



Seminar Model Driven Integration Engineering

Based vs. Driven vs. Oriented

Patrick Jähnichen

GEFÖRDERT VOM



Bundesministerium
für Bildung
und Forschung

Das Forschungs- und Entwicklungsprojekt OrViA wird mit Mitteln des Bundesministeriums für Bildung und Forschung (BMBF) gefördert, die innerhalb der zweiten Auswahlrunde der Forschungsoffensive „Softwareengineering 2006“ vergeben wurden, und vom Deutschen Zentrum für Luft- und Raumfahrt (DLR), Projektträger Informationstechnik/Softwaresysteme betreut.

- Beschreibung der Aufgabenstellung
- Bisherige Recherchen
 - Driven
 - Based
 - Oriented
- Ausblick

- Häufige Erwähnung der Begriffe Driven, Based und Oriented im Kontext des SE
- Fragestellung, was steht wofür?
- Abgrenzung der Begrifflichkeiten voneinander
- Hauptfokus liegt auf Modellen
 - Welche Modelle gibt es
 - Welcher Art sind sie
 - Wofür werden sie verwendet
- Tiefgehende Literaturrecherche steht noch bevor
- Ziel: Dokument, nach dessen Lektüre die Gemeinsamkeiten und die Unterschiede dieser Begriffe klar sind

- Es bestehen formal eindeutige Modelle verschiedener Abstraktionsebenen (3 Kernmodelle)
 - Platform-independent model (PIM)
 - Platform-specific model (PSM)
 - Code
- Strenge Trennung zwischen fachlichen und technischen Anteilen (Kommunikationsverbesserung zw. Fach- und Nichtfachleuten)
- Modelle sind untereinander inhaltlich getrennt (separation of concern)

- Modelltransformationen (meist von höherer zu geringerer Abstraktion)
- Modell „geringster“ Abstraktion (PSM) dient als Input zur Codegenerierung
- Änderungen im Design sollen nur im Modell vorgenommen werden
→ auch Nichtfachleute können auf Entwicklung Einfluß nehmen
- UML weit verbreitet

- Modelle werden anhand des Sollverhaltens des Systems erstellt
- Domänenmodell ergibt sich aus:
 - Prozeßmodell
 - Beinhaltet die einzelnen Entwicklungsschritte
 - Bezieht sich auf Koordination der Arbeitskraft
 - Hat Dokumentationscharakter
 - Produktmodell
 - Beschreibt den logischen Aufbau des angestrebten Systems
 - Bezieht sich auf die Entwicklungsschritte die in der Entwicklungsumgebung durchgeführt werden (Datenmodelle etc.)
 - Dient als Input für spätere Codegenerierung
 - Zerfällt noch in spezifischere Untermodelle

- Alle Phasen, Prozesse, Aktivitäten können im Modell eindeutig identifiziert werden
- Benefit: Zusammenspiel von Prozeß- und Produktmodell in Entwicklungsumgebungen
- Aber: oft sind Modelle am Anfang der Implementierungsphase schon veraltet (durch undokumentierte Änderungen)

- Bisher noch nicht sehr viel
- **Vermutung**, dass wesentliche Merkmale der Objektorientierung auch auf Modellorientierte Entwicklung zutreffen
- Noch keine weiterführende Literatur gefunden

- Weiterführende Recherchen zu Begrifflichkeiten
- Genaue Eingrenzung der Begriffe und ihrer Bedeutung
- Vor- und Nachteile der einzelnen Entwicklungsansätze
- Gemeinsamkeiten und Unterschiede (evtl. Synonyme?)
- Beispielszenarios zum besseren Verständnis