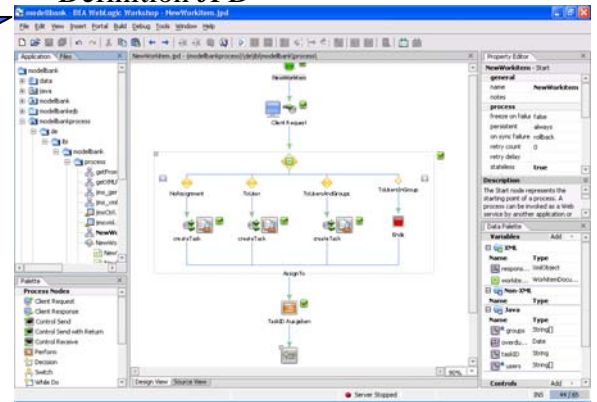
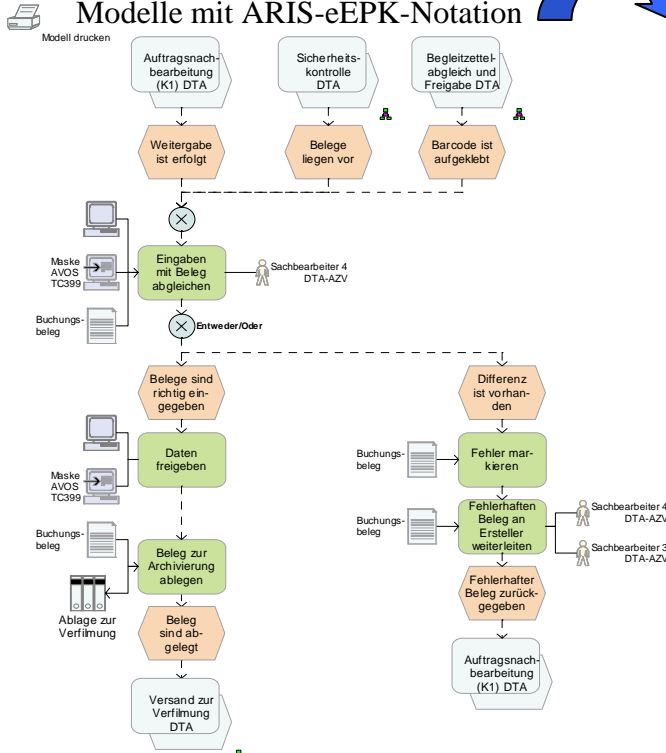


Erfahrungen mit der Model Driven Architecture (MDA) aus einem Bank-Labor: “Von ARIS über UML nach JPD in EPP und alles automatisch...”

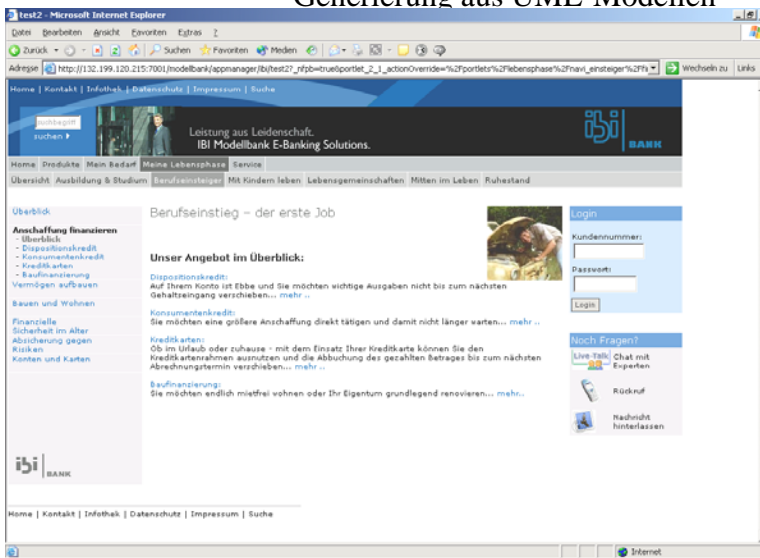
Dozenten: Rainer v. Ammon, Elke Jung, Stefan Tilkov
 Version 02/06/2006

UML-Activity Diagramme und Umwandlung in Java Process Definition JPD

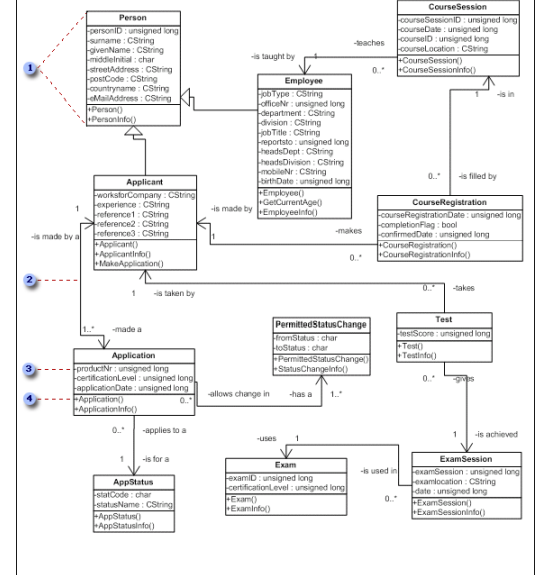
Vorliegende Geschäftsprozess- Modelle mit ARIS-eEPK-Notation



Automatische Applikations- Generierung aus UML-Modellen



UML Statische Klassendiagramme



Fachhochschule Oberösterreich, Standort Hagenberg

und

ibi academy an der Universität Regensburg

Inhalt

Im Mittelpunkt der Lehrveranstaltung steht die Idee der Model Driven Architecture (MDA), ausgehend von Geschäftsprozessen automatisch aus UML-Modellen Bankapplikationen zu generieren, die dabei im Sinne einer Service Orientierten Architektur (SOA) implementiert und „deployed“ werden. Dieses Konzept soll kritisch untersucht und beurteilt werden. Beispielsweise werden folgende Fragen behandelt:

- Ist die Idee der MDA wirklich im komplexen Umfeld einer (Groß-)Bank anwendbar?
- Wo ist die MDA- und SOA-Idee im gesamten Architektur-Management einer Bank einzuordnen und welcher Umfang der Gesamtproblematik wird davon maximal abgedeckt?
- Können Bank-Applikationen wirklich aus UML-Modellen generiert werden?
- Was wäre der Return Of Invest (ROI) für die Bank und wie lässt sich dieser bewerten?

Im Rahmen der Lehrveranstaltung werden folgende Komplexe bearbeitet:

- Typische reale Geschäftsprozesse einer Bank (am Beispiel HVB) werden vorgestellt und liegen als ARIS-Modelle (als eEPK) bei der Bank (am Beispiel der HVB) vor.
- Die „proprietären“ ARIS-Modelle werden in die Modelle des UML-Standards automatisch transformiert.
- Aus den UML-Modellen werden im Sinne der Model Driven Architecture (MDA) in einem wiederholbaren und modifizierbaren Prozess automatisch die Applikationen als Enterprise Portal generiert.
- Dieser zyklische, iterative, evolutionäre Prozess (Object Engineering Prozess OEP2.0) kann im Sinne eines Rapid Application Development (RAD) im direkten Dialog mit der Fachabteilung bis zur fertigen Applikation durchlaufen werden.
- Parallel zum MDA-Ansatz wird die Applikation mit einer Enterprise Portal Plattform (BEA Weblogic Enterprise Plattform) und einer mächtigen, aber teilweise nicht standardisierten, proprietären Entwicklungsumgebung (Weblogic Workshop) entwickelt. Daraus soll eine Einschätzung ermöglicht werden, welche Vor- oder Nachteile ein MDA-Ansatz bewirkt und auf welchen Ebenen des gesamten Architektur-Managements diese entstehen.

Zum Veranstaltungskonzept:

Die vorliegende Lehrveranstaltung ist konzipiert:

- als Kurs für Fachhochschule Oberösterreich, Standort Hagenberg
- als Kurs für Erasmus Incoming Students
- als zu exportierender Kurs, z.B. für Erasmus Teachers' Mobility Program
- als einwöchiger Industrie-Workshop der IBI Academy an der Universität Regensburg (Erasmus-Kooperation zwischen FH OÖ und Uni Regensburg)

Abstracts

Section 1:Enterprise Architecture, Business Architecture Models and Frameworks, Process- and IT-centric approach, Enterprise Architecture Tool Market – Reference models in finance sector

A successful architecture management requires an Enterprise Architecture approach which supports either the business strategy as well as the generic IT-requirements. That means the whole cycle from the product idea to implementation of business applications inclusive replacement of old applications is an essential part of the Enterprise Architecture. Enterprise Architecture Frameworks like Zachmann, Spewak, FEAF, TEAF, OMG (MDA) etc. describe business centric and IT centric aspects in a different manner. The challenge is to combine both architecture approaches to meet the requirements of business managers, process owner, project managers on a level with software engineers, data base system administrators etc. To ensure a sustainable description of Enterprise Architectures different tools like ASG, Casewise, Computas, IDS Scheer, Mega International, Popkin, Proforma, Ptech are more or less convenient. Based on reference models of banking sector different abstraction levels (process categories, value chains, eEPK) and content of architectures (architecture views) can be described. The extensive representation of typical "architecture views" for financial enterprises – "Business Model View", "Process-View", "IT-View", "Output View" and "Informations View" gives a deep insight into the architecture structures and standards of the reference model of HVB Group.

Enterprise Architecture, Business Architecture Models und Frameworks, Process- und IT-centric Approach, Enterprise Architecture Tool Market – Referenzmodelle im Finanzdienstleistungssektor

Ein erfolgreiches Architekturmanagement setzt den Einsatz von Enterprise-Architektur-Ansätzen voraus, die sowohl den Erfordernissen der Geschäftsstrategie eines Unternehmens entsprechen als auch die Gesamtheit der IT-Anforderungen von der Neuentwicklung von Anwendungen bis hin zur Ablösung von Altsystemen unterstützen. Für die Beschreibung von Enterprise Architekturen gibt es unterschiedliche Architektur-Frameworks wie Zachmann, Spewak, FEAF, TEAF, OMG (MDA) etc., die die einzelnen Business-zentrierten- und IT-zentrierten Aspekte der Architektur in einem unterschiedlichen Maß berücksichtigen.

Die Herausforderung besteht darin, diese Architekturansätze zu verbinden, um somit den Anforderungen aus den unterschiedlichen Bankbereichen, angefangen von Business Managern, Prozessverantwortlichen bis hin zu Projektleitern, Softwareentwicklern, Datenbankadministratoren etc. gerecht zu werden. Für eine kontinuierliche Fortschreibung von Enterprise Architekturen eignen sich unterschiedliche Tools wie ASG, Casewise, Computas, IDS Scheer, Mega International, Popkin, Proforma, Ptech.

Anhand von Referenzmodellen im Finanzdienstleistungssektor lassen sich die Betrachtungsebenen (Prozesskategorien, Wertschöpfungsketten, eEPK) und deren Inhalte ("Architektur-Views") des Enterprise Architekturmodells beschreiben. Eine umfassende Darstellung von typischen "Architektur-Views" im Bankensektor wie "Business Model View", "Process-View", "IT-View", "Output View" und "Informations View" liefern einen tiefen Einblick in Architekturstrukturen und Standards am Beispiel des Architektur-Referenzmodells der HVB Group.

Section 2: Model Driven Development of Service-oriented Systems Based on Business Process Definitions and UML Models

Model-driven development (MDD) of enterprise systems is becoming one of the most debated topics in the industry, with proponents and opponents equally convinced of their respective position. Our goal is to show both advantages as well as disadvantages and possible pitfalls when applying MDD to building modern, service-oriented software systems.

After an introduction of the basic concepts of model-driven software development in general and the OMG's MDA in particular, including standards such as UML, XMI, and MOF, we will contrast them with the CASE tool approaches of the 1980s and alternatives such as domain-specific languages and generic programming. We will then set up a typical tool chain composed of ARIS as the tool used for BPM, a UML tool for model refinement, a commercial software generator (innoQ's iQgen), and an integrated development environment, and demonstrate the typical process of model-driven development. To illustrate its benefits, we will first introduce changes to the business model and show how they can be applied without having to be concerned with the technical implementation details; secondly, we will change the underlying technology mapping without affecting the business logic. We will conclude with a look at different MDA approaches (elaborational vs. translational), the meta-data concept, and its role in service-oriented architectures.

Modellgetriebene Service-orientierte Systeme auf Basis von Geschäftsprozessdefinitionen und UML-Modellen

Die Modellgetriebene Software-Entwicklung (Model-Driven Development, MDD) von Unternehmensanwendungen entwickelt sich zu einem der am leidenschaftlichsten diskutierten Themen in der IT-Branche – mit Befürwortern und Gegnern, die von ihrer jeweiligen Argumentation gleichermaßen überzeugt sind. Ziel dieses Teils ist es, die Vor- und Nachteile beim Einsatz von MDD zur Entwicklung moderner, Service-orientierter Systeme aufzuzeigen sowie auf potenzielle Fallstricke hinzuweisen.

Nach einer Einführung in die Grundkonzepte modellgetriebener Software-Entwicklung im Allgemeinen und in die von der OMG definierte Model Driven Architecture (MDA) mit den Standards UML, XMI und MOF im Besonderen, werden diese den CASE-Tool-Ansätzen aus den 80er-Jahren und Alternativen wie domänenspezifischen Sprachen gegenübergestellt. Im Anschluss wird eine Werkzeug-Kette aufgesetzt, bestehend aus ARIS für die Geschäftsprozessmodellierung, einem UML-Werkzeug für die Modellverfeinerung, einem kommerziellen Software-Generator (innoQ iQgen) und einer integrierten Entwicklungsumgebung. Anhand dieser Kette wird der typische Prozess modellgetriebener Software-Entwicklung demonstriert. Um deren Vorteile zu illustrieren, werden zunächst Änderungen am Geschäftsmodell durchgeführt und gezeigt, wie diese ohne Beschäftigung mit den Implementierungsdetails angewandt werden können. Danach wird demonstriert, wie Änderungen an der Technologieabbildung ohne Auswirkung auf das fachliche Modell durchgeführt werden können. Abschließend werden unterschiedliche MDA-Ansätze (elaborational vs. translational) sowie das Konzept von Metadaten an sich und deren Rolle Service-orientierten Architekturen betrachtet.

Section 3: Realisation of Service-Oriented Architecture (SOA) using Enterprise Portal Platforms taking a typical Example of Banking Domains

As opposed to the MDA approach a typical bank application will be developed, based on an Enterprise Portal Platform and based on a powerful, but not standardized, proprietary IDE. This implementation shall allow estimating the (dis-) advantages of the MDA approach and also which tiers of the whole architecture management these (dis-) advantages will apply.

The most relevant technologies of an Enterprise Portal Platform are introduced and the present state of the art is shown. In terms of a tier model a reliable, scalable transaction platform is the base for the eBusiness. Depending on the programming model there are different middleware platforms like Web Application Servers for J2EE, TP-monitors like CICS and Tuxedo for C and Cobol, Object Transaction Monitors for C++ and Corba as well as platforms for integrating all these worlds. Based on these transaction infrastructures prebuilt EJB components allow rapid development of eBusiness applications. A Personalisation Server enables creating dynamic Web applications based on user profiles, a rules engine and a content management system. According to the standardized specification of the J2EE Connector Architecture a componentware market of plug&play adapters for Enterprise Application Integration came up the last years. Furthermore by a Workflow Management System business processes can be graphically modeled and implemented without hard coding. Also the processes of different business partners can be modeled and integrated to supply chains without any coding. Finally virtual market places can be easily built by defining collaboration hubs and downloadable enablers. Enterprise portals for Business to Consumers, Business to Employees and Business to Business support a single point of access on top of all these technics. All together is today called the eBusiness infrastructure. So called Integrated Service Environments (ISE) are provided for developing such complex enterprise portals respectively enterprise applications. A hands on workshop gives an impression of the developing process based on an ISE.

Realisierung von Service-Orientierten Architekturen (SOA) auf Basis von Enterprise Portal Plattformen am Beispiel einer typischen Bank-Applikation

In diesem Teil wird parallel zum MDA-Ansatz eine typische Bank-Applikation mit einer Enterprise Portal Plattform (am Beispiel BEA Weblogic Enterprise Plattform) und einer mächtigen, aber teilweise nicht standardisierten, proprietären Entwicklungsumgebung (am Beispiel BEA Weblogic Workshop) entwickelt. Daraus soll eine Einschätzung ermöglicht werden, welche Vor- oder Nachteile ein MDA-Ansatz bewirkt und auf welchen Ebenen des gesamten Architektur-Managements diese entstehen.

In diesem Zusammenhang werden die Technologien, die einer Enterprise Portal Plattform zugrunde liegen, eingeführt und der gegenwärtige Stand der Technik aufgezeigt. Im Sinne eines Schichtenmodells ist dabei eine ausfallsichere, skalierbare Transaktions-Plattform die Basis für die Anforderungen des eBusiness. Abhängig vom Programmiermodell gibt es verschiedene Middleware Plattformen wie Web Application Server für J2EE, TP-Monitore wie CICS und Tuxedo für C oder Cobol, Object Transaction Monitore für C++ und Corba sowie Plattformen für die Integration all dieser Welten. Darauf aufbauend sollen mit vorgefertigten Geschäftsprozessen und EJB Komponenten eBusiness-Anwendungen schnell entwickelt werden können. Ein Personalisation Server ermöglicht die Erzeugung dynamischer Web-Applikationen basierend auf

Benutzerprofilen, einer Regelmaschine für Personalisierungsregeln und einem Content Management System. Entsprechend dem Standard der J2EE Connector Architecture entstand ein Komponentenmarkt für vorgefertigte Adapter für Enterprise Application Integration. Außerdem können mittels integrierter Workflow Management Systemen Geschäftsprozesse graphisch modelliert und ohne „harte“ Codierung flexibel implementiert werden. Die Geschäftsprozesse zwischen Geschäftspartnern können z.B. als Supply Chains über Unternehmensgrenzen hinweg integriert werden sowie Virtuelle Marktplätze schnell und einfach eingerichtet werden. Oberhalb all dieser Techniken liefern Enterprise Portals für Business to Consumers-, Business to Employees- und Business to Business-Anwendungen einen Single Point of Access auf die Anwendungen und Informationen eines Unternehmens. Alles zusammen nennt man heute eine eBusiness Infrastruktur. Ein Hands On-Workshop vermittelt einen Eindruck von der Arbeitsweise und der Implementierung mit einer Entwicklungsumgebung, die derzeit als Integrated Service Environment (ISE) bezeichnet wird.

Your Course Instructors:

Dr. Elke Jung is IT Architect especially for business architecture in finance sector at the Bayerische Hypo-und Vereinsbank AG since 2002. Elke is responsible for conception and implementation of IT Architecture Models in the HVB Group. Before she worked as Architecture and Process Engineering Manager at the HVB Systems during the time of 2000 to 2002, she acted as Manager and System Owner for development and maintenance of investment banking application systems. Before it, in 1996 she was appeared as Customer Relations Manager and Senior Technical Support Engineer in the department of relational database system support at the Informix. Since 1988, after the study of electrical engineering and graduating in biomedical technology she was engaged in the research area and grappled with real time pattern recognition and statistic classification of biomedical signals. In addition Elke led European research projects dealing with conception, definition, standardization and prototyping of Smart Cards for diabetes care in the Research Centre GSF Munich.



Stefan Tilkov is managing director and co-founder of innoQ, a software consulting and services company with offices in Dusseldorf/Germany and Zurich/Switzerland. Stefan works with Fortune 500 companies to help them turn the theoretical advantages of modern approaches such as Model Driven Architecture (MDA) and Service-oriented architectures (SOA) into practical solutions. Before co-founding innoQ in 1999, he managed the application development team of MLC/PostCom, a subsidiary of the German Postal services. His experience includes architecting systems with a variety of technologies such as DCE RPC, CORBA, J2EE and Web services, using C, C++ and Java. Stefan maintains a personal weblog at <http://www.innoq.com/blog/st>.



Dr. Rainer v. Ammon is Professor for Software Engineering, especially for eBusiness infrastructures and distributed systems at the Upper Austria University of Applied Sciences since 2002. Also Rainer is research director at the institute for bank informatics, University of Regensburg in Germany. From 1998 to 2002 he worked as Principal Consultant and Manager for R+D Cooperations at BEA Systems (Central and Eastern Europe). Prior to this acting as Professor for Software Engineering in Dresden with focus on development of applications with event driven object oriented user interfaces and component based application development. Before this, he was acting as manager for Basic Systems at Mummert & Partners Consultancy/Hamburg. Since the beginning of the eighties he developed user interface management systems and CASE-tools in his own company. He graduated in Computer Linguistics from the University of Regensburg (Germany).



Agenda für Lehrveranstaltung FH Oberösterreich WS 2004/5:

Oktober 2004

“Einführung in Konzepte der SOA, Enterprise Portal Plattformen, Entwicklungsumgebungen/ISE, Plattform-Standards, Abgrenzung proprietärer Technologien“

Hands On Workshop „Implementierung Kreditberatungsprozess mit dem ISE BEA Weblogic Workshop“

Aufgabenstellung 1 „EPP-Entwicklung“ und Teameinteilung

November 2004

„Geschäftsprozesse und Referenzmodelle im Finanzdienstleistungs-Sektor“

Hands On Workshop „Modellierung von ausgewählten Geschäftsprozesse mit ARIS“

Aufgabenstellung 2 „GP-Modellierung“

Dezember 2004

„Modellgetriebene Entwicklung Service-orientierter Systeme auf Basis von Geschäftsprozessdefinitionen und UML-Modellen“

Hands On Workshop „MDA-Werkzeuge im praktischen Einsatz“

Aufgabenstellung 3 “GP-Transformation und Generierung”

Januar 2005

Präsentation und Diskussion der Projektergebnisse der Teilnehmer-Teams