

Auf dem Weg zu einer objektorientierten Architektur

Bericht über ein Technologieprojekt der
Barmenia Versicherungen und
Object Technology Practice, IBM

Hans-Ulrich Mörs, Abt. Leiter AE Leben, Sach, HUK

Dieser Erfahrungsbericht über das Barmenia „Architektur Projekt“ soll Sie über folgende Aspekte informieren!

- Ausgangssituation
- Projektziele
- Projektvorbereitung
- Projektverlauf
- Projektergebnisse
- Ausblick

Wir arbeiten derzeit nach dem auf unsere Belange angepaßten V-Modell mit strukturierten Methoden und dem Case-Tool Innovator.

- V-Modell 97
 - Softwareentwicklung (Analyse, Design)
 - Projektmanagement
 - Qualitätssicherung
 - Konfigurationsmanagement
- Strukturierte Methoden
 - SA, (S)ERM, SD/MD
- Innovator 6.1, MID GmbH, Nürnberg

Unsere Plattform für die Implementierung ist heute ein Host der Firma Bull mit Cobol 85, Codasyl-Datenbank und Transaktionssystem.

- Host
 - DPS 9000, Bull AG; Betriebssystem GCOS8
 - Cobol 85, Datenbank IDS2, TX-System TP8
 - Programmentwicklungssystem SOLD
 - Terminals
- Server/Clients
 - RS6000, PC; Betriebssysteme Unix/NT
 - Datenbank Informix
 - SAP

Wir möchten aus dieser „proprietären Welt“ in eine offene, standardisierte Entwicklungs- und Betriebsumgebung migrieren.

- Internationale- oder Markt-Standards
 - Methoden, Verfahren, Architektur
 - Werkzeuge
- Skalierbare Systeme
 - Software-Komponenten
 - Hardware
- Unternehmerische Freiheitsgrade
 - Make or Buy Entscheidungen
 - Verhandlungsspielraum

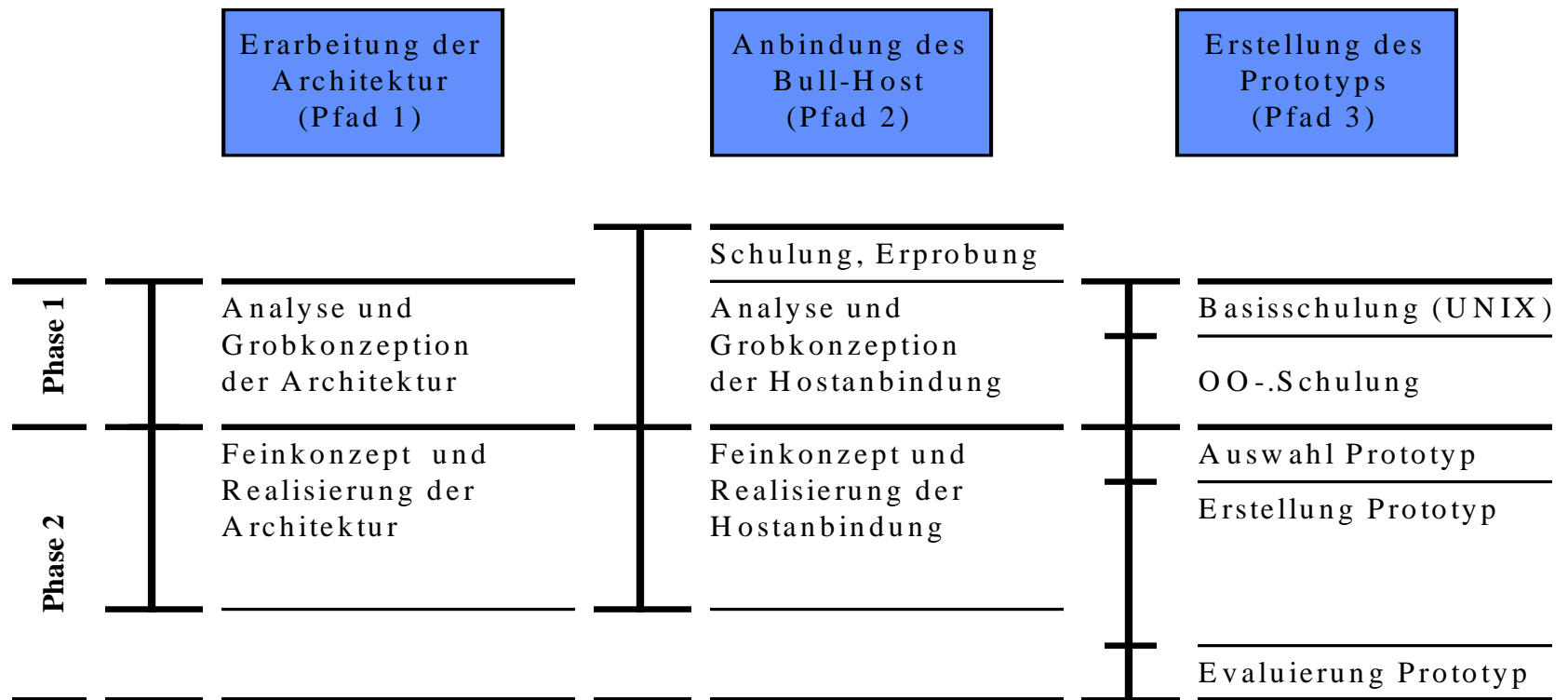
Um künftig die Wünsche unserer Kunden noch besser durch flexible, bedarfsorientierte Geschäftsprozesse befriedigen zu können, ...

- Kundenservice
 - Zeitliche Verfügbarkeit
 - Ortstransparenz
- Produktpalette
 - Individuelle Risikovorsorge
 - Kundennutzen
- Kundenzentrierung
 - Geschäftsprozesse
 - Informationsverarbeitung

... erarbeitet das „Architektur-Projekt“ ein stabiles, architekturgetriebenes Vorgehen für die Entwicklung von wiederverwendbaren SW-Komponenten.

- Architektur (Leitbild für Entwurf, tech. Frameworks)
- Vorgehen (Inhalt und Abfolge von Arbeitsschritten)
 - objektorientiert und strukturiert
- Methodik (Ausdrucksmittel für Modellbildung)
- Werkzeuge (Build- und Run-Time)
- Organisation (Projekt- und Aufbauorganisation)

Es wurde ein erster Projektplan erarbeitet, der das Projekt in zwei Phasen und drei Pfade (Architektur, Prototyp, Host-Anbindung) strukturiert.



Ausgewählte Werkzeuge wurden ohne vorherige Evaluierung eingesetzt und sollen durch die MA im Projekt auf ihre Eignung geprüft werden.

- Innovator 6.1
 - OO-Workstation (UML, Business Process)
- Visual Age for Java, Enterprise Edition 2.0
- Visual Age Persistenz
- Java Blend
- Visibroker 3.2
- Optimize It 2.0

Die Hardware für Client und Server, die Betriebssysteme, die Netzumgebung und die Datenbank waren vorgegeben.

- Bull Host
- Unix Server (AIX)
- Windows NT Clients
 - gute Ausstattung (Pentium, 128-256MB)
- 10 Mbit-Ethernet
- unternehmensweites Intranet (TCP/IP)
- Informix-Datenbank

Das Team von insgesamt 13 Mitarbeitern bestand aus drei Teilteams, dem Architekturteam, dem Prototypteam und dem Hostteam.

- Architekturteam
 - IBM Chef-Berater, (Architekt), Methodiker, Analytiker/Designer, Systementwickler
- Prototypteam
 - IBM Berater, Analytiker/Designer, Anwendungsentwickler, Systementwickler, Fachbereichsmitarbeiter
- Hostteam
 - Bull Berater, Systemprogrammierer

In Phase 1 (3 Monate) erhielten alle Teammitglieder eine Grundausbildung und anschließend weiterführende, rollenspezifische Schulungen ...

- Unix-Grundausbildung (3 Tage, alle)
- Konzepte der Objektorientierung (ca. 3 Tage, alle)
- Objektorientierte Analyse/Design (3 Tage)
- Java Sprachschulung (5 Tage)
- Java Werkzeugschulung (5 Tage)
- Verteilte Systeme (Corba) (1-2 Tage)
- „training on the job“

... anschließend wurden die Grundlagen für die zu erarbeitende Architektur und die Einbeziehung des Bull-Hostes gelegt.

- Architektur
 - Analyse der Architekturziele
 - Grobentwurf der Architektur
 - Systemarchitektur
 - technische Architektur
- Hostanbindung
 - Analyse der technischen Alternativen
 - Grobentwurf der Hostanbindung

In Phase 2 (ca. 8 Monate) wurde die Technische Architektur, die Systemarchitektur und die Hostanbindung ausspezifiziert, ...

- Architektur
 - Technische Architektur
 - ORB, Persistenz, Generator
 - Systemarchitektur
 - EGO, AGO, PGO
 - Modell, View, Controler
- Hostanbindung
 - JDBC-Treiber
 - CPI-C Kommunikationsschnittstelle

... die einzusetzenden Methoden und Verfahren beschrieben ...

- Angepaßtes V-Modell
- Analyse/Design (UML)
 - Geschäftsprozeßmodellierung
 - Use Cases, Szenarien
 - Objekt-Modelle
 - Interaktionsdiagramme
- Visuelle Java Programmierung
- Reviews, Inspektionen, Workshops

... weitere „Middleware“ selbsterstellt oder zum Einsatz gebracht ...

- Generator
 - aus Innovator Modellen
 - unter Berücksichtigung der Architektur
 - nach Visual Age for Java
- GUI Framework (Swing)
- Enterprise Java Beans
- Workflow

... und alle Aspekte im Rahmen des Prototyps erprobt.

- Geschäftsprozeß aus Vertragsverwaltung Leben
- Verlängerung/Verkürzung (5 Arbeitsschritte)
 - Antragseingang bearbeiten
 - Antragsprüfung durchführen
 - Prüfungsergebnis bewerten
 - Antrag durchführen
 - Antragsteller informieren
- Prozeßrahmen (Workflow)
 - vom Kundenwunsch bis zur Kundeninformation

Die Mitarbeiter sind durch Schulungen, „training on the job“ und Workshops in der Lage, eine produktive Anwendungsentwicklung zu beginnen.

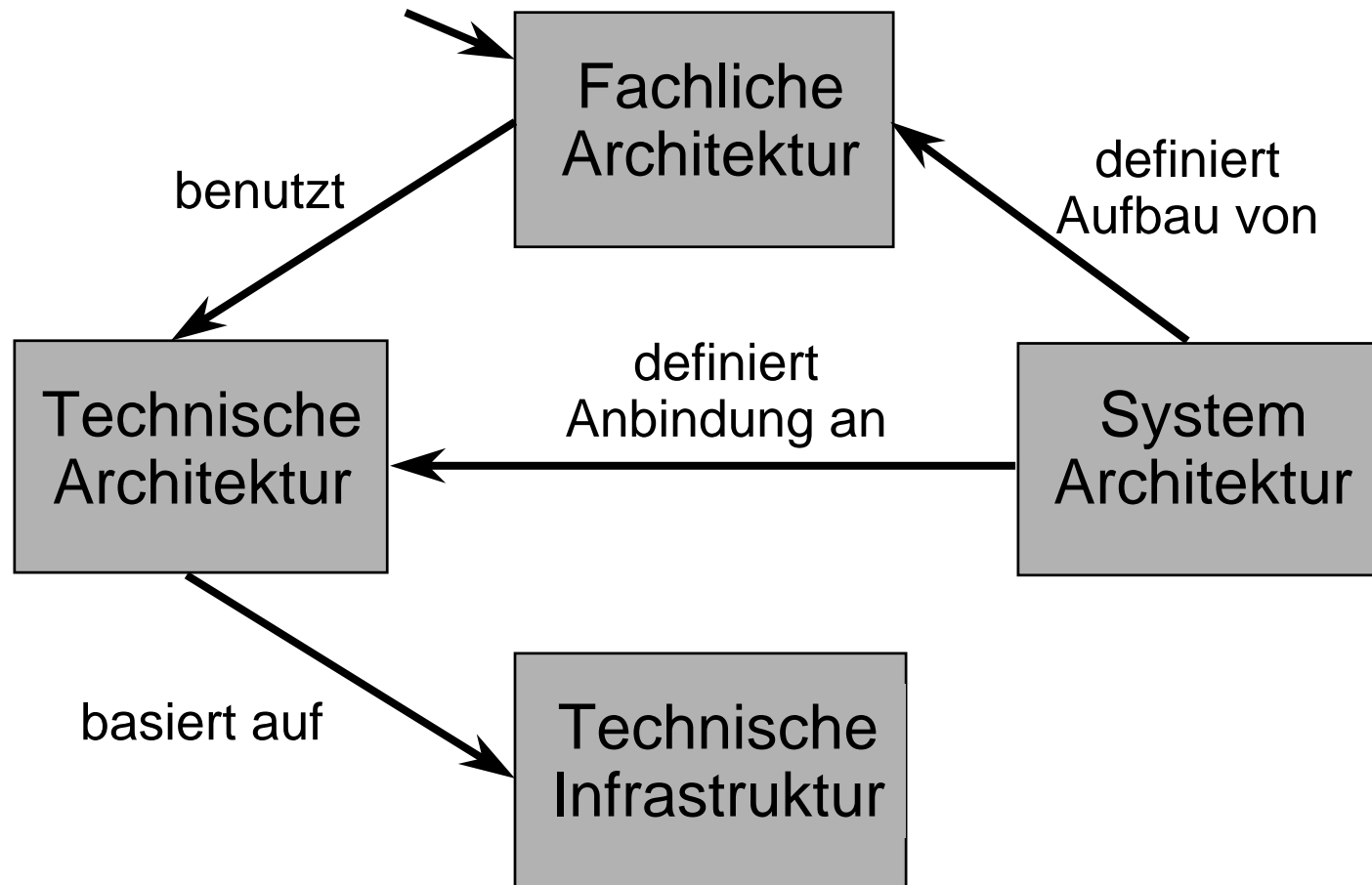
- Operatives Wissen
 - Vorgehen, Architektur, Werkzeuge ☺
- Strategisches Architekturwissen
 - Weiterentwicklung der Betriebs- und Entwicklungsumgebung ☹
- Methodisches Wissen
 - Modellierungs- und Programmiertechniken ☺
- Prozeßwissen ☺

Die Ergebnisse sind in allen Bereichen in Form einer schriftlichen Dokumentation festgehalten und werden von „document-ownern“ weiterentwickelt.



- Konzeptionsdokumente ☺
 - Architekturkonzeption, Technische Architektur, Systemarchitektur, Glossar
- Dokumente für Anwendungsentwickler ☺
 - Modellierungs-, Java-Programmier-richtlinien, Controller/Views, Namenskonventionen, GUI-Design
- Dokumente für Systementwickler ☺
 - Corba, Workflow, Modellabbildung, GUI-Framework, Persistenz
- Schulungsunterlagen ☺
 - Foliensätze, Schulungsbeispiel

Der Begriff „Architektur“ wird in technische Architektur, Systemarchitektur, Facharchitektur und technische Infrastruktur differenziert.

Projektergebnisse



Die erarbeitete Systemarchitektur gibt der Software, die die Fachlichkeit der Anwendung umsetzt, die innere Struktur vor.

- Modellebene 
 - kategorisierte Geschäftsobjekte
Entity- , Activity- und Process-GO's
 - vollständig definierte Elemente
Vererbung, Assoziationen, Attribute,
einfache und zusammengesetzte Typen,
Methoden, Metaphern, Leitbilder
- Implementierungsebene 
 - „thin client“ Muster
Präsentation, Modellogik, Datenhaltung
 - Generator-Ansatz
Modell - Kapselung - Implementierung

Was sind eigentlich EGO's ... ?

- Entitäts Geschäftsobjekte
 - Gegenstände des Geschäfts (z.B. Verträge, Produkte, Konten, Partner)
 - parametrisierbare, eigene „Intelligenz“
 - Arbeitsgegenstände für Aktivitäten und Geschäftsprozesse
 - mit Zustand (Lebenszyklus)
 - persistent
 - \approx Material
 - \approx Entity Beans

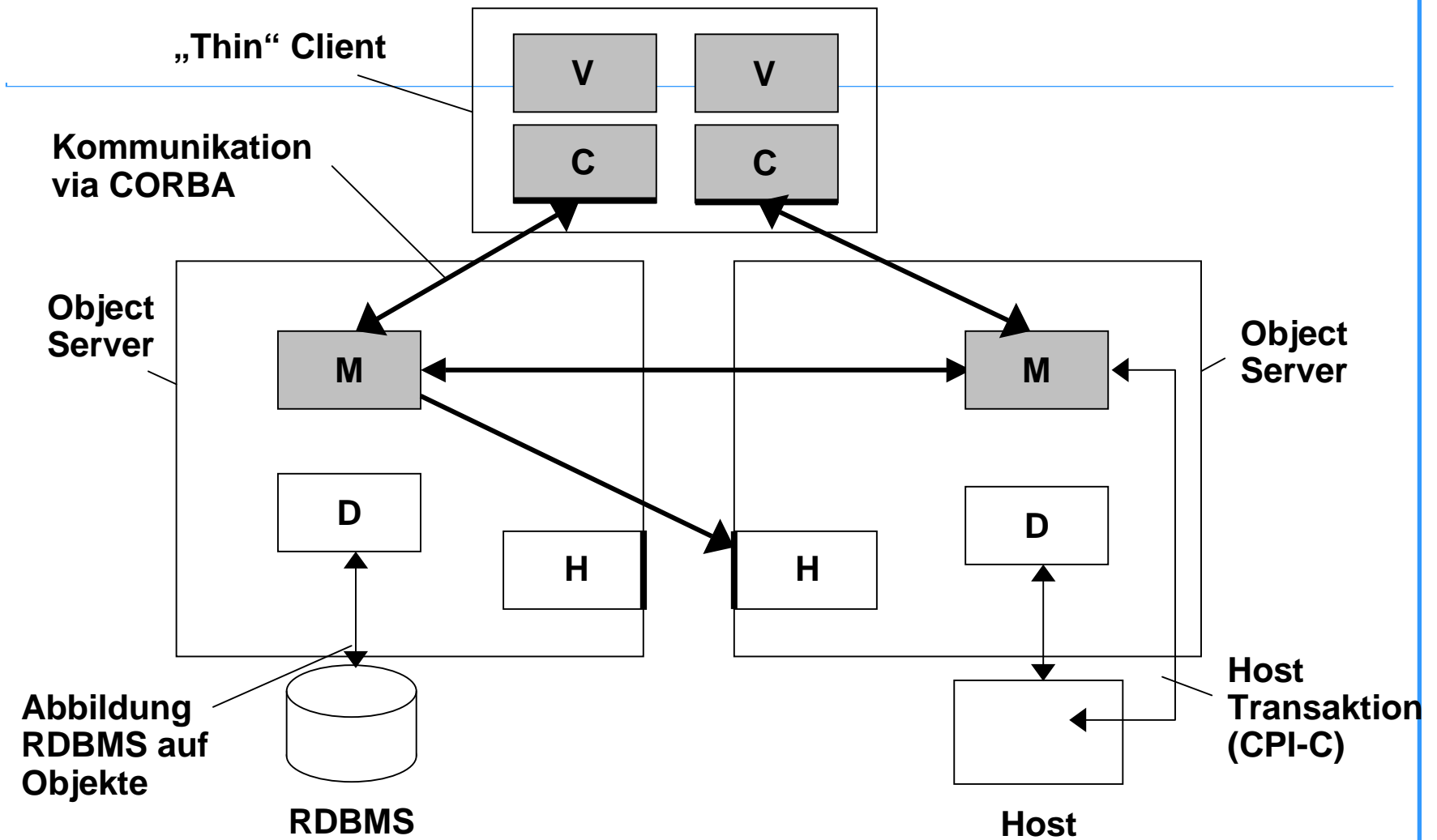
... und was sind dann AGO's ... ?

- **Aktivitäts Geschäftsobjekte**
 - parametrisierbare, elementare Ablaufeinheiten (z.B. rechtlicher Antragsprüfer, Kundensucher)
 - operieren auf EGO's
 - wiederverwendbare Komponenten
 - mit oder ohne Zustand
 - nicht persistent
 - \approx Werkzeug, \approx Automat
 - \approx Session Bean





... und was sind PGO's?

- Prozeß Geschäftsobjekte
 - Geschäftsprozesse oder -schritte (grob bis feingranular)
 - Kapselung von Prozess-, Ressourcen-, Zuständigkeits- und Kompetenzwissen (Kontext)
 - Arbeitsliste
 - mit Zustand (Lebenszyklus)
 - persistent
 - ≈ Werkzeug, ≈ Automat
 - ≈ Entity Bean




Die Architektur in einer groben Schemazeichnung.



Die erarbeitete technische Architektur orientiert sich weitgehend an den Standards CORBA und EJB.

- CORBA 
 - Verteilte Objekte
 - Legacy-Integration
 - Programmiersprachen-Integration
- Enterprise Java Beans 
 - Komponenten
 - write once - run everywhere
 - Service-Integration (Relationship, JTS) 
- Zusammenwirken von EJB und CORBA 




Sie bildet den Lebensraum („Technotop“) für alle Geschäftsobjekte ...

- Benutzungsoberfläche 
 - gekapselte Java Swing Klassenbibliothek
 - Komponenten für GO's (Vertrag, Partner)
 - Barmenia GUI Design Richtlinien
- Kommunikation und Verteilung 
 - generierte, gekapselte CORBA-Anbindung
 - Request-Optimierung Client/Server (XML)
 - Ziel: CORBA, EJB, RMI/IIOP (1999?)
- Datenhaltung 
 - Objektabbildung auf relationale Daten
 - Persistenz-Framework

... und berücksichtigt eine Vielzahl von technischen Aspekten, von denen einige nicht abschließend betrachtet wurden ...

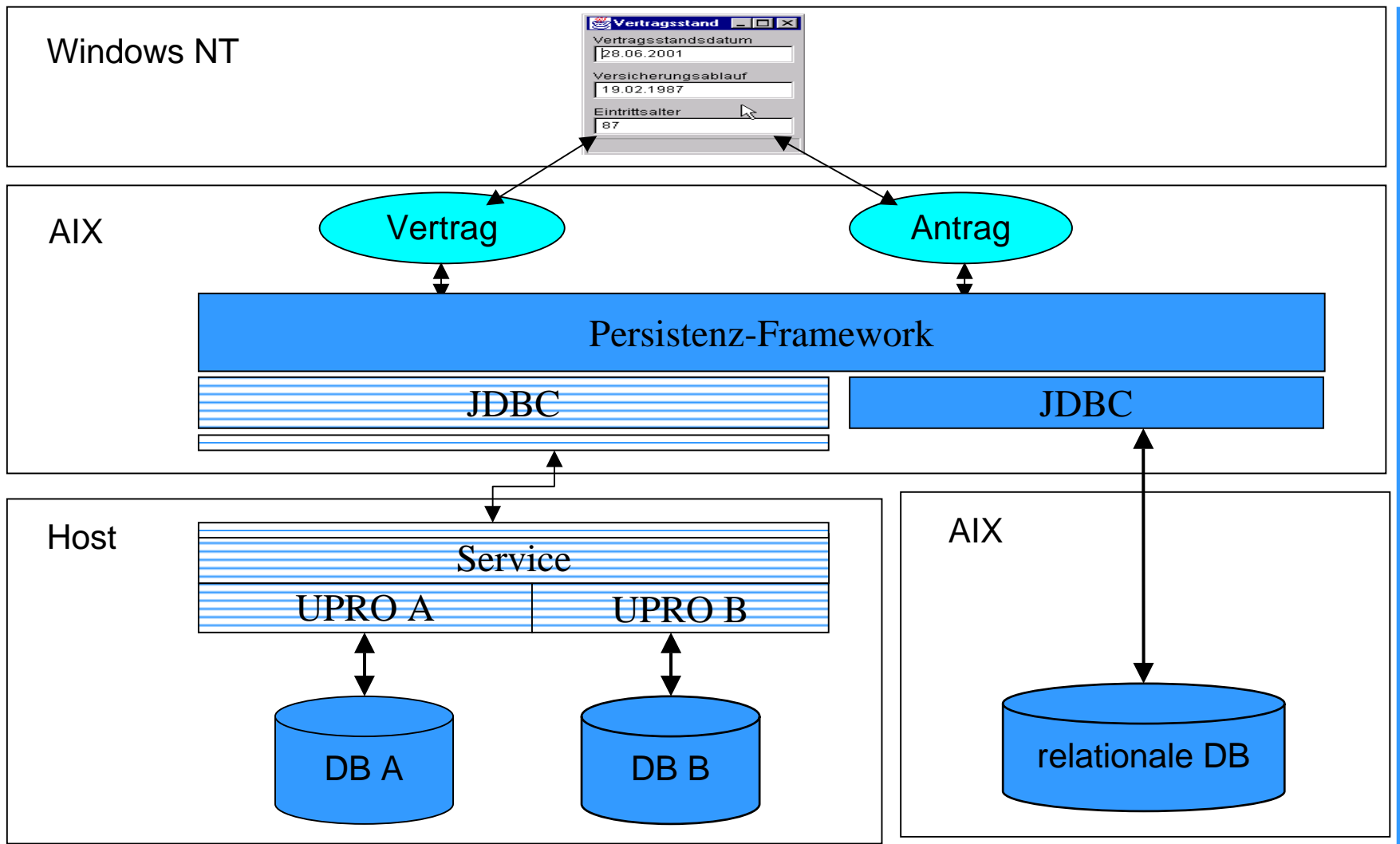
- Transaktionen ☹
 - Host-Transaktionsmonitor
 - Datenbank-Transaktionen
 - Ziel: OTS oder EJB TS mit JDBC-Treibern oder XA-Protokoll mit 2-phase commit
- Hostanbindung ☹
 - selbsterstellter JDBC Treiber ohne 2-phase commit mit Kommunikation über CPI-C
 - Ziel: Integration über TX-Coordinator
- Workflow ☹
 - PGO's
 - Ziel: Produkt integrieren (z.B. Verve Inc.)

... und andere noch einer näheren Untersuchung bedürfen.

- Internet 
 - Servlets mit HTML
- Security 
 - Verschlüsselung (secure sockets)
 - Berechtigungen (AGO's)
- Systems Management 
 - Logging/Tracing
 - Workload Balancing
 - Softwareverteilung

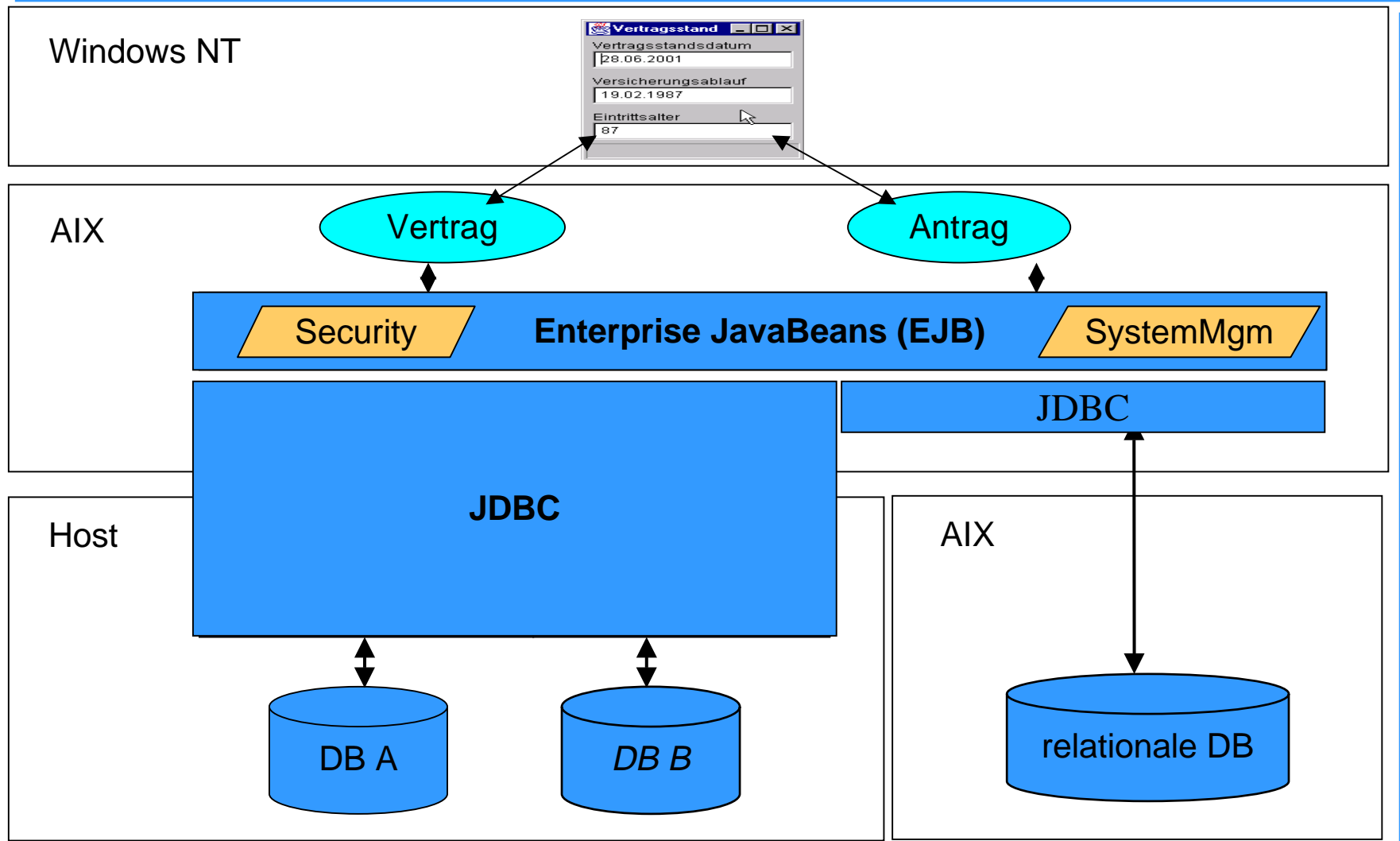
Das Hostsystem wurde erweiterbar integriert, ...

Projektergebnisse










... die Lösung wird ggf. technologisch angepaßt.

Projektergebnisse



Das vorhandene Vorgehensmodell wurde erweitert und einer objektorientierten Vorgehensweise angepaßt.

- iteratives, inkrementelles Vorgehen 
- Softwareentwicklung 
 - Analyse, Design
Geschäftsprozesse, Use Cases, GUI's, Objektdiagramme (statisch, dynamisch)
 - Implementierung 
 - Test (JavaStar, OptimizeIt)
Qualitätssicherung 
- Projektmanagement 
- Konfigurationsmanagement 
- Integration in Innovator 

Die Entwicklungsumgebung ist stabil und wird von den Entwicklern, trotz moderater Performance-Probleme, insgesamt positiv beurteilt.

- Case-Tool Innovator 6.1, MID ☺
- Visual Age for Java, Enterprise Ed. 2.0, IBM ☺
- Generator, Barmenia Versicherungen ☺
- VisiBroker 3.2, Inprise ☺
- Informix DB mit JDBC-Treiber, Intersolv ☹
- Visual Age Persistenz, IBM ☹
- Java Blend, Sun ☹
- OptimizeIt 2.0, Intuitive Inc. ☺

Wir planen eine Absicherung, Vertiefung und Weiterentwicklung der technologischen Basis und ein Projekt zur Entwicklung einer Fachanwendung.

- Technologie
 - Enterprise Java Beans
 - Application Server
 - Persistenz
 - verteilte Transaktionen
- Anwendungs-Projekt
 - Vertragsbearbeitung Leben
 - VAA/IAA als Facharchitektur
 - 24 Geschäftsprozesse

Vielen Dank für Ihre Aufmerksamkeit !!!

Beifall !!

Anmerkungen !?

Kritik ??