

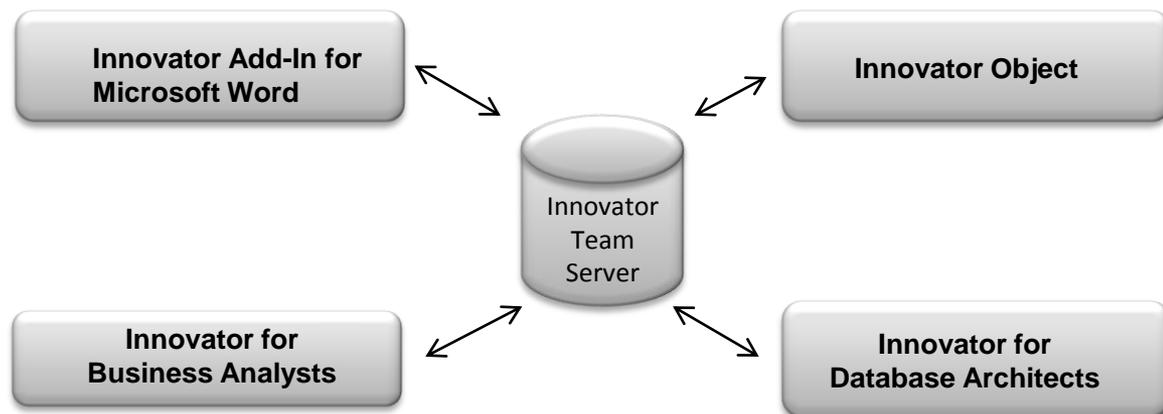


**Und es funktioniert doch! Erfahrungsbericht
über einen modellgetriebenen
Softwareentwicklungsprozess direkt aus der
Praxis einer großen Krankenversicherung**



Die MID GmbH

- **Beratungs- und Softwarehaus mit Hauptsitz in Nürnberg**
 - Weitere Standorte in Köln, Stuttgart und München
 - 110 Mitarbeiter
- **Modellierungsplattform Innovator**
 - IT-orientierte Geschäftsprozess-Analyse (BPM)
 - Anforderungs-Analyse für IT Projekte, modellbasiertes Testen
 - Business Intelligence (BI) Projekte auf Basis der Prozesse und Datenbanken
 - Modellgetriebene Software-Entwicklung
 - Planung und Konzeption von Service-Orientierten Architekturen (SOA)
 - Analyse für SAP R/3 Projekte



INHALT

1. **Probleme in der Softwareentwicklung
Lösung: Vorgehensmodell, Methodik**
2. **Modellieren der Fachlichkeit**
3. **Modelltransformationen**
4. **Modellieren der Technik**
5. **Codegenerierung**
6. **Fazit, Erfolgsfaktoren**

Was macht es so schwer die richtige Software zu erstellen?

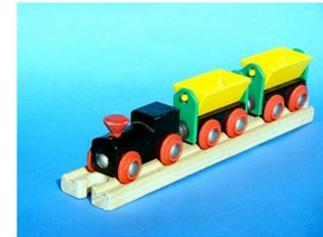
- Kommunikationsprobleme (vorausgesetzt es herrscht überhaupt Kommunikation)
 - FB und IT sprechen verschiedene Sprachen
 - Begriffsdefinitionen
 - FB sieht die Außensicht (Masken) und denkt vielleicht in Prozessen
 - Projektmanagement hat nochmals eine andere Sicht auf das Problem

- Prozess ist ständigen Änderungen unterworfen
 - Änderungen müssen nachvollziehbar sein
 - Dokumentation gibt Aufschluss was, wie und wo geändert wurde ... oder auch nicht?!
 - Fachlich
 - Pflichten- Lastenheft
 - Anforderungen
 - Technisch
 - Code
 - Inline- Doku
 - Manual, Totorial



Hilfe durch Modellierung

- Mit Hilfe von **Modellen** lassen sich komplexe Sachverhalte „einfach“ und Verständlich darstellen.
 - Ein Modell kann Aufschluss darüber geben, welche Funktionen bereits bestehen -> Wiederverwendung
- Es existieren verschiedene **Modellierungssprachen** um Sachverhalte darzustellen
 - UML
 - BPMN
 - SysML
 - ERM
 - *DSL*

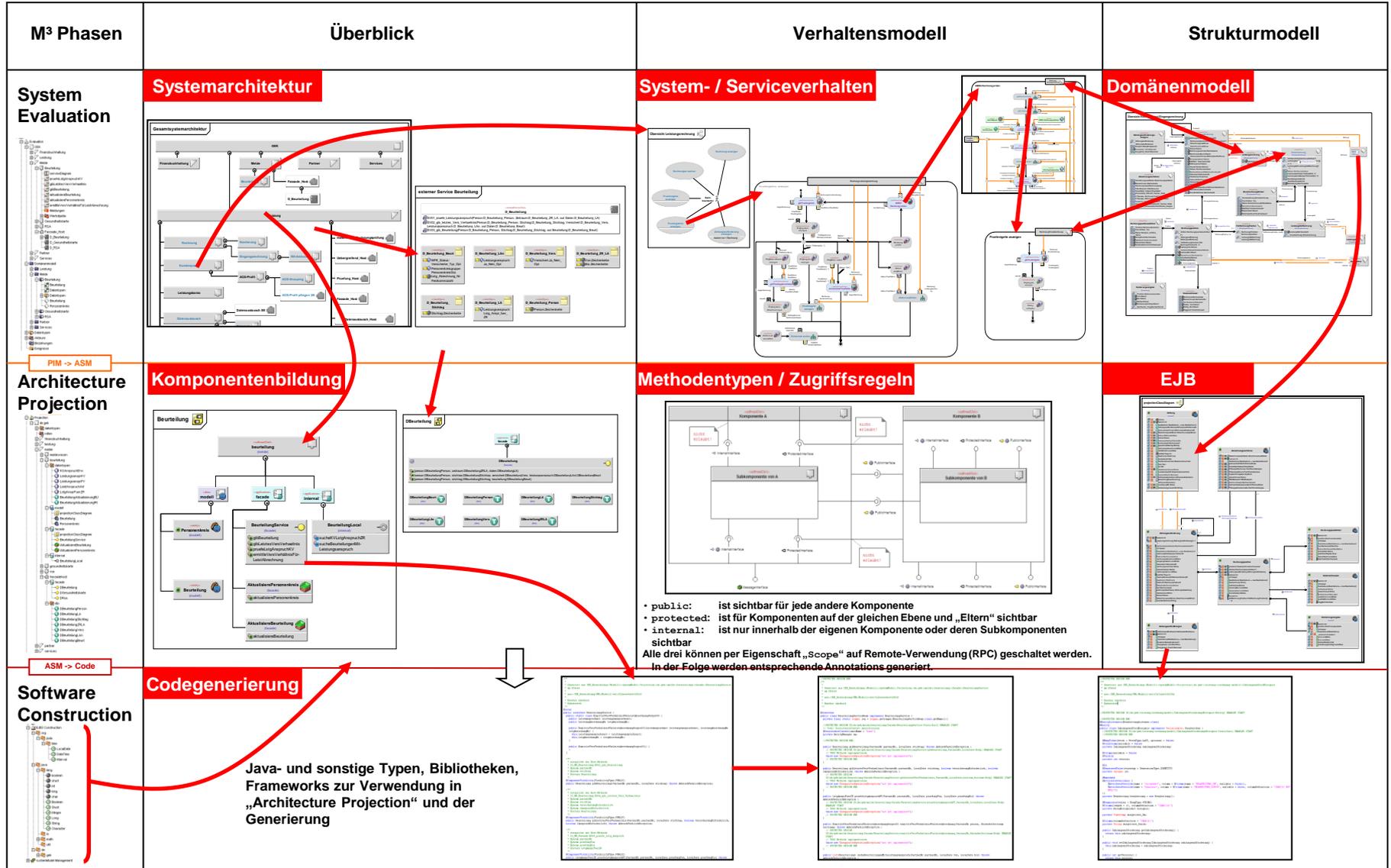


- Ein **Diagramm** gibt einen Überblick, in einer bestimmten Sicht auf relevante Modellausschnitte.
- Eine **Dokumentation**, die qualitätssicherbar ist entsteht während der Entwicklung

Setzen Sie Modellierung „sinnvoll“ ein. Sehen Sie die UML z.B. als Werkzeugkasten und nutzen Sie nur das was Mehrwert bringt.

Und es funktioniert doch!

Anwendung einer angepassten M³ bei der Barmer GEK





- Komplexe Infrastruktur
- Mitte der 90er: Servicegedanke wurde geprägt durch technische Services auf Großrechner (COBOL) und Windows Clients
 - Ein eigenes Kommunikationsverfahren auf RPC über TCP/IP wurde entwickelt
- Seit 2000 nutzt die GEK Innovator und die UML zur Modellierung und Generierung
 - Durch aufkommen der OO wurde COOL entwickelt (Cobol Objectoriented Light)
 - OO wurde auch von fachlich geprägten Mitarbeitern akzeptiert
 - Analyse und Design von Hostprogrammen wird in Innovator modelliert
- 2008 wurde Java EE eingeführt ... neue Services werden in Java entwickelt
 - Zugriff über .NET /C#, Webanwendungen und Legacyanwendungen
- Heute: Wenige gut ausgebildete Techniker kümmern sich um Frameworks, Generatoren, Architektur, Vorgehen und Qualität

Der Prozess

Evaluierungsphase – Modellieren der Fachlichkeit

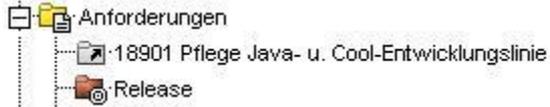
Am Anfang steht die Anforderung

- Anforderungen werden meist vom Fachbereich formuliert
 - Word
 - Excel
 - Anforderungsmanagementsystem
 - ...

- Anforderungen werden mit den Modellen über Beziehungen verbunden
 - Traceability
 - Release- Changemanagement



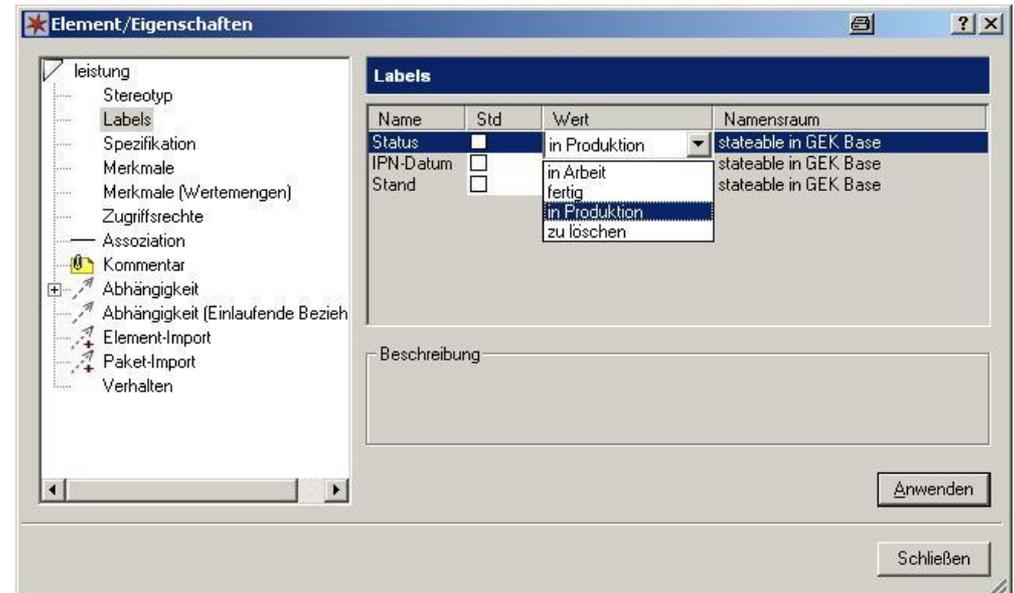
Aus einem Anforderungssystem werden Aufgaben importiert



Mit Modellreferenzen wird Nachverfolgbarkeit sichergestellt

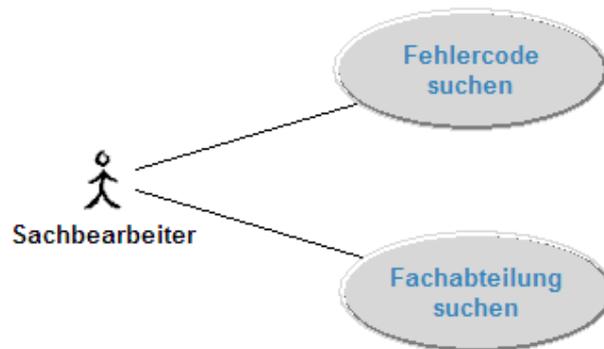
▲	Status	Name	Typ
1	OR A	leistung	«segment» Komponente
2	OR AT	fassadehost	«externalUnit» Komponente

Durch Stati wird ein Releasemanagement ermöglicht

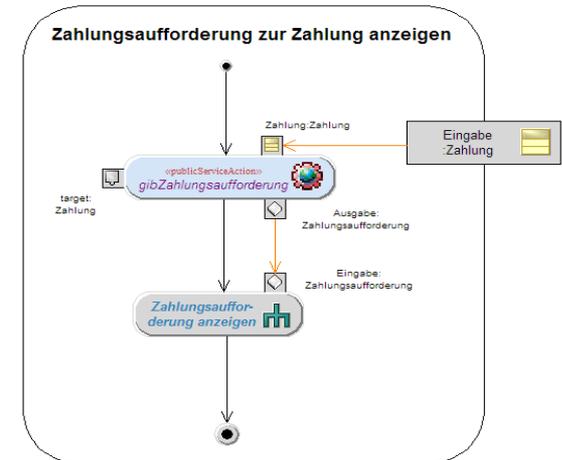


Modellierung von Anwendungsfällen

- Usecases bilden eine abstrakte Sicht auf den Nutzen, welchen der Anwender durch das System erhält.
- Gemeinsame Basis für die Kommunikation von Fachbereich und IT



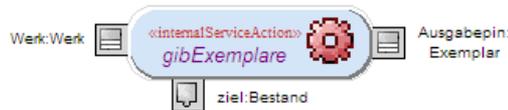
Verfeinerung der Anwendungsfälle durch Aktivitäten



Domänen und Schnittstellen



Aufruf öffentlicher Service



Aufruf interner Service



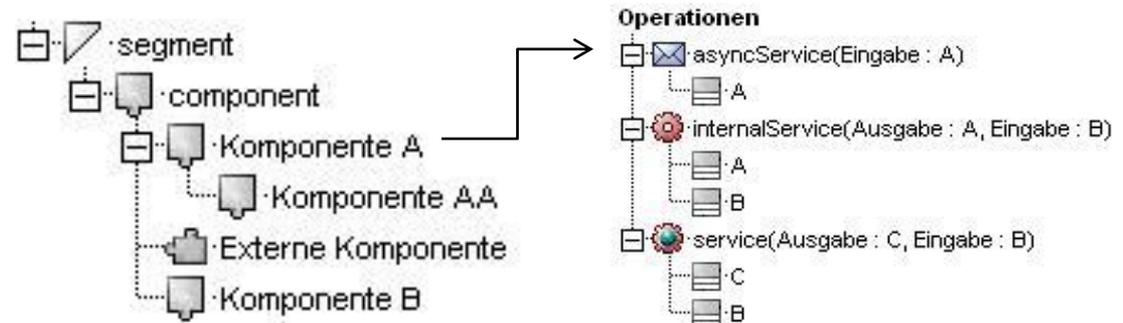
Aufruf externer Service (RPC-Service)



Asynchroner Aufruf eines Service (Verarbeitung einer Message)

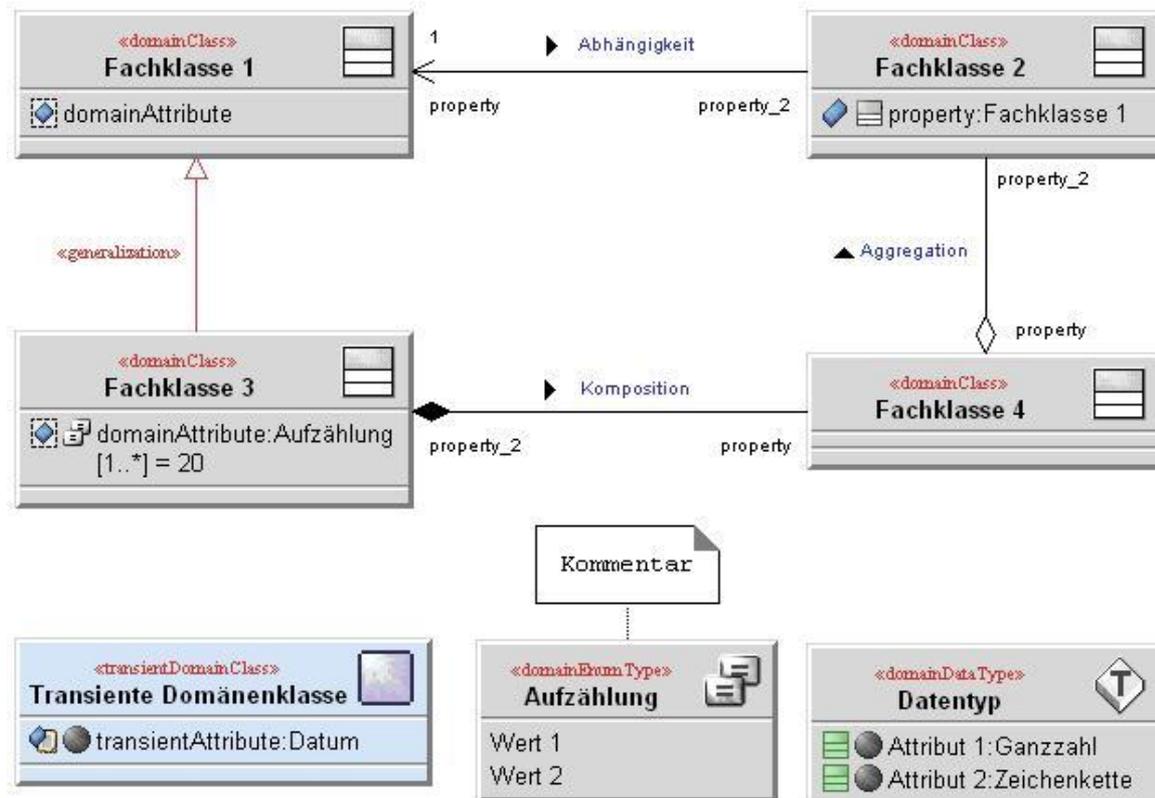


Operationsaufruf eines Domänenobjekts



Fachklassendesign

- Klassendiagramme der Fachlichkeit beschreiben die Geschäftsobjekte mit ihren Attributen -> statische Sicht
- Gibt einen Überblick über Fachklassen, deren Eigenschaften und Beziehungen untereinander
- Fachliche Datentypen und Aufzählungen werden ebenfalls modelliert

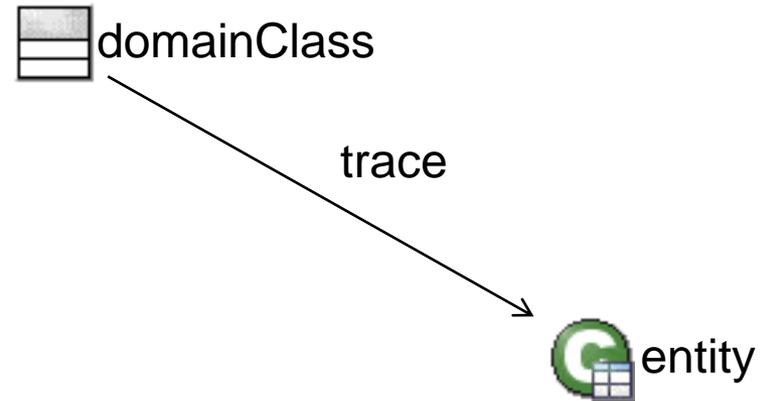


Der Prozess

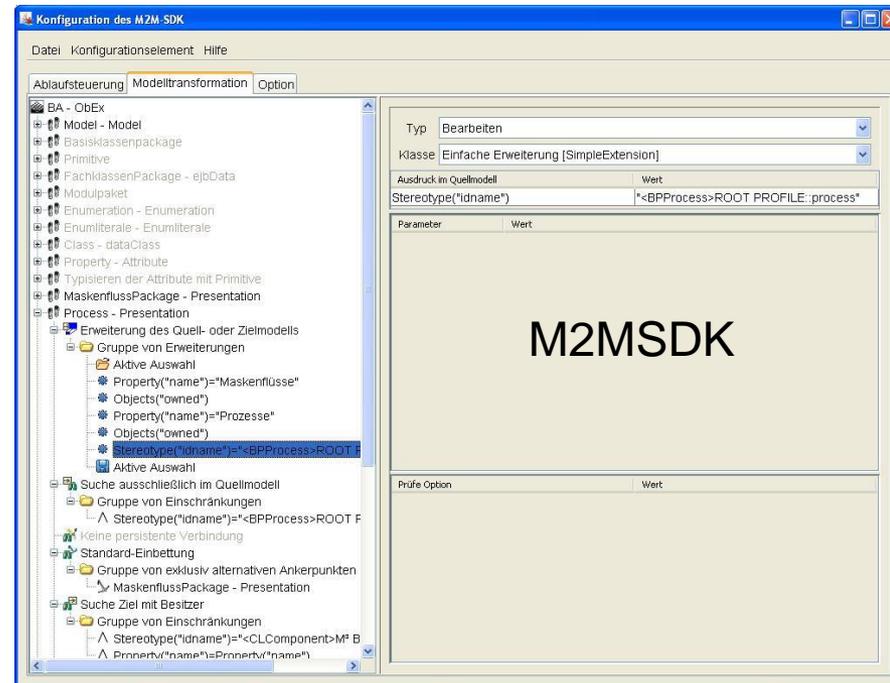
Modelltransformationen – Übergang in die technische Welt

Modelltransformationen

- Modell-zu-Modell-Transformationen
- Modellmodifikationen (In-Place-Transformationen)
- Modell-zu-Text-Transformationen
 - Codegenerierung



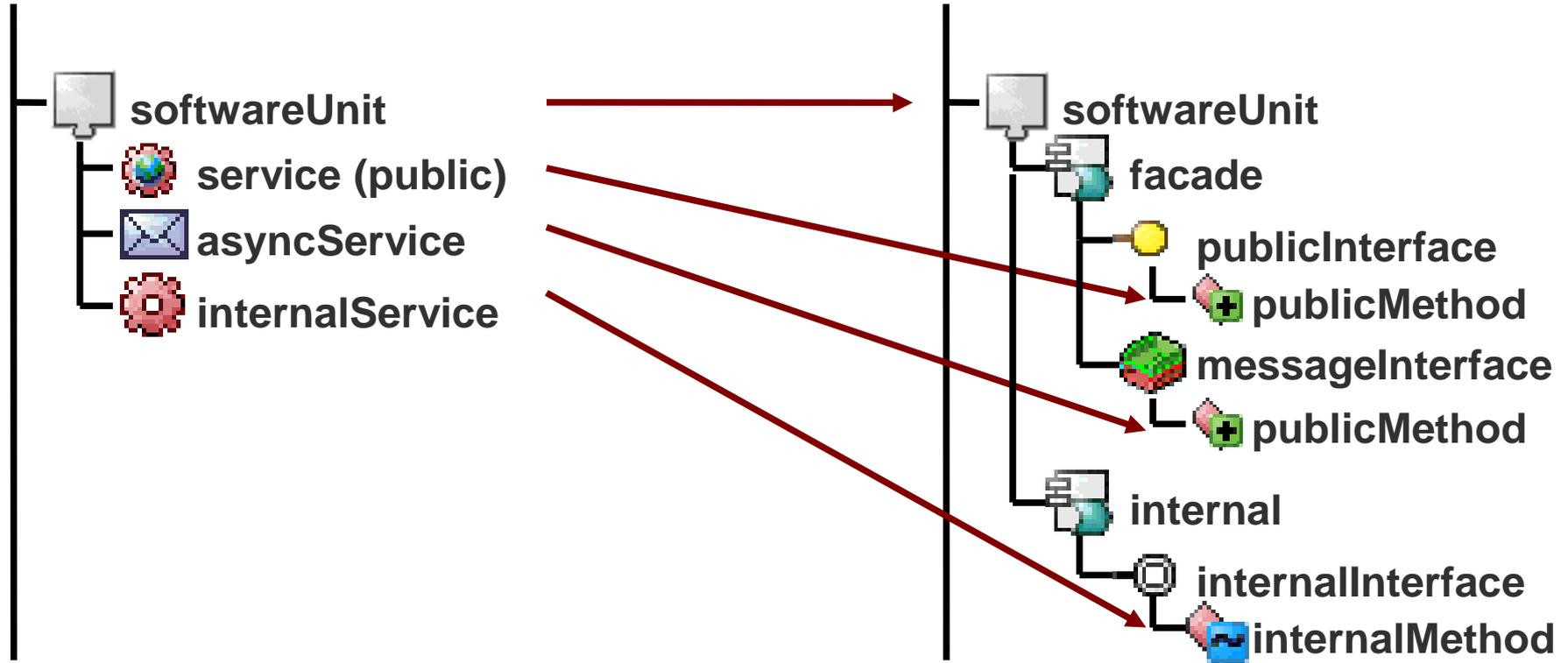
- Traceability
- Wiederholbar



Und es funktioniert doch!

Der Entwicklungsprozess – Übergang in die Technik

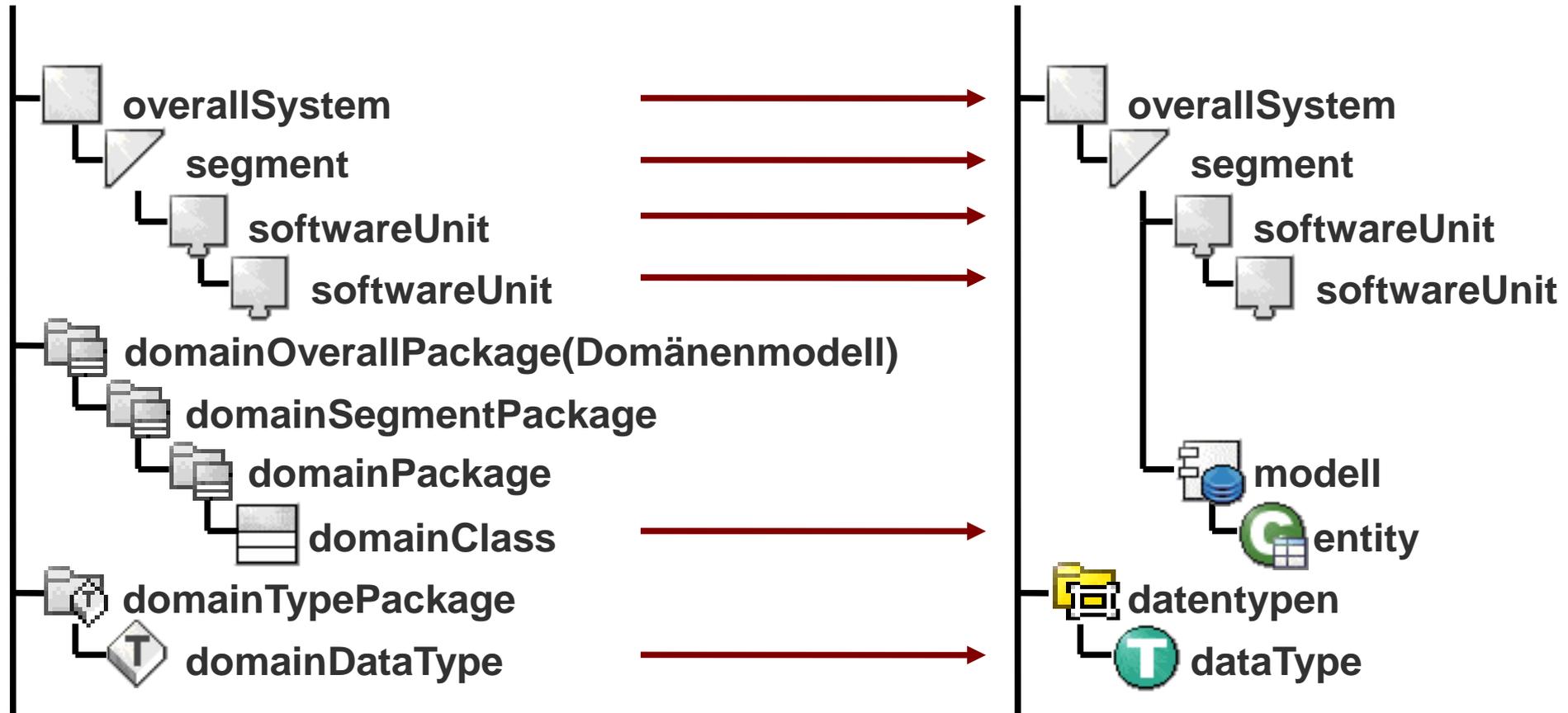
Modelltransformation: Von der Fachlichkeit zur Technik



Und es funktioniert doch!

Der Entwicklungsprozess – Übergang in die Technik

Modelltransformation: Von der Fachlichkeit zur Technik



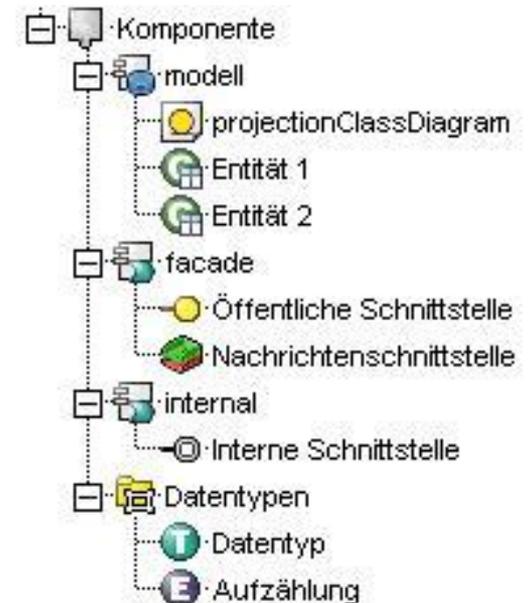
Der Prozess

Designphase – Modellieren der Technik

Designphase

- Anreicherung von technischen Informationen
 - Service-Interfaces umbenennen, anlegen
 - Service-Methoden verschieben
 - Sichtbarkeit an den Service-Methoden genauer spezifizieren
 - Technische Service-Merkmale setzen
 - Datentypen genauer spezifizieren
 - JPA Merkmale an Entities und deren Attributen setzen
 - Übersichtsdiagramme erstellen

- In-Place-Transformationen



Entities

■ Generierte Standardattribute



Element/Eigenschaften

Person

- Stereotyp
- Labels
- Stereotypeigenschaften
- Spezifikation
- Merkmale
- Merkmale (Wertemengen)

Name	Std	Wert	Namensraum
DAW Protected Region: metaTags	<input checked="" type="checkbox"/>	<input type="checkbox"/>	codeable in Object eXcellence
ExcludeSuperClassListeners	<input checked="" type="checkbox"/>	<input type="checkbox"/>	mappable in GEK ASM
Entity.name	<input checked="" type="checkbox"/>	<input type="checkbox"/>	entity in GEK ASM
Table	<input checked="" type="checkbox"/>	<input type="checkbox"/>	entity in GEK ASM

```
@EntityListeners(BearbeitungListener.class)
@Entity
public class Entity implements Bearbeitbar, Serializable {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;

    @Column(nullable = false)
    @Version
    private int version;

    @Embedded
    @AttributeOverrides( {
        @AttributeOverride(name = "zeitpunkt", column = @Column(name = "BEARBEITUNG_UM", nullable = false)),
        @AttributeOverride(name = "benutzer", column = @Column(name = "BEARBEITUNG_DURCH", nullable = false,
            columnDefinition = "CHAR(9) NOT NULL"))
    })
    private Bearbeitung bearbeitung = new Bearbeitung();
}
```

Entitiy: JPA- spezifische Attribute

Element/Eigenschaften

nachname : String

- Stereotyp
- Labels
- Stereotypeigenschaften**
- Spezifikation
- Merkmale
- Merkmale (Wertemengen)
- String
- Kommentar
- Abhängigkeit

Name	Std	Wert	Namensraum
TemplateArgs	<input checked="" type="checkbox"/>		typeable in GEK ASM
Column.name	<input checked="" type="checkbox"/>		embeddable in GEK ASM
Column.unique	<input checked="" type="checkbox"/>	<input type="checkbox"/>	embeddable in GEK ASM
Column.nullable	<input type="checkbox"/>	<input type="checkbox"/>	embeddable in GEK ASM
Column.insertable	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	embeddable in GEK ASM
Column.updatable	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	embeddable in GEK ASM
Column.columnDefinition	<input checked="" type="checkbox"/>		embeddable in GEK ASM
Column.table	<input checked="" type="checkbox"/>		embeddable in GEK ASM
Column.length	<input type="checkbox"/>	35	embeddable in GEK ASM
Column.precision	<input checked="" type="checkbox"/>	0	embeddable in GEK ASM
Column.scale	<input checked="" type="checkbox"/>	0	embeddable in GEK ASM
TemporalType	<input checked="" type="checkbox"/>	Keiner	embeddable in GEK ASM
Read-only access	<input checked="" type="checkbox"/>	<input type="checkbox"/>	embeddable in GEK ASM
RpcDataColumn.name	<input checked="" type="checkbox"/>		embeddable in GEK ASM
EnumMapping	<input checked="" type="checkbox"/>	Literalname	embeddable in GEK ASM
Basic	<input type="checkbox"/>	<input type="checkbox"/>	persistent in GEK ASM
Basic.fetch	<input type="checkbox"/>	LAZY	persistent in GEK ASM
Basic.optional	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	persistent in GEK ASM
NamedQuery	<input checked="" type="checkbox"/>	<input type="checkbox"/>	persistent in GEK ASM
Lob	<input checked="" type="checkbox"/>	<input type="checkbox"/>	persistent in GEK ASM
Version	<input checked="" type="checkbox"/>	<input type="checkbox"/>	persistent in GEK ASM
Field-based access	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	persistent in GEK ASM

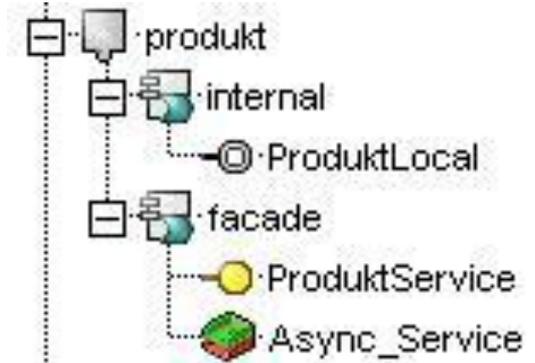
Interfaces

■ Externe Schnittstelle

- Stellt Services für andere Komponenten und Dialoge bereit
- Nutzt hierzu in der Regel andere interne Services

■ Interne Schnittstelle

- Realisiert die komplette (Fach-)Logik:
 - Objekte erzeugen, löschen, modifizieren
 - Objektmengen selektieren (JPA Zugriffe, Named Queries, etc.)
- Aufruf i.d.R. nur aus der gleichen Komponente heraus erlaubt!



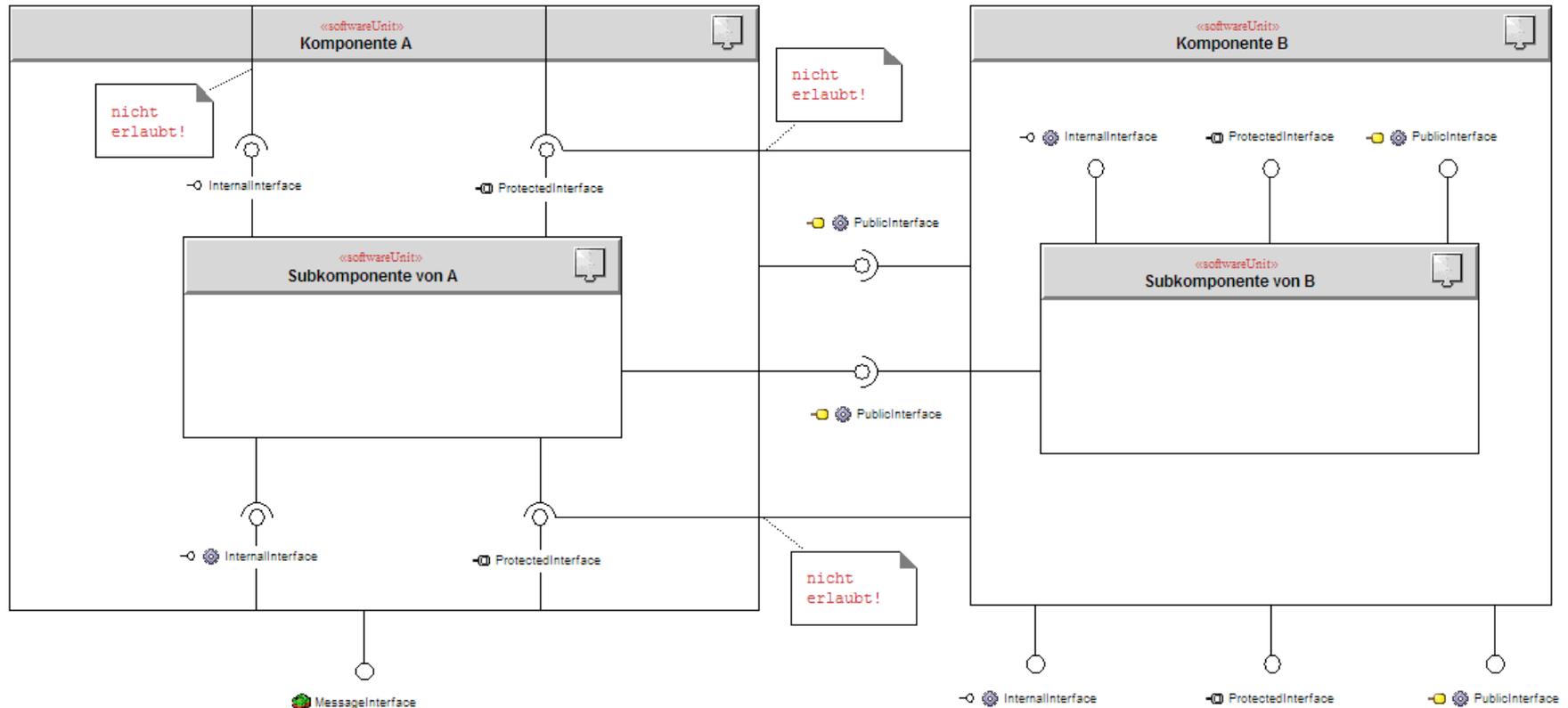
■ Asynchrone Schnittstelle

- Systemübergreifende asynchrone Kommunikation
- Transaktionssicher!
- Hoher Durchsatz (Massendatenverarbeitung)
- Lose Kopplung



Übersichtsdiagramme: Komponenten und Schnittstellen

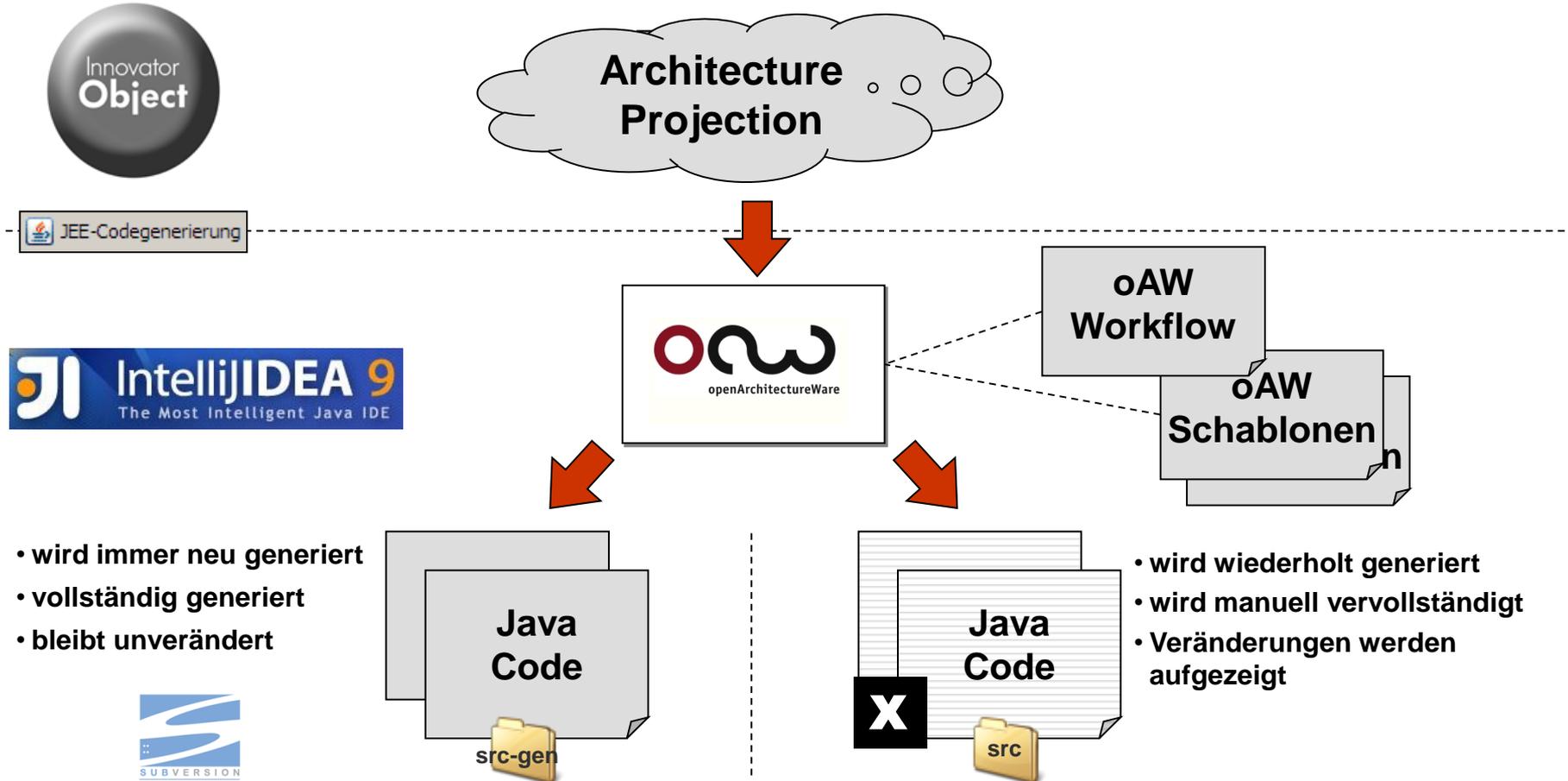
- Komponentendiagramme Kompositionsstrukturdiagramme
- Gibt einen Blick auf die Gesamtarchitektur



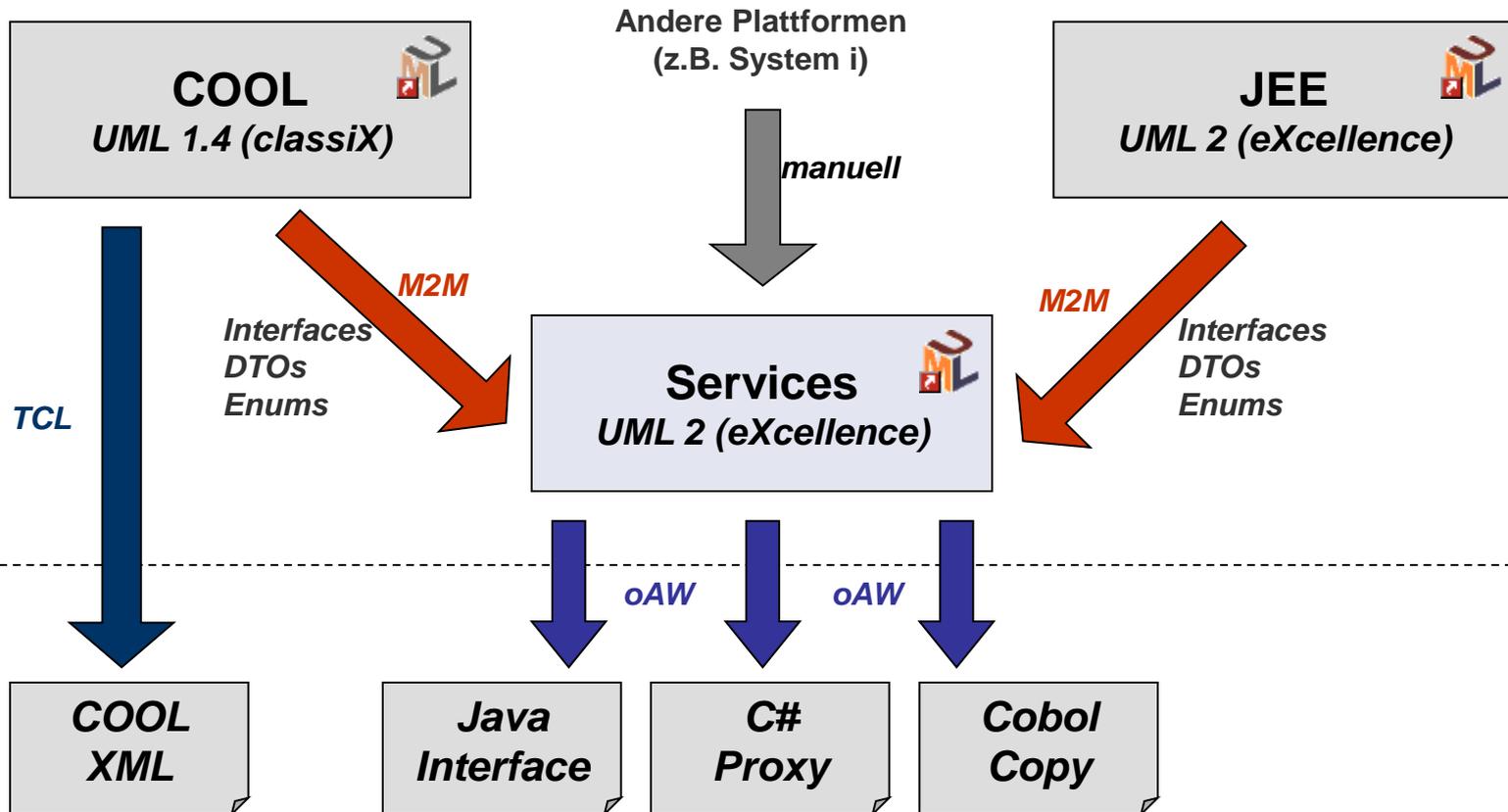
Und es funktioniert doch!

Der Entwicklungsprozess – System Construction (Generierung von Artefakten)

Codegenerierung



Service repository



Prüfmanager

The screenshot shows the MID PRÜFMANAGER application window. It has two tabs: 'Prüfmanager' and 'Prüfungen'. The 'Prüfungen' tab is active.

Anzeige

- Radio buttons: Elemente des Auftrags (QueryEngine nicht definiert), Elemente des Typs
- Dropdowns: Elementtyp, STEREOTYP
- Radio buttons: Liste ergänzen, Liste ersetzen
- Button: Anzeigen

Ausführen

- List box: ok Prüfung (checked), Prüfung: EntityCheck. (checked), Prüfung: MappingTraceCheck. (checked), Prüfung: SpecificationCheck. (checked), Prüfung: UniqueNameCheck. (checked)
- Button: Prüfen

Elementliste [Elemente: 53 / fehlerhafte Elemente: 9 / Fehler gesamt: 24 / ungeprüfte Elemente: 0]

Nr.	Element Type	Stereotype	Name	Kurzname	Hinweise	Warnungen	Fehleranzahl
19	Komponente	softwareUnit	partner		0	16	1
20	Komponente	softwareUnit	institution		0	0	0
21	Klassendiagramm	projectionClassDiagram	Schnittstellen		0	0	0
22	Klassendiagramm	projectionClassDiagram	projectionClassDiagram		0	0	0
23	Klasse	entity	Person		0	2	0
24	Klasse	entity	LstgAbrechnungNr		0	14	0
25	Klasse	entity	KVPraxisArzt		0	2	0
26	Klasse	entity	Institution		0	2	0
27	Klasse	entity	IKGruppe		0	1	0
28	Klasse	mappedSuperclass	Partner		0	0	0
29	Datentyp	dataType	Personensuche		0	0	0
30	Datentyp	dataType	Personendaten		0	0	0
31	Datentyp	dataType	PartnerNamen		0	0	0

Ergebnisliste [0. von 0 Ergebnis(sen) zum selektierten Element]

Buttons: Fehler, Warnung, Hinweis

Text: Stereotypeigenschaft DiscriminatorColumn selektiert, aber DiscriminatorColumn.Name leer.

Button: Schließen

Fazit

- Entwickler können sich auf die Kommunikationsaufrufe der Schnittstellen konzentrieren
- Schnittstelle zur Kommunikation mit FB wird vereinfacht
- Weniger fehleranfällig
 - Höhere Qualität
 - Zwischengelagerte Qualitätssicherungsmaßnahmen
- Schnellere Entwicklung
 - Viel „stupides“ Codieren bleibt erspart
- Hohe Entwicklerakzeptanz
 - Optimierungsvorschläge „noch und nöcher“
 - Weiter gestecktes Aufgabenfeld (vielseitigere Arbeit) – mehr Spaß

Erfolgsfaktoren!



Ansprechpartner: Martin Löffler (m.loeffler@mid.de)



Hauptverwaltung Nürnberg

MID GmbH

Kressengartenstraße 10

90402 Nürnberg

Telefon: +49 (0)911 - 968 36-0

Telefax: +49 (0)911 - 968 36-100

Niederlassung Stuttgart

MID GmbH

Silberburgstraße 187

70178 Stuttgart

Telefon: +49 (0)711 - 633 859-0

Telefax: +49 (0)711 - 633 859-10

www.mid.de