Hawlitzek Consulting

J2EE-Einsatz in größeren Projekten

Florian Hawlitzek

V1.0

J2EE-Erfahrungen
Übersicht

Hawlitzek Consulting

Erfahrungen mit J2EE

- in Projekten mit 25-40 Mitarbeitern
- Was beinhaltet J2EE?
- Deklarieren statt Programmieren
- Toolunterstützung
- Limitationen

Hawlitzek Consulting

JUGM Mailingliste

- Rene Kern:
  - "Extrem hilfreich wäre eine kurze Klärung des Begriffs Container bzw. EJB. Wozu benötigt man diese und welche Vorteile haben diese gegenüber Java Beans bzw. POJOs?
    [...]
    Mit Servlets, JSP, EJB, JMS, JCA hatten wir bisher noch keinen Kontakt, was sich nun hoffentlich ändert."

Hawlitzek Consulting

JUGM Mailingliste

- Aus den Antworten:
  - „Eine kurze Klärung von EJB dauert ungefähr eine Woche“
  - „Entity Beans sind IMHO ziemlich sch, außer in EJB 3.0“
  - „Ehrlich gesagt, wißt Ihr, auf was Ihr Euch da einlaßt? Hinter diesen Keywords steckt mordsmäßig viel Arbeit und Lernkurve.“
  - „An Eurer Stelle würde ich die Applikation mit nem modernen Webframework, Tomcat und Hibernate bauen; nix EJB, JMS, JCA.“
  - „Firmen wie Siemens, die selbst das Aufhängen von Kantinenplänen prozessoptimiert haben, mögen diese Komplexität bewältigen können und vielleicht sogar benötigen, um andere Probleme jenseits meines Erkenntnishorizonts, zu umgehen.“

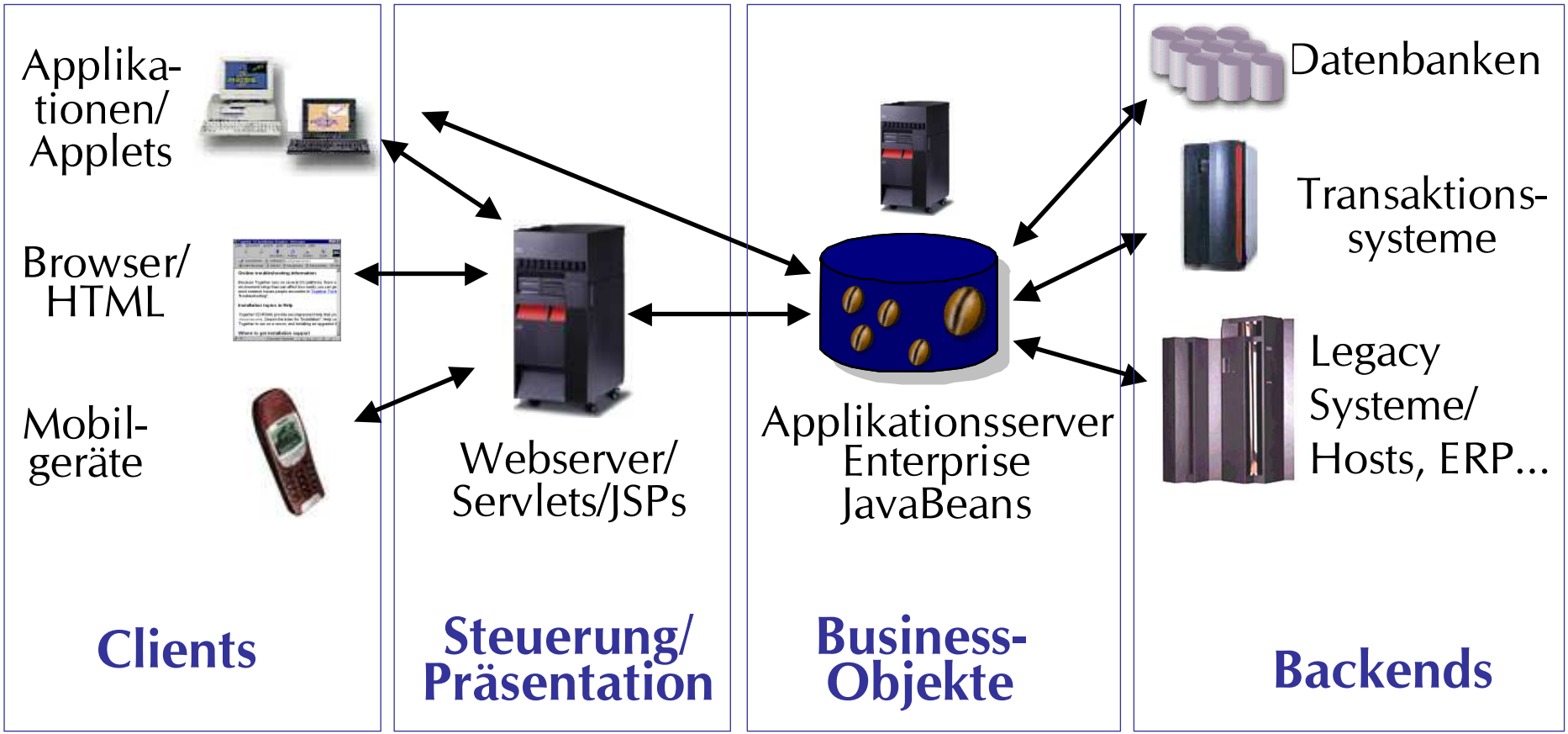
Hawlitzek Consulting

JUGM Mailingliste

- Fazit Rene Kern:
  - "Sehen das die anderen auch so das J2EE im Moment eher am aussterben ist, weil zu komplex?"
- Andreas Haug:
  - Ich kann nicht sagen, dass ich EJB Projekte kenne, die einen nennenswerten Umfang haben (also mehr als den Status Prototyp oder "wollten wir halt mal ausprobieren").

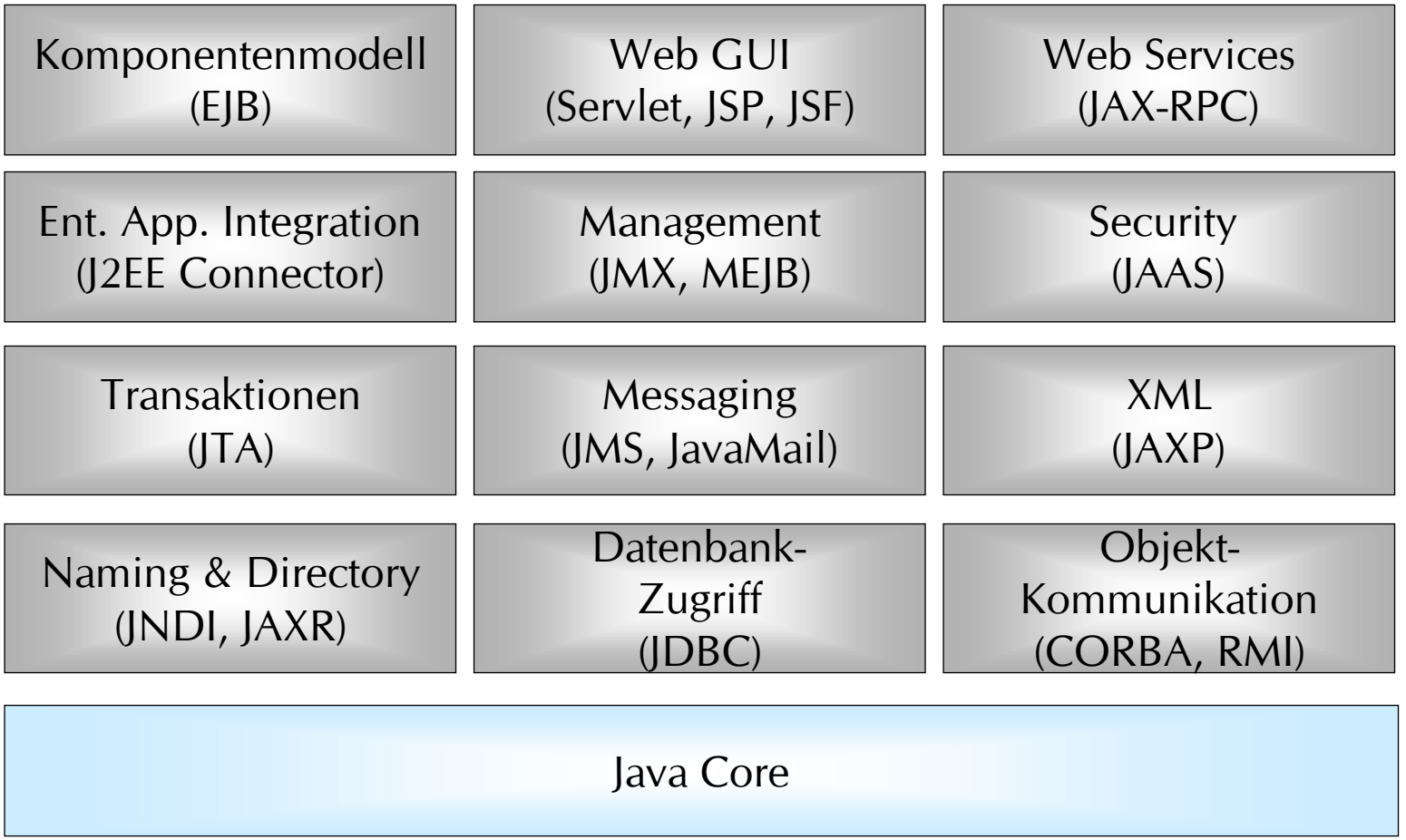
Mehrschicht-Architektur

Hawlitzek Consulting



J2EE-Erfahrungen
J2EE 1.4

Hawlitzek Consulting



J2EE-Erfahrungen
Projekte

Hawlitzek Consulting

Beispielprojekte

- Reporting (Bank)
  - 2000-2003
  - WebSphere, VisualAge for Java, Rational Rose
- Kreditanwendung (Bank)
  - 2003-2004
  - WebSphere, WSAD, TogetherJ
- Wetter (Medien)
  - 2004-2005
  - JBoss, MyEclipse IDE

J2EE-Erfahrungen Reporting-Projekt

Hawlitzek Consulting

Internes Kundenengagement-System

- Analyse von Geschäftsvorfällen nach verschiedenen Kriterien
- z. B. nach Unternehmensbereichen, Kundengruppen, Geschäftsarten, Laufzeiten...
- Ziel
  - Reduzierung des Zeitaufwandes für eine Engagementübersicht von 2 Wochen auf 5 sec

J2EE-Erfahrungen Reporting-Projekt

Hawlitzek Consulting

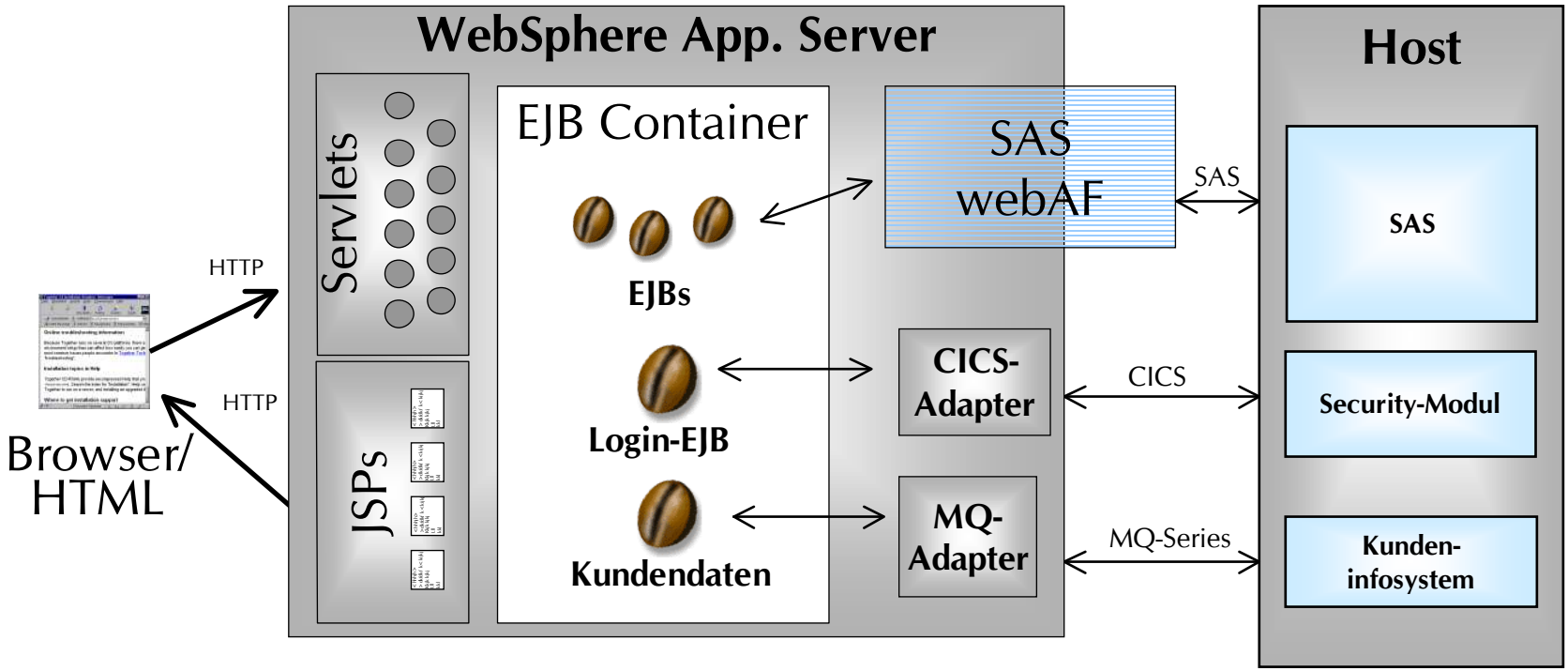
Internes Kundenengagement-System

- Reports
  - Feste, dynamische und Favoriten
  - Online-Recherchen nach bestimmten Filtern
- Technisch
  - J2EE: WebSphere Application Server (AIX)
  - EJB als Legacy-Wrapper und Sitzungsverwaltung
  - Servlets/JSPs für Frontend
  - SAS als OLAP-Server (Host)

J2EE-Erfahrungen Reporting-Projekt

Hawlitzek Consulting

Internes Kundenengagement-System



J2EE-Erfahrungen Kreditanwendung

Hawlitzek Consulting

Interne Kreditanwendung

- Komplette Unterstützung des Kreditprozesses
  - Front-Office + Back-Office
  - Workflow mit mehreren beteiligten Abteilungen
- Ziele
  - Vereinheitlichung des Prozesses und der SW-Landschaft
  - Modernisierung der Risikoprüfung

J2EE-Erfahrungen Kreditanwendung

Hawlitzek Consulting

Interne Kreditanwendung

- Bestandteile
  - Beantragung
  - Wiedervorlagen
  - Risikoprüfung
  - Vertragsvorbereitung
  - Überleitung an Backends
  - Protokollierung / Dokumentation

J2EE-Erfahrungen Kreditanwendung

Hawlitzek Consulting

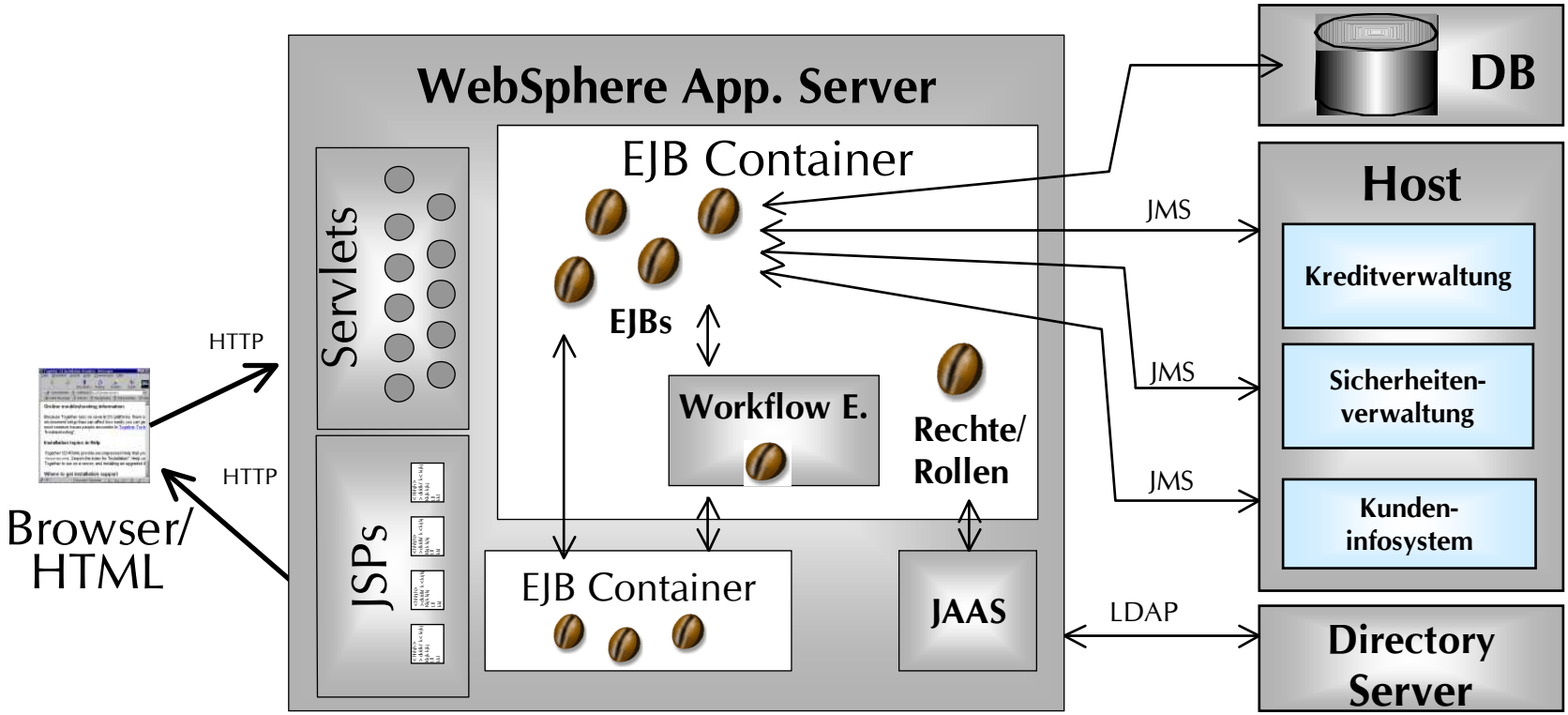
Interne Kreditanwendung

- Technisch
  - J2EE: WebSphere Application Server (Solaris)
  - EJB als Legacy-Wrapper
  - EJB-basierte Workflow Engine (Carnot)
  - Servlets/JSPs für Frontend
  - JMS für Hostkommunikation +Events
  - JAAS für Single-Login (Smartcards)

J2EE-Erfahrungen Kreditanwendung

Hawlitzek Consulting

Interne Kreditanwendung



Clients **Applikationsserver** **Backend**

Hawlitzek Consulting

JUGM Mailingliste

- Aus den Antworten:
  - Freilich kann ich auf den EJB Container verzichten und mir alle Funktionalitäten nach mix&match und best-of-breed etc. zusammensuchen.
  - Jemand hatte JBoss mit einer Linux-Distro verglichen. Guter Vergleich.
  - Das wird auch funktionieren und maybe ein schönes System ergeben. Der Komplexitätsgrad reduziert sich dadurch aber nicht zwingend!
  - Warum gibt es Linux-Distros ? Weil es mühsam ist alle Pakete einzeln zusammenzustopseln.

J2EE-Erfahrungen
Vorteile von J2EE

Hawlitzek Consulting

J2EE ist...

- eine Sammlung von APIs
  - Breite Abdeckung von Servertechnologien
  - DER Standard für Java Application Server
  - Deklarieren statt Programmieren
  - Umsetzung erprobter Pattern
- eine Zertifizierung
  - Für kompatible Server
  - Für Entwickler-KnowHow
- Optimiert auf
  - Skalierung, Verteilung und Management

J2EE-Erfahrungen Technologien 1/6

Hawlitzek Consulting

Eingesetzte APIs

- JDBC
  - praktisch konkurrenzlos
  - Connection Pooling
  - Probleme bei Verbindungsabbrüchen (nicht transparent)
- XML/JAXP
  - sehr große Verbreitung
  - gute Erfahrung, da lesbar und prüfbare Syntax (gegenüber Properties-Dateien und Binärformaten)

J2EE-Erfahrungen Technologien 2/6

Hawlitzek Consulting

Eingesetzte APIs

- JavaMail
  - sehr einfache Handhabung
- JMS
  - einfache Handhabung
  - ideal für effizienten Datenaustausch zwischen heterogene Systemen (z.B. Unix ↔ Host)
  - gute Integration mit EJB
    - Eventgesteuerte Message Driven Beans

J2EE-Erfahrungen Technologien 3/6

Hawlitzek Consulting

Eingesetzte APIs

- JAAS
  - Autorisierung früher
    - sehr proprietär
    - tiefe Eingriffe ins System (z.B. Integration RACF)
  - heute
    - externe Berechtigungssysteme leicht einbindbar
    - LDAP
    - Single Sign On: Smartcards

J2EE-Erfahrungen Technologien 4/6

Hawlitzek Consulting

Eingesetzte APIs

- Servlet
  - heute: Controller im MVC-Modell
  - praktisch konkurrenzlos
    - jedoch häufig nur als 1 Frontcontroller
    - Funktionalität häufig in Webframeworks verlagert (z.B. Struts...)
- JavaServer Pages
  - notwendiges Übel
    - häufig geänderte APIs
    - fehlerträchtig
  - durch TagLibs erträglich

J2EE-Erfahrungen Technologien 5/6

Hawlitzek Consulting

Eingesetzte APIs

- Enterprise JavaBeans
  - als zentrale Business Logik gut verwendbar
  - Stateless Session Beans
    - service-orientiert
    - für Kapselung von Remotezugriffen ideal
  - Stateful Session Beans
    - besseres Handling/Performance als Sessiondaten in Servlets
  - Message Driven Beans + Timer
    - endlich nicht nur synchrones Verhalten

J2EE-Erfahrungen Technologien 6/6

Hawlitzek Consulting

Eingesetzte APIs

- Entity Beans
  - Ansatz gut
    - leicht generierbar
    - Abstraktion von transaktionalem/relationalem Modell in Objekte
  - Umsetzung schlecht
    - starke Änderungen 1.0 → 1.1 → 2.x → 3.0
    - EJB QL mangelhaft
    - Vererbung/Relationen
    - suggeriert leichtes Handling, erfordert aber viel Wissen

J2EE-Erfahrungen
Tools 1/2

Hawlitzek Consulting

Server und Werkzeuge

- Application Server
  - WebSphere
  - JBoss
- Entwicklungsumgebungen
  - VisualAge for Java
  - WebSphere Studio App. Developper (WSAD, jetzt Rational App. Dev)
  - MyEclipse IDE

J2EE-Erfahrungen
Tools 2/2

Hawlitzek Consulting

Server und Werkzeuge

- UML-Designer
  - Rational Rose, TogetherJ
- Change Management / Testing
  - Excel, Mantis, TestDirector, Change Synergy
  - SilkTest, JProbe, JUnit

J2EE-Erfahrungen
Learned Lessons 1/3

Hawlitzek Consulting

Was hat sich bewährt?

- Enterprise Java Beans
  - hervorragend geeignet für den Aufbau unternehmenseigener Frameworks
    - Kapselung von Legacy Systemen (Cobol Copystrecken durch Java SOA ersetzen)
    - Standard-Schnittstellen für fachliche Systeme (z.B. Stammdatenverwaltung, Reporting...)
  - gute Monitoringmöglichkeiten
  - verbirgt Komplexität von Transaktionen, nebenläufiger Programmierung, Monitoring, Security

J2EE-Erfahrungen
Learned Lessons 2/3

Hawlitzek Consulting

Was hat sich bewährt?

- Iterative Vorgehensweise
  - zügige Integration:
    Daily Builds – ca. alle 2 Wochen ein Release
- starke Kapselung
  - aller Fremdsystem-Aufrufe
  - aller technischer Systeme
  - klare Abhängigkeiten, lose Kopplung
- konsequente Anwendung der J2EE Pattern

J2EE-Erfahrungen
Learned Lessons 3/3

Hawlitzek Consulting

Was hat sich bewährt?

- Klare Zuordnungen
  - für jede Schnittstelle
    - 1 Person + 1 Stellvertreter
    - Pair Programming in kritischen Bereichen
- Qualitätsmanagement (QM)
  - Code Walks
  - Change Management
  - Monitoring → punktuelles Tuning
  - Refactoring
  - Unittests

J2EE-Erfahrungen
Bitter EJB

Hawlitzek Consulting

Was nervt?

- Komplexes Classloading
- viel Schreibaufwand bei EJBs
  - Home Interfaces, Deployment Deskriptoren
  - Abhilfe: IDEs, xDoclet
- EJB QL
- DB-Transaktion nicht in-place behebbar
- Scriplets in JSPs
- API-Wechsel mit massiven Umstellungen (EJB, JSP)

Hawlitzek IT-Consulting
IBM Business Partner

- Anschrift:

  Hawlitzek IT-Consulting GmbH
  Landsberger Str. 155
  80687 München
  info@hawlitzek-consulting.de

http://www.hawlitzek.de

- Anbieter von:
  - Schulungen und Consulting
    zu Java, Eclipse und WebSphere