

DAIMLERCHRYSLER

Web Services – Vision, Potential und Standards

Mario Jeckle

DaimlerChrysler Forschungszentrum Ulm

mario.jeckle@daimlerchrysler.com

mario@jeckle.de

www.jeckle.de

Gliederung

- Web Services: Grundidee und Einsatzgebiete
- Architektonische Aspekte
- Rolle und Bedeutung von Standards
- Existierende Standards und Umsetzungen

Web Services: Grundidee und Einsatzgebiete

Was ist ein Web Service?

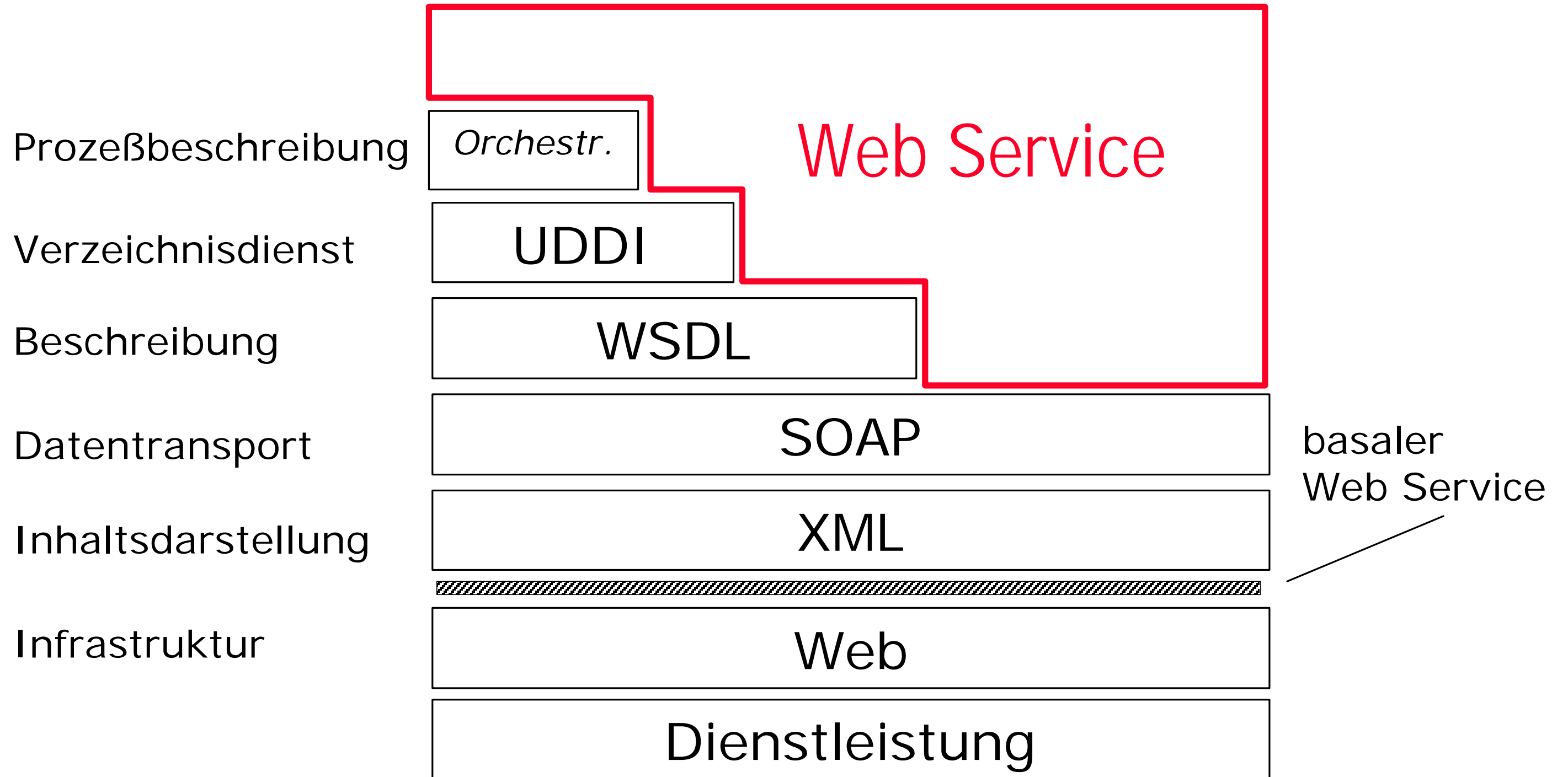
- Intel: Web Services sind eine Technik zur Kommunikation zwischen heterogenen Geräten.
- Microsoft: Über das Internet auslieferbare Softwarelösung auf der Basis von XML.
- SUN: Applikationsfunktionalität die über das Web zugreifbar wird.
- ... sowie ca. 50 weitere Definitionen

Web Services: Grundidee und Einsatzgebiete

Was ist ein Web Service?

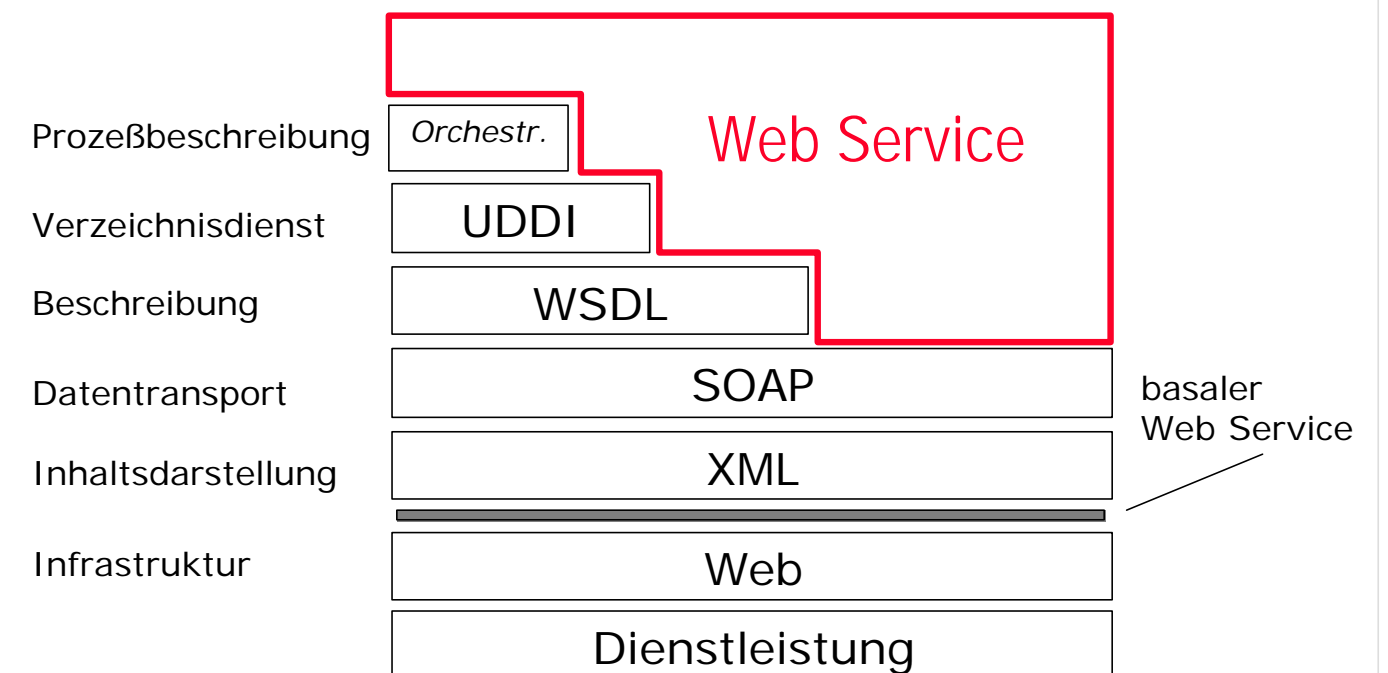
- Definitionsdimensionen
 - Strikt technisch
 - Gefahr des vorzeitigen Veraltens
 - Strikt funktional
 - Gefahr der mangelnden (technischen) Trennschärfe
- Einbezug des kulturellen Kontexts:
 - US-Amerikanische Definitionen: eher funktional
 - Europäische Definitionen: eher technisch

Web Services: Grundidee

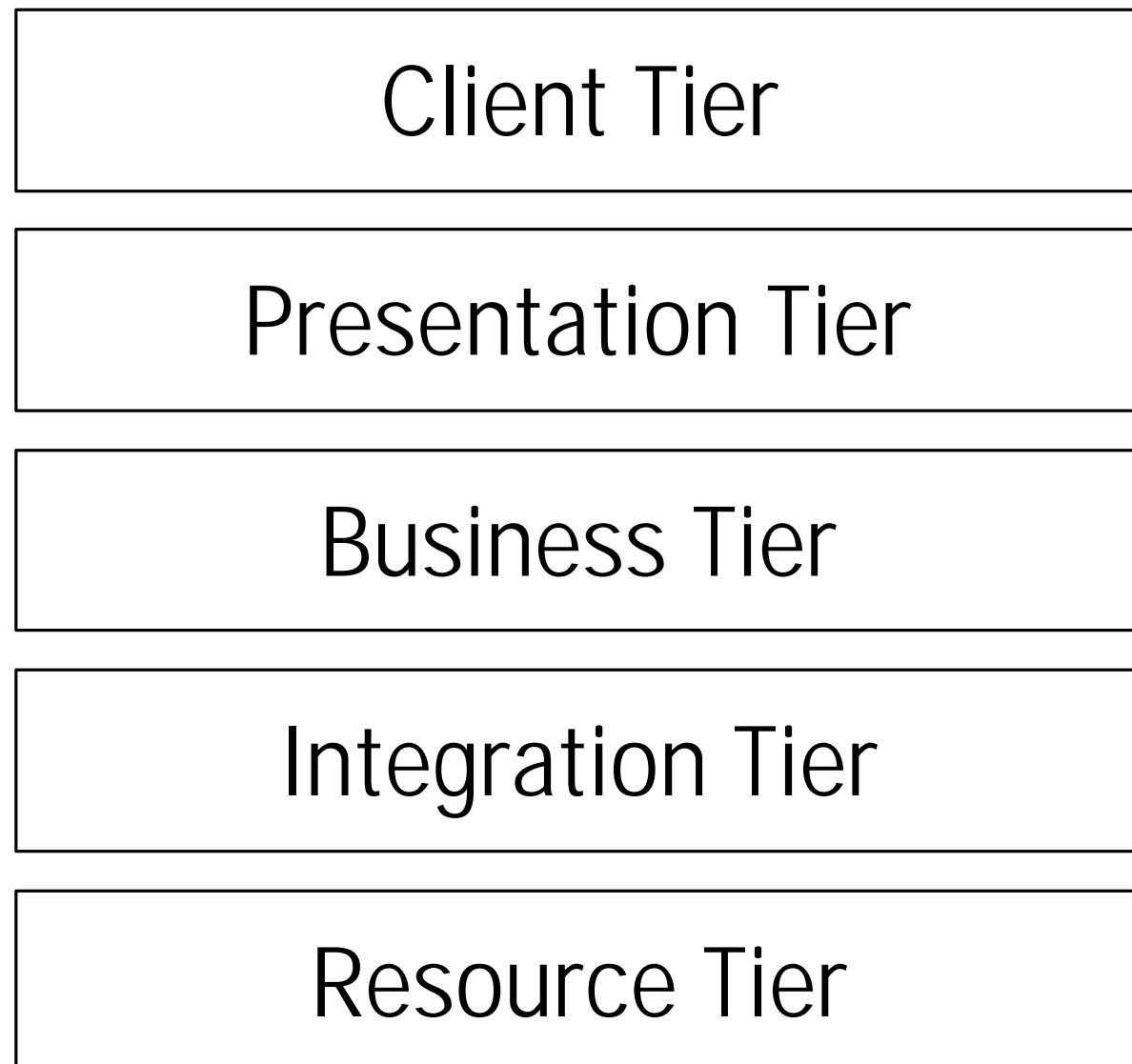


Web Services: Einsatzgebiete

- Datenaustausch zwischen heterogenen Applikationen (A2A)
- Integration verschiedener Systeme in-house und extern (B2Bi)
- Zugriff auf (ausgewählte) unternehmensinterne Daten (B2E)
- Flexible Ad-hoc Kollaboration (B2B)
- Kundenbeziehungen (B2C)

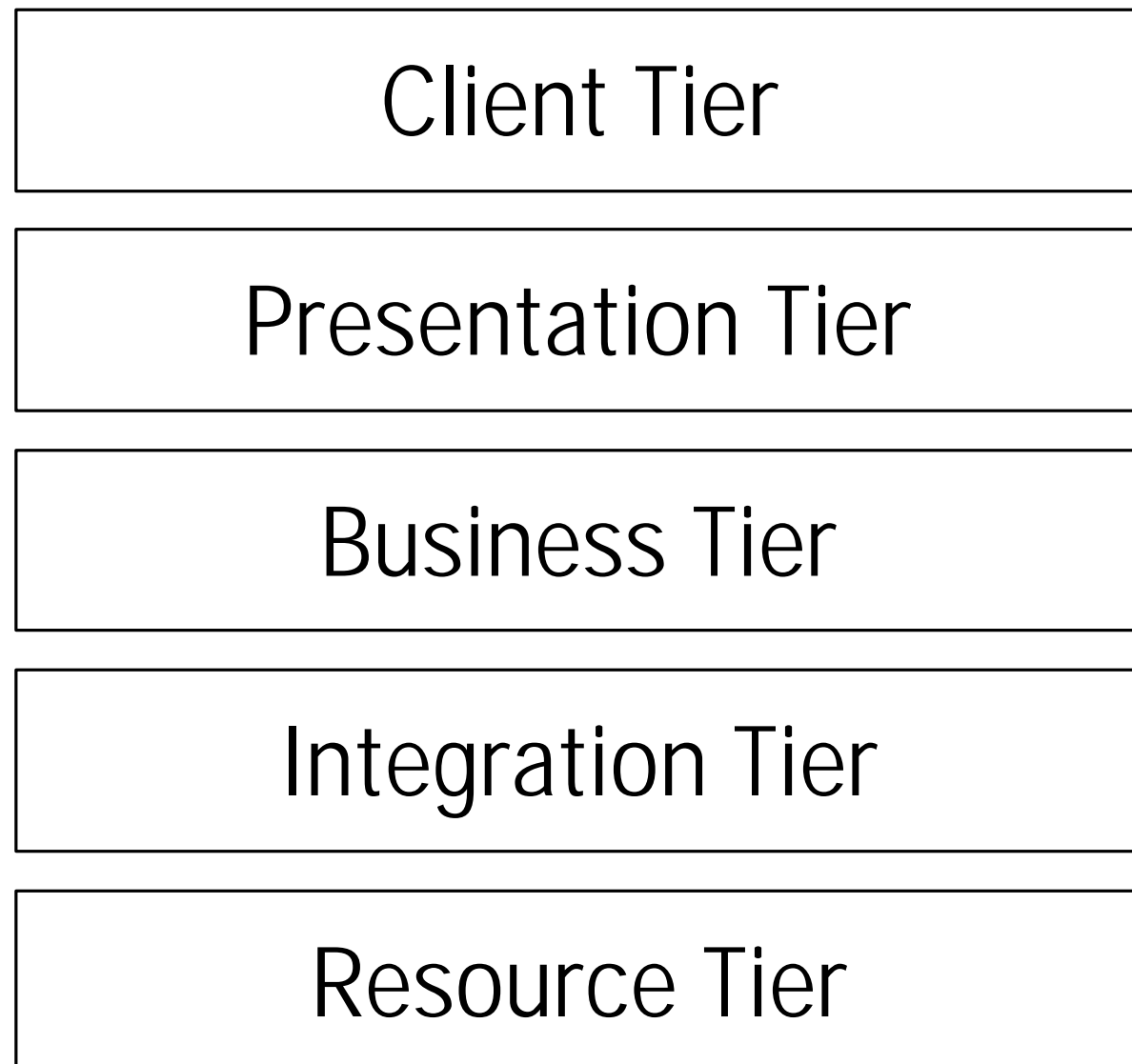


Architektonische Aspekte



Implementierung der Präsentationslogik, die benötigt wird um native Darstellungen für angebundene Clients zu erzeugen (z.B. Servlets, JSPs, Publishing Framework, **Web Services**)

Architektonische Aspekte



Verarbeitung der für die Clients zugreifbaren Businesslogik und Businessdaten (z.B. EJB, COM-Komponenten, **Web Services**)

Architektonische Aspekte

Client Tier

Presentation Tier

Business Tier

Integration Tier

Resource Tier

Kommunikation mit den Datenhaltung,
Reformatierung und Syndikatisierung der
Businessdaten
(z.B. ODBC, J2EE Konnektoren, **Web Services**)

Architektonische Aspekte

Client Tier

Presentation Tier

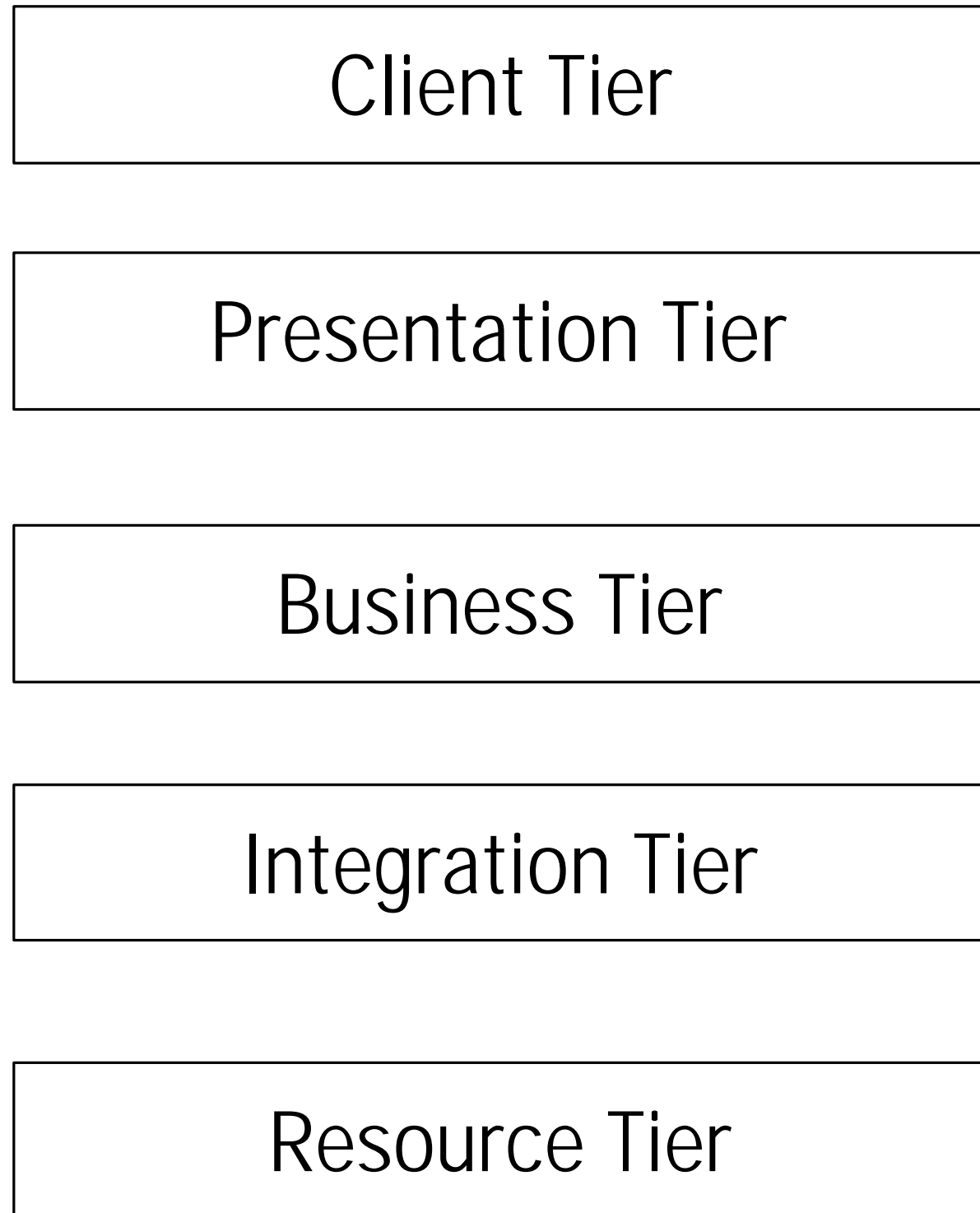
Business Tier

Integration Tier

Resource Tier

Physischer Zugriff auf Businessdaten
(z.B. Legacy Integration, B2B, **Web Services**)

Architektonische Aspekte



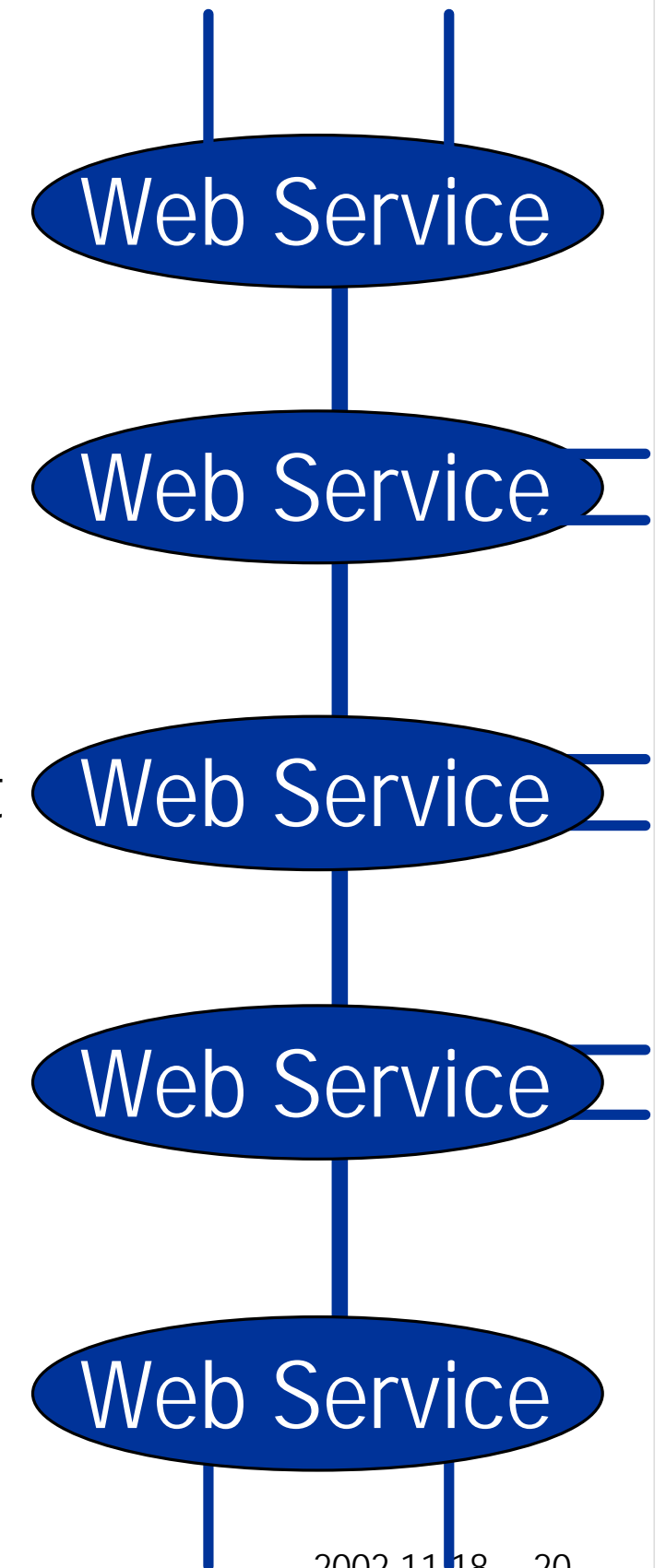
Client „spricht“ SOAP

Clientspezifische Datenaufbereitung

Businesslogik durch Web Services implementiert oder gekapselt

Sichtenbildung auf Inhalte

Zugriff auf Wirksysteme, Legacyintegration



Web Service Architektur

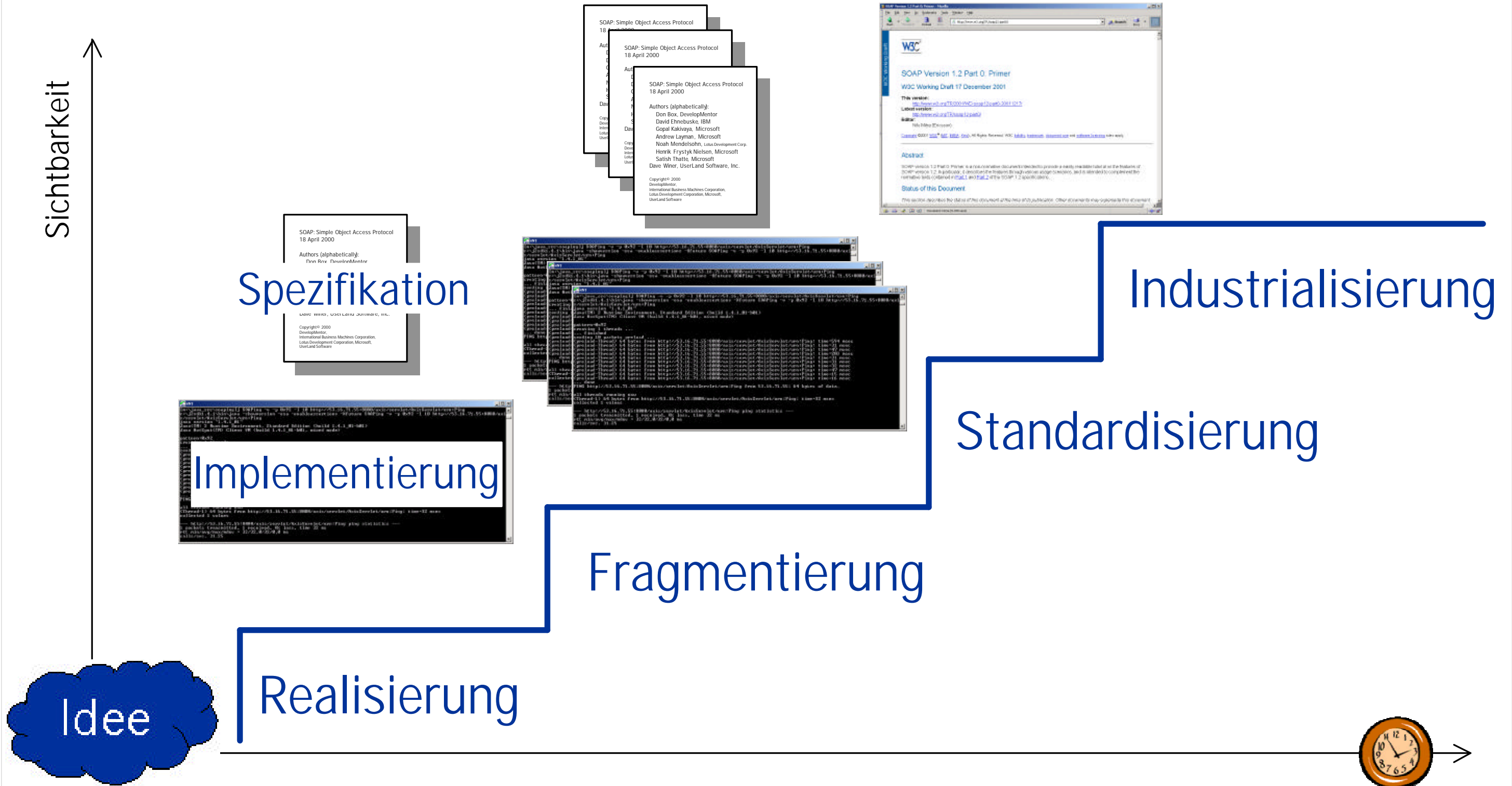
- Web Services bilden ein verteiltes System
- Web Services sind (zumeist) in eine existierende Systemwelt integriert oder zu integrieren
- Web Services sind kein kommerziell verfügbares Produkt
- Web Services sind nicht genau eine homogene integrierte Architektur
- Web Services beruhen auf einer Reihe eigenständiger Standards und Ideen, die in eine Architektur zu integrieren sind

Rolle und Bedeutung von Standards

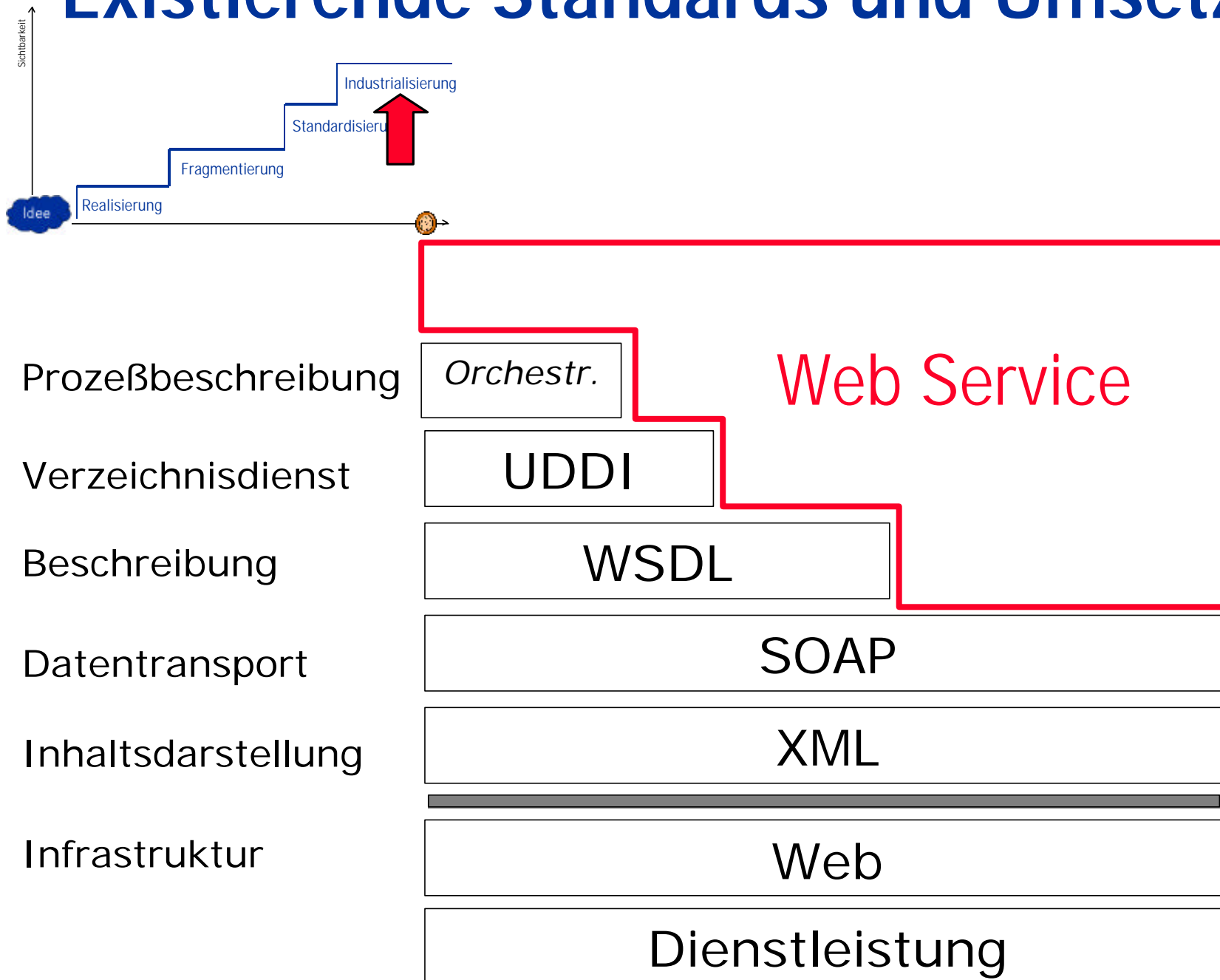
- Standard sind im Web Services-Umfeld ...
 - ... die Initialzündung (→ SOAP)
 - ... unabdingbar (→ Interoperabilitätsgarantie)
 - ... breit unterstützt (→ SOAP, WSDL)
 - ... ein Politikum (→ Web Service Interoperability Organization)
 - ... immer zu spät (→ SOAP, WSDL, UDDI, Orchestration)
 - ... kostenfrei (→ W3C, OASIS, Firmen)
 - ... leichtgewichtig/praxisnah (→ SOAP, WSDL)

Rolle und Bedeutung von Standards

Sichtbarkeit

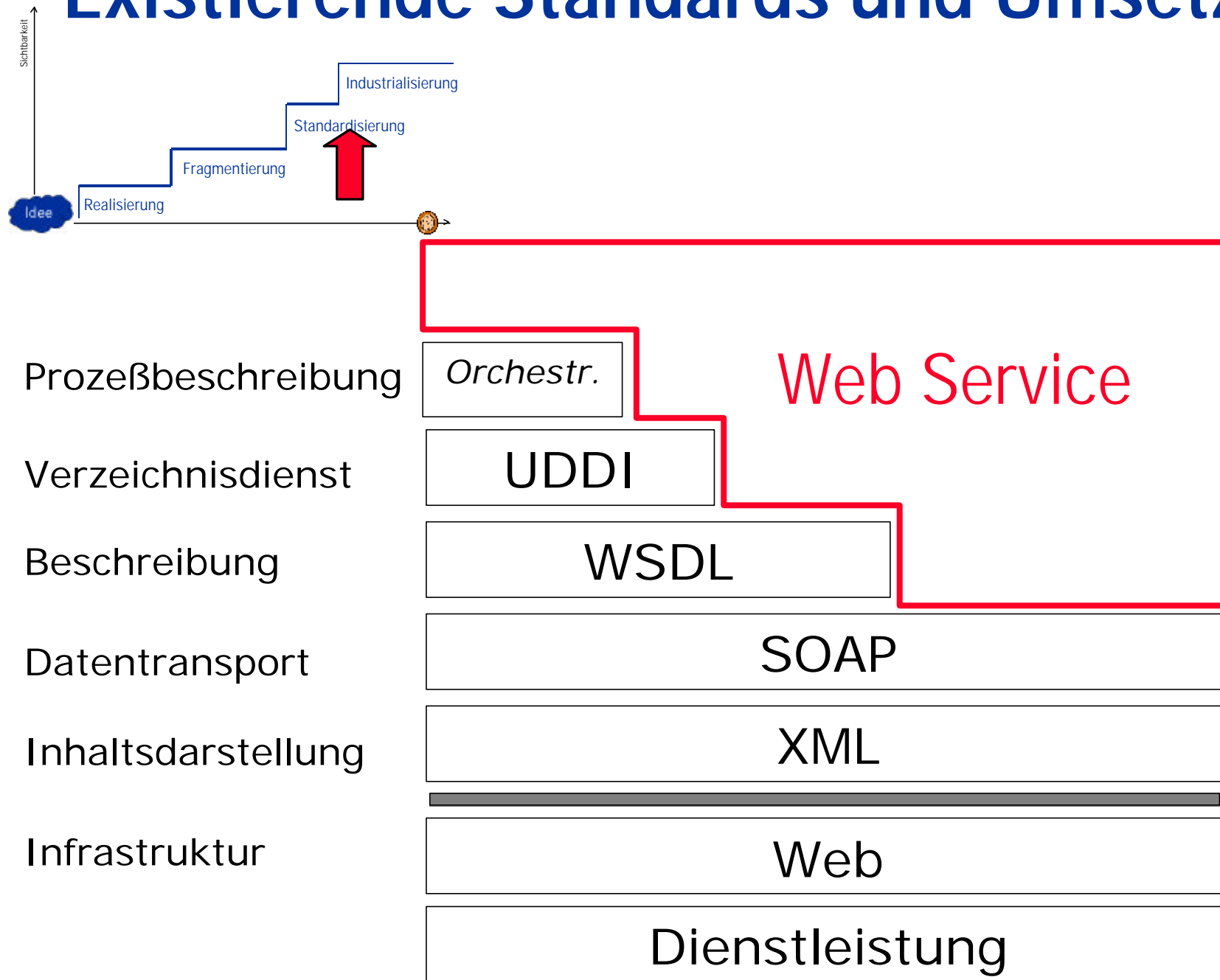


Existierende Standards und Umsetzungen



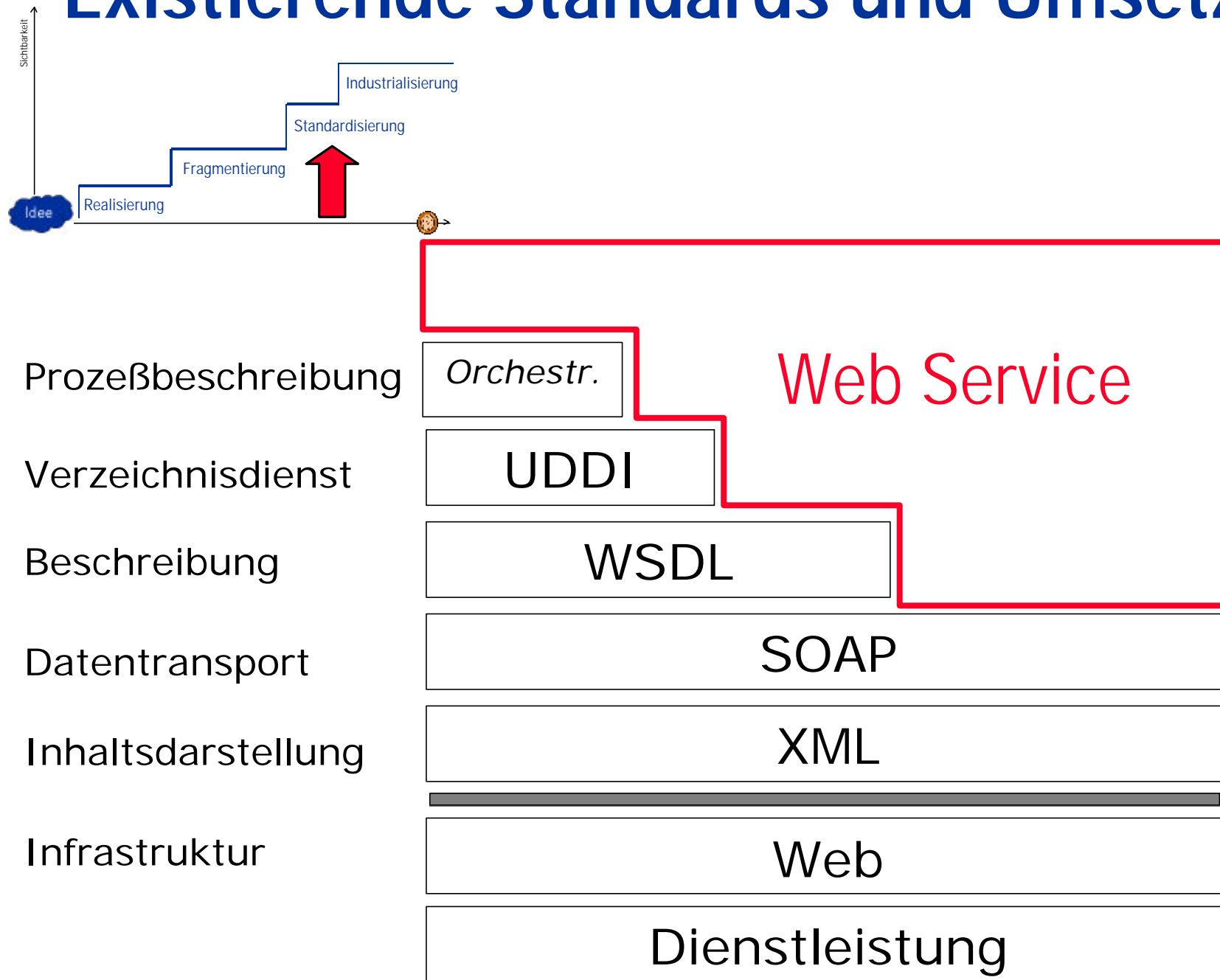
- XML v1.0 2nd edition
- Namensräume
- XML Schema
- Implementierungen verfügbar

Existierende Standards und Umsetzungen



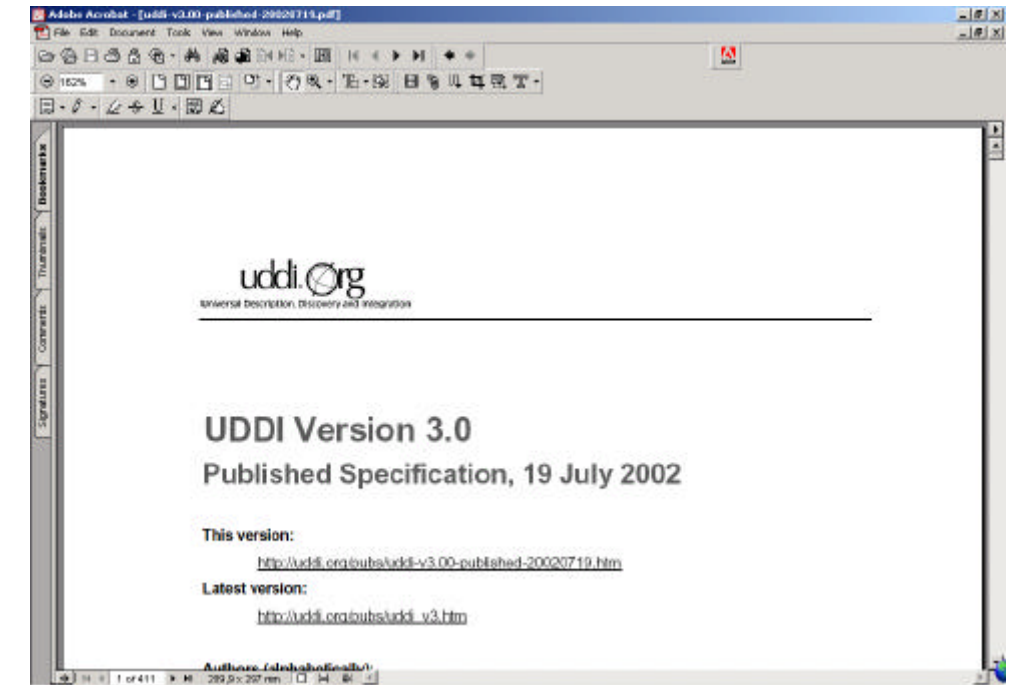
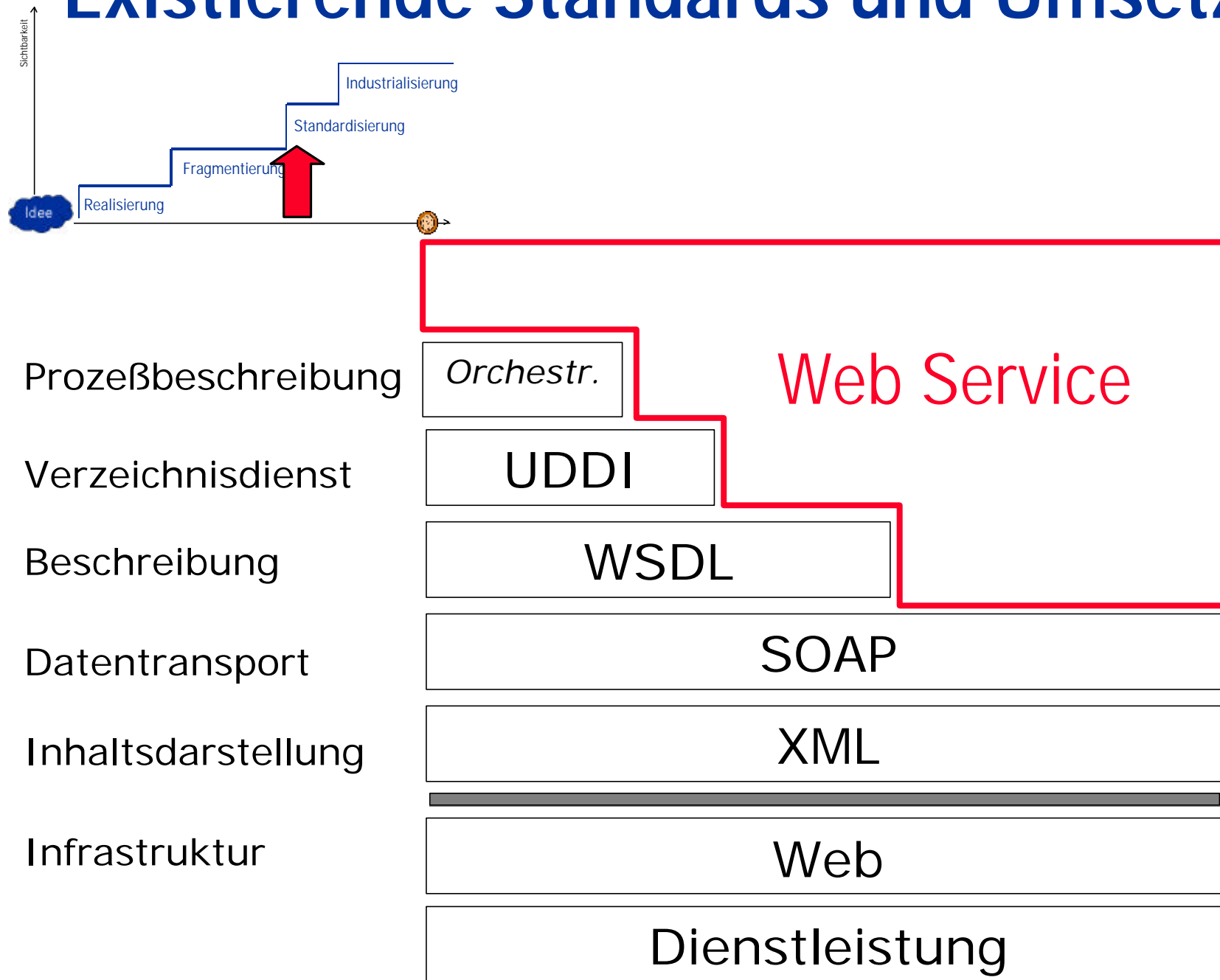
- SOAP v1.2 (W3C)
- Transaktionen (OASIS)
- Sicherheit (W3C, et al.)
- Implementierungen verfügbar

Existierende Standards und Umsetzungen



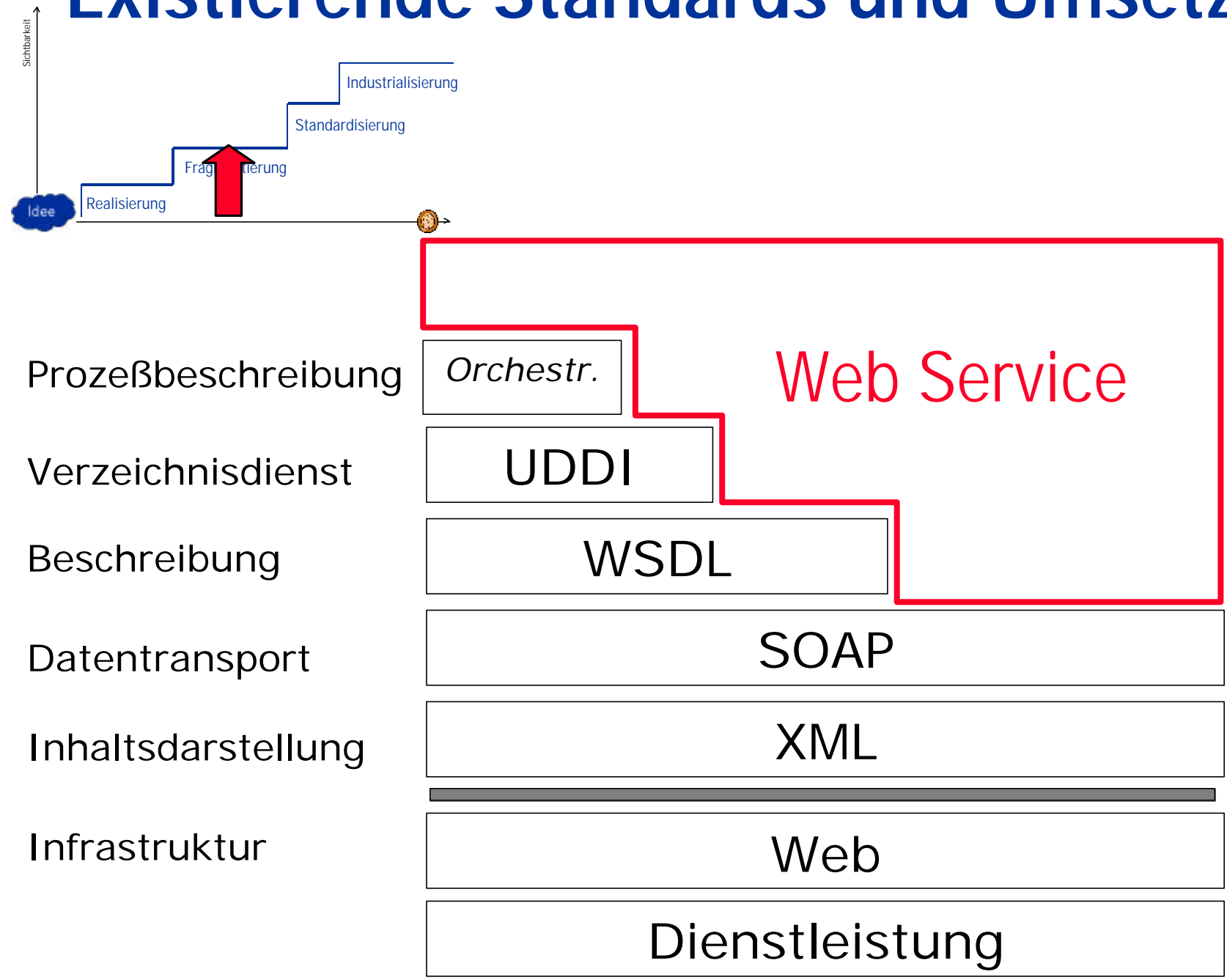
- WSDL v1.2 (W3C)
- Implementierungen verfügbar

Existierende Standards und Umsetzungen



- UDDI v3 (OASIS)
- Repositories verfügbar

Existierende Standards und Umsetzungen



- Einige verschiedene offen am Markt konkurrierende Vorschläge
- Derzeit (noch keine) Standardisierung
- Kaum Umsetzungen