

Überblick



Agenda

- Umfeld/Rahmenbedingungen
- Systemmodellierung mit UML
- Projektdatenbank
- Automatisierung
- Tooling
- Lessons learned
 - ◆ Probleme/Hinweis
 - ◆ Positive Erfahrung

Motivation MDA



Rahmenbedingungen

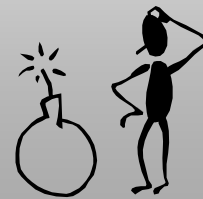
- Entwicklung sehr komplexer Systeme
 - ◆ Entwicklung/Integration unterschiedlicher Anwendungssysteme, HW, Betriebssysteme
 - ◆ Hohe Parallelität und Realzeitanforderungen
- Berücksichtigung beim Vorgehen
 - ◆ Teile des System als Unterauftrag
 - ◆ Verschiedene Entwicklungsstandorte
 - ◆ Entwicklungsstandards (V-Modell)
- Lange Einsatzdauer
- Aufsetzen auf (Alt-)Systemen

Motivation MDA



Was ist oft verbesserungswürdig?

- Nur unpräzise Systemspezifikation und Softwarevorgaben
- Auswirkungen von Änderungen unklar
- Dokumentengetriebene Entwicklung
- Komplexität schwer handhabbar
- "local heroes" halten Wissen
- Unklares Test/Nachweiskonzept
- Fehler treten erst bei Integration auf



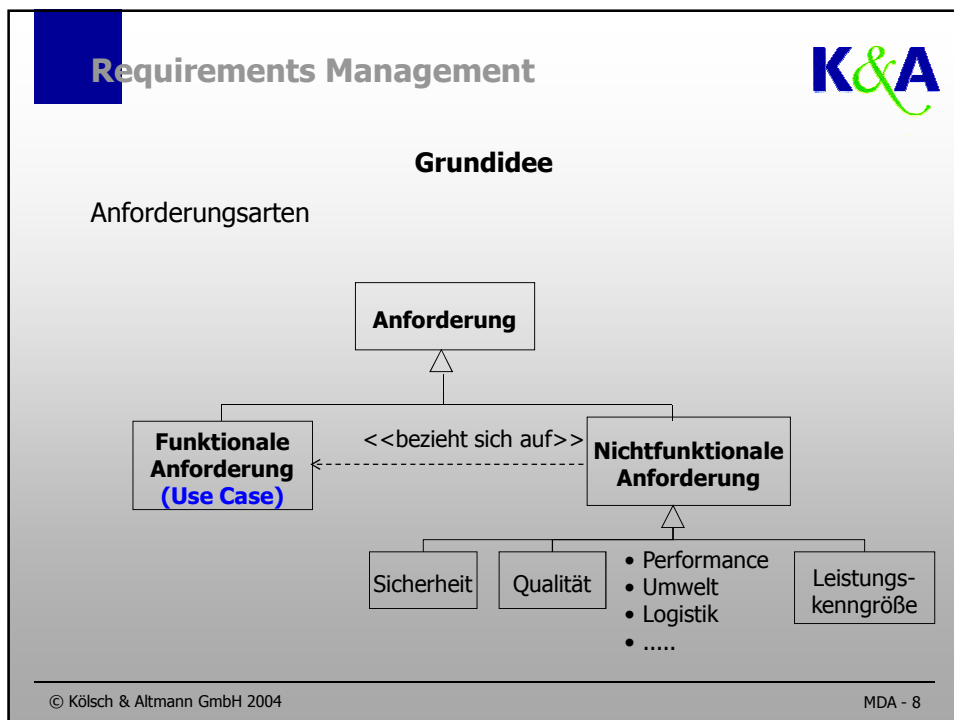
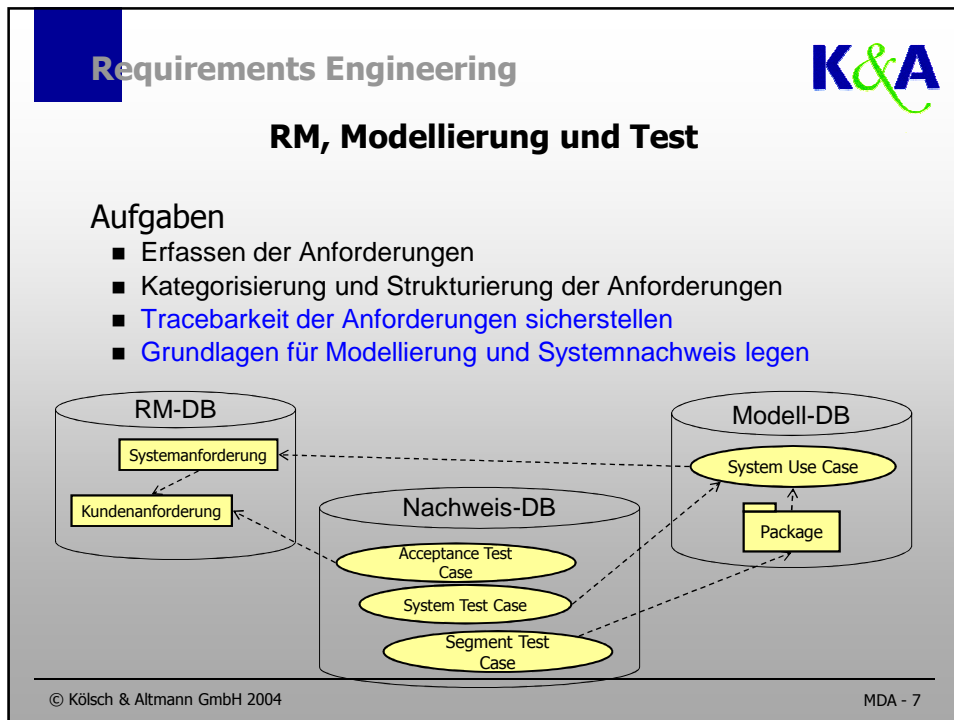
Motivation MDA




Lösungsansatz

- RE/RM
- Inkrementelles Vorgehensweise (Ausbaustufen)
- Systemmodellierung mit UML
 - ◆ Verschiedene Abstraktionsstufen des Systems
- Aufbau einer Projektdatenbank
- Toolunterstützung
- Automatische Dokumentengenerierung

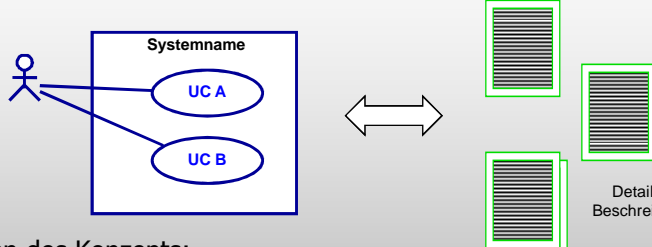




Requirements Management

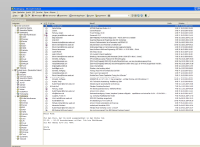


Konzept der Use Cases



Nutzen des Konzepts:

- Zielgruppe (Anwender) **beteiligen**
- Beschreibung aus Sicht der Anwender **durchführen**
- Anforderungen **strukturieren**
- Systemgrenzen **festlegen**
- Modell der Anforderungen **darstellen**




GUI Prototyp

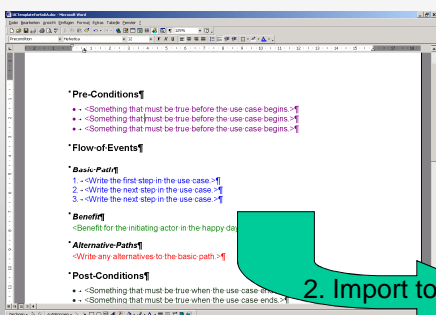
© Kölsch & Altmann GmbH 2004

MDA - 9


Konzept der Use Cases

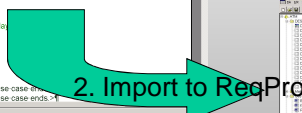


Use Case Beschreibung - Vorgehen



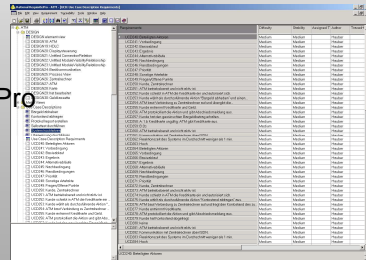
3. Change Management
unter ReqPro
(History, Discussions)





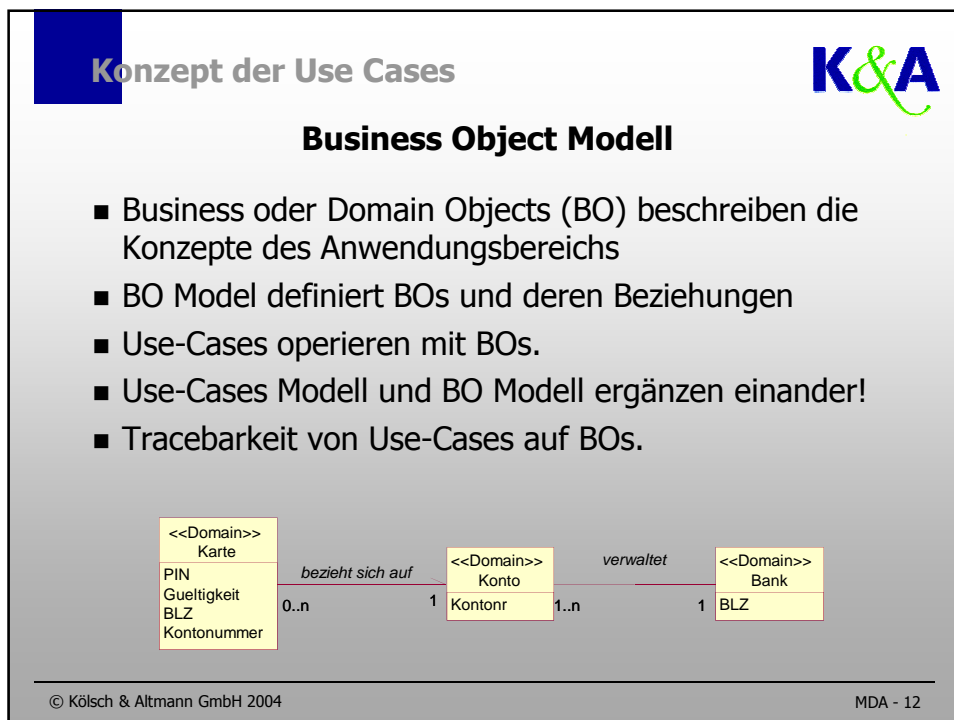
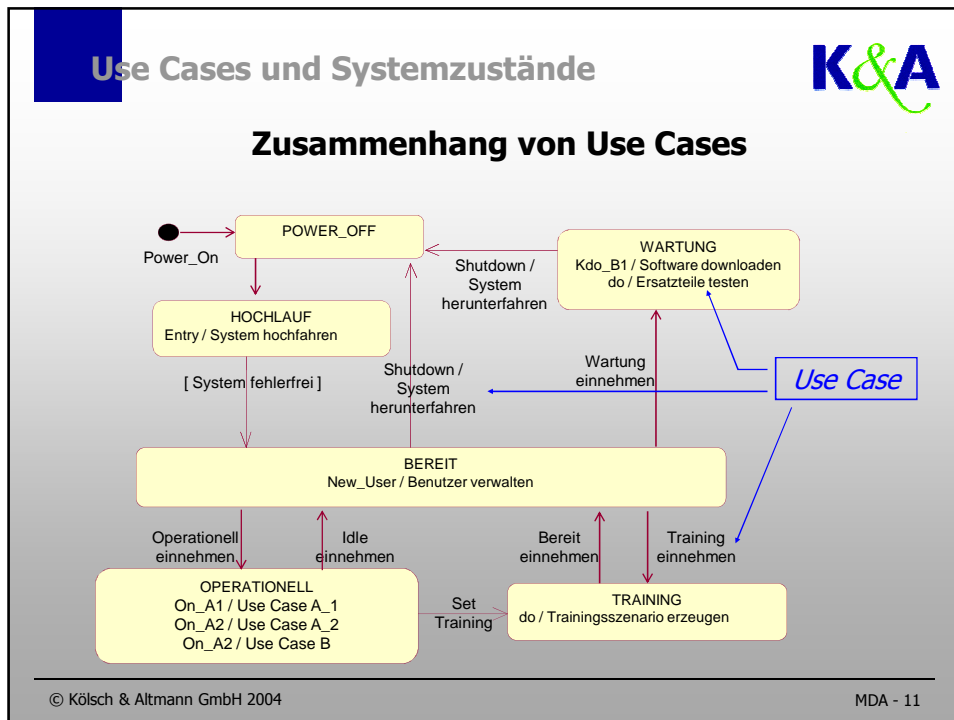
2. Import to ReqPro

1. Beschreibung mittels Word-Template



© Kölsch & Altmann GmbH 2004

MDA - 10



Konzept der Use Cases **K&A**

Einsatz des Use Case-Modells

Use Case-Modell dient zusätzlich als Grundlage für die

- Identifikation von Objekten
- Beschreibung des dynamischen Verhaltens
- Manuale, Dokumentation
- Qualitätsprüfung (Testfälle)
- Projektmanagement

© Kölsch & Altmann GmbH 2004 MDA - 13

Konzept der Use Cases **K&A**


Nichtfunktionale Anforderungen

Tracebarkeit der nichtfunktionalen Anforderungen bis auf Code-Ebene dank UML

Id	UREQ
UREQ 103	Geldausgabe des ATM innerhalb max. 10 Sekunden
UREQ 231	Die PIN ist im SPY.X Verfahren verschlüsselt
UREQ 250	ATM muß bis -10 C störungsfrei funktionieren


© Kölsch & Altmann GmbH 2004 MDA - 14

Motivation MDA




Lösungsansatz

- RE/RM
- **Inkrementelle Vorgehensweise (Ausbaustufen)**
- Systemmodellierung mit UML
- Aufbau einer Projektdatenbank
- Toolunterstützung
- Automatische Dokumentengenerierung

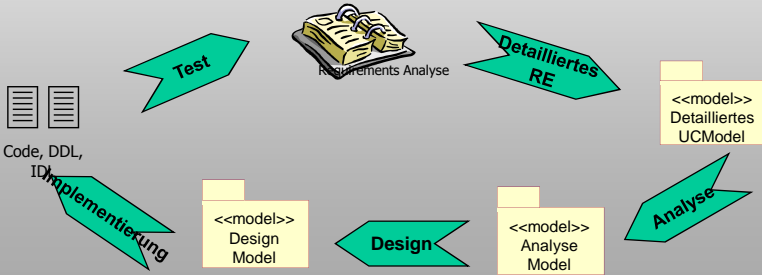


© Kölsch & Altmann GmbH 2004 MDA - 15

Iterativ, inkrementell Systementwicklung

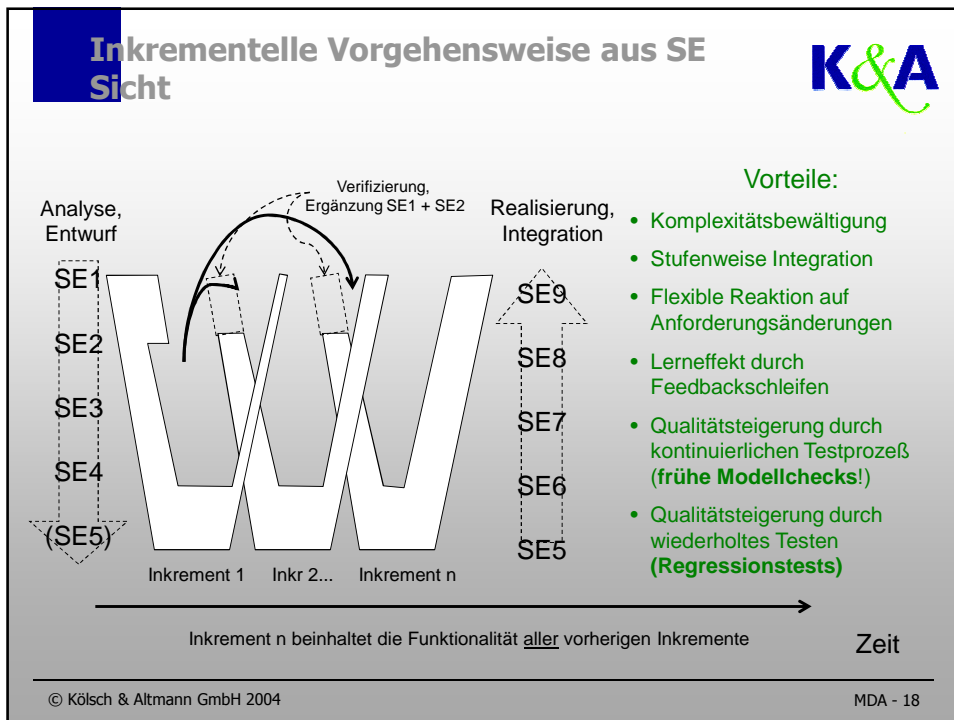
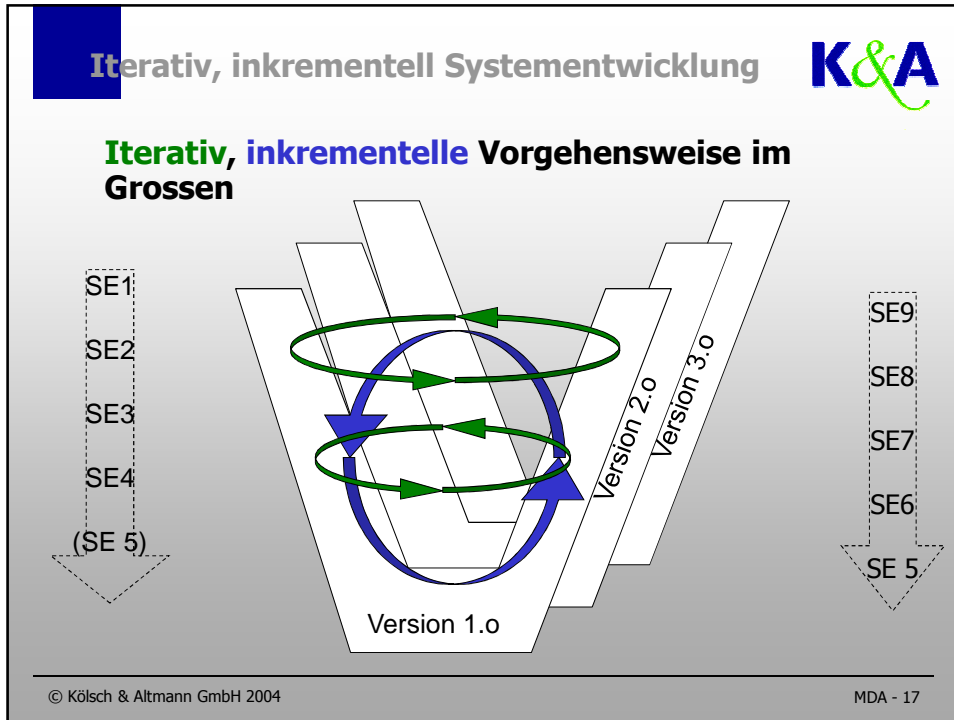


- Mehrere lauffähige Ausbaustufen
- Erweiterungen werden schrittweise durchgeführt
- Laufende Erweiterung der Toollandschaft möglich
- Risiko-minimierend
- Schnelles Feedback
- Erfahrungsgewinn



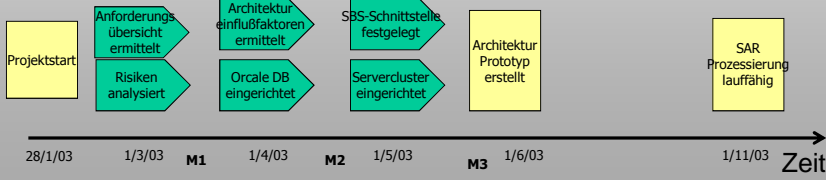
The diagram illustrates an iterative development cycle. It starts with a 'Requirements Analyse' (represented by a book icon) leading to a 'Detailliertes RE' (Requirements Engineering) model. This model then undergoes 'Analyse' to produce an '<<model>> Analyse Model'. This model is then used for 'Design' to create a '<<model>> Design Model'. The design model is implemented ('Implementierung'), resulting in 'Code, DDL, ID'. Finally, the code is tested ('Test') to produce 'Code, DDL, ID' again, which feeds back into the 'Requirements Analyse' stage.

© Kölsch & Altmann GmbH 2004 MDA - 16



Inkrementelles Vorgehen aus PM Sicht

- aktives Risikomanagement
- kontrollierbarere Planungszeiträume
- sichtbarer Entwicklungsfortschritt (**Zwischenergebnisse**)
- Soviel Planung wie nötig, sowenig wie möglich
 - ◆ Roadmap
 - ◆ Feinplanung für 4-8 Wochen (**z.B. auf Use Case Basis**)
- definierte Ziele (**basierend auf Modellinformationen**)




© Kölsch & Altmann GmbH 2004 MDA - 19

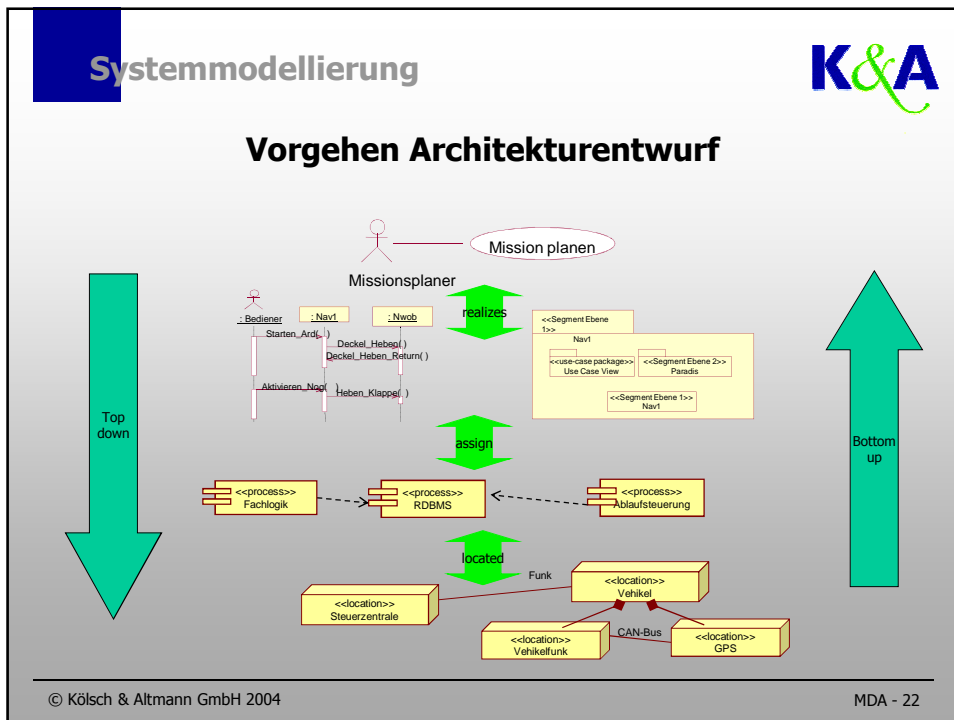
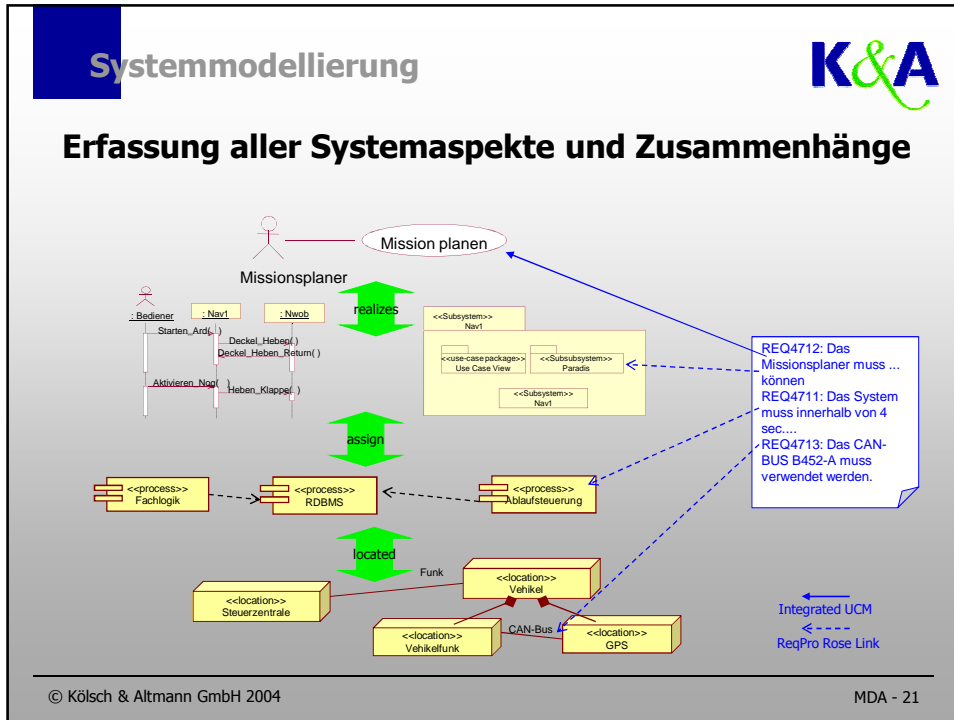
Motivation MDA

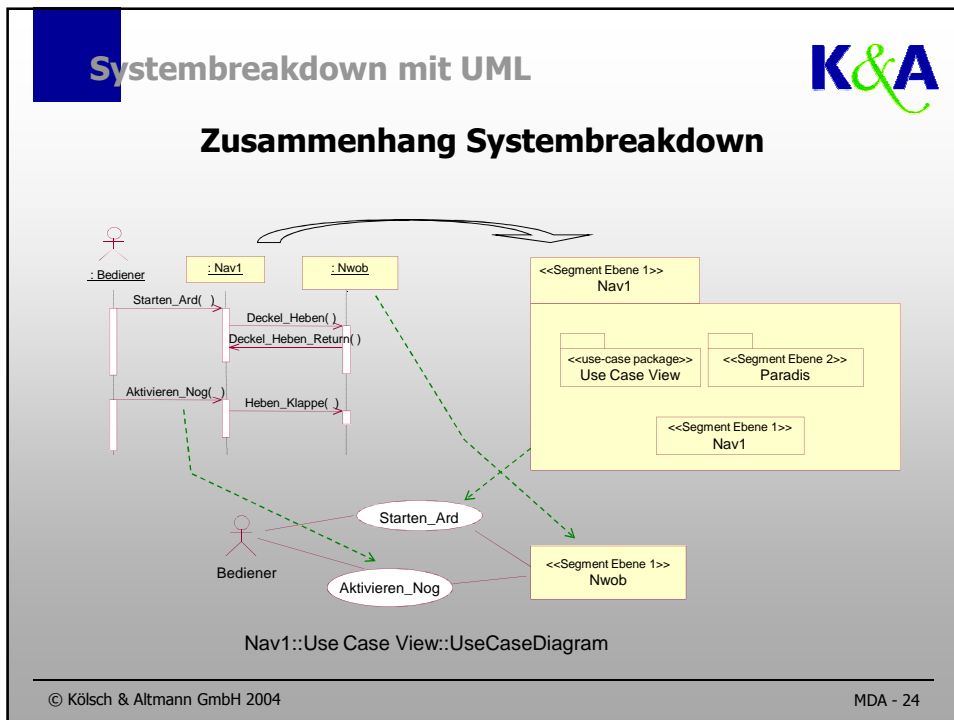
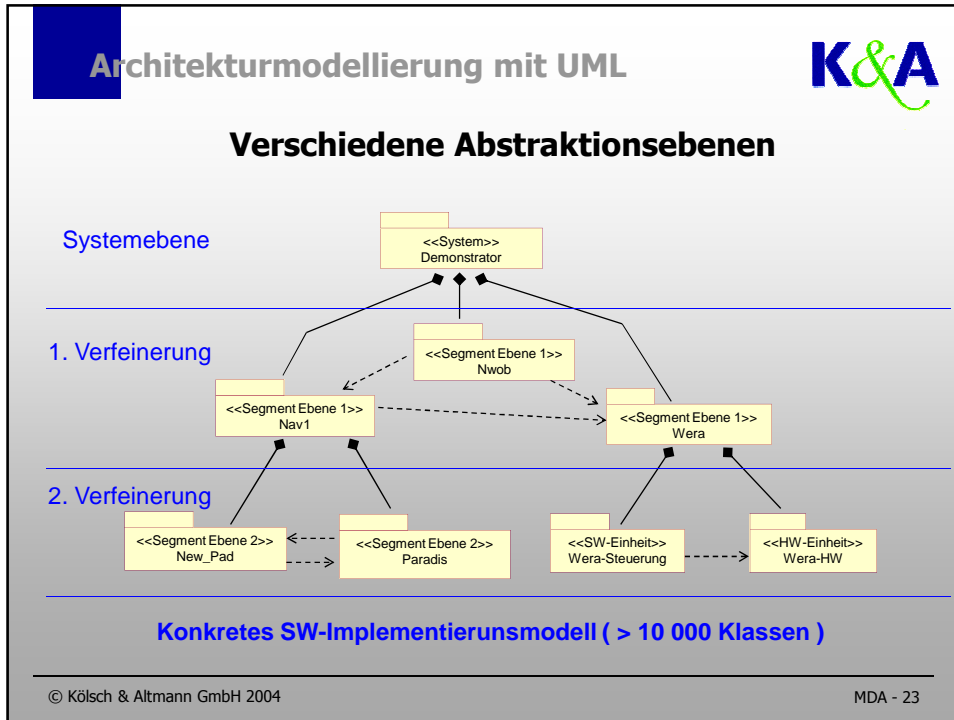
Lösungsansatz


- RE/RM
- Iterativ, inkrementelle Vorgehensweise (Ausbaustufen)
- **Systemmodellierung**
- Aufbau einer Projektdatenbank
- Toolunterstützung
- Automatische Dokumentengenerierung



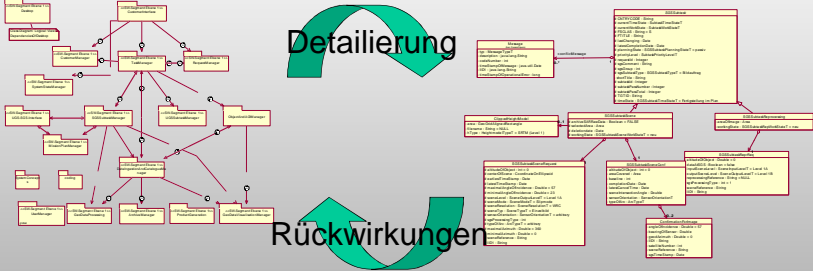
© Kölsch & Altmann GmbH 2004 MDA - 20






Systemmodellierung 

Durchgängige Systemmodellierung



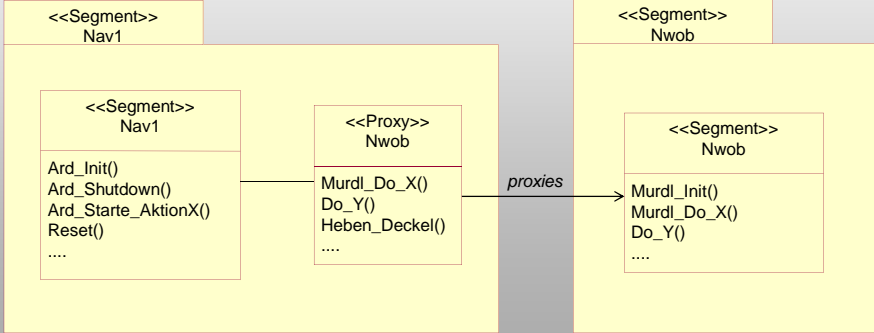
■ Frühe Erkennung von konzeptuellen Fehlern
➤ Klare Richtlinien notwendig!

© Kölsch & Altmann GmbH 2004 MDA - 25


Systemmodellierung 

Schnittstellenprüfung

Proxy-Elemente




© Kölsch & Altmann GmbH 2004 MDA - 26


Motivation MDA 

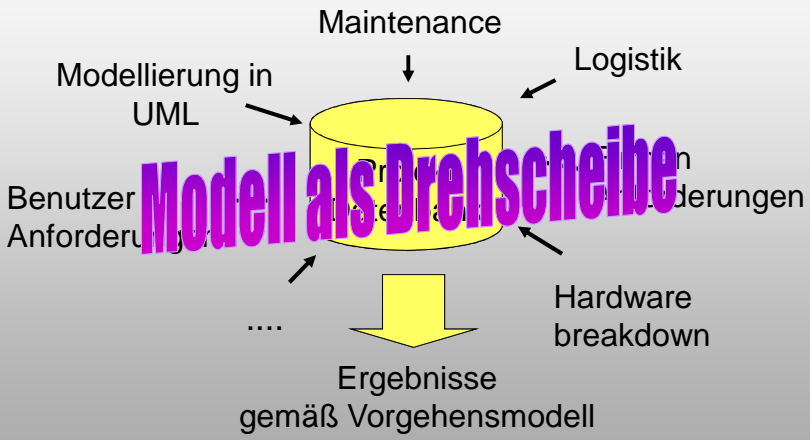
Lösungsansatz

- RE/RM
- Iterativ, inkrementelle Vorgehensweise (Ausbaustufen)
- Systemmodellierung mit UML
- **Aufbau einer Projektdatenbank**
- Toolunterstützung
- Automatische Dokumentengenerierung



© Kölsch & Altmann GmbH 2004 MDA - 27

Lösungsansatz Projektdatenbank 



The diagram illustrates a central yellow cylinder representing a 'Modell als Drehscheibe' (Model as a Hub). Arrows point towards this hub from 'Modellierung in UML', 'Benutzer Anforderungen', and 'Logistik'. An arrow points away from the hub towards 'Hardware breakdown'. A large yellow arrow points downwards from the hub to the text 'Ergebnisse gemäß Vorgehensmodell'. The text 'Maintenance' is positioned above the hub, and 'Ergebnisse gemäß Vorgehensmodell' is below it. The central text 'Modell als Drehscheibe' is written in large, pink, stylized letters across the hub.

© Kölsch & Altmann GmbH 2004 MDA - 28

Lösungsansatz Projektdatenbank

K&A

Nutzung des Systemmodells/ Automatisierungsmöglichkeiten

- Konsistenzprüfungen
- Tracing
- Dokumentationsgenerierung
- Testfallgenerierung
- CORBA-Unterstützung
- Generierung von Datenzugriffsschicht
- Roundtrip-Engineering
- Metriken

© Kölsch & Altmann GmbH 2004 MDA - 29


Konsistenzprüfungen

K&A

Konsistenzprüfungen

- syntaktisch Prüfungen
 - ◆ Korrekte UML Modellierung?
- semantische Prüfungen
 - ◆ Einhaltung von Konventionen
 - ◆ Einhaltung von Modellierungsrichtlinien
 - ◆ Schnittstelleneinhaltung
 - ◆ Tool-übergreifende Konsistenz
- automatisiert

© Kölsch & Altmann GmbH 2004 MDA - 30


Toolunterstützung 

Aktueller Stand

- ✓ Vernünftige Integration aller (!) Werkzeuge
- ✓ Vernünftiges Arbeiten möglich
- ✓ Umfangreiche Automatisierungsmöglichkeiten


- HW Entwurf nicht gut unterstützt
- Round Trip Engineering nur bedingt
- Einarbeitungsaufwand in Gesamttoolpalette

© Kölsch & Altmann GmbH 2004 MDA - 31


Lessons Learned 

Probleme/Hinweise

- Disziplinierte Prozess-Anwendung nötig
 - ◆ Klare Richtlinien für Anwendung jeder Methode/Tools
- Regelmäßige modellbasierte Reviews
 - ◆ Wöchentliche Fortschrittskontrolle
- Modellpflege auf allen Ebenen
- Lernaufwand
 - ◆ Mentoring durch Poweruser
- Abhängigkeit von Verfügbarkeit der IT-Infrastruktur
- Konzeptentwicklung jeweils 1 Phase vor Umsetzung




© Kölsch & Altmann GmbH 2004 MDA - 32


Lessons Learned 

Positive Erfahrung

- Hoher Wiederverwendungsgrad auf allen Ebenen
 - ◆ Richtlinien, Architekturkonzepte, Anforderungsvarianten, Patterns, Subsysteme, Design
- Mannigfache Modellnutzung für Sekundäraktivitäten
 - ◆ QS/Nachweis, Informationsquelle/PM, Hilfesysteme
- Hoher Anteil von generiertem Code
- Schnelle Einarbeitung auf Modellebene
- Relative problemlose Modellaktualisierung auf Codeebene
- Aktuelle Dokumentation



© Kölsch & Altmann GmbH 2004 MDA - 33

Lessons Learned 

Referenzprojekte

- SAR-Lupe
- EFA 2000 – MDB
- Sostar – X
- GAST
- Roland LS
- Polyphem
- Trifom
- Zeus

© Kölsch & Altmann GmbH 2004 MDA - 34