



Praktische Informatik für Wirtschaftsmathematiker,
Ingenieure und Naturwissenschaftler I
(PIWIN I, 3 V + 1 Ü)
WS 2002/03

15. Vorlesungswoche

Softwaretechnik: Methoden der objektorientierten Analyse

Unterlagen:

Gumm/Sommer, Kapitel 11

Helmut Balzert: Lehrbuch der Software-Technik 1 / 2,

Spektrum Akademischer Verlag, Heidelberg u.a. 2001

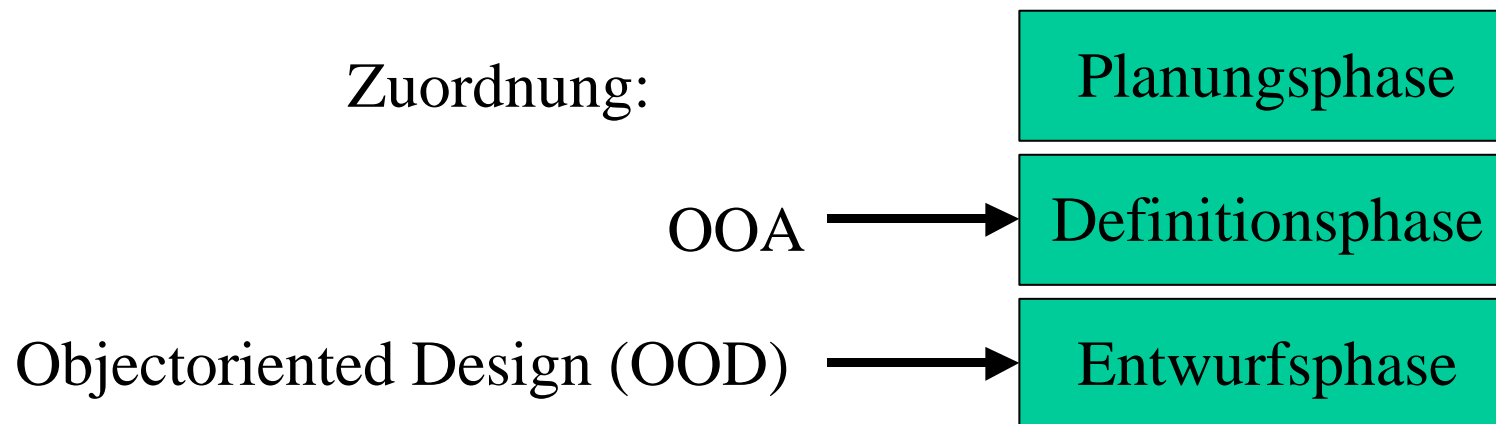
UB: Lehrbuchsammlung L Sr 389-1

Folien nach Heinecke (FH Gelsenkirchen), Heyer (Uni Leipzig)



Methoden der Objektorientierten Analyse

- Grundbegriffe
- Anwendungsfalldiagramme
- Klassendiagramme
- Zustandsdiagramme
- Aktivitätsdiagramme
- Vorgehensweise bei Objektorientierte Analyse
- Anwendung der OOA





Vorgehensweise bei Objektorientierte Analyse

- Grobanalyse
- Statisches Modell
- Dynamisches Modell



Grobanalyse

- Hauptfrage
 - Welche Elemente des Problembereichs gehören in das Modell und welche werden ausgeklammert?
- Vorgehen
 - Anwendungsfall beschreiben
 - Relevanten Problemausschnitt beschreiben
 - Teilsysteme (Pakete) abgrenzen
 - Schnittstellen zwischen Paketen definieren
- Restriktionen
 - zeitlich
 - personell
 - finanziell
 - Sachmittel / Infrastruktur



Grobanalyse (2)

- Beschreibung eines Anwendungsfalls
 - zunächst verbal
 - relativ abstrakt, wird später verfeinert
- Leitfragen für die Ermittlung von Akteuren und Prozessen
 - Welcher Akteur löst Anwendungsfall aus?
 - Welche Akteure sind am Anwendungsfall beteiligt?
 - Welche Aufgaben sind im Anwendungsfall zu erfüllen?
 - Wer ist verantwortlich für Planung, Durchführung, Kontrolle der Aufgaben?
 - Welche Ereignisse treten im Anwendungsfall auf?
 - Welche Bedingungen sind zu beachten?
 - Was sind die Ergebnisse des Anwendungsfalls?
 - Welche Beziehungen gibt es zu welchen anderen Anwendungsfällen?



Grobanalyse (3)

- Abgrenzung des relevanten Problemausschnitts
 - Aufbau auf Anwendungsfallbeschreibung
 - widerstreitende Forderungen aus fachlicher und software-technischer Sicht möglich
- Leitfragen für die Festlegung der Systemgrenzen
 - Welche Ziele werden mit dem Anwendungsfall verfolgt?
 - Welche Rollen sind an der Aufgabe beteiligt?
 - Kann die Aufgabenerfüllung durch organisatorische Umgestaltung optimiert werden?
 - Ist das betrachtete Element für die Ziele und Aufgaben von Bedeutung?
 - Welche Beziehungen und Abhängigkeiten hat dieses Element
 - zu anderen Elementen innerhalb des Systems?
 - nach außerhalb des Systems?
 - Welche Auswirkungen hat der Verzicht auf dieses Element?



Grobanalyse (4)

- Abgrenzung von Teilsystemen
 - Reduzierung der Komplexität
 - Gliederung des Gesamtsystems möglich
 - inhaltlich nach Eigenschaften von Aufgaben und Prozessen
 - prozeßorientiert nach Reihenfolge der Aufgaben
 - lokal nach Ort der Aufgaben
 - funktional nach Rollen
- Leitfragen für die Festlegung von Paketen

--->



Grobanalyse (5)

- Leitfragen für die Festlegung von Paketen
 - Ergeben sich Kriterien für Paketbildung aus Organisation des Prozesses (Ablauf, Struktur)?
 - Können durch Änderung der Organisation Pakete gebildet werden?
 - Kann die Aufgabe inhaltlich, prozeßorientiert, lokal oder funktional gegliedert werden? Ist sie überhaupt teilbar?
 - Ist das betrachtete Element ein isolierbares Subsystem?
 - Welche Beziehungen und Abhängigkeiten hat dieses Subsystem
 - zu anderen Elementen innerhalb des Systems?
 - nach außerhalb des Systems?
 - Wie stark sind die Abhängigkeiten zwischen Subsystemen?



Grobanalyse (6)

- Definition von Schnittstellen zwischen den Paketen
 - Austausch von Nachrichten und Diensten
 - Abbildung aller Beziehungen zwischen Paketen und ihren Elementen
 - flexibel in Bezug auf Änderungen
- Leitfragen zur Festlegung der Schnittstellen
 - Welche Kommunikationsbeziehungen bestehen
 - zwischen Paketen?
 - zwischen Elementen verschiedener Pakete?
 - Welche Abhängigkeitsbeziehungen bestehen
 - zwischen Paketen?
 - zwischen Elementen verschiedener Pakete?
 - Welche Daten werden über die Schnittstellen ausgetauscht?
 - Werden die Schnittstellen bei anderer Paketaufteilung einfacher?



Statisches Modell

- Hauptfrage
 - Welche Klassen gibt es
und welche Beziehungen bestehen zwischen ihnen?
- Vorgehen
 - zuerst Klassen mit Attributen und Methoden identifizieren
 - dann parallel
 - Klassen und Attribute beschreiben
 - Vererbungsstrukturen ermitteln
 - Assoziationen ermitteln



Statisches Modell (2)

- Informationsquellen zur Identifikation von Klassen
 - Formulare, Listen, Verträge etc.
 - Dokumentationen von Abläufen, Projekten etc.
 - Organigramme, Stellenbeschreibungen etc.
 - vorhandene Software und ihre Dokumentation
 - Bildschirmmasken
 - Datenbankaufbau etc.
 - Interviews mit Akteuren
 - Modelle vergleichbarer Problembereiche
 - Fachliteratur
 - durchgeführte Projekte
 - andere Firmen / Abteilungen etc.
- Faustregel
 - gegenständliche Substantive -> häufig Klassen
 - beschreibende Substantive (Eigenschaften) -> häufig Attribute
 - Verben -> häufig Methoden



Statisches Modell (3)

- Leitfragen zur Identifikation von Klassen
 - Welche Akteure sind beteiligt?
 - Welche Gegenstände werden für die Aufgaben benötigt?
 - Welche Informationen werden von den Akteuren verarbeitet?
 - An welchen Orten werden die Aktivitäten durchgeführt?
 - Ähneln sich bestimmte Aufgaben in ihrer Struktur?
 - Welche Ereignisse treten auf?
 - Welche Hardware / Software wird bereits verwendet?
- Leitfragen zur Überprüfung gefundener Klassen
 - Werden die Informationen in der Klasse für den Anwendungsfall benötigt?
 - Was ändert sich beim Weglassen der Klasse
 - an der Abbildung des Problembereichs?
 - am Objektmodell?
 - Unterscheidet sich die Klasse deutlich von den anderen?
 - Kann man die Klasse als Attribut einer anderen modellieren?
 - Stimmt die Zuordnung zum Paket?



Statisches Modell (4)

- Bestimmung von Attributen
 - Informationsquellen wie bei Klassen
 - vorläufige Festlegung des Datentyps
 - vorläufige Festlegung der Sichtbarkeit
- Leitfragen für die Festlegung von Attributen
 - Welche Daten beschreiben Objekte der betrachteten Klasse?
 - Welche Informationen muß die Klasse für andere zur Verfügung stellen?
 - Welche Zustände kann / soll die Klasse einnehmen?
 - Ist das betrachtete Attribut Gegenstand von Operationen?
 - Verlangen diese Operationen einen bestimmten Argument-/Ergebnistyp?
 - Was ändert sich beim Weglassen des Attributs
 - an dieser Klasse?
 - an anderen Klassen?
 - Ist das Attribut so komplex, dass eine eigene Klasse sinnvoller ist?
 - Ist die Klassendefinition noch übersichtlich?



Statisches Modell (5)

- Bestimmung von Methoden
 - Informationsquellen wie bei Klassen
 - vorläufige Festlegung der Parameter
 - Leitfragen analog denen für Attribute
- Überprüfung von Attributen und Methoden
 - in Hinblick auf Vererbung
 - Vollständigkeit aufwärts / abwärts
 - sinnvolle Sichtbarkeit
 - in Hinblick auf Assoziationen
 - zyklisch in Zusammenhang mit
 - Beschreibung der Klassen
 - Ermittlung der Vererbungsstruktur
 - Ermittlung der Assoziationen



Statisches Modell (6)

- Beschreibung von Klassen und Attributen
 - über die Klassendefinition hinaus
 - als verbale Dokumentation
 - mit Angabe von
 - Klassenname
 - Paketname
 - vollständige Vererbungsstruktur (alle Oberklassen)
 - Name, Typ, Sichtbarkeit und Bedeutung von Attributen
 - Name, Parameter und Zweck von Methoden
 - zyklisch ergänzt und erweitert
 - wichtig in Hinblick auf Wartung der zu entwickelnden Software



Statisches Modell (7)

- Ermittlung von Vererbungsstrukturen
 - Generalisierung
 - Finden gleichartiger Attribute und Methoden
 - Bilden einer gemeinsamen Oberklasse
 - Spezialisierung
 - Untersuchung von Sonderfällen
 - Feststellen zusätzlich nötiger Attribute / Methoden
 - Bilden von Unterklassen
- Leitfragen zum Finden von Vererbungsstrukturen
 - Gibt es Gemeinsamkeiten zwischen identifizierten Klassen?
 - Gibt es spezielle Ausprägungen von Klassen?
 - Gibt es im Problembereich Bezeichnungen desselben Sachverhalts auf unterschiedlichem Abstraktionsgrad?
 - Gibt es hierarchische Strukturen? Mit Ähnlichkeiten von Elementen?



Statisches Modell (8)

- Überprüfung von Vererbungsstrukturen
 - Werden alle vererbten Attribute in der Unterklasse tatsächlich benötigt?
 - Geben die vererbten Methoden wirklich das Verhalten der Instanzen der Unterklasse wieder?
 - Liegt statt Vererbung eine andere Beziehung vor?
 - Vermeidung zu tiefer Vererbungshierarchien
 - Vermeidung von Mehrfachvererbung



Statisches Modell (9)

- Identifikation von Assoziationen
 - Relevante Beziehungen ermitteln
 - Ermittelte Beziehungen auf Aggregation / Komposition prüfen
 - Multiplizitäten bestimmen
- Leitfragen zum Auffinden von Assoziationen
 - Welche Elemente tauschen Nachrichten aus?
 - Welche Elemente bieten Dienste für andere an?
 - Setzt sich ein Element aus anderen zusammen?
 - Hängt die Existenz eines Elements von der eines anderen ab?
 - Lassen sich beide Elemente in einer Klasse zusammenfassen?
 - Gibt es zusammengehörige organisatorische Einheiten?
 - Gibt es Elemente, die bestimmte Rollen einnehmen?
 - Gibt es Beziehungen der Art „hat ein“, „ist ein“, „besitzt“, „gehört zu“, „benutzt“, „verwaltet“, „plant“, „führt aus“, „kontrolliert“, „beschreibt“?



Dynamisches Modell

- Hauptfrage
 - Wie arbeitet das System?
 - Ziel: genaue Abbildung der realen Vorgänge
- Vorgehen
 - erst Interaktionen identifizieren
 - Sequenzdiagramme und Kollaborationsdiagramme
 - dann parallel
 - Zustände und Übergänge definieren
 - Zustandsdiagramme, wenn nötig
 - Ereignisse identifizieren
 - Aktivitäten identifizieren
 - Aktivitätsdiagramme, wenn nötig
 - dann Methoden spezifizieren
 - aufbauend auf Grobanalyse und statischem Modell



Dynamisches Modell (2)

- Informationsquellen zur Identifikation von Interaktionen
 - Verteilerlisten
 - Organigramme, Stellenbeschreibungen
 - Telefon-, Fax-, Email-Verzeichnisse
 - Berichtswesen
 - Ablagesystem
 - Dokumentenmanagementsysteme
 - Controllingsysteme und -berichte
 - Projektberichte
 - Interviews
 - vorhandene Netzwerkstrukturen
 - Dokumentation vorhandener Hard- und Software



Dynamisches Modell (3)

- Leitfragen zur Identifikation von Interaktionen
 - Welche Assoziationen enthält das statische Modell?
 - Welche Objekte arbeiten an einer gemeinsamen Aufgabe?
 - Welche Objekte sind Dienstleister für andere?
 - Welche Objekte benötigen Informationen von anderen?
 - Welche Objekte kommunizieren miteinander?
 - Was ist das Ziel der Kommunikation zweier Objekte?
 - Über welche Schnittstellen tauschen die Objekte Informationen aus?
 - Welche Informationen werden ausgetauscht?
 - In welchen Zuständen werden Informationen ausgetauscht?
 - Unterliegt die Kommunikation bestimmten Bedingungen?
 - Welchen Lebenszyklus durchlaufen die beteiligten Objekte?
 - Wann werden Objekte erzeugt / gelöscht?



Dynamisches Modell (4)

- Definition von Zuständen und Übergängen
 - Lebenszyklus von Objekten / Klassen
 - Zustandsdiagramme nur bei komplexem Verhalten erforderlich
 - Vorgehen
 - Anfangszustand nach Entstehung eines Objekts ermitteln
 - die möglichen Zustände des Objekts beschreiben
 - Übergänge feststellen
 - Endzustände ermitteln
 - zyklisch verfeinern zusammen mit Ermittlung von Ereignissen



Dynamisches Modell (5)

- Leitfragen zu Zuständen und Übergängen
 - Wann wird ein Objekt erzeugt?
 - Welche Anfangswerte besitzt es?
 - Was ist der Anfangszustand des Objekts?
 - Welche Schnittstellen besitzt das Objekt?
 - Welche eigenen Methoden ändern Attributwerte (Zustände)?
 - Wodurch werden diese Methoden aktiviert (Ereignisse)?
 - Welche konkreten Ereignisse führen zu einem Zustandsübergang?
 - Unter welchen Bedingungen führt ein Ereignis zu einen Zustandswechsel?
 - Kann das gleiche Ereignis unterschiedliche Reaktionen / Übergänge auslösen?
 - Was ist der Endzustand des Objekts?
 - Wann endet der Lebenszyklus des Objekts?
 - Sind alle Zustände des Diagramms erreichbar?



Dynamisches Modell (6)

- Leitfragen zur Identifikation von Ereignissen
zusätzlich zu den Fragen zu Zuständen und Übergängen
 - Welche Beziehungen hat das Objekt?
 - Welche Zustände sind bereits ermittelt?
 - Wie unterscheiden sich die Zustände?
 - Welche externen Methoden greifen auf das Objekt zu?
 - Betrifft ein bestimmtes Ereignis mehrere Objekte?



Dynamisches Modell (7)

- Identifikation von Aktivitäten
 - enger Zusammenhang mit Zuständen
 - zusätzliche Informationsquellen
 - Beobachtung bei der Aufgabenbearbeitung
 - Analyse von Protokollen
- Leitfragen zu Aktivitäten
 - Welche Verantwortlichkeiten bestehen?
 - Welche internen Prozesse laufen in einem Objekt ab?
 - Welche Bedingungen / Ergebnisse gehören zu einem Prozeß?
 - Wo gibt es Alternativen bei Zustandsübergängen? Welche Bedingungen?
 - Wo führt ein Übergang zu mehreren verschiedenen Aktivitäten?
 - Wo werden mehrere Aktivitäten zu einer Folgeaktivität zusammengeführt?
 - Wo gibt es Beziehungen zwischen Aktivitäten und Objektzuständen?
 - Mit welchen Objekten werden Signale ausgetauscht? Bedingungen?



Dynamisches Modell (8)

- Beschreibung von Methoden
 - Dokumentation mit
 - Name
 - Parametern / Datentypen
 - Zweck
 - Vorgehensweise
 - Resultat
 - verbale Beschreibung
 - Spezifikation durch
 - Pseudocode
 - Programmiersprache
 - Programmablaufplan / Struktogramm



Dynamisches Modell (9)

- Leitfragen zur Beschreibung von Methoden
 - zusätzlich zu den bereits genannten Angaben
 - Wodurch wird die Methode ausgelöst?
 - Welche Daten benötigt die Methode?
 - Wie führt sie ihre Aufgabe aus? Teilaufgaben? Reihenfolge?
 - Welche internen Attribute werden gelesen / verändert?
 - Welche Informationen / Dienste von anderen Objekten werden benötigt?
 - Sendet die Methode eine Antwort / ein Ergebnis an das aufrufende Objekt? An andere Objekte?
 - Welche Sichtbarkeit hat die Methode?
 - Überprüfung des Objektmodells: Hat die Methode Zugriff auf alle von ihr benötigten Informationen?



Anwendung der OOA

- Vorgehen
 - Leitbilder für Klassenaufteilung
 - Material - Werkzeug - Aspekt
 - Orientierung
 - an Informationen (Daten)
 - an Funktionalität (Verhalten)
 - gemischt
 - Modell
 - Spiralmodell
 - Prototyping



Anwendung der OOA (2)

- Ergebnisse
 - verschiedene Modelle für ein Problem möglich
 - mehrere zutreffende Modelle möglich
 - nicht alle zutreffenden Modelle gleich effizient
 - unzutreffende Modelle nicht immer sofort erkennbar
 - ausführliche Dokumentation wesentlich
 - grafische Darstellung heute vereinheitlicht (UML)
 - nicht immer übersichtlich
 - zusätzliche textuelle Ergänzungen erforderlich
 - Wiederverwertbarkeit nicht automatisch gegeben
 - sehr konkrete Modellierung nur für Einzelfall, aber effektiv
 - sehr abstrakte Modellierung großer Aufwand, aber universell
 - Abhilfe durch Entwurfsmuster



Zusammenfassung

OOA

- verwendet Konzepte aus OO Programmierung:
 - Objekt, Klasse, Vererbung, Polymorphie, späte Bindung, Schnittstellen
- verwendet Konzepte aus ER Modellen:
 - Assoziation, Kardinalität
- und weitere:
 - Rollen
 - Aggregation, Komposition
 - Design Patterns, Muster
 - Anwendungsfälle, Szenarien
- ist verbunden mit UML Diagrammen als Menge von Notationen
 - Klassendiagramm,
 - Aktivitätsdiagramm, Zustandsdiagramm
 - Sequenzdiagramm, Kollaborationsdiagramm
 - Anwendungsfalldiagramm
 - Paketdiagramm

für die verschiedenen Sichten auf ein System