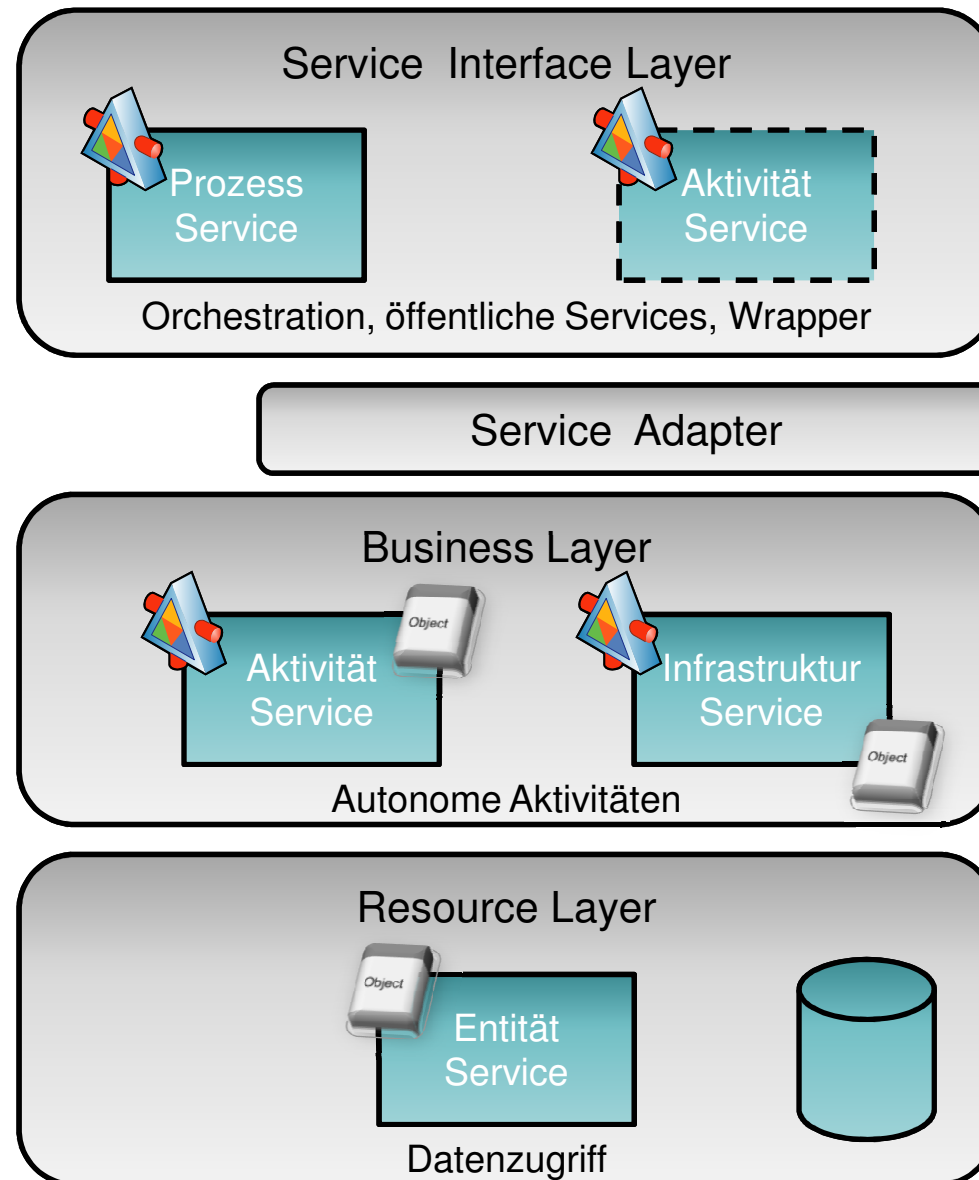
## ... zur Architektur



# Dimensionen, Kategorien und Anforderungen



- Wie und von wem wird der Service genutzt?
  - ❑ extern oder intern
  - ❑ Trennung Implementation & Interface/Fassade
- Integration von existierenden Komponenten?
  - ❑ Heterogenes oder homogenes Umfeld?
- Welche Anforderungen existieren?
  - ❑ Abstimmung Anforderungen-Dimensionen
- Konzept der Implementation sollte sich immer nach Service-orientierten Prinzipien richten!



- SOA ist eine Philosophie bzw. ein Paradigma
- Lose Kopplung und die Nachrichten/Dokumenten-Metapher sind die wichtigsten Prinzipien der SO
- Sorgfältiges Abwägen der Kopplungsdimensionen und Anforderungen
- Aufbau einer Service-Taxonomie und Spezifikation der technologischen Umsetzung
- Prinzipien und die Service-Taxonomie bestimmen in den Grenzen der Anforderungen die Architektur

**Danke!**



## Fragen?

Hartmut Wilms

innoQ Deutschland GmbH

Halskestraße 17

D-40880 Ratingen

Tel +49 2102 77 1620

Fax +49 2102 77 1601

<http://www.innoq.com>

<http://www.innoq.com/blog/hw>

- Grundsätze der Service-Orientierung
  - ❑ <http://msdn.microsoft.com/msdnmag/issues/04/01/indigo/default.aspx>
- Kopplungsdimensionen
  - ❑ <http://www.manageability.org/blog/stuff/loosely-coupled-dimensions>
- Architektur & Design
  - ❑ Contract-First
    - <http://msdn.microsoft.com/msdnmag/issues/05/05/ServiceStation/>
  - ❑ Daten
    - <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnbda/html/dataoutsideinside.asp>
  - ❑ Service Factory
    - <http://msdn.microsoft.com/msdnmag/issues/06/12/ServiceStation/default.aspx>