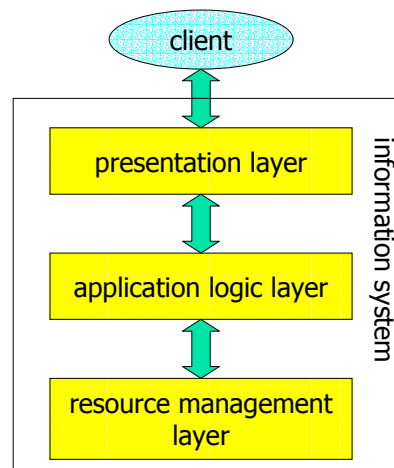


Chapter 1 – Motivation



Layers of an Information System

- Separation of functionality into three conceptual layers
 - presentation
 - application logic
 - resource (e.g., data) management
- Architecture of an IS
 - layers can be combined and distributed in different ways
 - 1-tier, 2-tier, 3-tier, n-tier



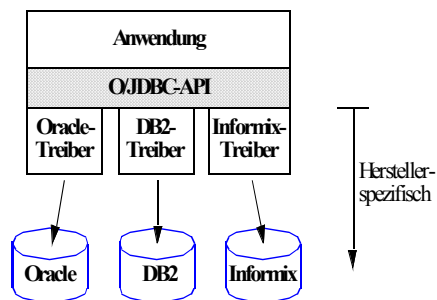
Middleware

- Middleware
 - supports the development, deployment, and execution of complex information systems
 - facilitates interaction between and integration of applications across multiple distributed, heterogenous platforms and data sources
- Two major aspects
 - middleware as a programming abstraction
 - middleware as infrastructure
- Principles
 - Make distribution transparent
 - Support standardized APIs/languages/data formats to overcome platform heterogeneity
 - Transform data and/or operations/requests to bridge structural and semantic heterogeneity
 - Application logic independent from infrastructure code
 - Powerful programming abstractions



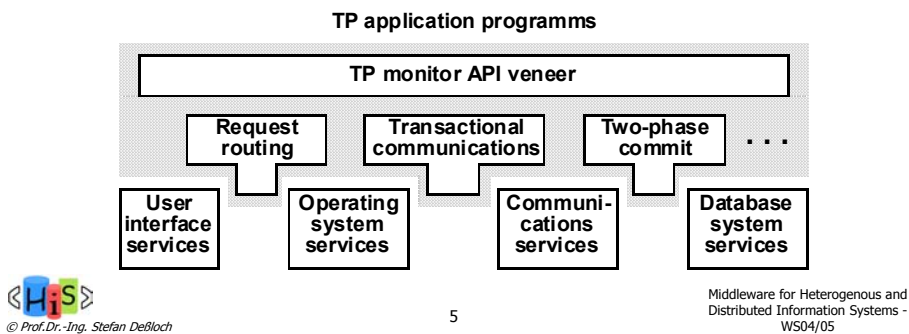
Datenbank-Gateways

- Einheitlicher Zugriff auf Daten
 - Anfragesprache
 - Metadaten (z.T. Datenmodell)
 - Programmierschnittstelle
- Dynamisches, spätes Binden an DBS
 - Aufrufchnittstelle (CLI)
 - keine herstellerspez. Pre-Compiler
 - Dynamisches Binden von Laufzeitbibliotheken
 - Späte Anfrageübersetzung
- Gleichzeitiger Zugriff auf mehrere DBMS
 - Architektur unterstützt arbeiten mit (mehreren) herstellerspez. Treibern
 - Koordination durch Treibermanager
- Herstellerspezifische Erweiterungen möglich

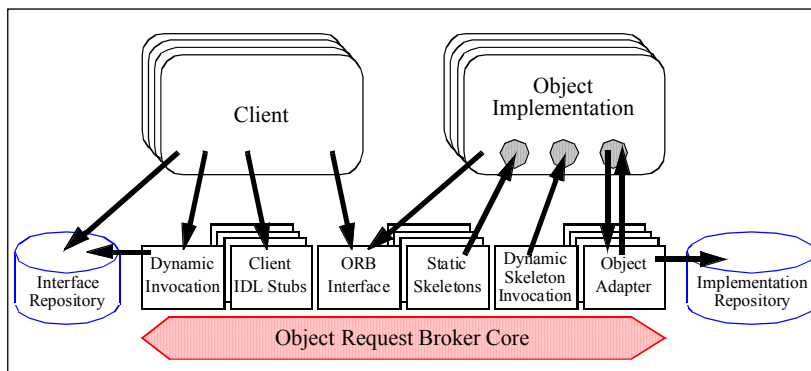


Aufgaben des TP-Monitors

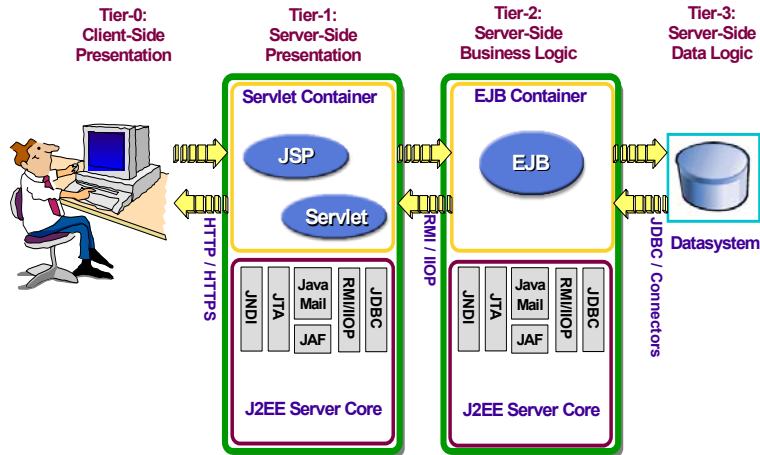
- Bewältigung der Heterogenität
- Kontrolle der Kommunikation
- Verwaltung der Terminals
- Präsentationsdienste
- Kontextverwaltung
- Start/Restart
- Programmverwaltung
- Konfigurationsverwaltung
- Lastbalancierung
- Autorisierung
- Bereitstellung von Administrationsschnittstellen



CORBA – Distributed Object Computing



4-Tier Distributed Computing in J2EE



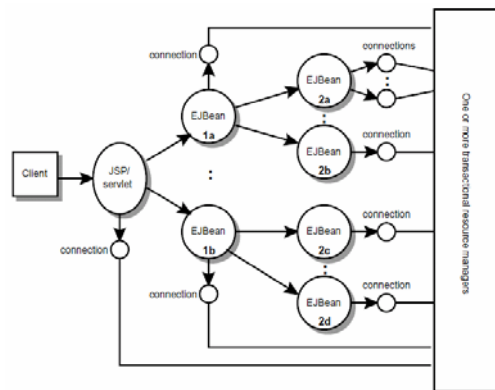
© Prof. Dr.-Ing. Stefan Deßloch

7

Middleware for Heterogenous and Distributed Information Systems - WS04/05

Important Services for Distributed IS

- Transactions
 - explicit
 - implicit/declarative
- Data Access
 - persistence
 - relationships
 - query
- Security



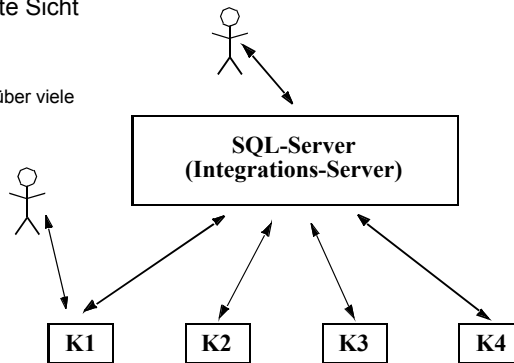
© Prof. Dr.-Ing. Stefan Deßloch

8

Middleware for Heterogenous and Distributed Information Systems - WS04/05

Datenintegration

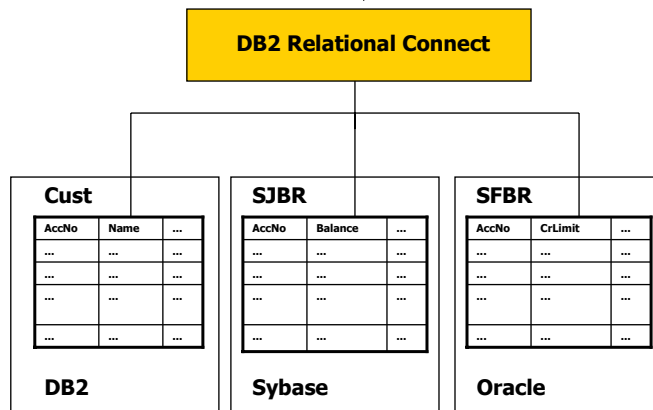
- Ziel: homogenisierte, integrierte Sicht der Daten
 - eine logische Datenbank
 - (Join-) Anfrage transparent über viele Datenquellen verteilt
- erfordert:
 - Wrapper
 - Mechanismen zur Datenintegration



Datenintegration - DB2 Relational Connect

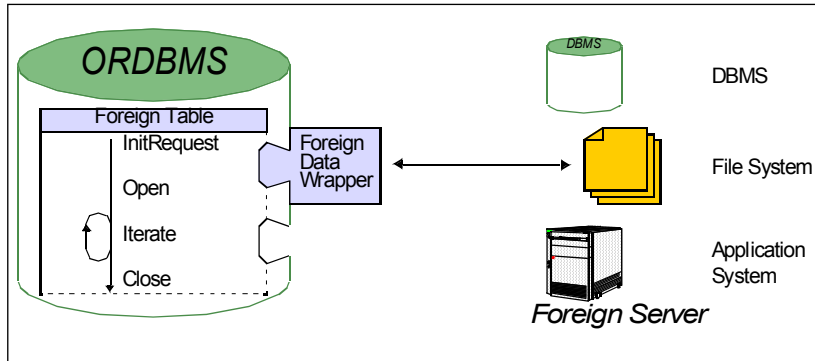
```

Select *
From Cust, SJBR, SFBR
Where Cust.Acct No = SJBR.Acct No
And SJBR.Acct No = SFBR.Acct No
    
```



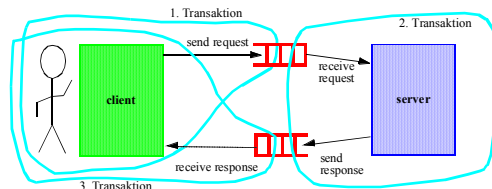
Datenintegration – SQL/MED

- 'Foreign Data Wrapper' in 'SQL/MED'



Message-Oriented Middleware (MOM)

- Botschaftenorientierte Interoperabilität
 - Programmiermodell: Kommunikation durch Austausch von Botschaften
- Bereitstellung persistenter Warteschlangen
 - zur Abwicklung asynchroner Kommunikation
 - als zuverlässige Nachrichtenpuffer
- Durchsatzoptimierung anstelle von Antwortzeitminimierung
- Lose Kopplung von Anwendungskomponenten
 - "Client" ist während der Abarbeitung einer Anforderung nicht blockiert
 - "Server" kann Zeitpunkt der Anforderungsbearbeitung flexibel wählen
 - ist mglw. zum Zeitpunkt des Client Enqueue garnicht verfügbar

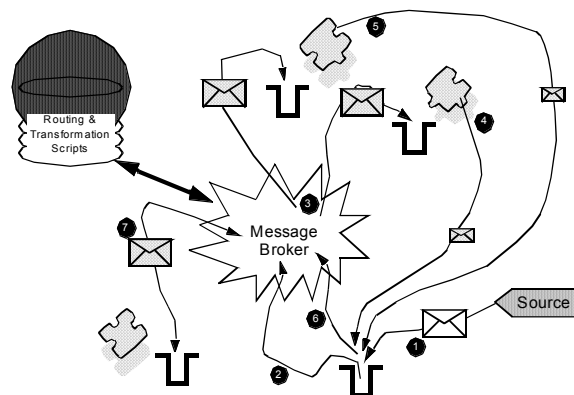


Enterprise Application Integration

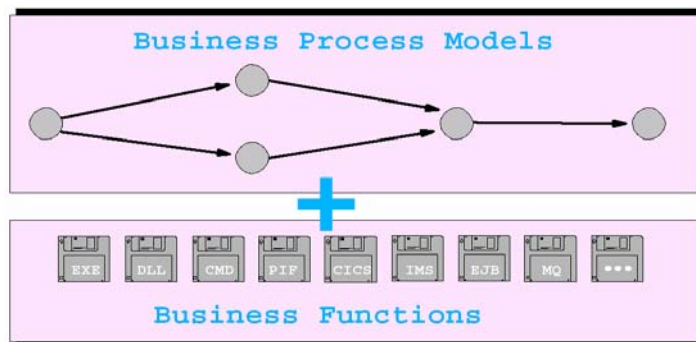
- Focus on application integration within an enterprise (vs. development of new application)
 - integration across different middleware platforms
 - major shift towards asynchronous interactions
- Message Brokers
 - based on MOM
 - hub-and-spoke (instead of point-to-point)
 - publish and subscribe model to link applications together
- Business Process Modeling and Workflow Management Systems
 - make integration logic explicit, easy to modify/extend
 - "programming in the large"



Message Brokering – Processing Model



Workflow-Based Applications: Structure

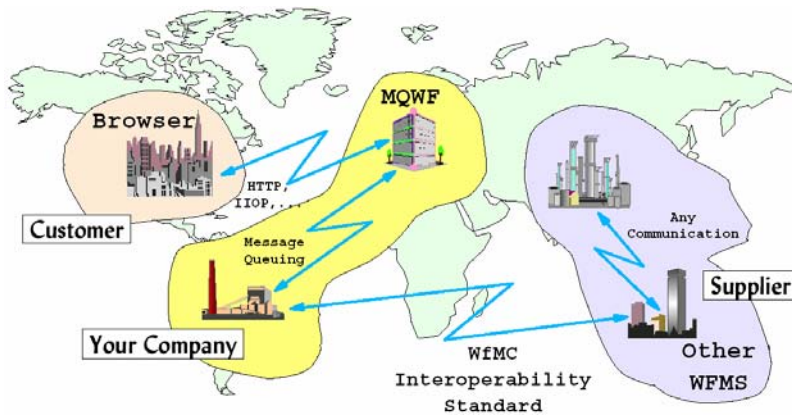


Workflows And External Communications



- Customers invoke company's applications to perform certain steps of the business process
 - E.g. place on order, inquire status,...
 - Company's applications must get a browser-based front-end for that purpose ("web-up")
- Workflow activities may directly communicate with the outside
 - Send e-mail, faxes, messages,...
- Workflow activities may trigger actions in another company
 - Simple invocation of program or start of another workflow ("subprocess" from invokers point-of-view)
 - Such "business-to-business" scenarios are the base for realizing sophisticated "supply chains"

Virtual Enterprise: Scenario



Business-To-Business (B2B) Integration

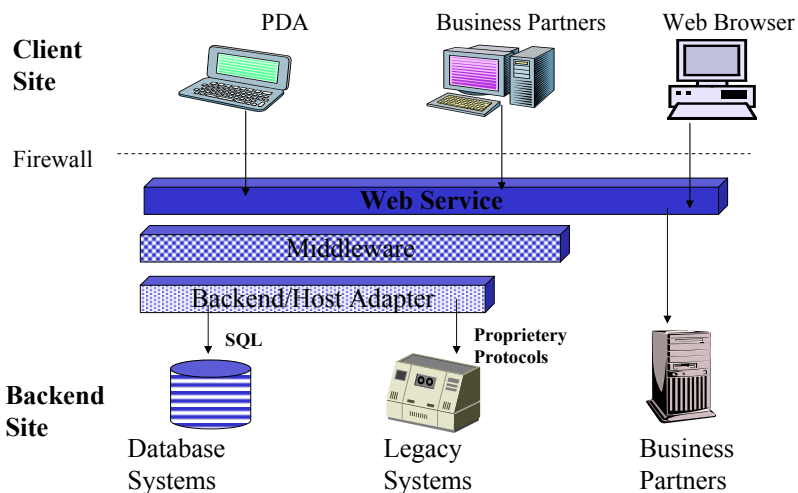
- Goal: facilitate interaction among trading partners, across companies
 - Establish relation between processes of different enterprises
 - Predominant are relation to suppliers, and customer relations to other enterprises like industrial consumers, retailers, banks
- Traditional B2B has focused on well-defined, standard message formats and protocols (e.g., RosettaNet, cXML)
 - Ad hoc B2B occurs today via XML over HTTP
- How to publish business functions to customers, partners and suppliers?
 - E.g. access to reservation systems, quote systems
 - Programmatic access to a service, independent of underlying implementation and client software
- Technologies such as Corba, DCOM, EJBs, etc. barely present in this context



Web Services

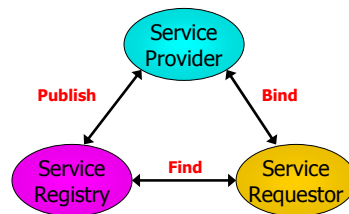
- New distributed computing platform built on existing infrastructure including XML & HTTP
 - Web services are for B2B what browsers are for B2C
- Self-contained, self describing, modular service that can be published, located and invoked across the web
 - Refer to open standards and specifications:
 - component model (WSDL)
 - inter-component model communication (SOAP)
 - discovery (UDDI)
 - Platform- and implementation-independent access
 - Described, searched, and executed based on XML
 - E.g. credit card validation, airline schedules, rental car.
- Enable component-oriented applications
 - Loose coupling from client to service
 - Enable to integrate legacy systems into the web
 - Useful for other distributed computing frameworks such as Corba, DCOM, EJBs

Web Service System Architecture



Service-Oriented Architecture (SOA)

- Service Requestor
 - Finds required services via Service Broker
 - Binds to services via Service Provider
- Service Provider
 - Provides e-business services
 - Publishes availability of these services through a registry
- Service Registry
 - Provides support for publishing and locating services
 - Like telephone yellow pages

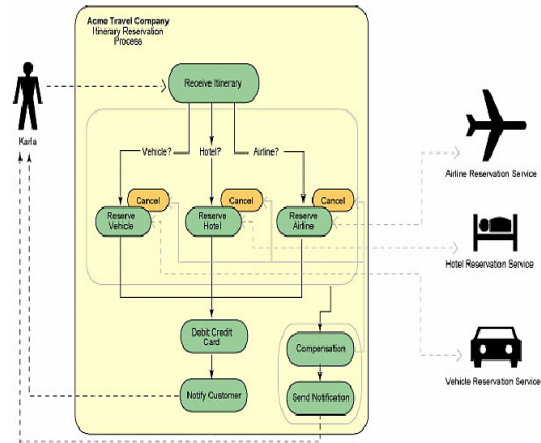


Standards

- UDDI
 - Universal Description, Discovery and Integration
 - Registry of and search for web services
- SOAP
 - Simple Object Access Protocol
 - Communication protocol
- WSDL
 - Web Services Description Language
 - Description of a service's functionality
- XML
 - eXtensible Markup Language
 - Underlying basic representation approach

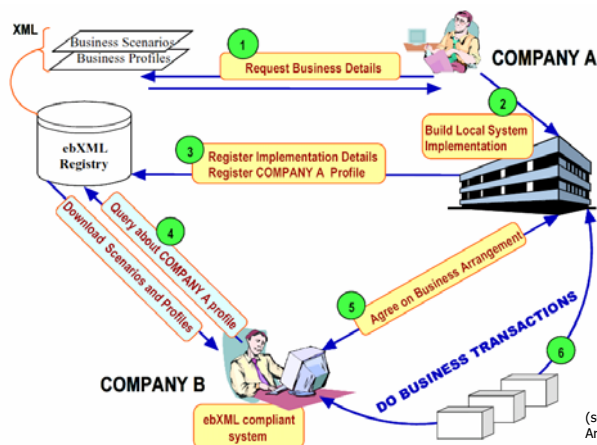
Web Services & Business Processes

- Business process making use of web services
- Business process externalized as a web service
- Long-running transactions
- Compensation
- Correlation
- Dynamic Binding of business partners and web services



e-Business Collaboration

- Example: ebXML



(source: ebXML Technical Architecture Specification)

