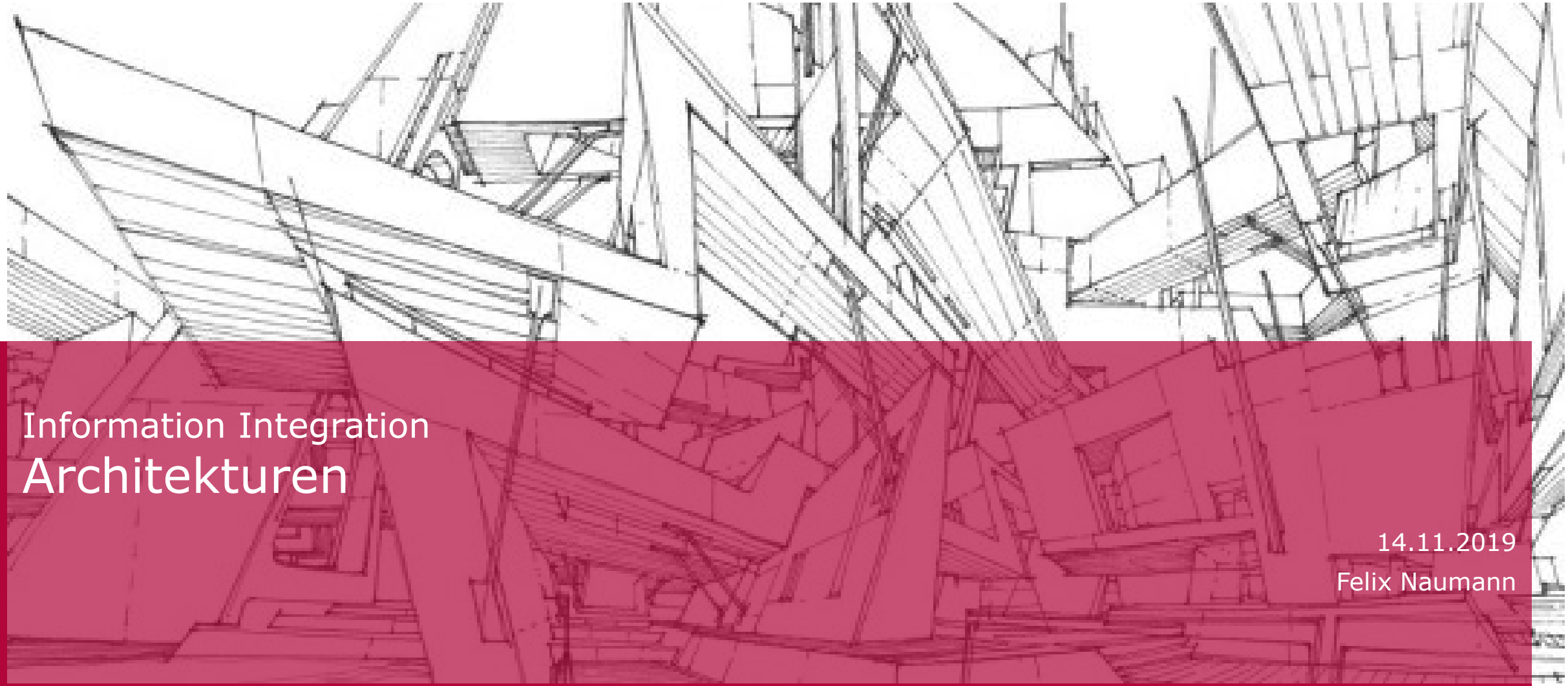


Information Integration

Architekturen

14.11.2019

Felix Naumann

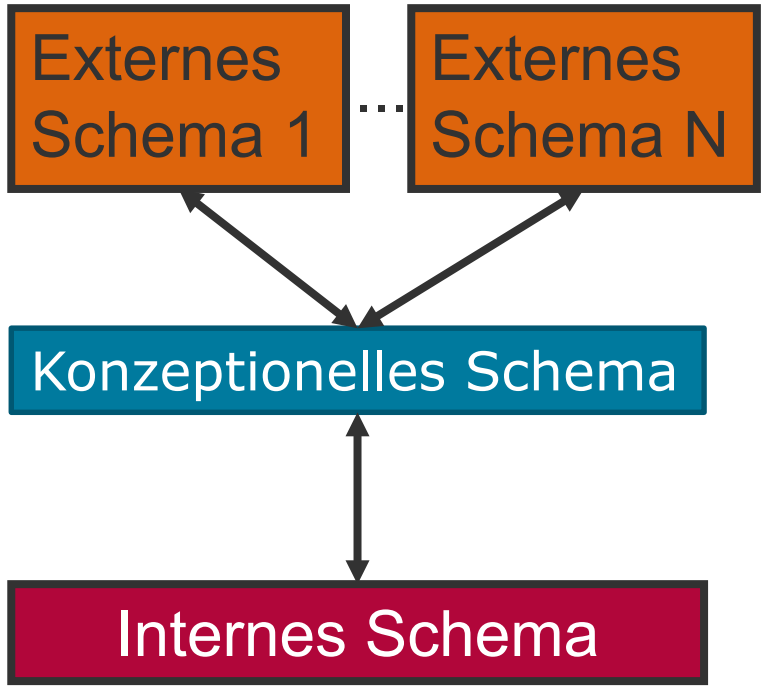
Überblick

1. **Architekturen**
   - 3 Schichten Architektur
   - 4 Schichten Architektur
   - 5 Schichten Architektur
2. Mediator-Wrapper Architektur
   - Gio Wiederholds Definitionen
   - Konfigurationen
   - Mediatoren
   - Wrapper
3. Peer-Data-Management
   - Architektur
   - Anwendungen

Felix Naumann
Information Integration
Winter 2019/20

3-Schichten Architektur

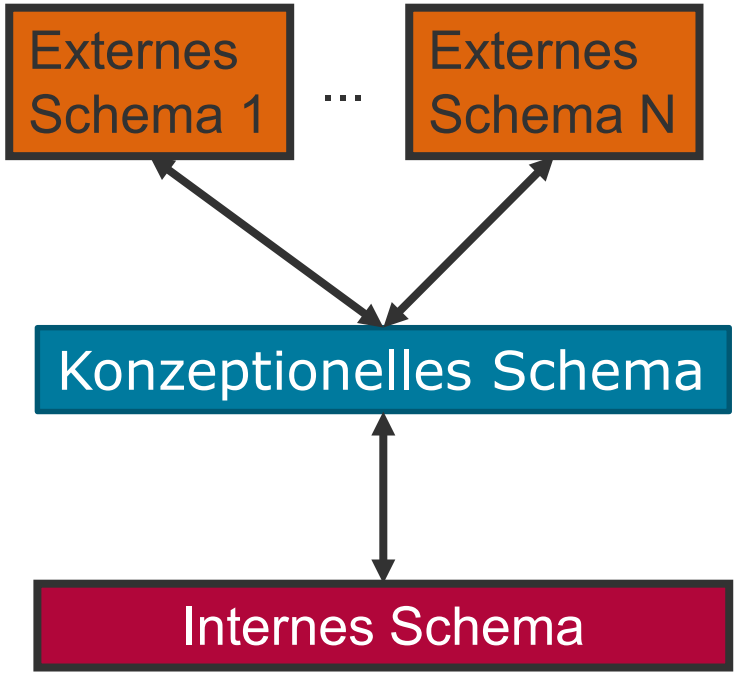
ANSI/SPARC
3-Schichten Architektur für zentralisierte DBMS



Felix Naumann
Information Integration
Winter 2019/20

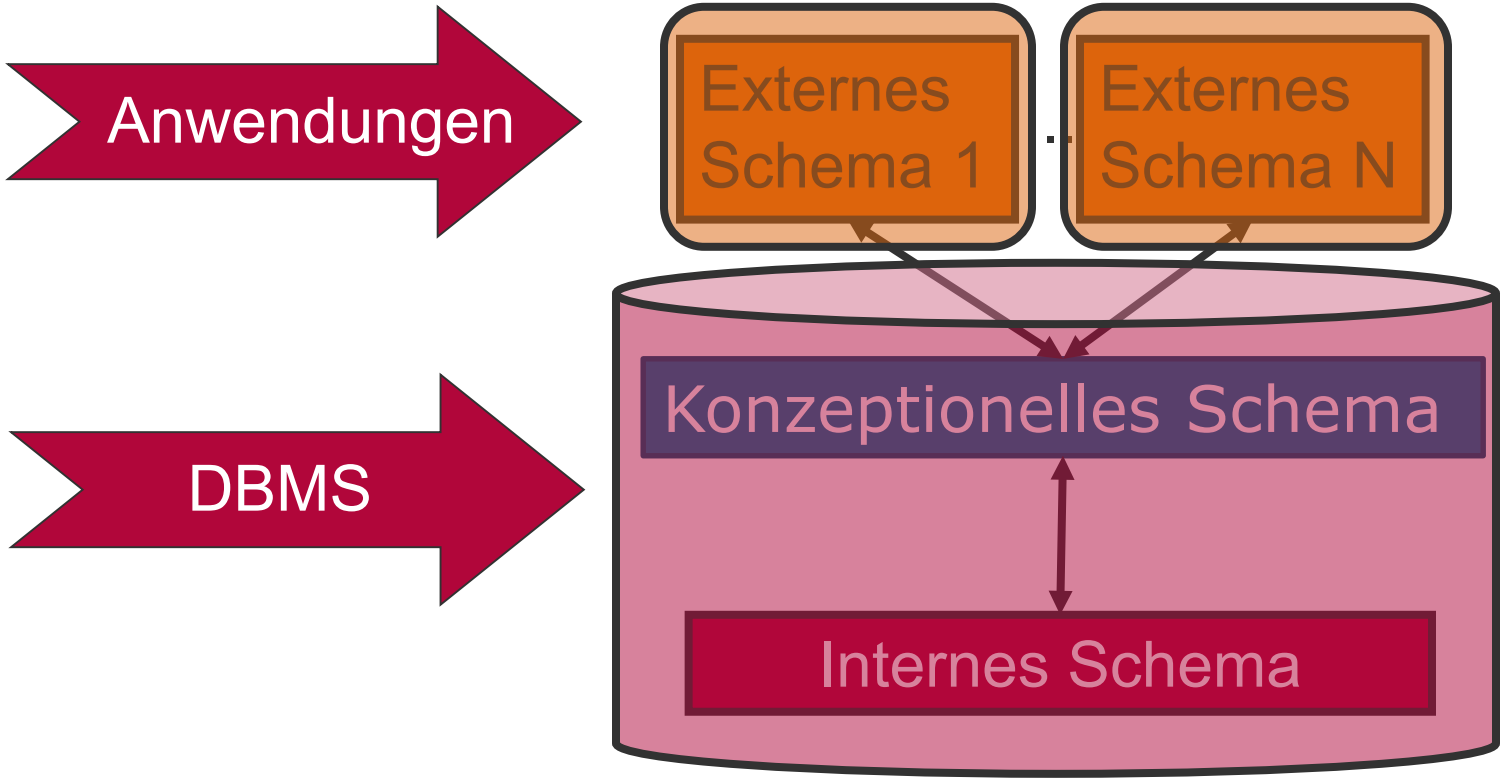
Wdh: Das Schichtenmodell

- Interne (physische) Sicht
  - Speichermedium (Tape, Festplatte)
  - Speicherort (Zylinder, Block)
- Konzeptionelle (logische) Sicht
  - Unabhängig von physischer Sicht
  - Definiert durch Datenmodell
  - Stabiler Bezugspunkt für interne und externe Sichten
- Externe (logische) Sicht
  - Anwendungsprogramme
  - Nur auf die relevanten Daten
  - Enthält Aggregationen und Transformationen



Felix Naumann
Information Integration
Winter 2019/20

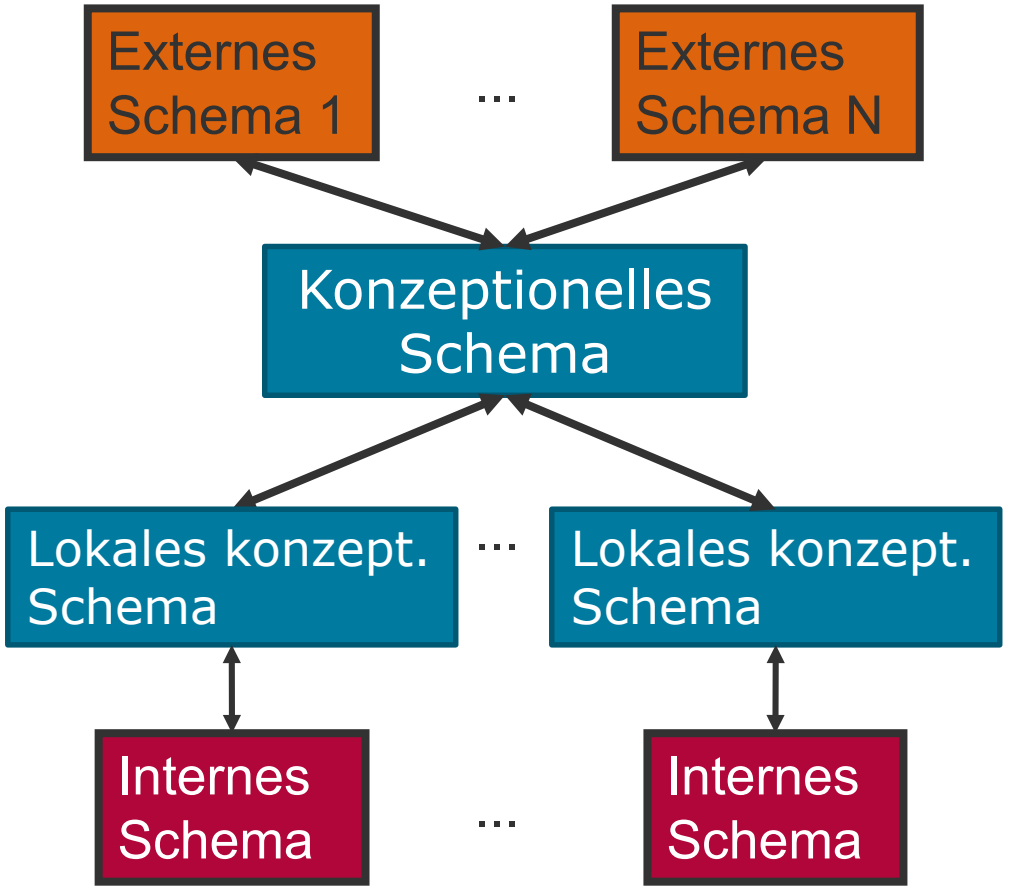
3-Schichten Architektur



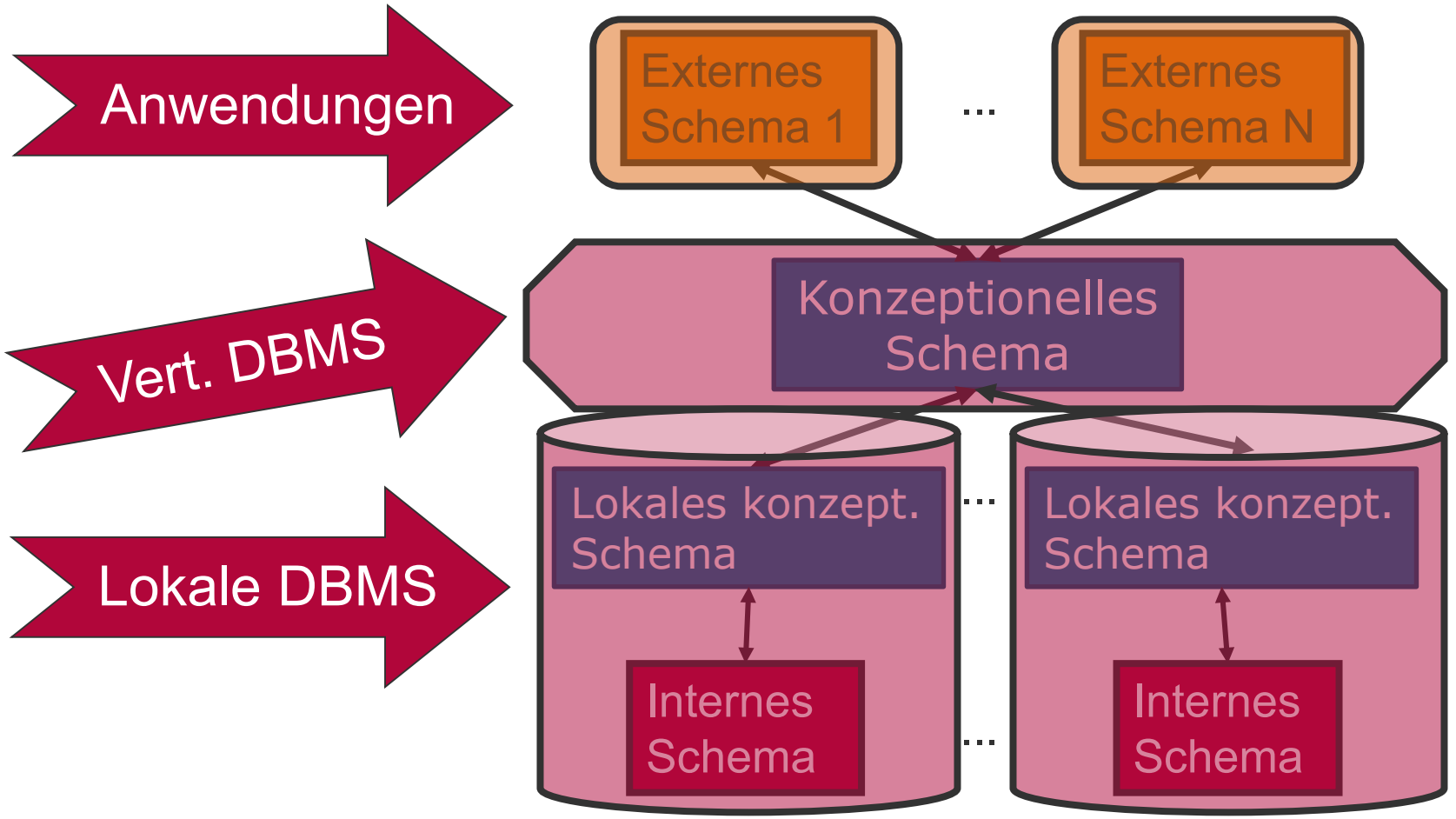
Felix Naumann
Information Integration
Winter 2019/20

4-Schichten Architektur

- Für verteilte DBMS
- Neu: Trennung lokales vs. globales konzeptionelles
- Globales Konzeptionelles Schema ist integriert aus den lokalen konzeptionellen Schemata.
- Lokales und globales konzept. Schema kann gleich sein.

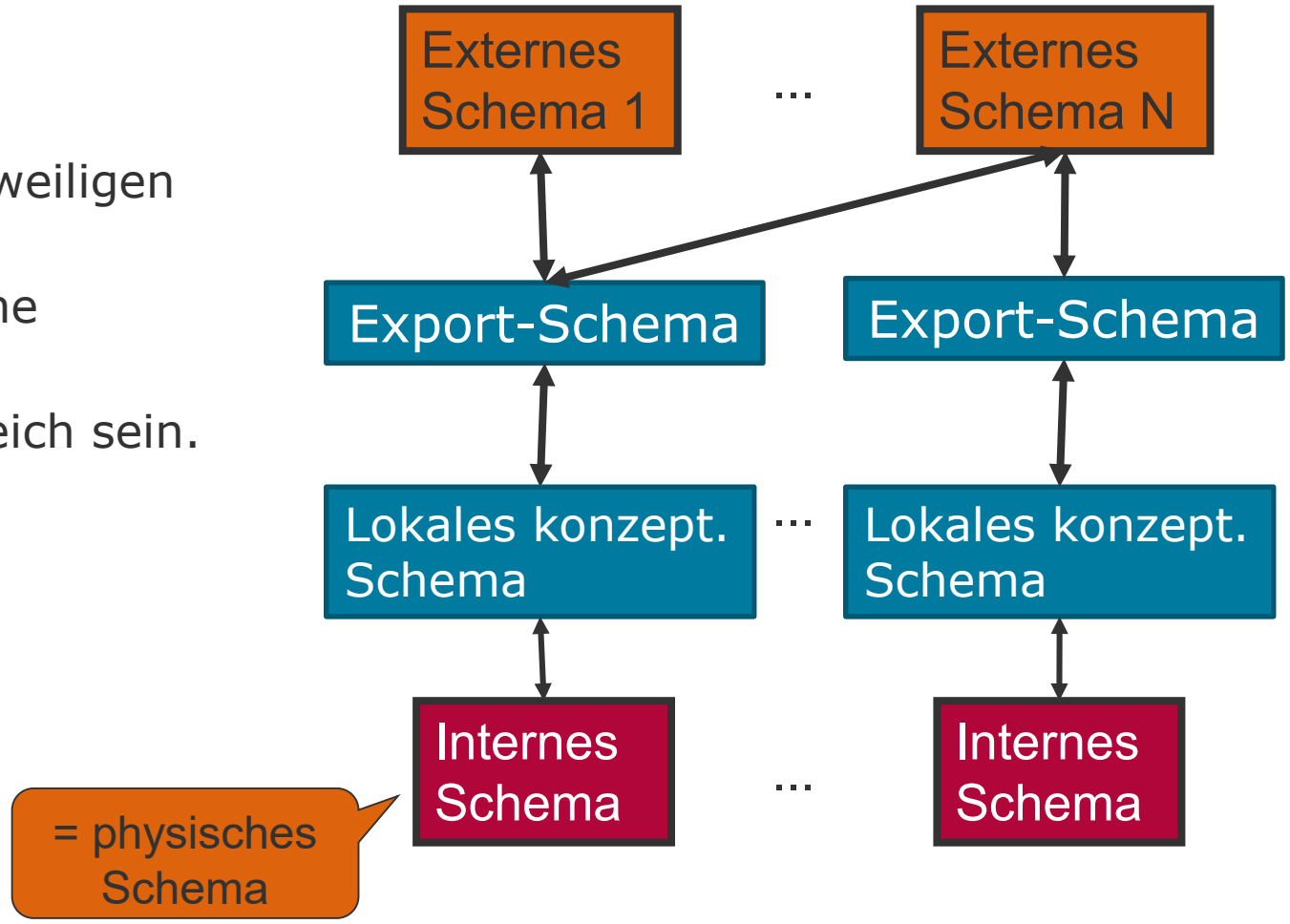


4-Schichten Architektur

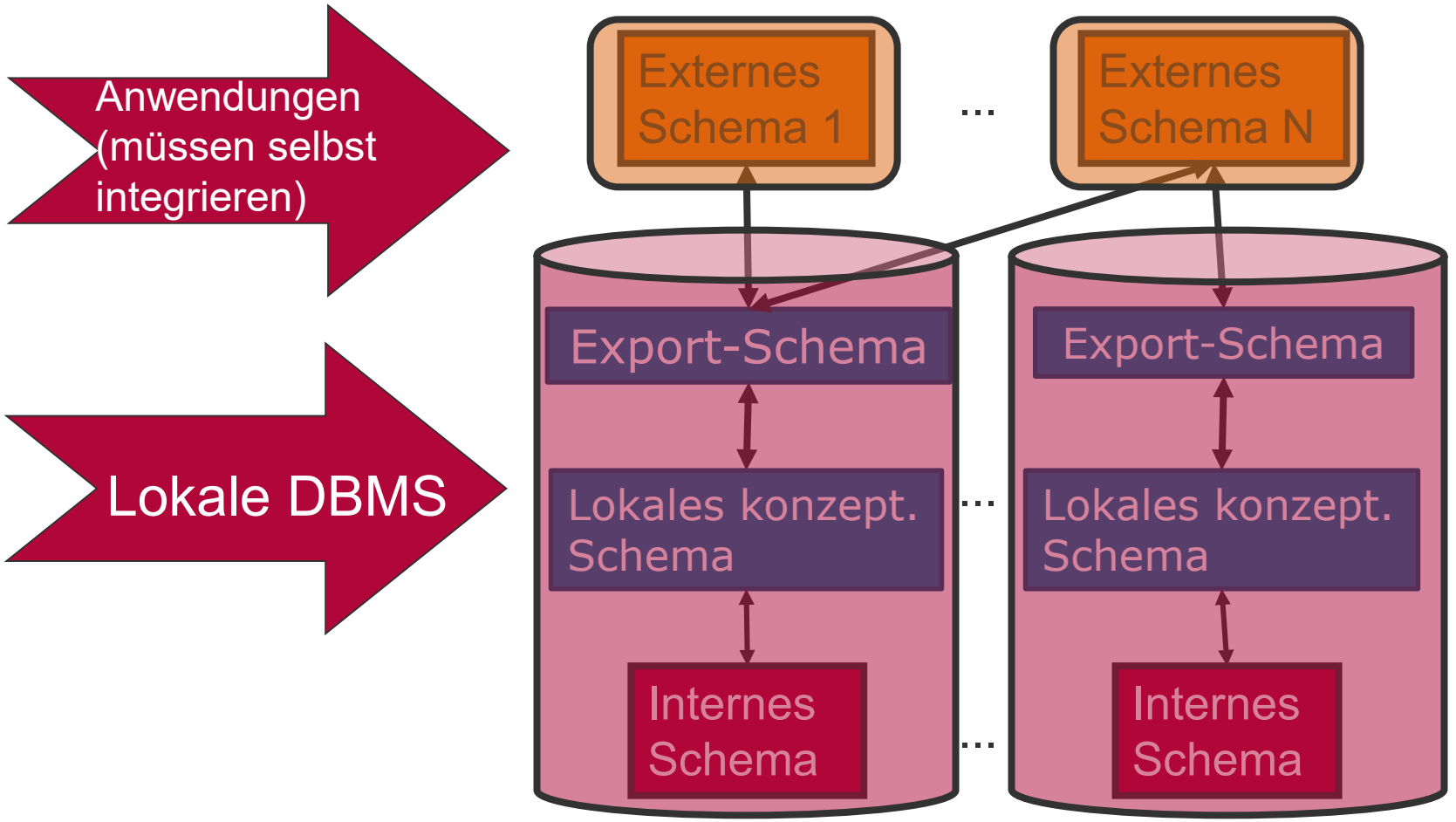


4-Schichten Architektur

- Auch: Multidatenbank-architektur [LMR90]
- Voraussetzung
  - Nutzer kennen die jeweiligen Schemata
  - Multidatenbanksprache
- Lokales Schema und Exportschema kann gleich sein.
- Lose Kopplung



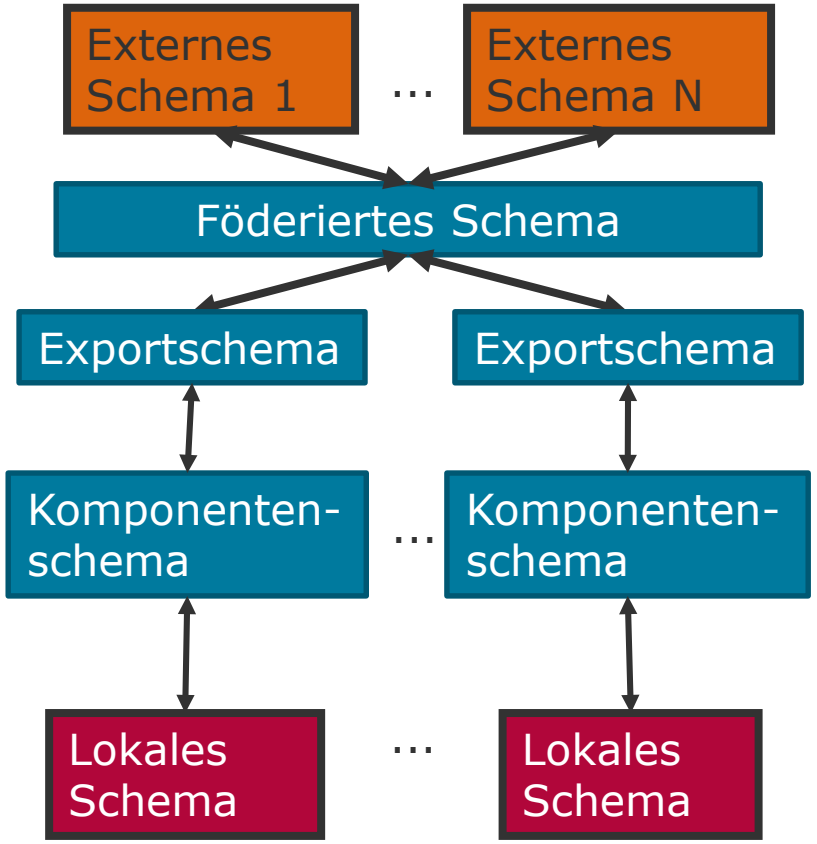
4-Schichten Architektur



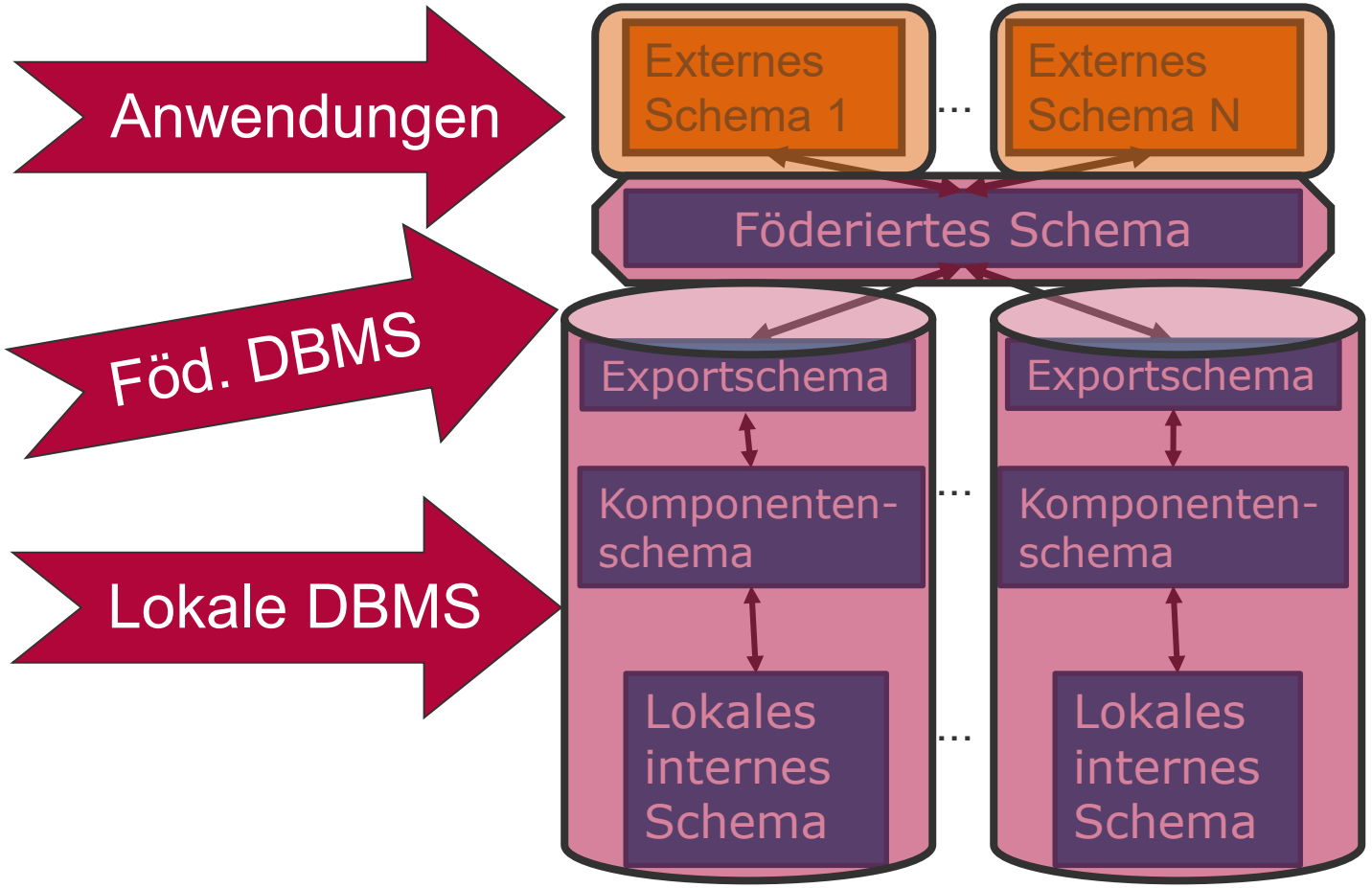
Felix Naumann
Information Integration
Winter 2019/20

5-Schichten Architektur [SL90]

- Neu:
  - Interne Schemata werden nicht mehr betrachtet.
  - Exportschemata
  - Integriertes, föderiertes Schema
- Terminologie
  - Komponentenschema = lokales konzept. Schema
  - Föderiertes Schema = globales konzept. Schema

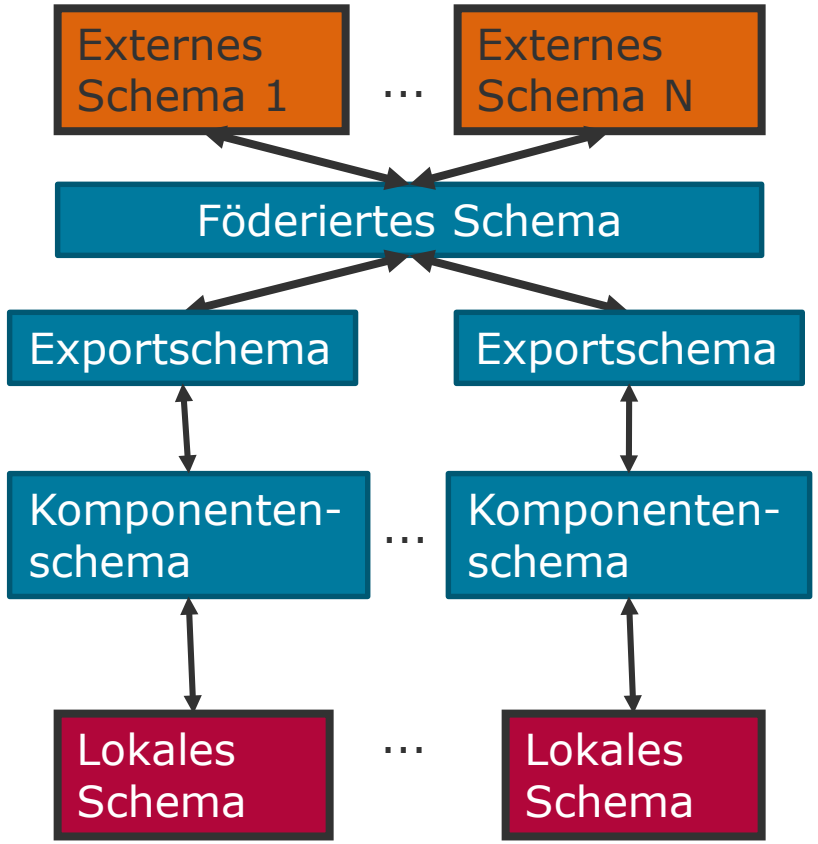


5-Schichten Architektur [SL90]



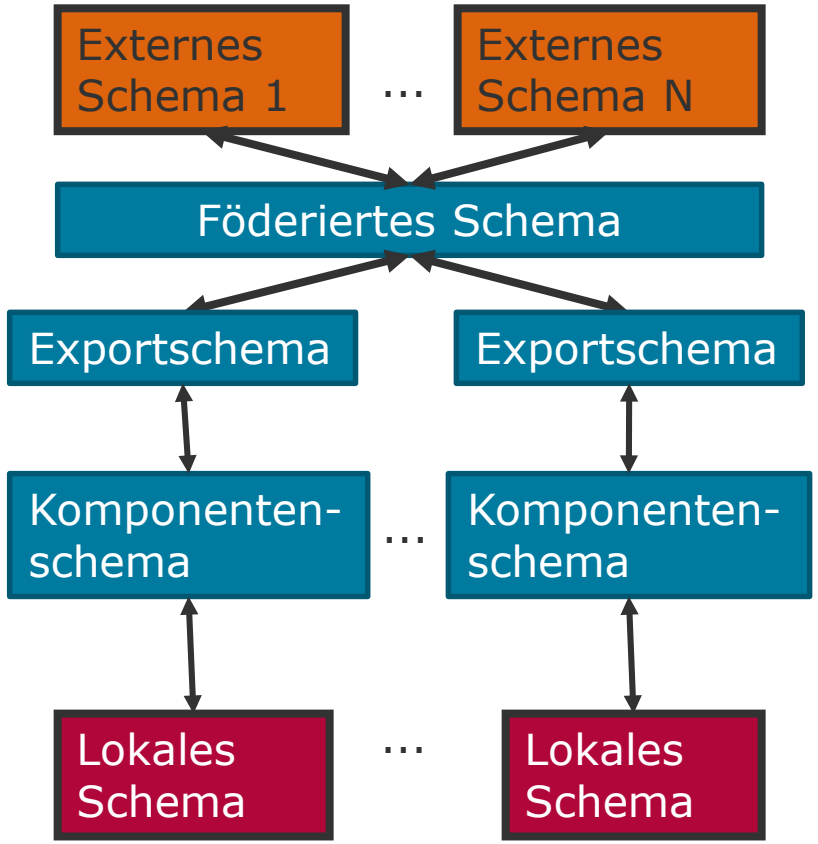
5-Schichten Architektur [SL90]

- Lokale Schemata
  - Konzeptionell
- Komponentenschemata
  - Kanonisches Datenmodell
  - Fügt fehlende Semantik hinzu.
  - Übergang durch Mappings.
- Exportschemata
  - Teilmenge des Komponentenschemata
  - Verwaltet Zugangsberechtigungen
  - Gleiche Rolle wie externe Schemata in klassischer 3-Schichten-Architektur



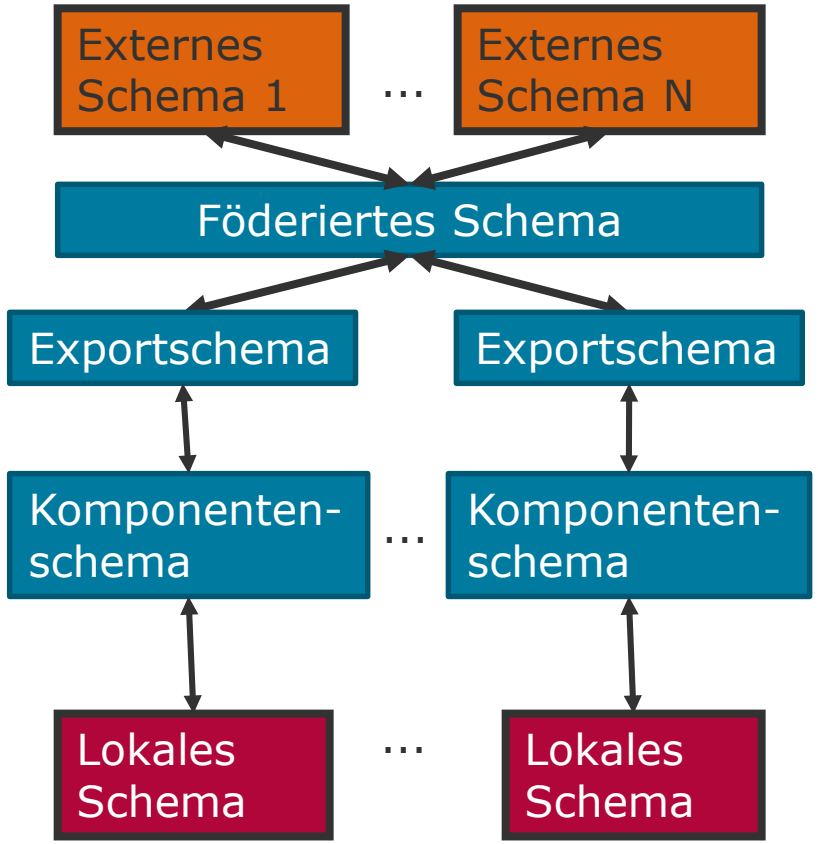
5-Schichten Architektur [SL90]

- Föderiertes Schema
  - Integriert aus den Exportschemata
  - Kennt Datenverteilung
  - Andere Namen:
    - Import Schema
    - Globales Schema
    - Enterprise Schema
    - Unified Schema
    - Mediator Schema
- Externes Schema
  - Föderiertes Schema kann sehr groß sein → Vereinfachung im externen Schema
  - „Schema Evolution" leichter
  - Zusätzliche Integritätsbedingungen
  - Zugangskontrollen



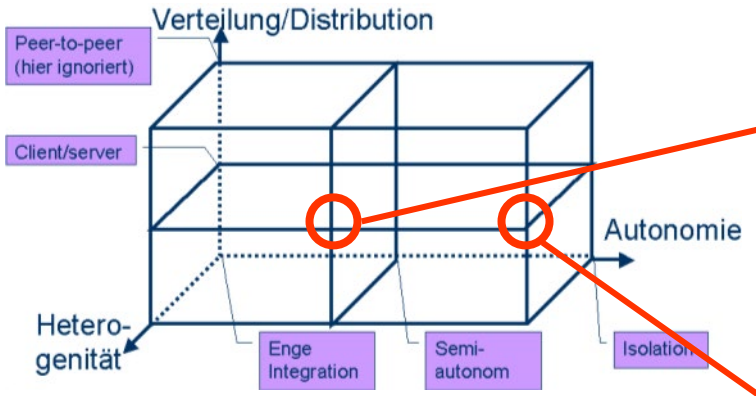
5-Schichten Architektur [SL90]

- Mischformen
  - Einige Schichten nicht immer nötig.
    - Z.B. wenn lokales und Komponentenschema gleich sind.
    - Z.B. wenn komplettes Komponentenschema exportiert werden soll.
  - Ein Komponentenschema kann mehrere Exportschemata haben.
  - Große FDBS können mehrere föderierte Schemata haben.
- Föderation!
  - Nur semi-autonom
  - Lokale DBMS müssen bereits kanonisches Datenmodell unterstützen.

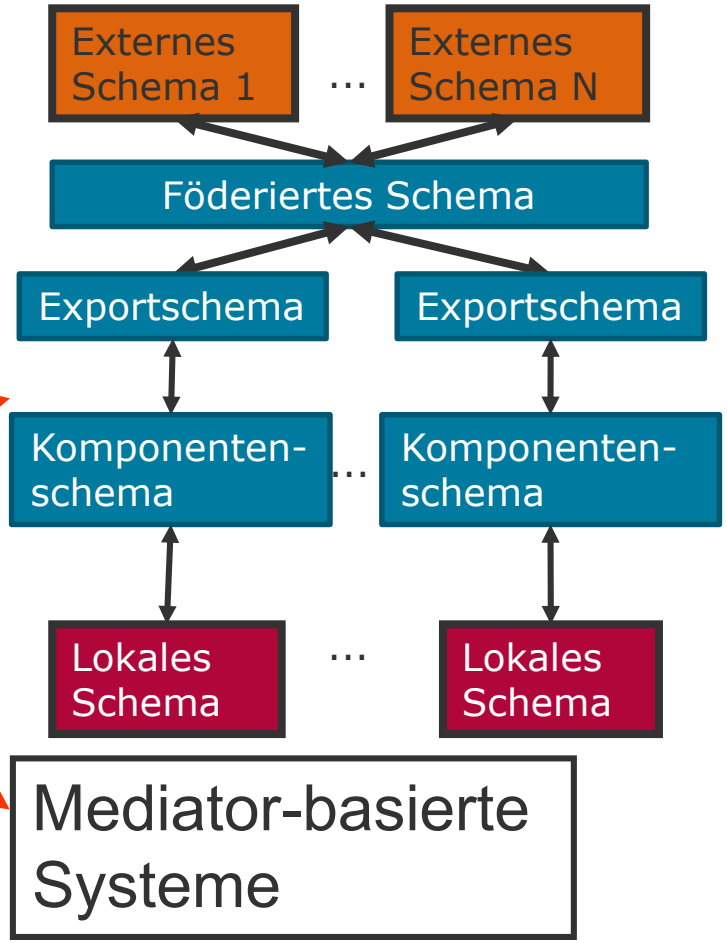


Zusammenfassung

1.



2.

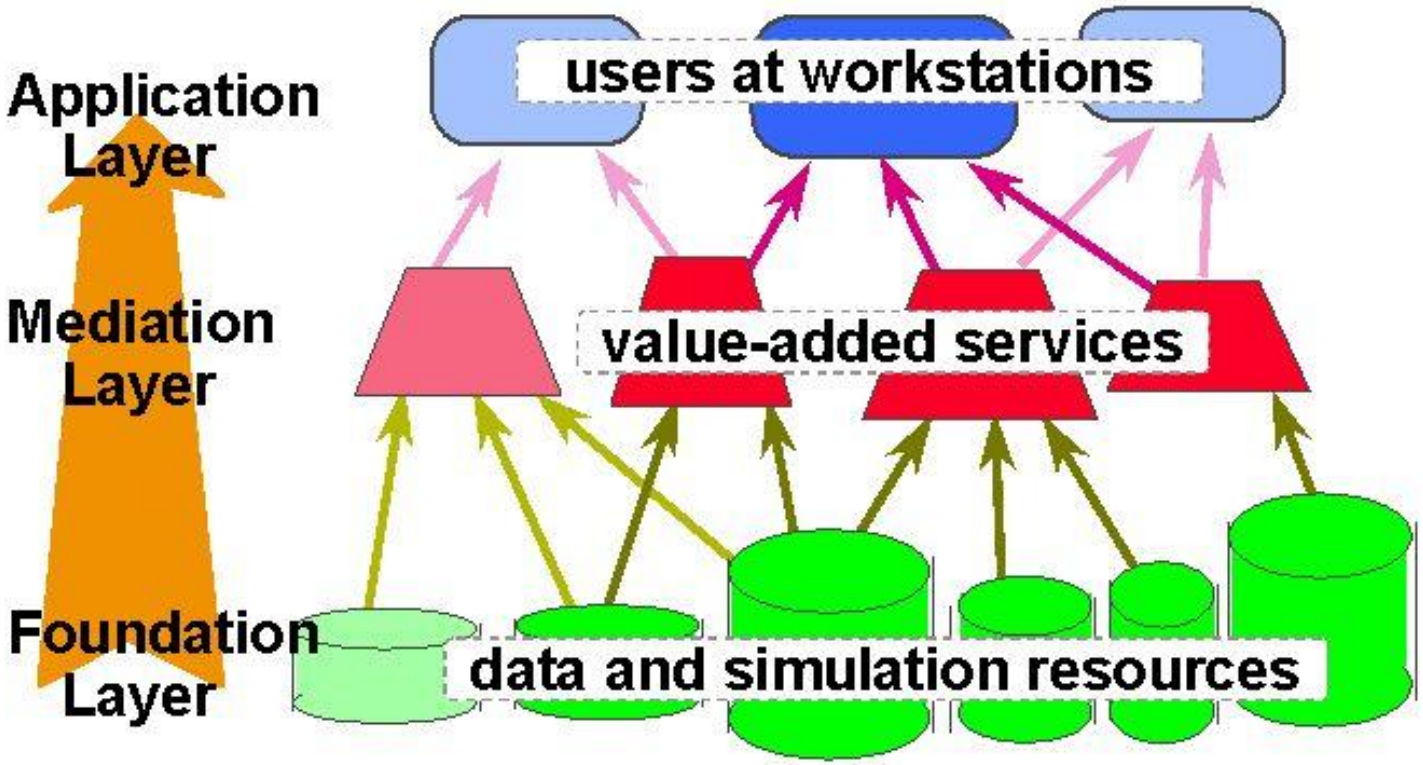


Überblick

1. Architekturen
   - 3 Schichten Architektur
   - 4 Schichten Architektur
   - 5 Schichten Architektur
2. Mediator-Wrapper Architektur
   - **Gio Wiederholds Definitionen**
   - Konfigurationen
   - Mediatoren
   - Wrapper
3. Peer-Data-Management
   - Architektur
   - Anwendungen

Felix Naumann
Information Integration
Winter 2019/20

Daten werden zu Informationen



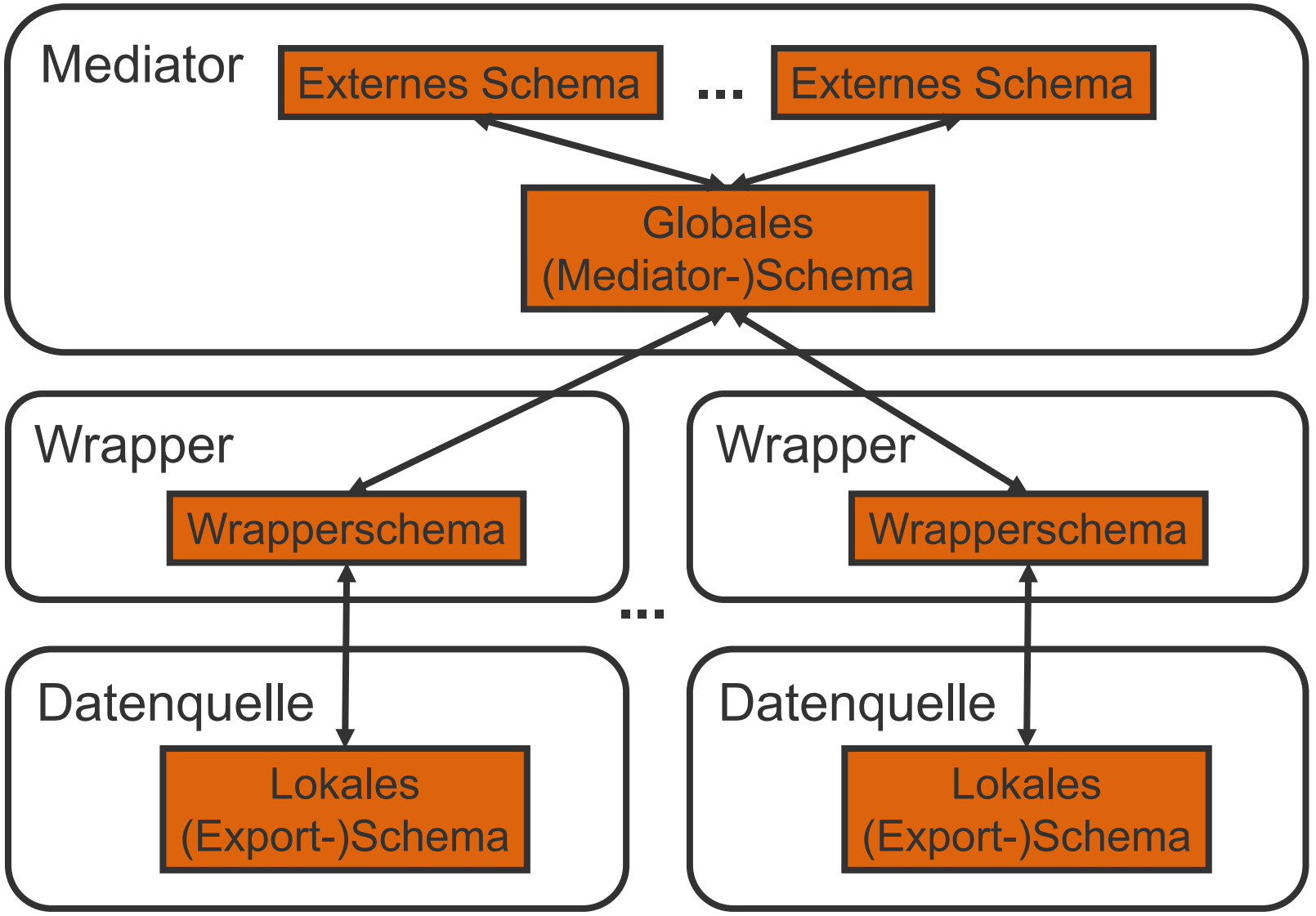
Gio Wiederhold 1999 15

Mediatoren

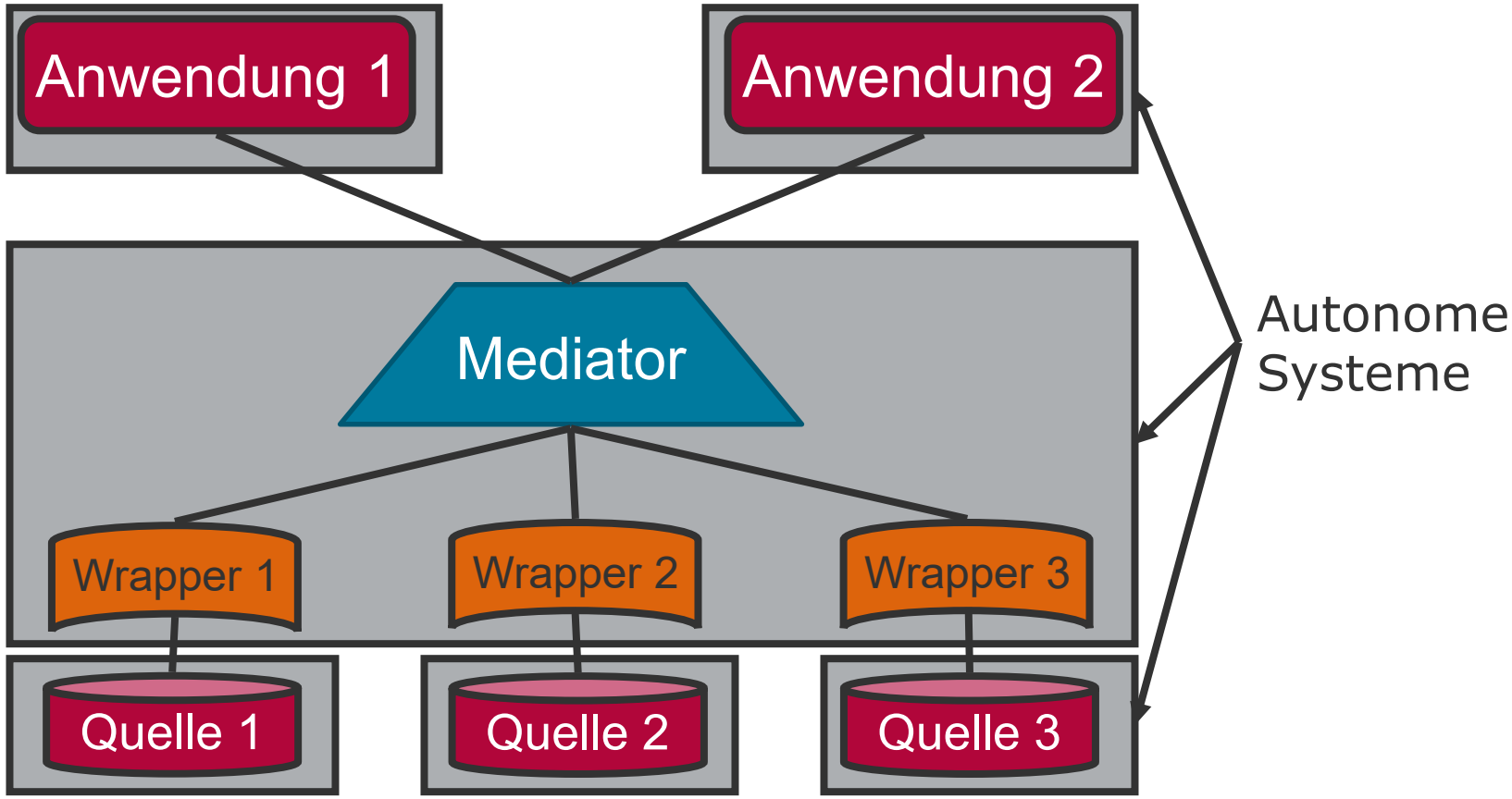
„A mediator is a software module that exploits encoded knowledge about certain sets or subsets of data to create information for a higher layer of applications“ [Wie92]

Ein Mediator ist eine Softwarekomponente, die Wissen über bestimmte Daten benutzt, um Informationen für höherwertige Anwendungen zu erzeugen.

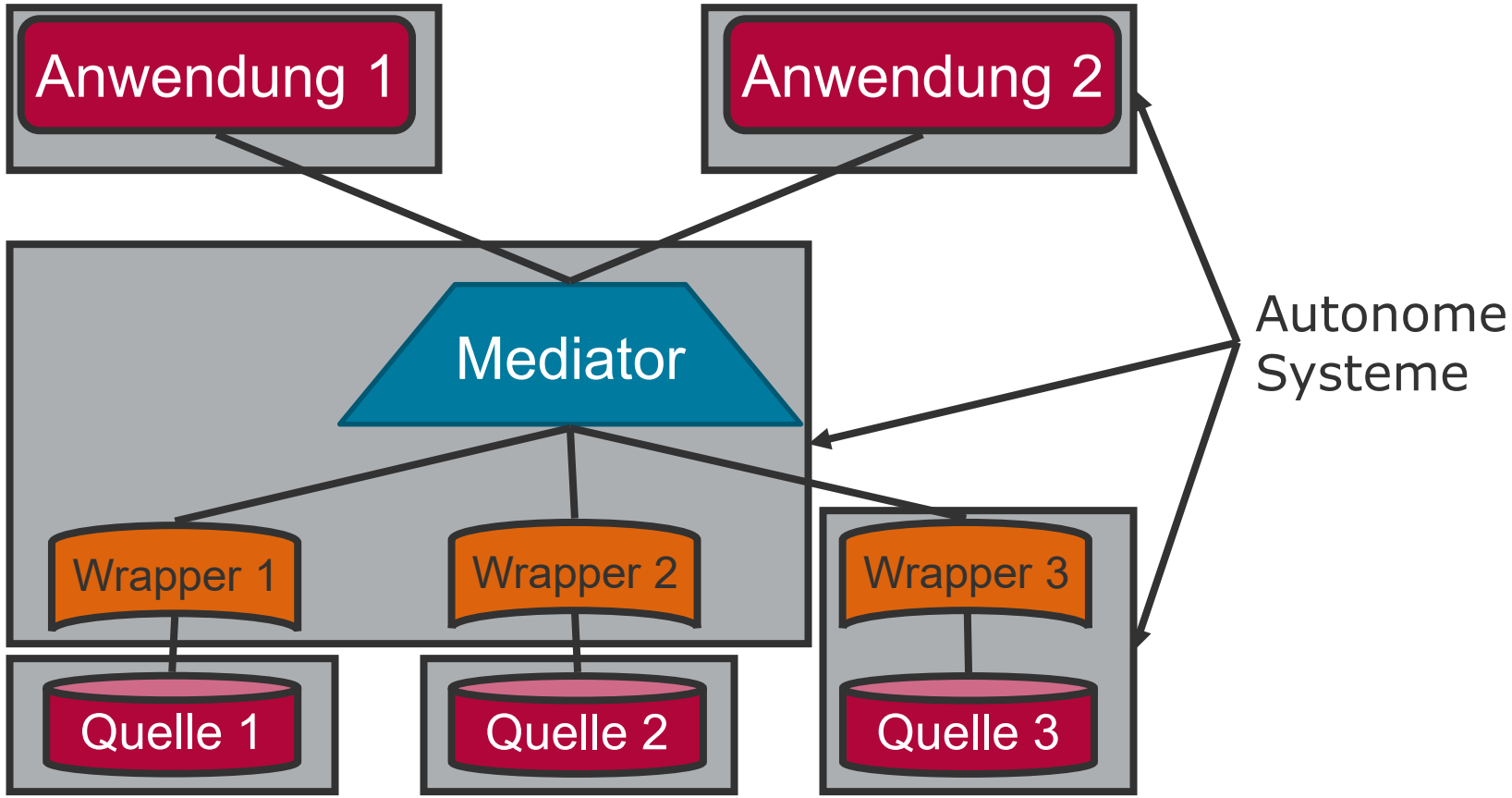
Die Architektur



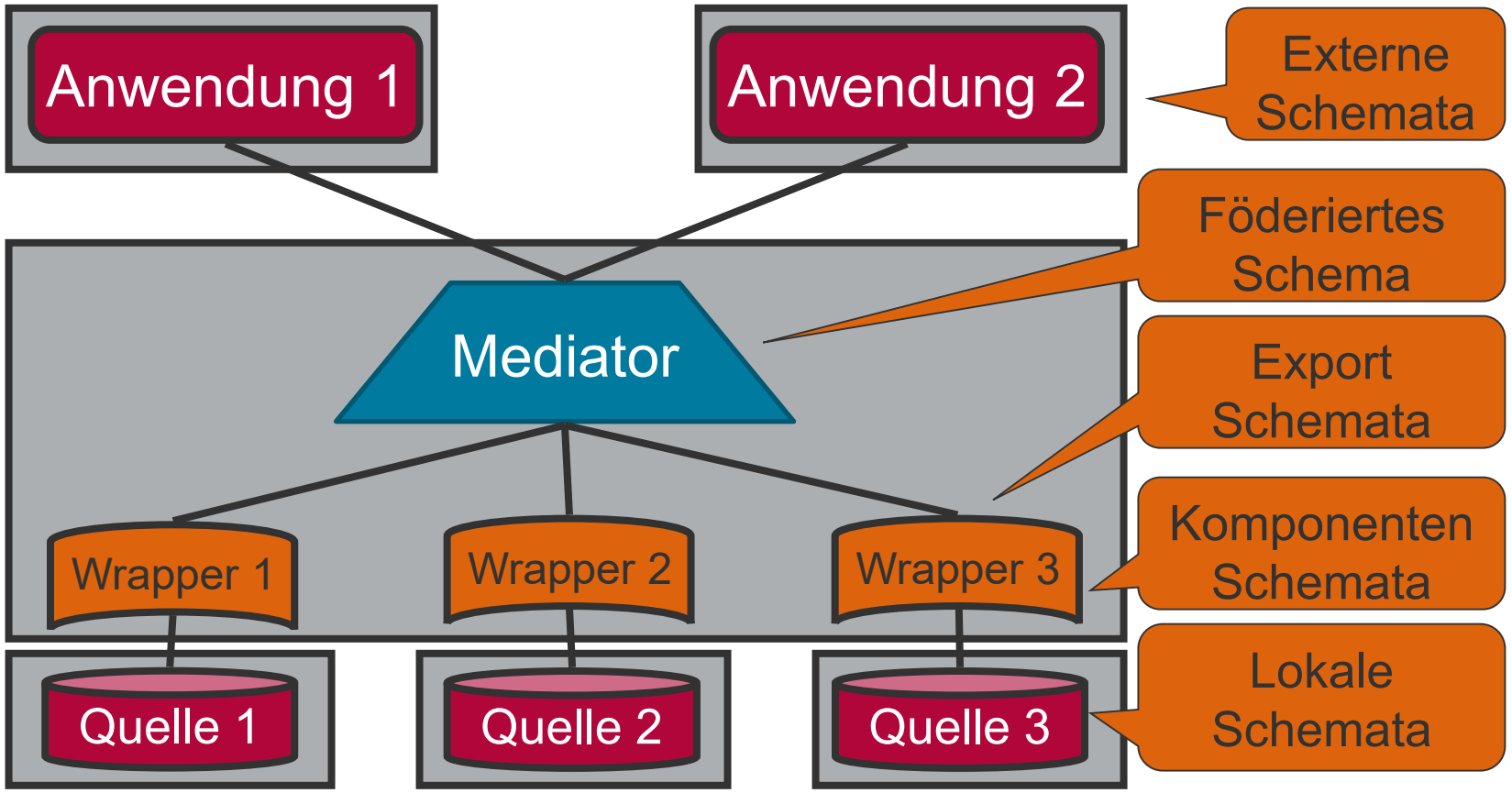
Mediator-Wrapper Architektur



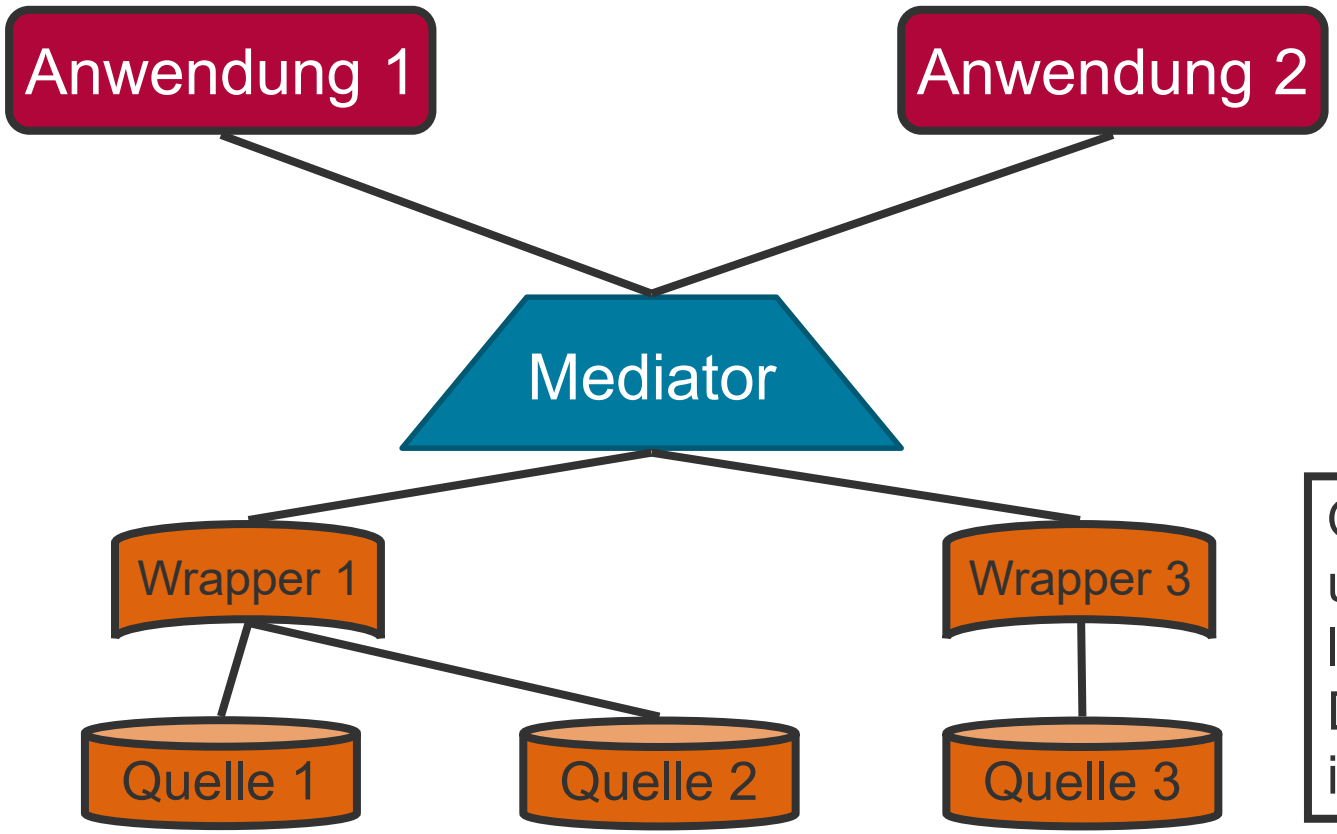
Mediator-Wrapper Architektur



Mediator-Wrapper Architektur

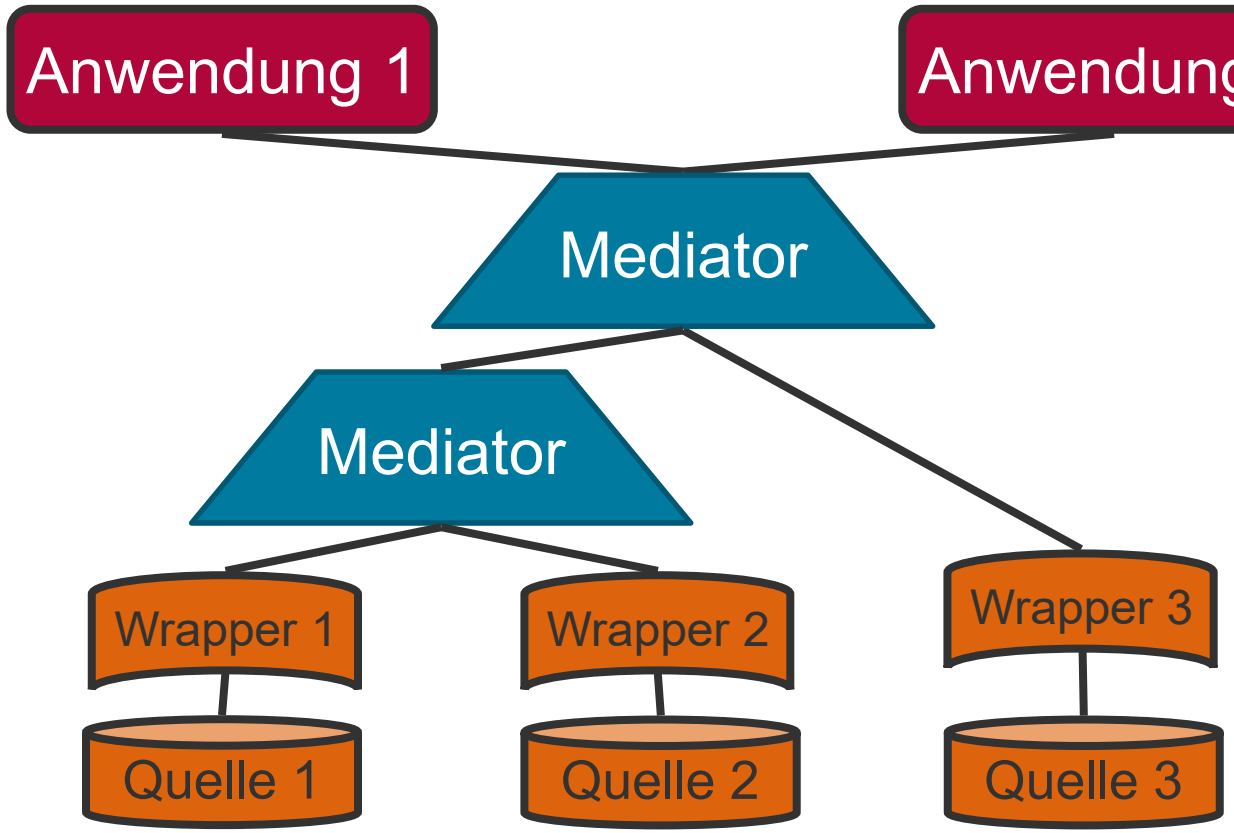


Mediator-Wrapper Architektur



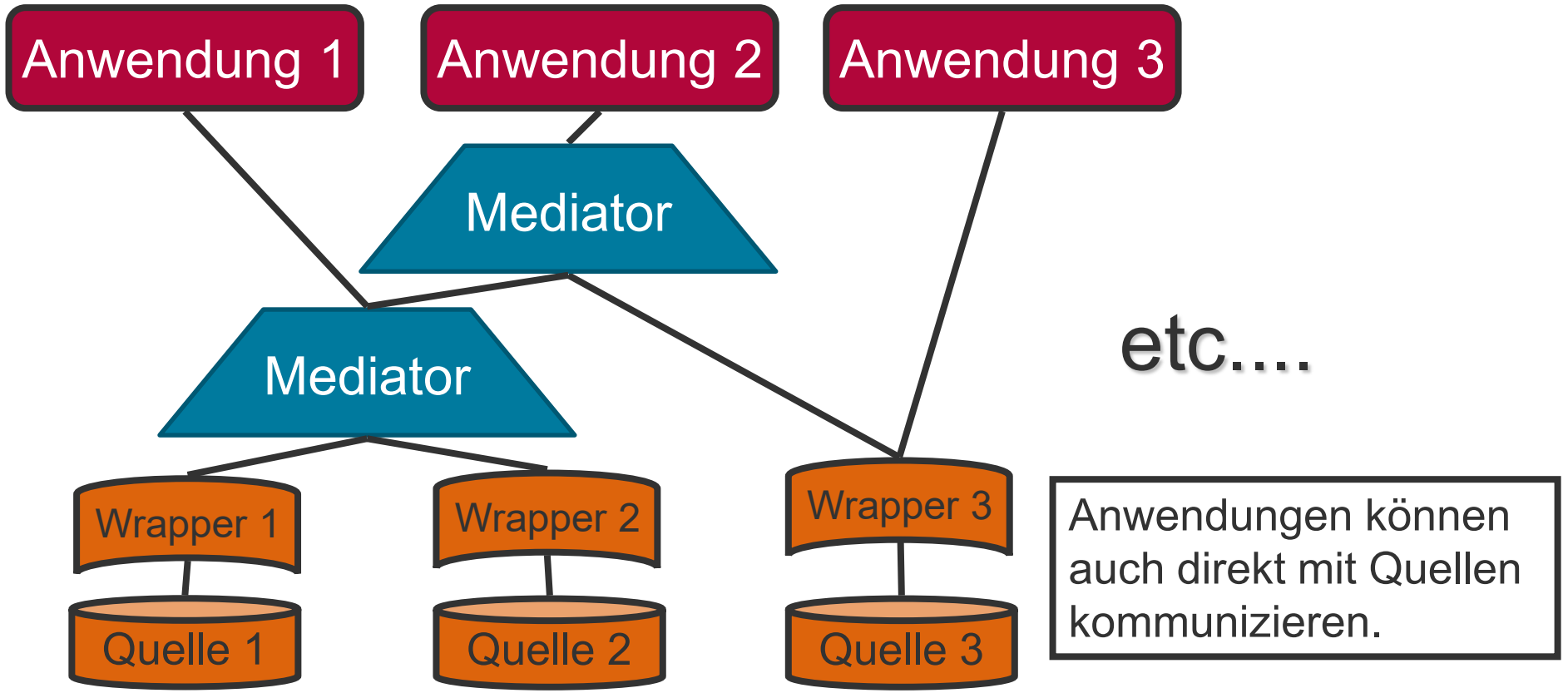
Quelle 1 und Quelle 2 unterscheiden sich nur leicht, z.B. zwei Oracle Datenbanken mit identischen Schemata.

Mediator-Wrapper Architektur

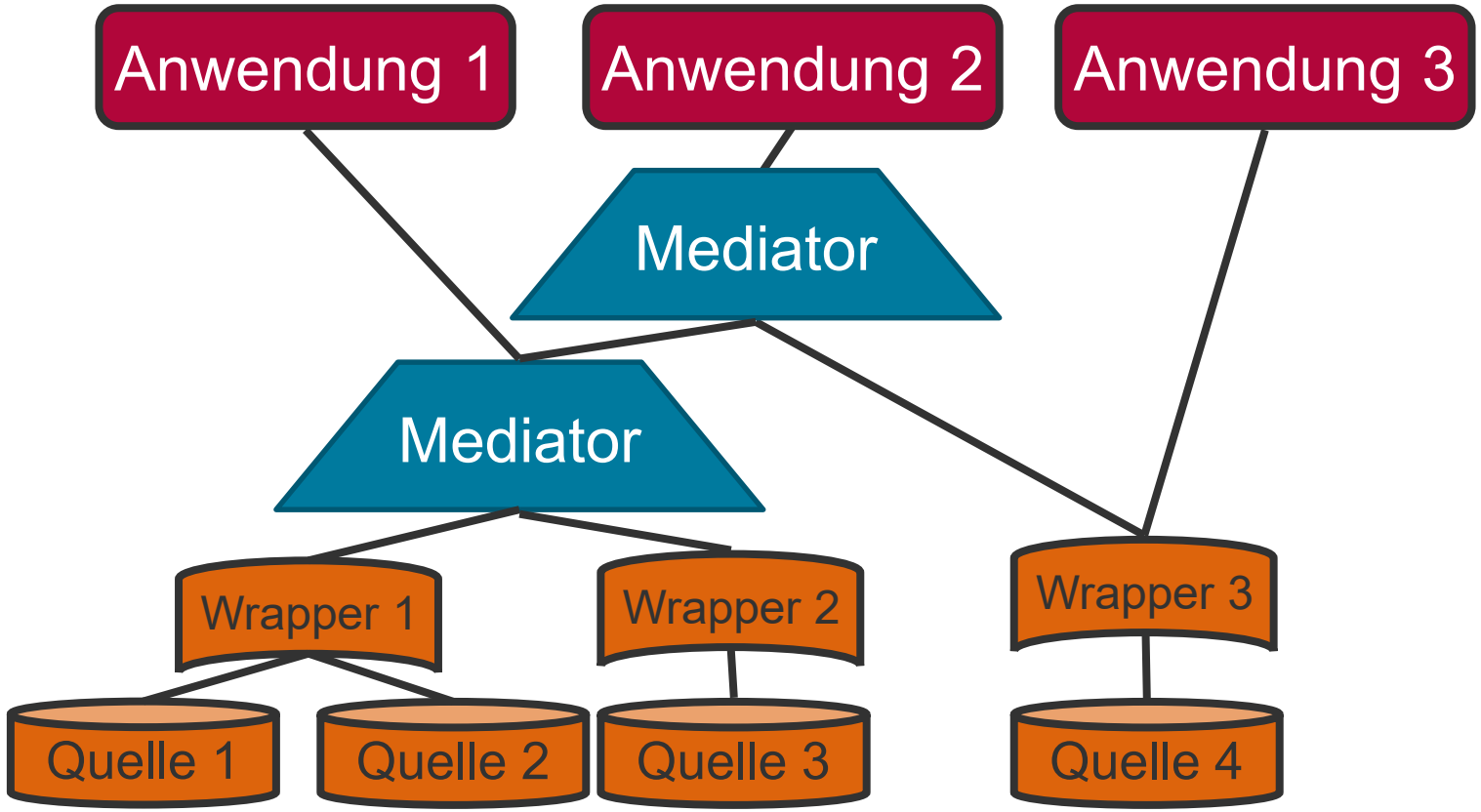


Mediatoren dienen als Quellen für andere Mediatoren. Stufenweise Added-Value.

Mediator-Wrapper Architektur



Mediator-Wrapper Architektur



Überblick

1. Architekturen
   - 3 Schichten Architektur
   - 4 Schichten Architektur
   - 5 Schichten Architektur
2. Mediator-Wrapper Architektur
   - Gio Wiederholds Definitionen
   - Konfigurationen
   - **Mediatoren**
   - Wrapper
3. Peer-Data-Management
   - Architektur
   - Anwendungen

Felix Naumann
Information Integration
Winter 2019/20

Einfache Mediatoren

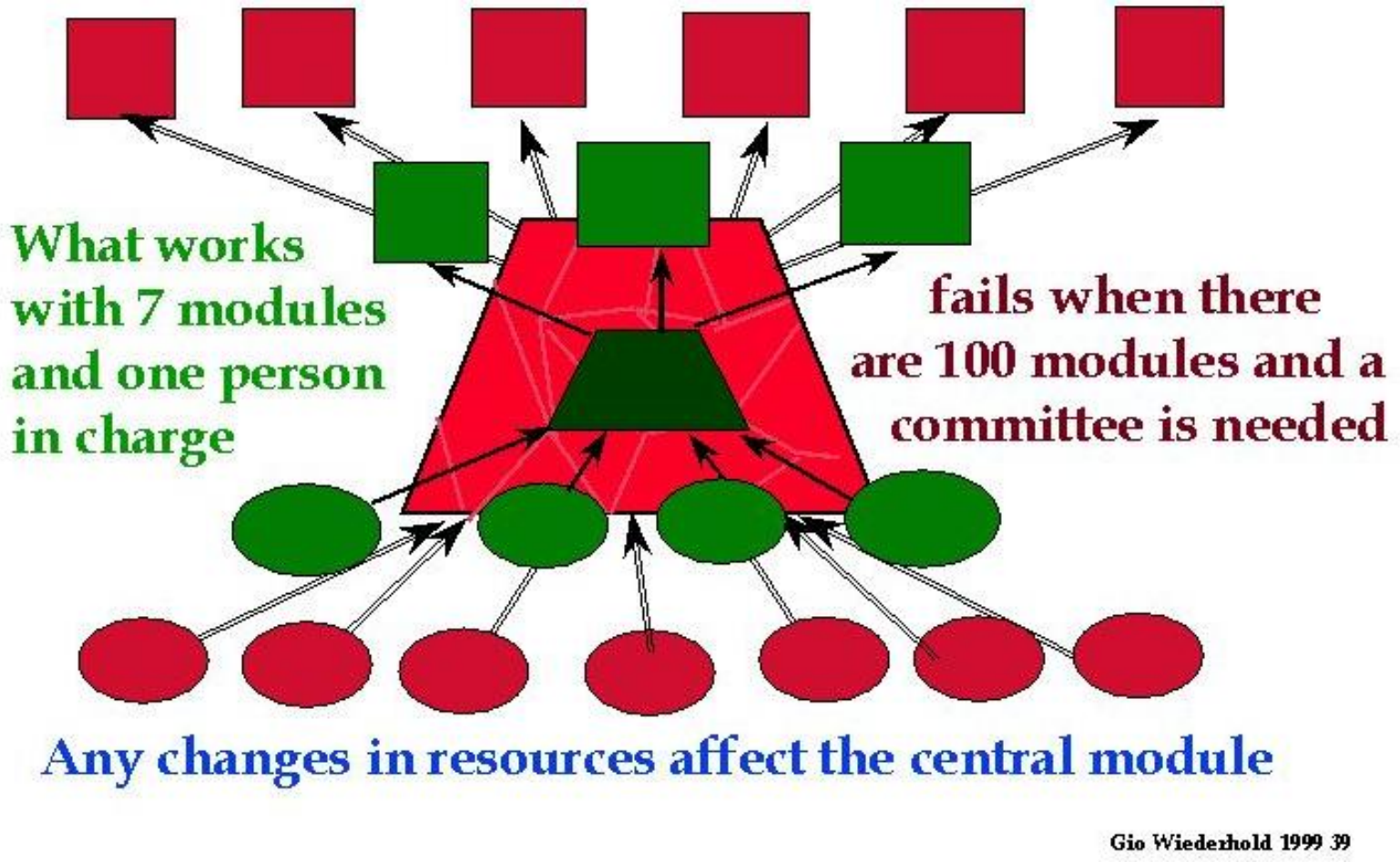
- „[A mediator] should be small and simple, so that it can be maintained by one expert or, at most, a small and coherent group of experts.“ Wiederhold `92

- Ein Mediator sollte klein und einfach genug sein, um durch einen einzigen oder höchstens eine kleine Gruppe von Experten gewartet werden zu können.

- D.h.: Einfaches föderiertes Schema, begrenzte Domäne, einfache Schnittstellen

- Erfahrung: Suchmaschinen ändern monatlich ihre HTML Schnittstelle und häufig ihre API.

Einfache Mediatoren

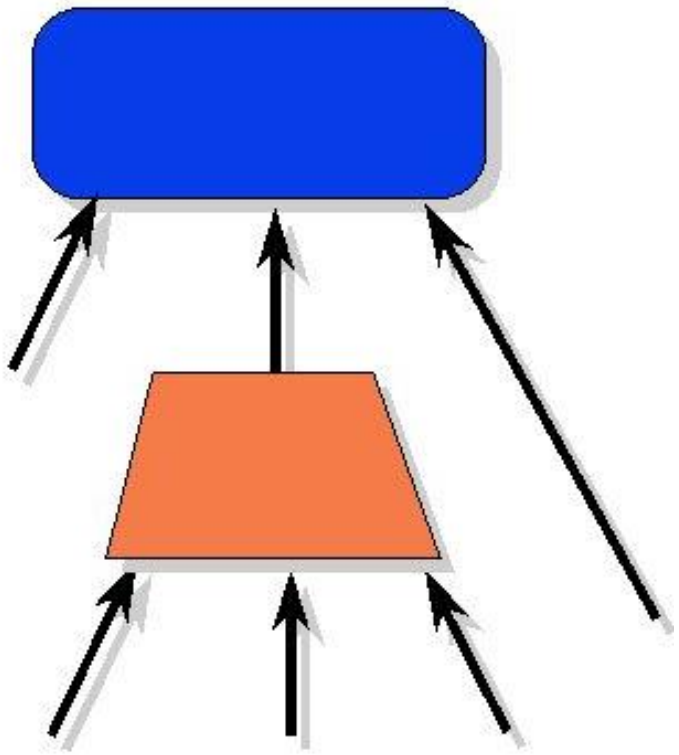


Application

- Informal, pragmatic
- Client-control
- Use up to 7±2 mediators

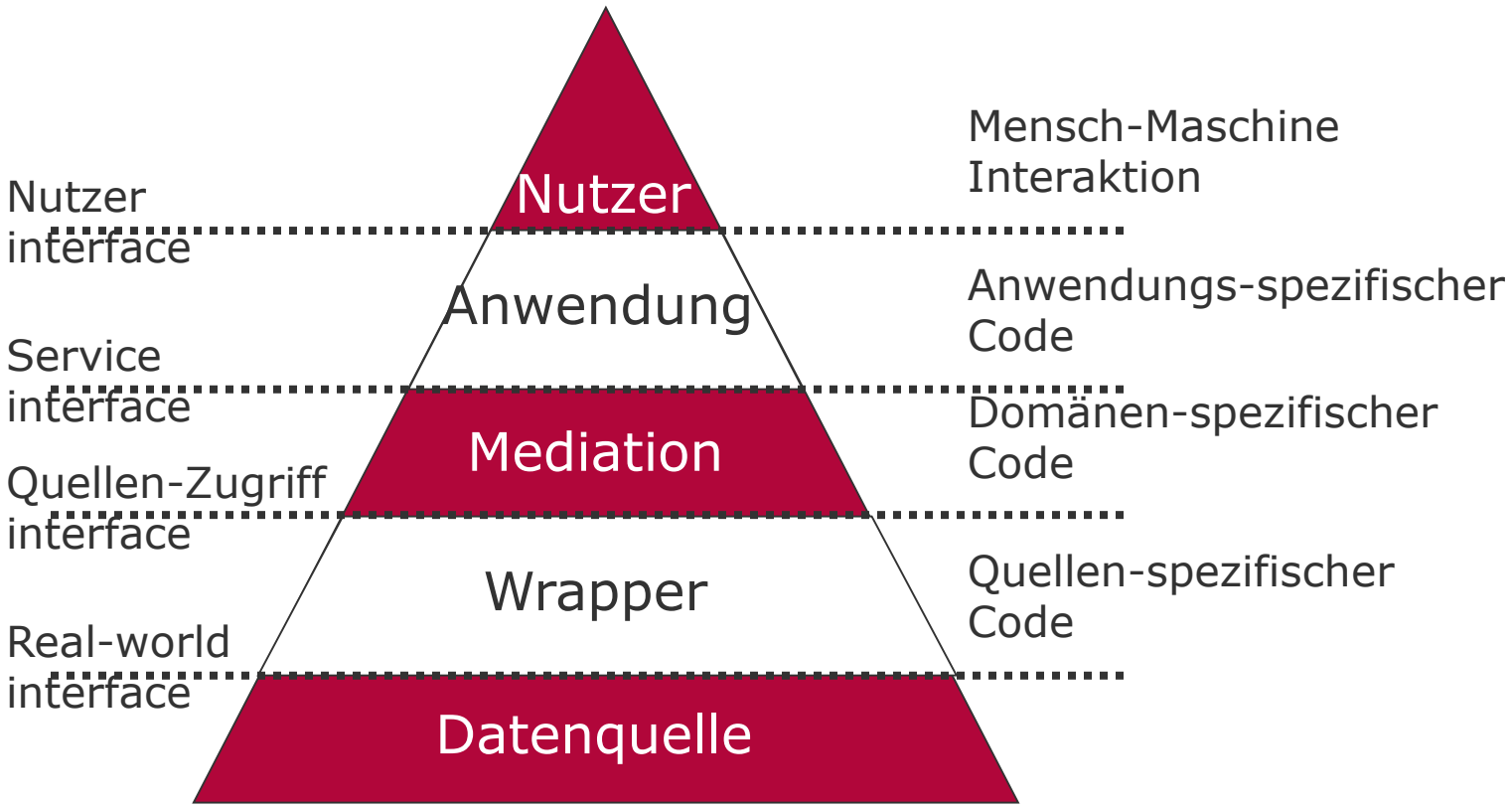
Mediation

- Formal, reliable service
- Domain-Expert control
- Use up to 7±2 sources

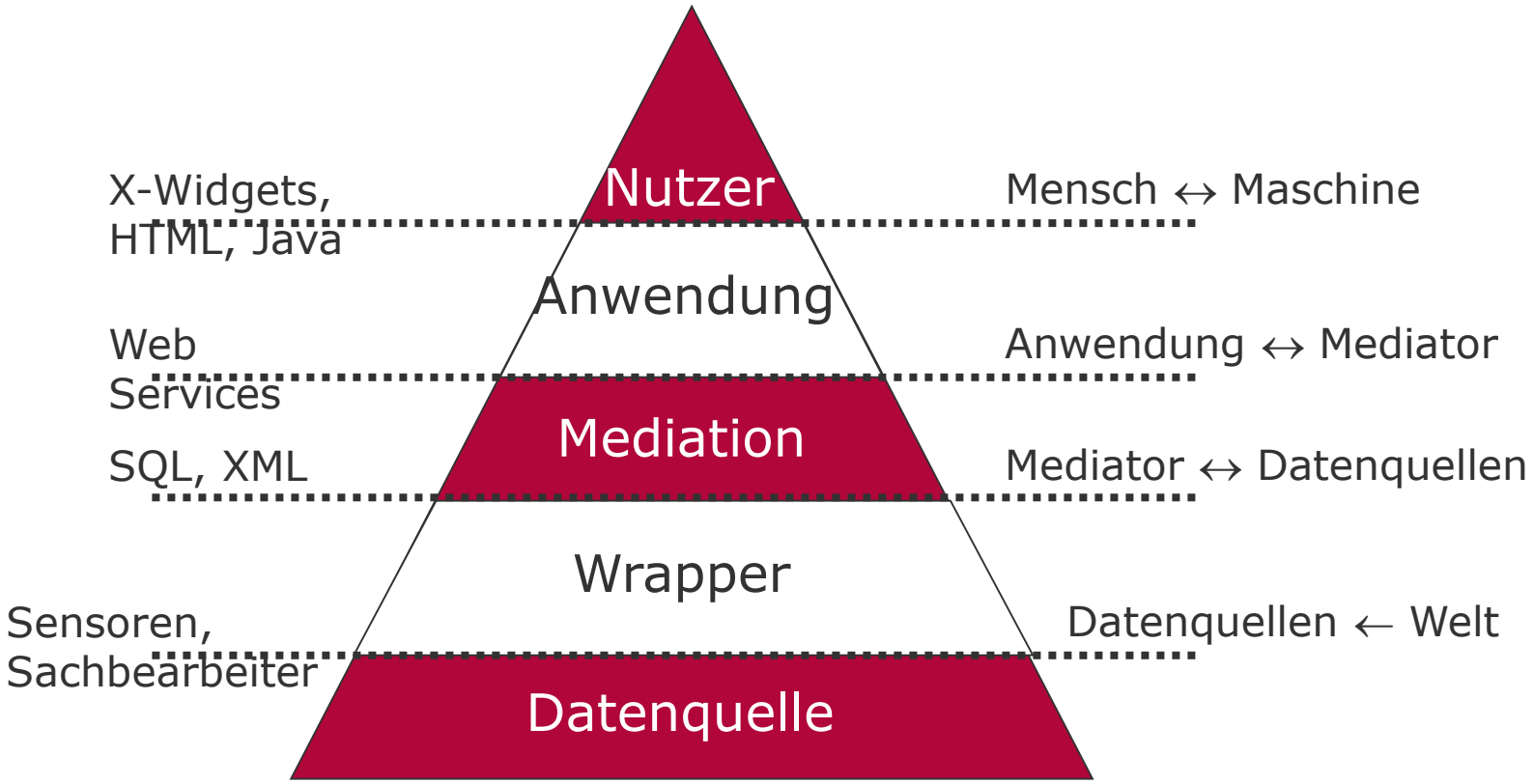


Gio Wiederhold 1999 41

Funktionale Schichten



Schnittstellen



Funktionen der Mediation

- Erbracht durch Domänen-Experten
  - Suche und Auswahl relevanter Informationsquellen
    - „Source selection“
  - Transformationen zur Konsistenzerhaltung
  - Metadaten zur Verarbeitung
  - Abstraktion zum Verständnis
  - Integration verschiedener Quellen
  - Zusammenfassung zur Präsentation
- All dies transformiert Daten zu Informationen.

Dicke und Dünne Mediatoren

Felix Naumann
Information Integration
Winter 2019/20

Überblick

1. Architekturen
   - 3 Schichten Architektur
   - 4 Schichten Architektur
   - 5 Schichten Architektur
2. Mediator-Wrapper Architektur
   - Gio Wiederholds Definitionen
   - Konfigurationen
   - Mediatoren
   - **Wrapper**
3. Peer-Data-Management
   - Architektur
   - Anwendungen

Felix Naumann
Information Integration
Winter 2019/20

Wrapper

- Wrapper sind Softwarekomponenten, die die Kommunikation und den Datenfluss zwischen Mediatoren und Datenquellen herstellen.
- Wrapper sind jeweils spezialisiert auf eine Ausprägung autonomer, heterogener Quellen.
- Wrapper vermitteln zwischen Mediator und Quelle.

Wrapper – Aufgaben

- Lösen Schnittstellenheterogenität
  - technisch
  - SQL, HTML Formulare, http, CORBA, ...
  - Mächtigkeit der Anfragesprache
- Lösen Datenmodellheterogenität
- Lösen schematische Heterogenität
  - Liefern globales Schema
- Reduzieren Anzahl der Datenmodelle (mit denen das IIS umgehen muss)
- Reduzieren Anzahl der Schemata
- Unterstützen globale Optimierung
  - Kostenmodell
  - Anfragefähigkeiten

Wrapper – Anforderungen

- Sollten schnell implementiert werden können
  - (< 1 Woche)
- Sollten wiederverwendbar sein
- Lokale Wartung (bei föderierten Systemen)

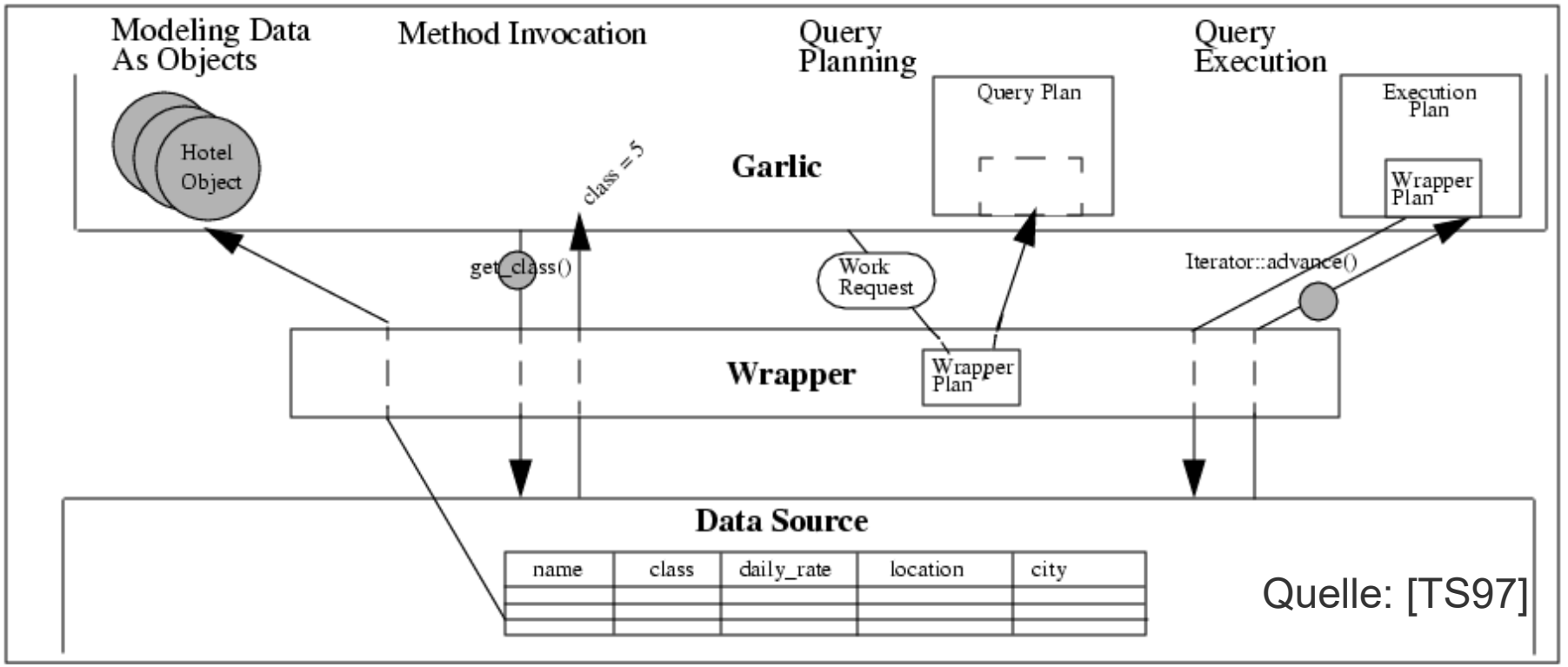
- An den Wrappern scheitern viele Projekte!
  - Deshalb Forschung zur schnellen oder sogar automatischen Wrappergenerierung.
  - Wrapperbibliotheken

Garlic Wrapper Generierung nach [TS97]

- Praktische Anforderungen aus [TS97]
  - Start-up Kosten gering (Stunden)
  - Erweiterbarkeit
    - Einfacher Start
    - Später Fähigkeiten der Quellen hinzufügen
  - Flexibilität
    - Möglichst breites Spektrum an Quellen abdecken
    - Neue Quellen stören Architektur nicht.
  - Optimierung
    - Nicht durch Autoren sondern durch IIS (hier: Garlic)

Garlic Wrapper

- Vier Grund-Services
  1. Modellierung und Zugriff auf die Daten in der Quelle
  2. Aufruf von Methoden in der Quelle
  3. Mithilfe bei der Anfrageplanung
  4. Anfrageausführung



Quelle: [TS97]

Garlic Wrapper

- Modellierung und Zugriff auf die Daten
  - Garlic nutzt OO Modell
  - Wrapper stellt Daten als Objekte mit Interface (globales Schema) und Implementierung (lokales Schema) dar.
  - Stellt Identität von Objekten her.

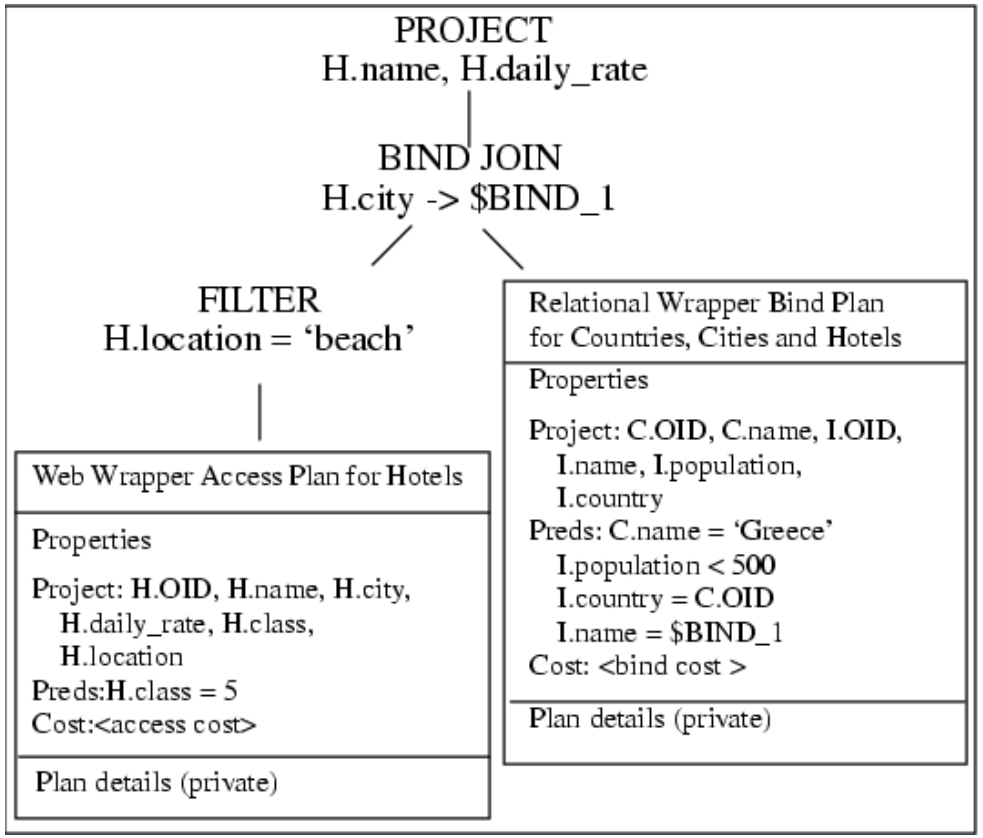
- Aufruf von Methoden in der Quelle
  - Implizit immer: Get_attr() für jedes Attribut
  - Implizit immer: Set_attr() für jedes nicht-read-only Attribut
  - Um auch Quellen abzudecken, die nur über Methoden zu erreichen sind.
  - Um besondere Fähigkeiten von Quellen auszuschöpfen.
  - Beispiel: display_Image(ImageID)

Garlic Wrapper

- Mithilfe bei der Anfrageplanung (query planning)
  - Garlics Anfrageplanung betrachtet alternative Pläne und sucht den besten heraus.
    - Kostenbasiert
  - Mediator verschickt „Teilaufgaben“ an Wrapper.
    - Wrapper kann Teile davon ablehnen (je nach Fähigkeiten der Quelle).
    - Mediator gleicht aus: Preprocessing und Postprocessing
  - Wrapper liefert null oder mehr Teilpläne zurück.
  - Teilpläne werden in Gesamtplan eingebaut.

Garlic Wrapper

- Anfrageausführung (query execution)
  - Mediator produziert Operatorbaum.
  - Wrapper-Teilpläne sind Blätter in dem Baum.
  - Pläne werden in Iteratoren umgewandelt
  - Pipelining

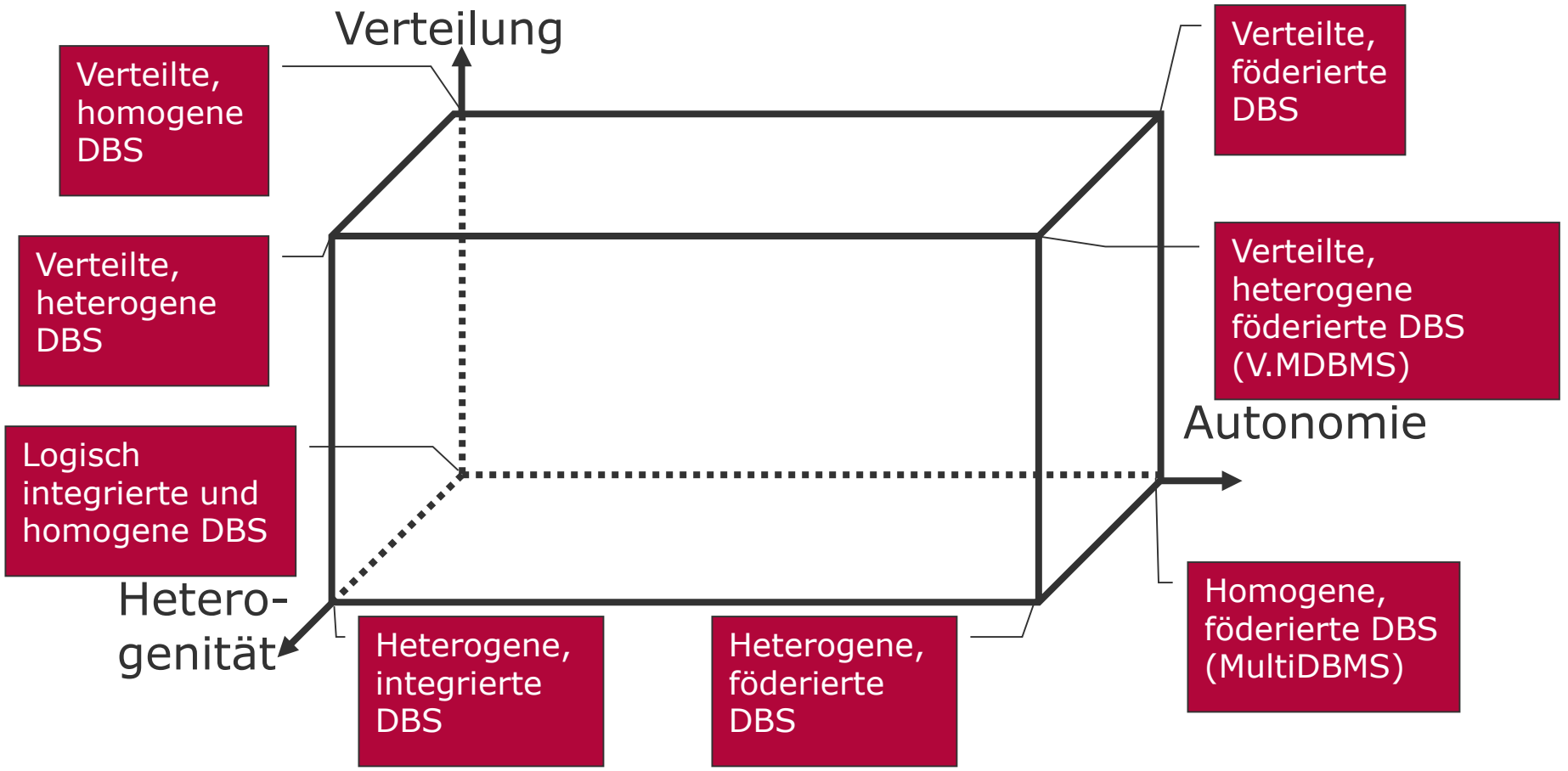


Überblick

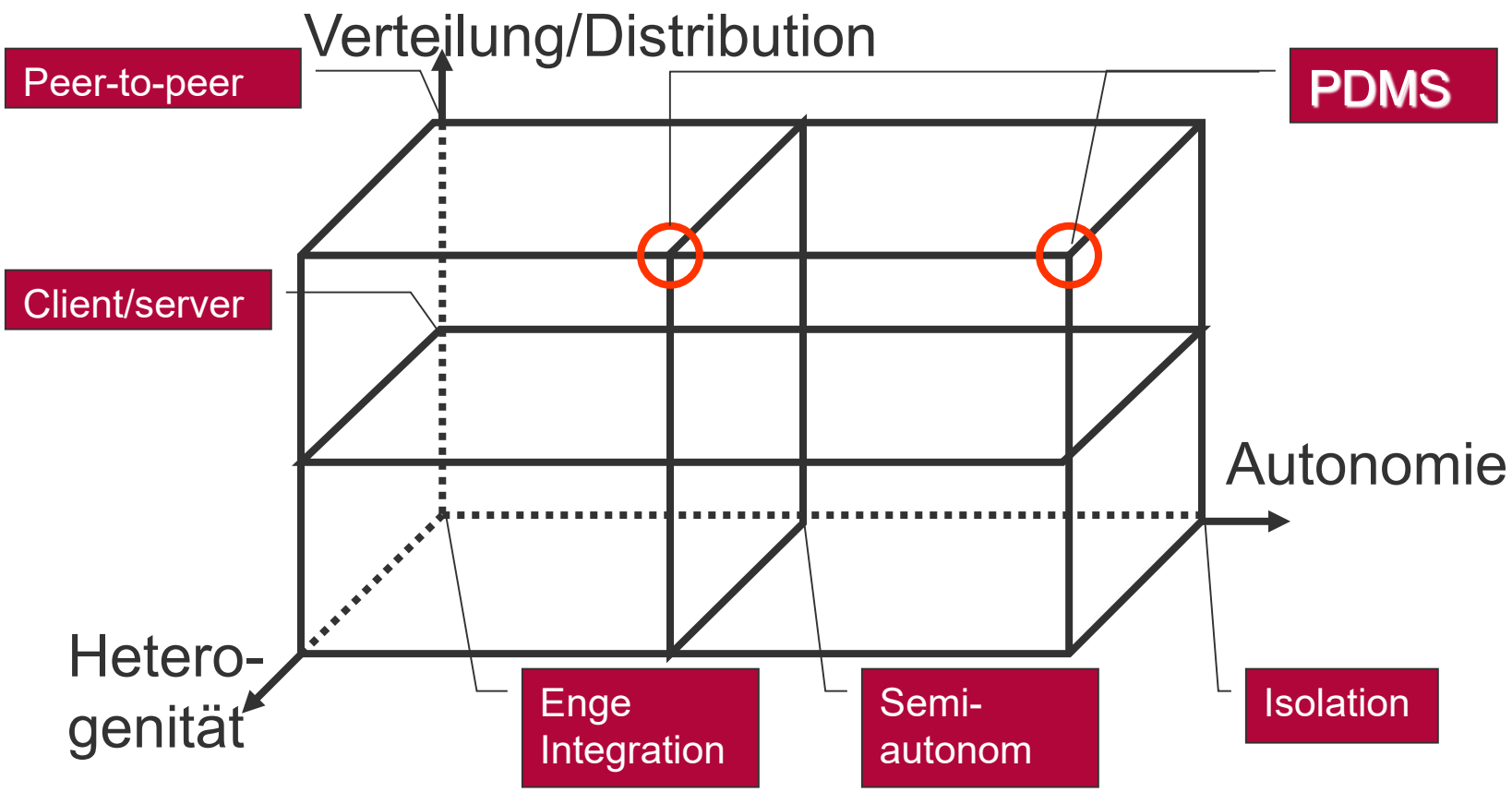
1. Architekturen
   - 3 Schichten Architektur
   - 4 Schichten Architektur
   - 5 Schichten Architektur
2. Mediator-Wrapper Architektur
   - Gio Wiederholds Definitionen
   - Konfigurationen
   - Mediatoren
   - Wrapper
3. **Peer-Data-Management**
   - Architektur
   - Anwendungen

Felix Naumann
Information Integration
Winter 2019/20

Wdh: Klassifikation von Informationssystemen nach [ÖV91]



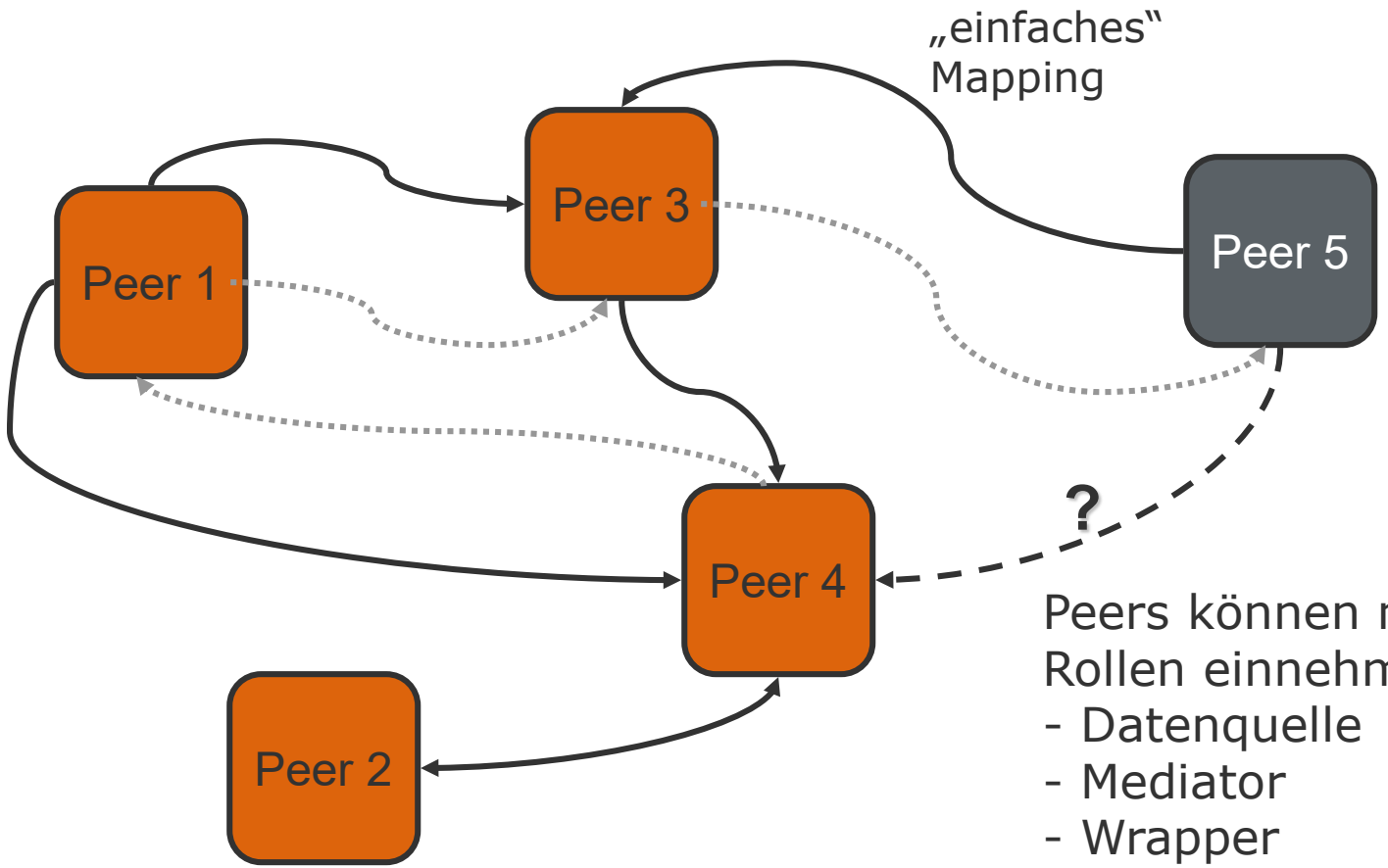
Wdh: Erweiterung der Klassifikation nach [ÖV99]



PDMS – Idee

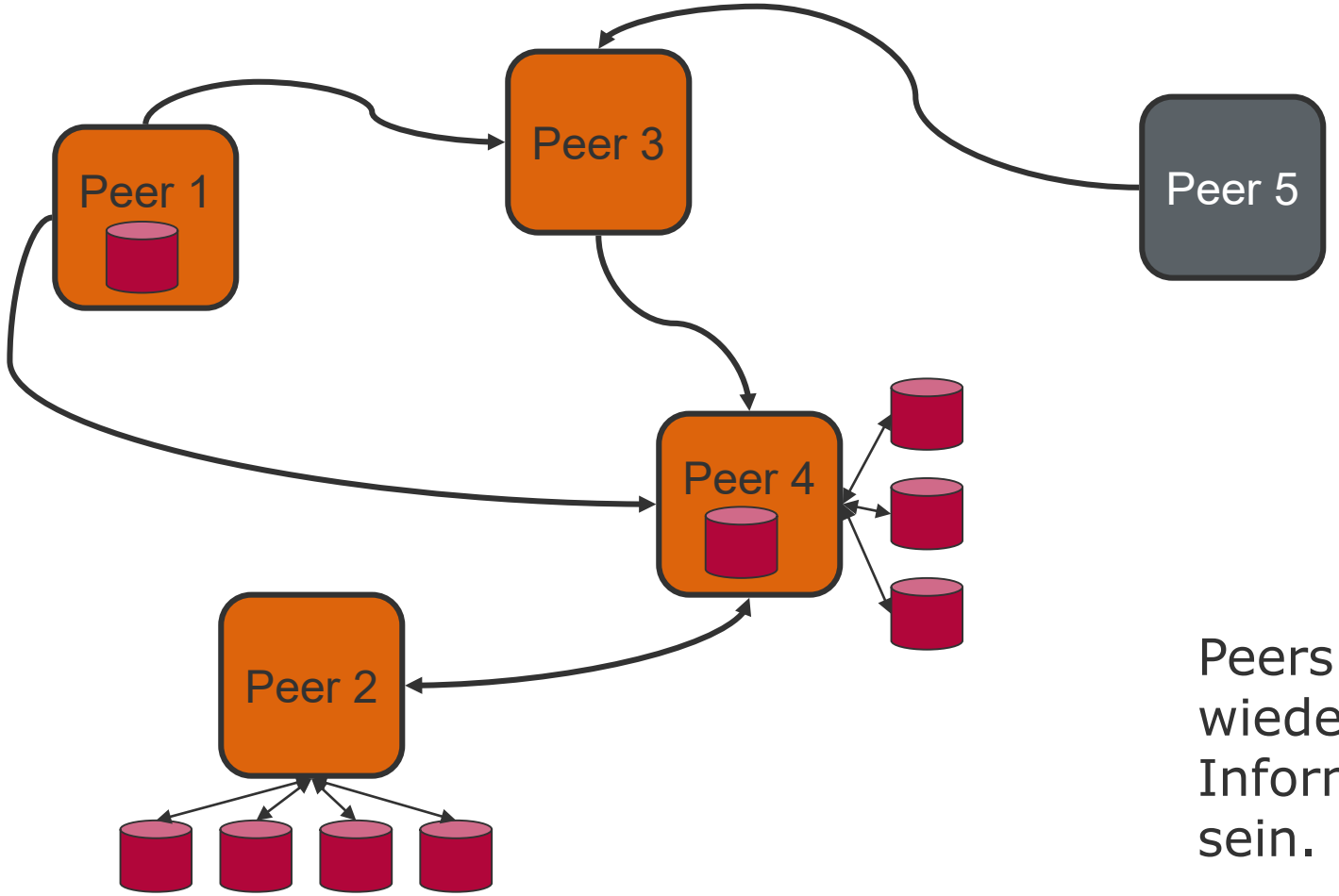
- Idee: Peer Netzwerk (P2P)
  - [HIST03], [HIMT03], [BGK+02]
- Jeder Peer kann
  - Daten exportieren (= Datenquelle)
  - Sichten auf Daten zur Verfügung stellen (= Wrapper)
  - Anfragen anderer Peers entgegennehmen und weiterleiten (= Mediator)
  - Anfrage stellen
- Verknüpfungen nicht zwischen lokalen und globalem Schema, sondern zwischen Paaren von Peers.

Peer-Data-Management Systeme (PDMS)



Peers können mehrere Rollen einnehmen:
- Datenquelle
- Mediator
- Wrapper
- Anfrager

Peer-Data-Management Systeme (PDMS)



Peers können selbst wiederum integrierte Informationssysteme sein.

PDMS vs. P2P Dateiaustausch

P2P

1. Nur ganze Dateien (grobe Granularität)
2. Einfachste Anfragen
   - Dateinamen, Keywords
3. Unvollständige Anfrageergebnisse (top-N)
4. Einfaches Schema
5. Hoch dynamisch
6. Millionen Peers
7. Datenübertragung direkt

PDMS

1. Objekte (feine Granularität)
2. Komplexe Anfragen
   - Anfragesprache (SQL, etc.)
3. Vollständige Anfrageergebnisse (zumindest erwartet)
4. (Normales) Schema
5. Kontrollierte Dynamik
6. Zig peers
7. Datenübertragung entlang des Mapping-Pfads

PDMS vs. FDBMS

Vorteile

- Nutzer/Anwendungen müssen nur das eigene Schema kennen.
- Alle Daten sind erreichbar (über die transitive Hülle der Mappings).
- Neue Schemata und Peers können leicht hinzugefügt werden (inkrementell).
- Mappings nur zu ähnlichsten Schemata

Nachteile / Probleme

- Mapping-Erstellung ist schwierig.
- Mapping Komposition ist schwierig.
- Viele Mappingschritte durch das Netzwerk:
  - Effizienz
  - Skalierbarkeit
  - Datenqualität

Offene Fragen

- Effiziente Platzierung von Daten?
- Updates?

Literatur

- Wichtige Literatur
  - [ÖV99] Principles of Distributed Database Systems. M. Tamer Özsu, Patrick Valduriez, Prentice Hall, 1999.
  - [SL90] Amit P. Sheth and James A. Larson, Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases, ACM Computing Surveys, Vol. 22(3), pp183-236, 1990.
  - [Con97] Stefan Conrad, Föderierte Datenbanksysteme. Springer, Heidelberg 1997.
  - [HIST03] Alon Y. Halevy, Zachary G. Ives, Dan Suciu, Igor Tatarinov. Schema Mediation in Peer Data Management Systems, ICDE conference 2003
  - [Wie92] Mediators in the Architecture of Future Information Systems, Gio Wiederhold, IEEE Computer Journal, 25(3), 38-49, 1992
- Weitere Literatur zu Architekturen
  - [LMR90] W. Litwin, L. Mark, N. Roussoupoulos, Interoperability of Multiple Autonomous Databases, ACM Computing Surveys, Vol. 22(3), pp267-293, 1990.
  - [HM85] Dennis Heimbigner, Dennis McLeod: A Federated Architecture for Information Management. ACM Trans. Inf. Syst. 3(3): 253-278 (1985)

Literatur

- Weitere Literatur zu Mediator Wrapper Architektur
  - [Gar95] Michael J. Carey, Laura M. Haas, Peter M. Schwarz, Manish Arya, William F. Cody, Ronald Fagin, Myron Flickner, Allen Luniewski, Wayne Niblack, Dragutin Petkovic, Joachim Thomas II, John H. Williams, Edward L. Wimmers: Towards Heterogeneous Multimedia Information Systems: The Garlic Approach. RIDE-DOM 1995: 124-131
  - [JS03] Querying XML data sources in DB2: the XML Wrapper, Vanja Josifovski and Peter Schwarz, ICDE conference, Bangalore, India, 2003
  - [TS97] Don‘t Scrap It, Wrap It! A Wrapper Architecture for Legacy Data Sources, Mary Tork Roth and Peter Schwarz, VLDB Conference 1997 Athens, Greece, 1997.
- Weitere Literatur zu PDMS
  - [BGK+02] Philip A. Bernstein, Fausto Giunchiglia, Anastasios Kementsietsidis, John Mylopoulos, Luciano Serafini, Ilya Zaihrayeu: Data Management for Peer-to-Peer Computing : A Vision. WebDB 2002: 89-94
  - [ÖV91] & [ÖV99] Principles of Distributed Database Systems. M. Tamer Özsu, Patrick Valduriez, Prentice Hall, 1991/1999.
  - [HIMT03] Alon Y. Halevy, Zachary G. Ives, Peter Mork, Igor Tatarinov. Peer Data Management Systems: Infrastructure for the Semantic Web. WWW Conference, 2003.
  - [NOTZ03] W.S.Ng, B.C. Ooi, K.L. Tan und A. Zhou: PeerDB: A P2P-based System for Distributed Data Sharing. ICDE 2003
  - [Len04] Maurizio Lenzerini: Quality-aware data integration in peer-to-peer systems. Invited talk at IQIS workshop, Paris, 2004.