

Modellbasierte Werkzeugintegration

Seminar Modellgetriebene Softwareentwicklung
Abschlusspräsentation

Name: Ngoc Diep Phan

Gliederung

- **Problemstellung**
- **Zielsetzung und thematische Abgrenzung**
- **Ansätze zur Werkzeugintegration**

Problemstellung

- Bestimmte Werkzeuge unterstützen nur ausgewählte Phasen der SW-Entwicklung.
- Die Modelle sind konzeptionell stark mit einander verzahnt, können aber nicht gemeinsam benutzt werden.
- Gleiche Spezifikationen müssen oft mehrmals erfasst werden
=> ineffizient und fehleranfällig

Lösung:

- Kopplung der Entwicklungswerkzeuge, damit die Modelle reibungslos ausgetauscht werden können.
- Werkzeugintegration: Die gemeinsame Nutzung unterschiedlicher Werkzeuge
- Interoperabilität: Mehrere Werkzeuge können dasselbe Modell verwenden, um den reibungslosen Austausch von Daten und Informationen zu erreichen.

Zielsetzung und thematische Abgrenzung

- **Recherche nach Ansätze zur Werkzeugintegration (WI) im MDSD-Umfeld**
- **Klassifikation der Ansätze**
- **Thematische Abgrenzung: reine Recherche, kein Vergleich der verschiedenen Ansätze**

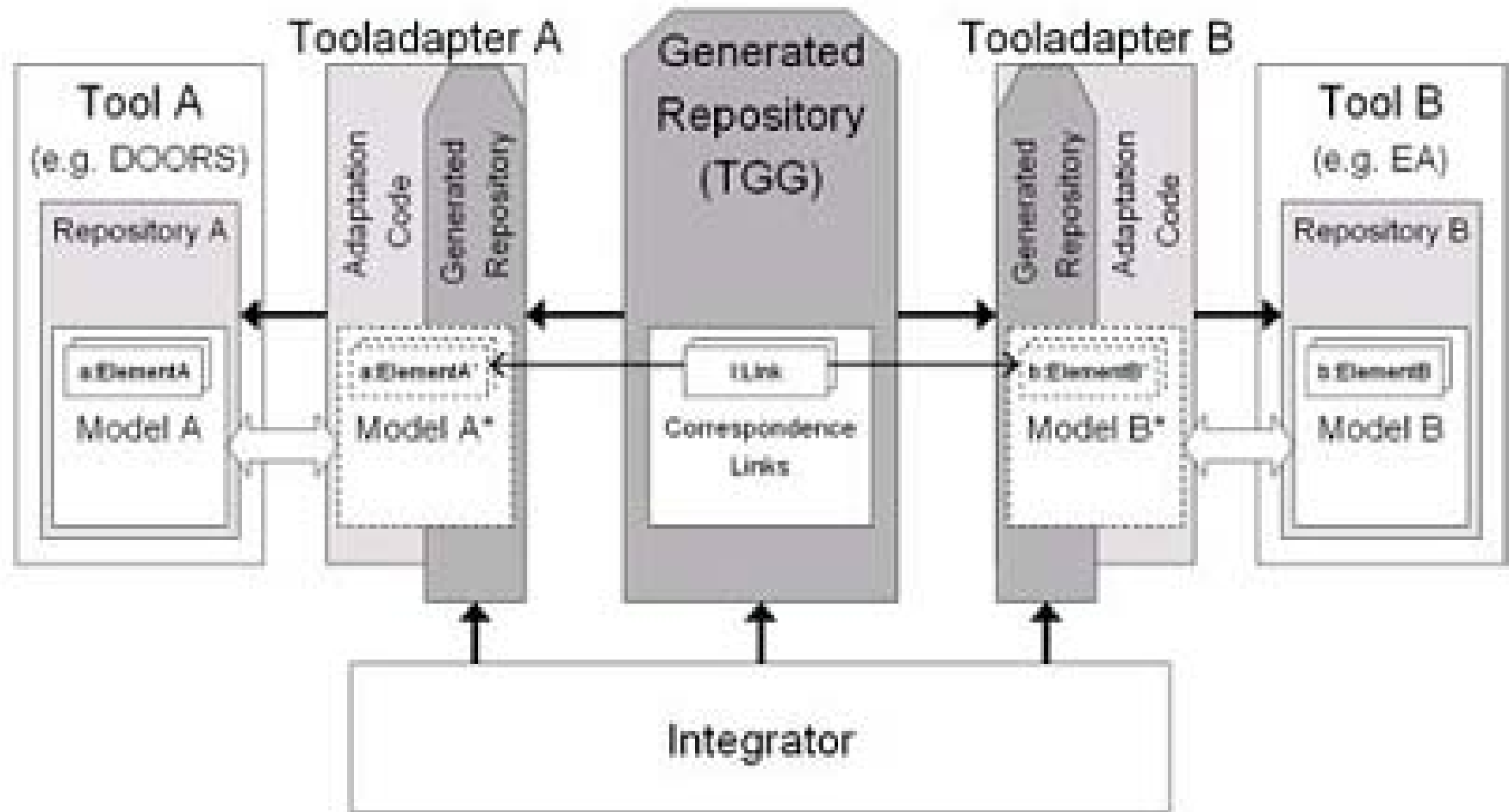
Klassifikation der Ansätze

- **WI mittels (Meta)-Modell-Bridging, z.B. MOFLON**
- **WI unter Verwendung der auf MOF/XMI-basierten Modelltransformation.**
- **WI unter Verwendung der Metamodellierung.**
 - ▶ WI, die auf integrierte Modelle basiert, z.B. ModelBus
 - ▶ WI mittels Prozess-Flows: z.B. ModelCVS

WI mittels (Meta)-Modell-Bridging am Bsp. von MOFLON

- **Ziel:** Erstellung eines Modell-Bridging zwischen den Werkzeugen
- **Ansatz:**
 - ▶ Verwendung des *Triple Graph Grammers (TGG)-Editors*, um die Korrespondenzen zwischen Elementen von Source- und Target-Metamodellen festzustellen
 - ▶ Mapping die Korrespondenzen zwischen Linktypen in Klassen und Assoziationen sowie deklarative TGG-Regeln in operationale Regeln
 - ▶ eines TGG-Repositorys, welches die Mappingsvorschriften (z.B. operationale Regeln) zur Integration beider Werkzeuge beinhaltet.
 - ▶ Generierung für jedes Werkzeug ein adaptiertes Repository zur unmittelbaren Modifikation der Werkzeugdaten

WI mittels (Meta)-Modell-Bridging am Bsp. von MOFLON



- **Ziel: Definition eines Standards (Metametamodell) von allen Metamodellen der Werkzeugen. Alle Metamodelle, die durch diesen Standard darstellbar sind, können damit problemlos ausgetauscht werden.**
- **Standardformate:**
 - ▶ **XMI (XML Metadata Interchange)**
 - ▶▶ MOF als „Übermodell“ der OMG
 - ▶▶ XMI transportiert beliebige MOF-Instanzen, d.h. Ausprägungen eines Meta-Metamodells
 - ▶ **CDIF (CASE Data Interchange Format)**
 - ▶▶ Das CDIF-Konsortium löste sich 1999 auf, da XMI gegenüber CDIF als Erfolg versprechender angesehen wurde
 - ▶ **GXL (Graph eXchange Language)**
 - ▶▶ eine *XML-Subsprache* zur Beschreibung von Graphen
 - ▶▶ 2001 als Standard-Austauschformat für das Reengineering anerkannt
 - ▶▶ Eine Fusion von GRAX, TA, Graphformat von PROGRES usw.

WI, die auf integrierte Modelle basiert

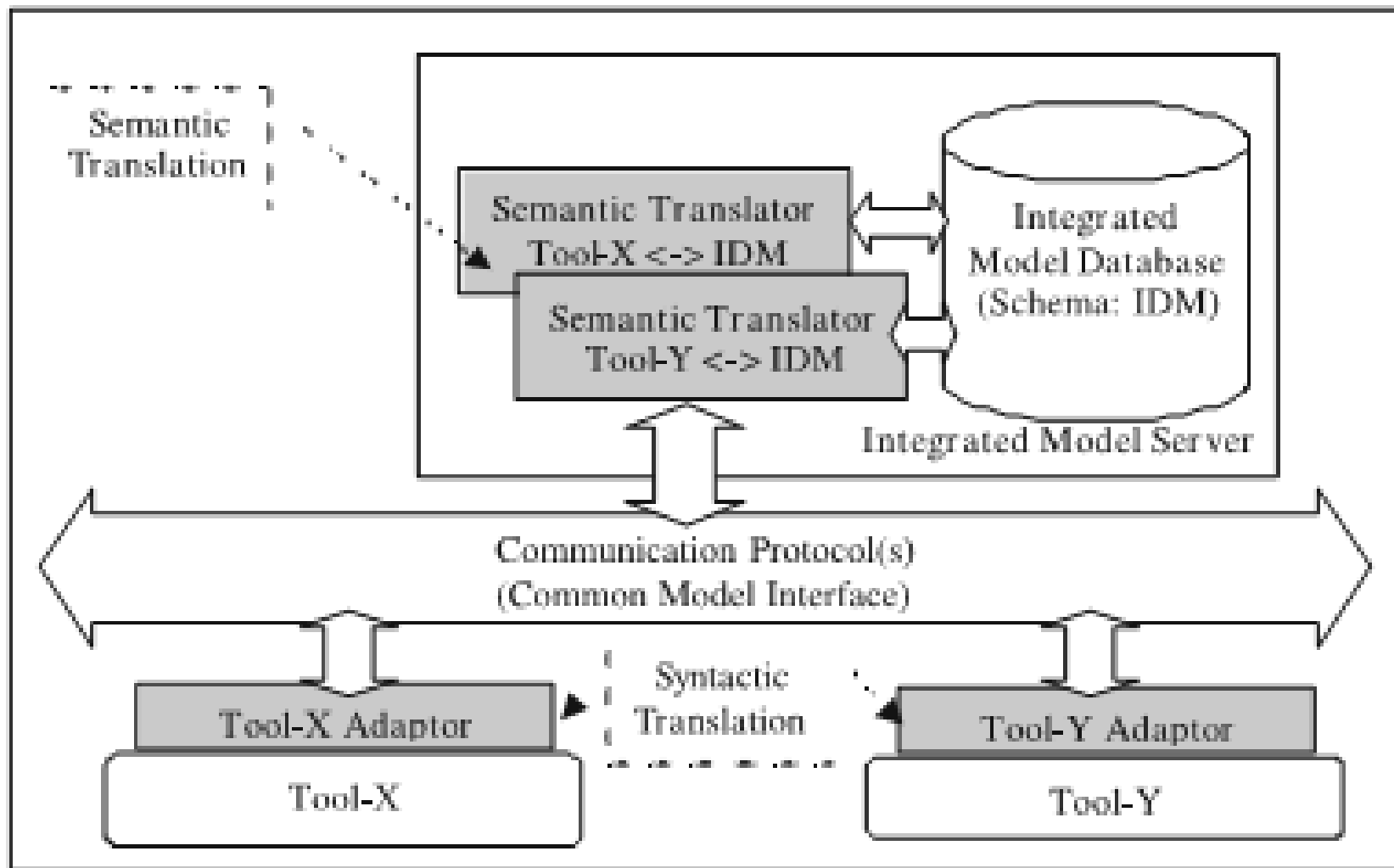
■ Problemstellung:

- ▶ die Werkzeuge haben verschiedenen Funktionen
- ▶ jedoch gemeinsamer Artefakt
- ▶ signifikante Überlappung zwischen den Konzepten der einzelnen Werkzeugen

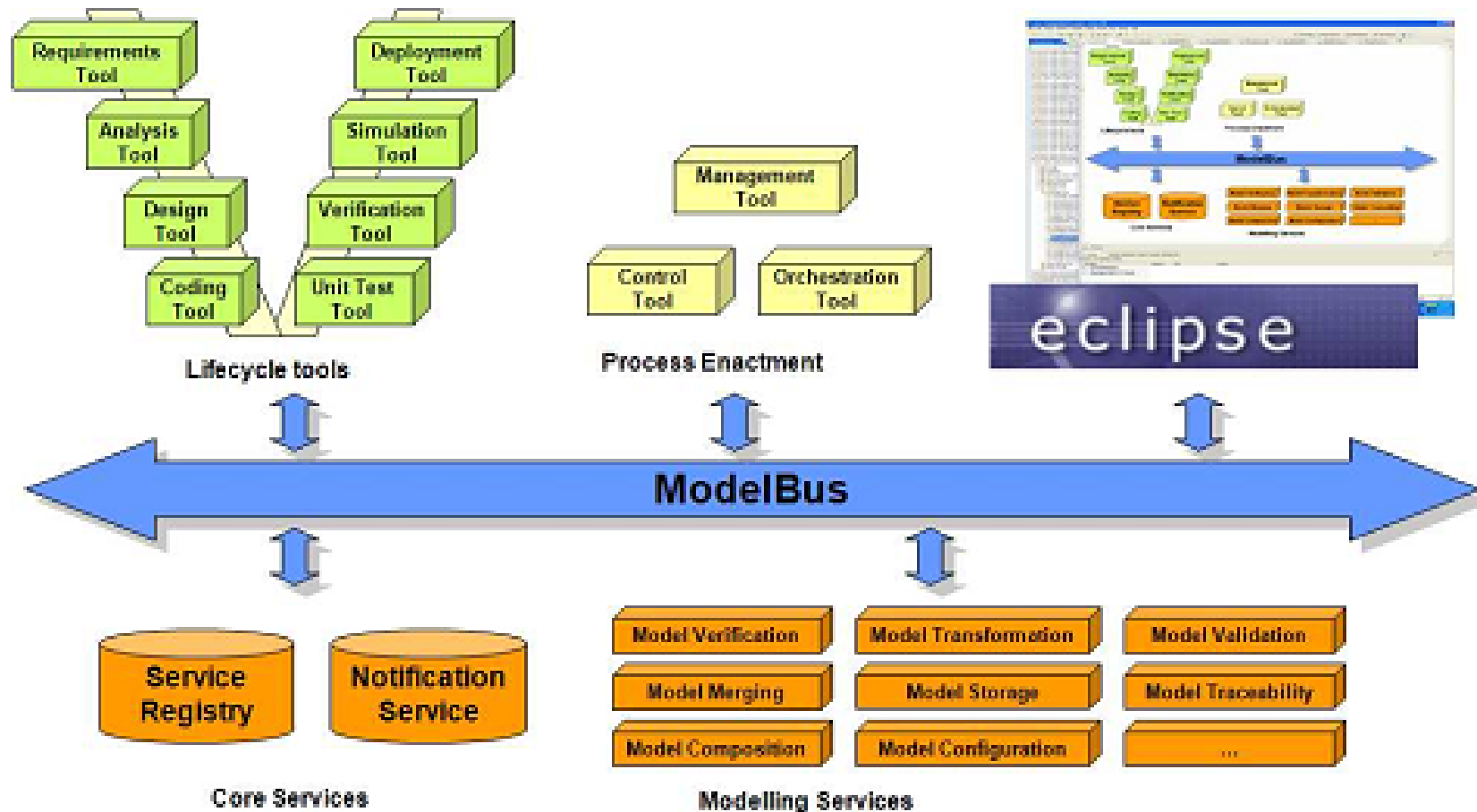
Ansatz:

- ▶ **Metadata-Repository mit einem integrierten Datenmodell**
 - ▶ „reich“ genug für die Darstellung aller möglichen Modelle von jedem beliebigen Werkzeug
- ▶ **Integrated Model Server**
 - ▶ stellt Services zum semantischen Mapping für die Werkzeuge
- ▶ **Tool Adaptors**
 - ▶ unmittelbare Kopplung am Werkzeug
 - ▶ Lesen und Schreiben von Werkzeugdaten in das vom Werkzeug generiertes Format
- ▶ **syntaktisches Mapping zwischen Daten des Werkzeuges und Daten des integrierten Modellserver**

WI, die auf integrierte Modelle basiert



WI, die auf integrierte Modelle basiert, am Bsp. von ModelBus

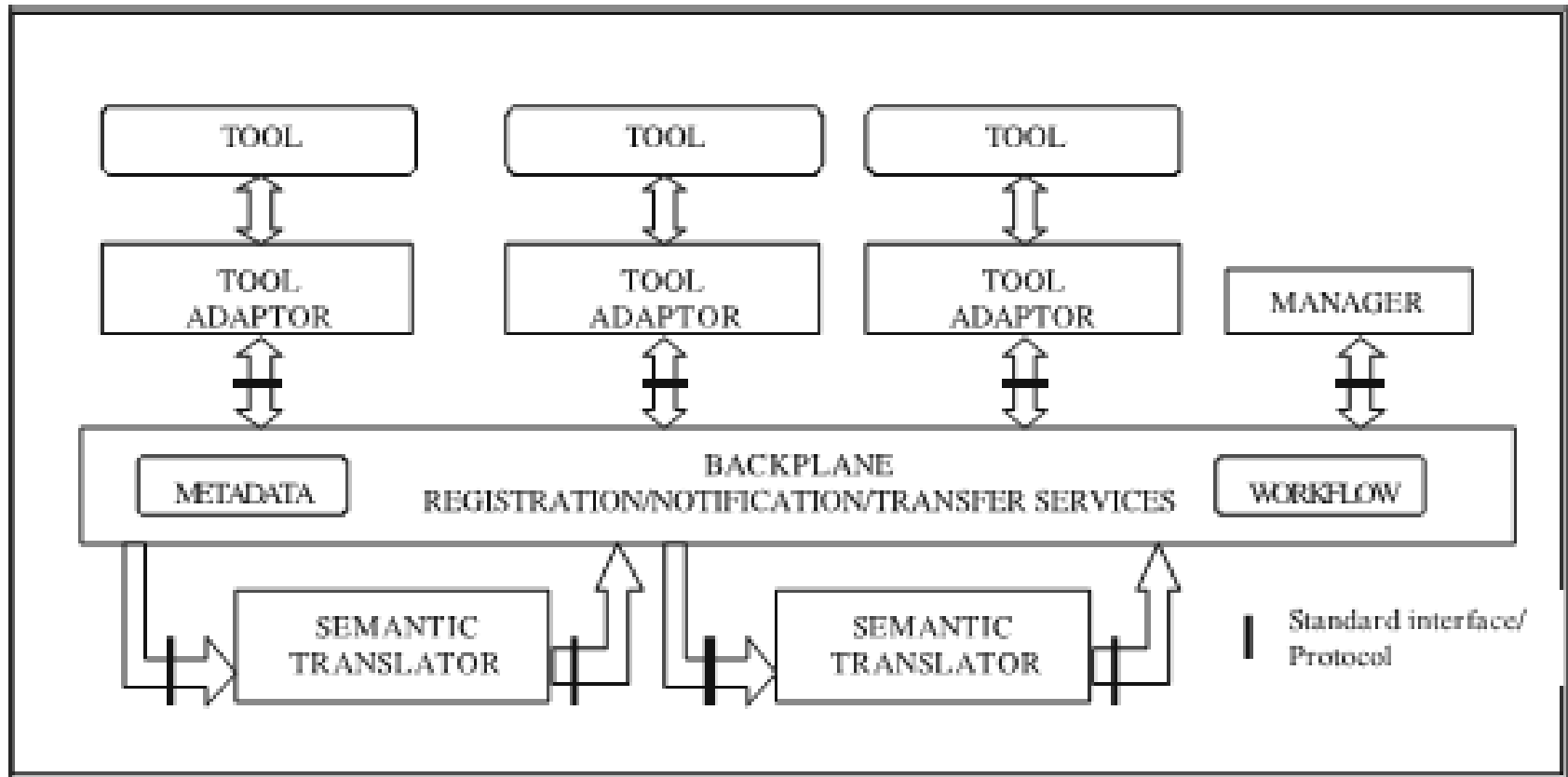


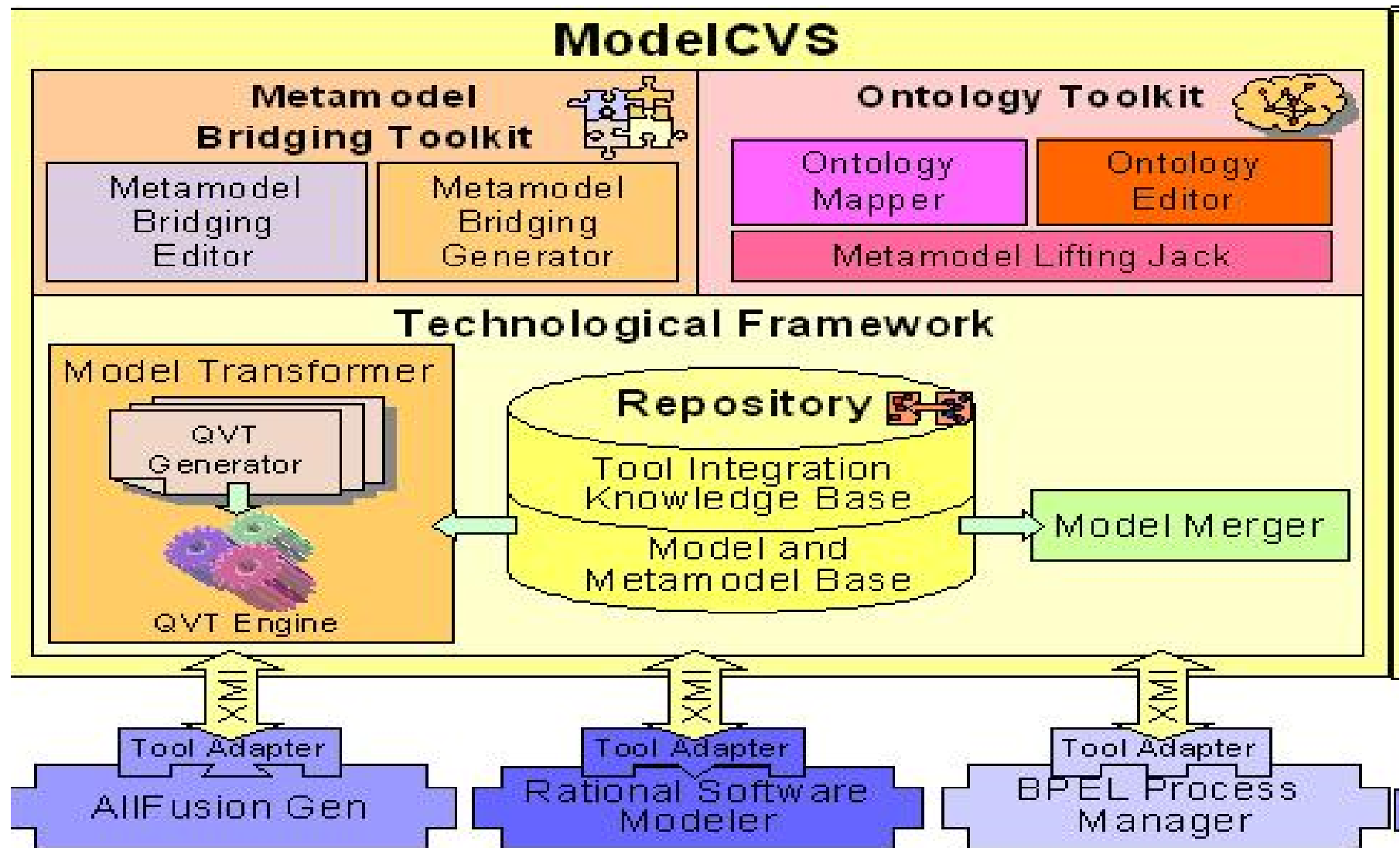
■ Problemstellung:

- ▶ typisch in eingebetteten Systems: verschiedene Softwarewerkzeuge, viele Personen in unterschiedlichen Rollen
- ▶ verschiedene Modelle: Komponenten-, System-, Analyse, lauffähige Modelle usw.
- ▶ fehlende Konsistenz der Artefakte, fehlende Automatisierung und die fehlende Interoperabilität zwischen den Werkzeugen
- ▶ eine Punkt-zu-Punkt-Integration wird gefragt.

■ Ansatz:

- ▶ **Ersatz des zentralen semantischen Translators durch verteilte semantische Translatoren**
- ▶ **Backplane-Komponent**
 - ▶ Datenverteilung erfolgt per nachrichtenbasierte Methode
- ▶ **Manager**
 - ▶ konfiguriert die Backplane und überwacht ihre Operationen





Danke für Ihre Aufmerksamkeit!