

Wie werden Software-Systeme entwickelt?

Bevor mit der Implementierung begonnen wird,  
werden die Anforderungen ermittelt und beschrieben.

Danach wird ausgehend von der Beschreibung der Anforderungen eine Struktur festgelegt, nach der SW-System gebaut wird.

Erst dann beginnt die Implementierung.

Dabei begleiten ständig Tests und Dokumentation die Arbeit.

Was ist Software-Engineering“? (siehe weitere Definitionen)

Software-Engineering ist

- die effektive und effiziente Entwicklung und Weiterentwicklung  
komplexer SW-Systeme
- sowie begleitender Dokumente
- in einem bewusst arbeitsteilig gestalteten Prozess
- unter Anwendung bewährter Prinzipien, Methoden und Modellen.

Warum haben Analyse und Definition von Anforderungen an das SW-System große Bedeutung im Entwicklungsprozess?

Auch heute noch werden nur die Hälfte aller SW-Projekte wie geplant realisiert.

60% der Fehler im SW-system resultieren aus Fehlern in der Analysephase, d.h. aus Fehlern in der Definition der Anforderungen.

Die Behebung von Fehlern aus der Analysephase sind vergleichsweise sehr teuer.

Jochen Ludewig / Horst Lichter „Software Engineering“  
Grundlagen, Menschen, Prozesse, Techniken

Auflage 1, dpunkt.verlag 2007 3-89864-268-2, S. 51f

Software Engineering in der Informatik



Hygiene in der Medizin

Jochen Ludewig / Horst Lichter „Software Engineering“  
Grundlagen, Menschen, Prozesse, Techniken

Auflage 1, dpunkt.verlag 2007 3-89864-268-2, S. 51f

Software Engineering in der Informatik



Hygiene in der Medizin

nützt nichts  
sondern verhindert vielmehr Schäden  
→ sollte generell beachtet werden

„Software Engineering ist – wie die Hygiene in der Medizin –  
langweilig und frustrierend für Leute,  
die die Abwehr von Fehlschlägen und Katastrophen  
nicht als positive Leistung betrachten.“

Problem der SW-Entwicklung am Ende der 60-iger Jahre:

veränderte Rahmenbedingungen  
beim Einsatz und der Entwicklung  
von Software-Systemen

Strategien bei der SW-Entwicklung ändern



Was kennzeichnet ein System?

Warum sind Software-Systeme ganz spezielle Systeme?

Welchen Unterschied gibt es zwischen Modellen und Prototypen bei der Software-Entwicklung?

Welche Grundprinzipien finden in der SW-Entwicklung Anwendung?

## → VDI 3633 Blatt 1: System ( VDI = Verein Deutscher Ingenieure)

Ein System ist eine abgegrenzte Anordnung von Komponenten, die miteinander in Beziehung stehen.

Es ist gekennzeichnet durch:

- die Festlegung seiner Grenze gegenüber der Umwelt (**Systemgrenze**), mit der es über **Schnittstellen** Materie, Energie, Informationen austauschen kann.  
(System**ein**- und System**ausgangsgrößen**)
- die Komponenten, die bei der Erhöhung der Auflösung selbst wiederum Systeme darstellen (**Subsysteme**) **oder** aber als nicht weiter zerlegbar angesehen werden (**Elemente**).
- die **Ablaufstruktur** in den Komponenten, die durch spezifische Regeln und konstante oder variable Attribute charakterisiert wird.
- die **Relationen**, die Systemkomponenten miteinander verbinden (**Aufbaustruktur**)
- die **Zustände der Komponenten**, die jeweils durch Angabe der Werte aller konstanten und variablen Attribute (Zustandsgrößen) beschrieben werden, von denen im allgemeinen nur ein kleiner Teil untersuchungsrelevant ist.
- die **Zustandsübergänge der Komponenten** als kontinuierliche oder diskrete Änderungen mindestens einer Systemvariablen aufgrund des in dem System ablaufender **Prozesses**.

→ VDI 3633 Blatt 1: **Prozess** ( VDI = Verein Deutscher Ingenieure)

---

Ein Prozess ist die Gesamtheit von aufeinander einwirkenden **Vorgängen** in einem System, durch die Materie, Energie oder auch Informationen **umgeformt, transportiert oder auch gespeichert werden.**

Was kennzeichnet ein System?

Ein System besteht aus Komponenten, die miteinander in Beziehung stehen.

Eine Grenze trennt das System von seiner Umgebung (auch Kontext genannt).

Schnittstellen (Verbindungsstellen) verbinden das System mit seiner Umwelt.

Ein System kann Subsysteme enthalten. Ein System befindet sich zum Zeitpunkt  $t$  in einem def. Zustand.

Warum sind Software-Systeme ganz spezielle Systeme?

Welchen Unterschied gibt es zwischen Modellen und Prototypen bei der Software-Entwicklung?

Welche Grundprinzipien finden in der SW-Entwicklung Anwendung?



Was kennzeichnet ein System?

Ein System besteht aus Komponenten, die miteinander in Beziehung stehen.

Eine Grenze trennt das System von seiner Umgebung (auch Kontext genannt).

Schnittstellen (Verbindungsstellen) verbinden das System mit seiner Umwelt.

Ein System kann Subsysteme enthalten. Ein System befindet sich zum Zeitpunkt  $t$  in einem def. Zustand.

Warum sind Software-Systeme ganz spezielle Systeme?

SW-Systeme sind „immateriell“ und komplex; sie haben keine natürliche Lokalität.

SW-Systeme sind aus einem Werkstoff hergestellt, der von sich aus keine Strukturierung im „Großen“ erfordert.

Welchen Unterschied gibt es zwischen Modellen und Prototypen bei der Software-Entwicklung?

Welche Grundprinzipien finden in der SW-Entwicklung Anwendung?

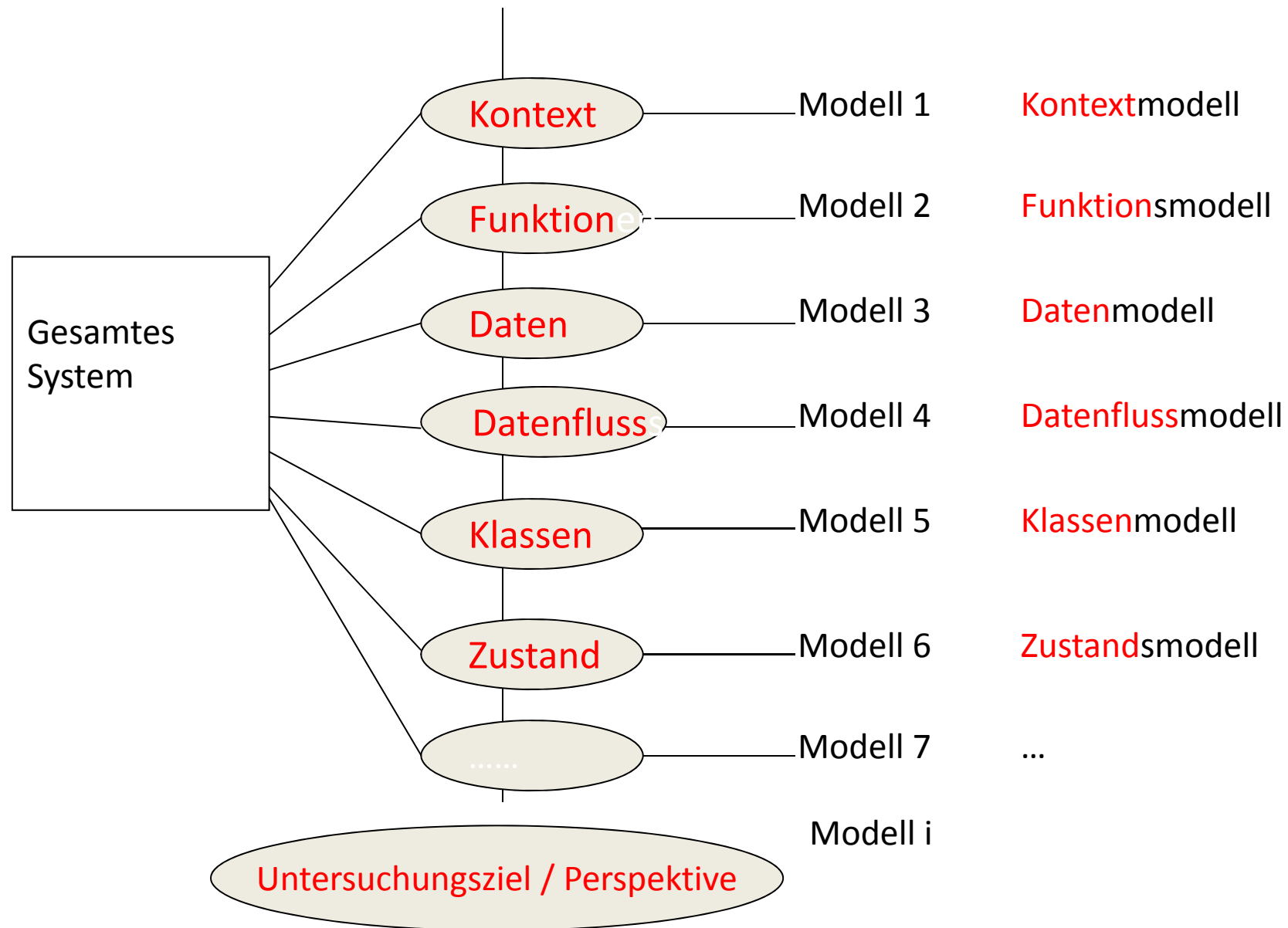
## → VDI 3633 Blatt 1: Modell ( VDI = Verein Deutscher Ingenieure)

---

Ein Modell ist eine vereinfachte **Nachbildung** eines geplanten oder real existierenden Originalsystems mit seinen Prozessen in einem anderen begrifflichen oder gegenständlichen System.

Es unterscheidet sich hinsichtlich der untersuchungsrelevanten Eigenschaften nur innerhalb eines **vom Untersuchungsziel abhängigen** Toleranzrahmens vom Vorbild.

Grundsätze des Modellierens  
www. ....



Was kennzeichnet ein System?

Ein System besteht aus Komponenten, die miteinander in Beziehung stehen.

Eine Grenze trennt das System von seiner Umgebung (auch Kontext genannt).

Schnittstellen (Verbindungsstellen) verbinden das System mit seiner Umwelt.

Ein System kann Subsysteme enthalten. Ein System befindet sich zum Zeitpunkt  $t$  in einem def. Zustand.

Warum sind Software-Systeme ganz spezielle Systeme?

SW-Systeme sind „immateriell“ und komplex; sie haben keine natürliche Lokalität.

SW-Systeme sind aus einem Werkstoff hergestellt, der von sich aus keine Strukturierung im „Großen“ erfordert.

Welchen Unterschied gibt es zwischen Modellen und Prototypen bei der Software-Entwicklung?

Modelle bilden ab; Prototypen sind lauffähig.

Welche Grundprinzipien finden in der SW-Entwicklung Anwendung?

Was kennzeichnet ein System?

Ein System besteht aus Komponenten, die miteinander in Beziehung stehen.  
Eine Grenze trennt das System von seiner Umgebung (auch Kontext genannt).  
Schnittstellen (Verbindungsstellen) verbinden das System mit seiner Umwelt.  
Ein System kann Subsysteme enthalten. Ein System befindet sich zum Zeitpunkt  $t$  in einem def. Zustand.

Warum sind Software-Systeme ganz spezielle Systeme?

SW-Systeme sind „immateriell“ und komplex; sie haben keine natürliche Lokalität.  
SW-Systeme sind aus einem Werkstoff hergestellt, der von sich aus keine  
Strukturierung im „Großen“ erfordert.

Welchen Unterschied gibt es zwischen Modellen und Prototypen bei der Software-Entwicklung?

Modelle bilden ab; Prototypen sind lauffähig.

Welche Grundprinzipien finden in der SW-Entwicklung Anwendung?

# Grundprinzipien der SW-Entwicklung

Abstraktion

Strukturierung

Zerlegung

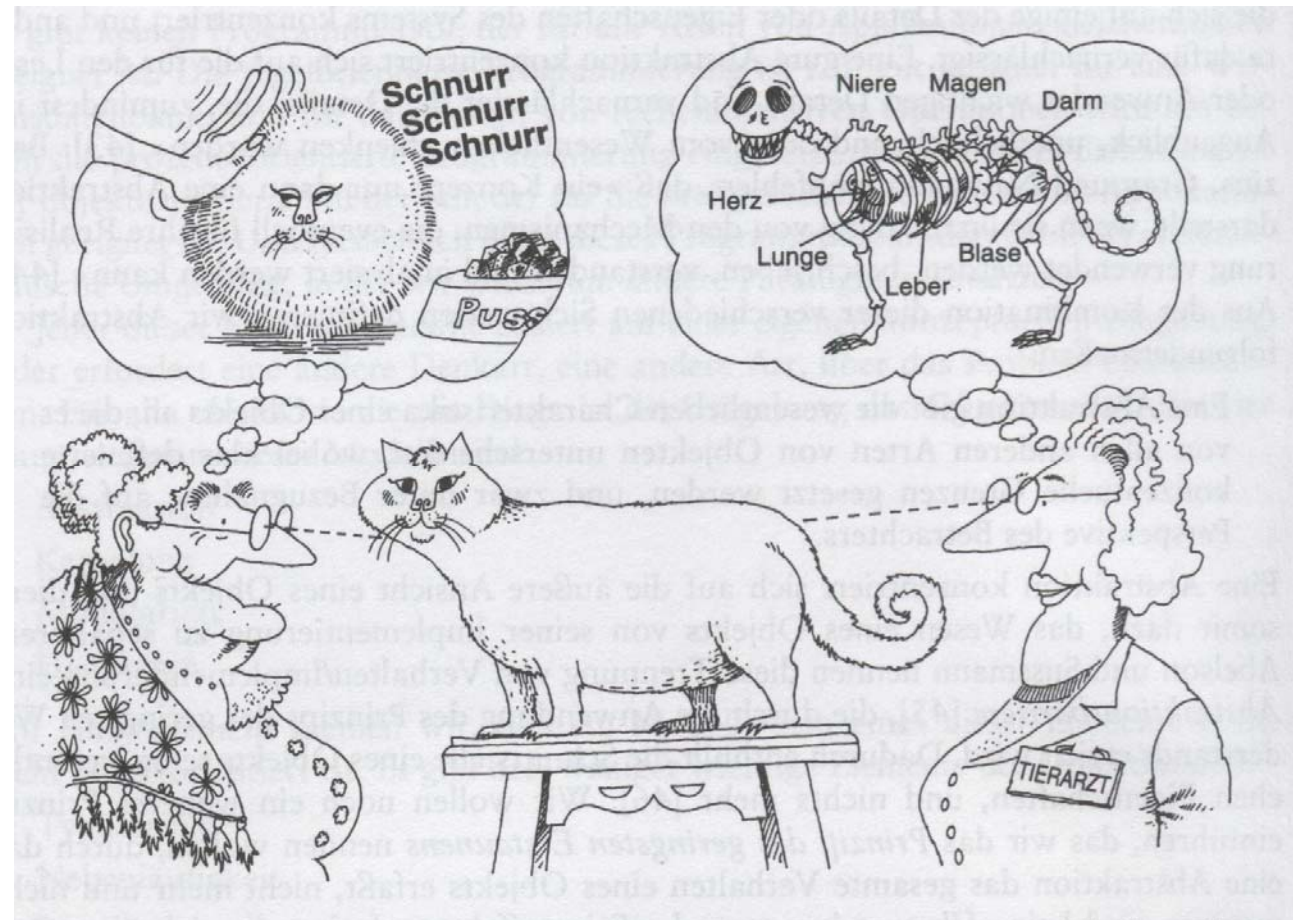
Kapselung

Hierarchisierung

Standardisierung

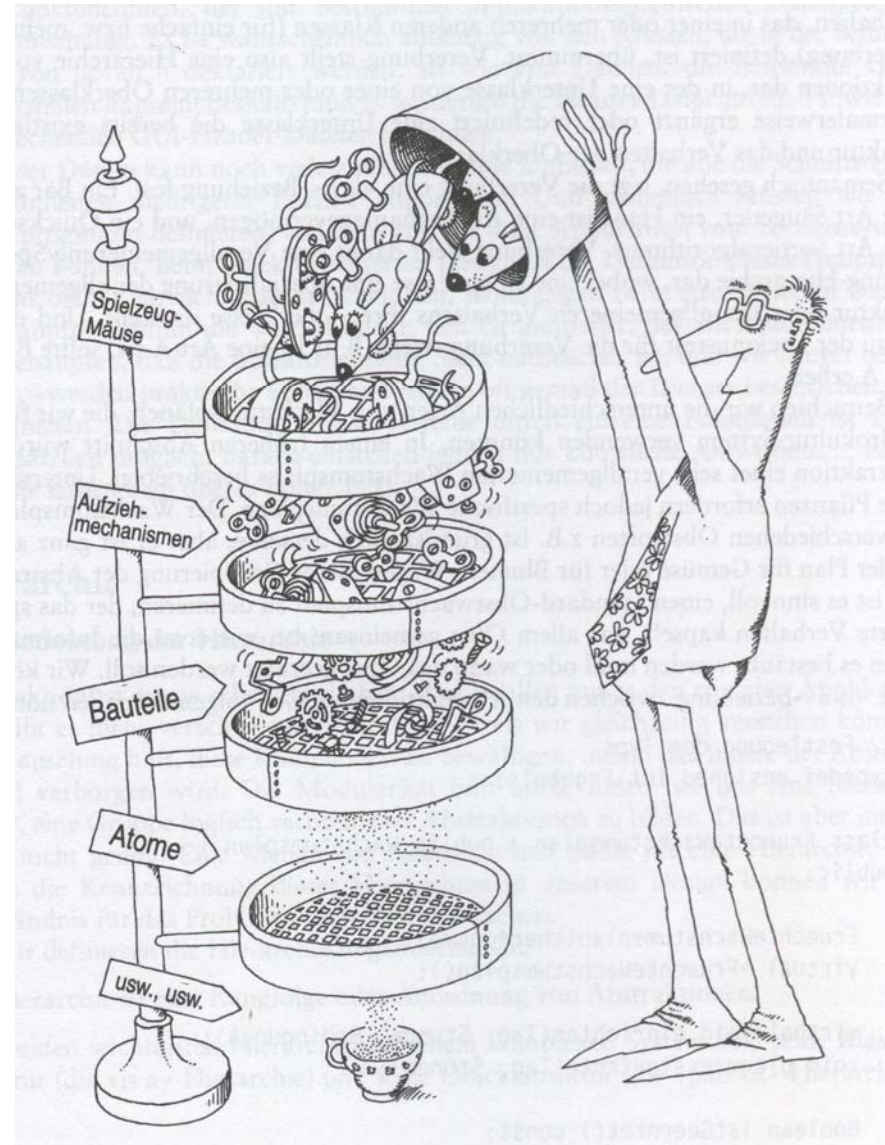
(integrierte) Dokumentation

## Abstraktion



Booch, Grady „Objektorientierte Analyse und Design“ Addison Wesley, S. 62

## Hierarchisierung

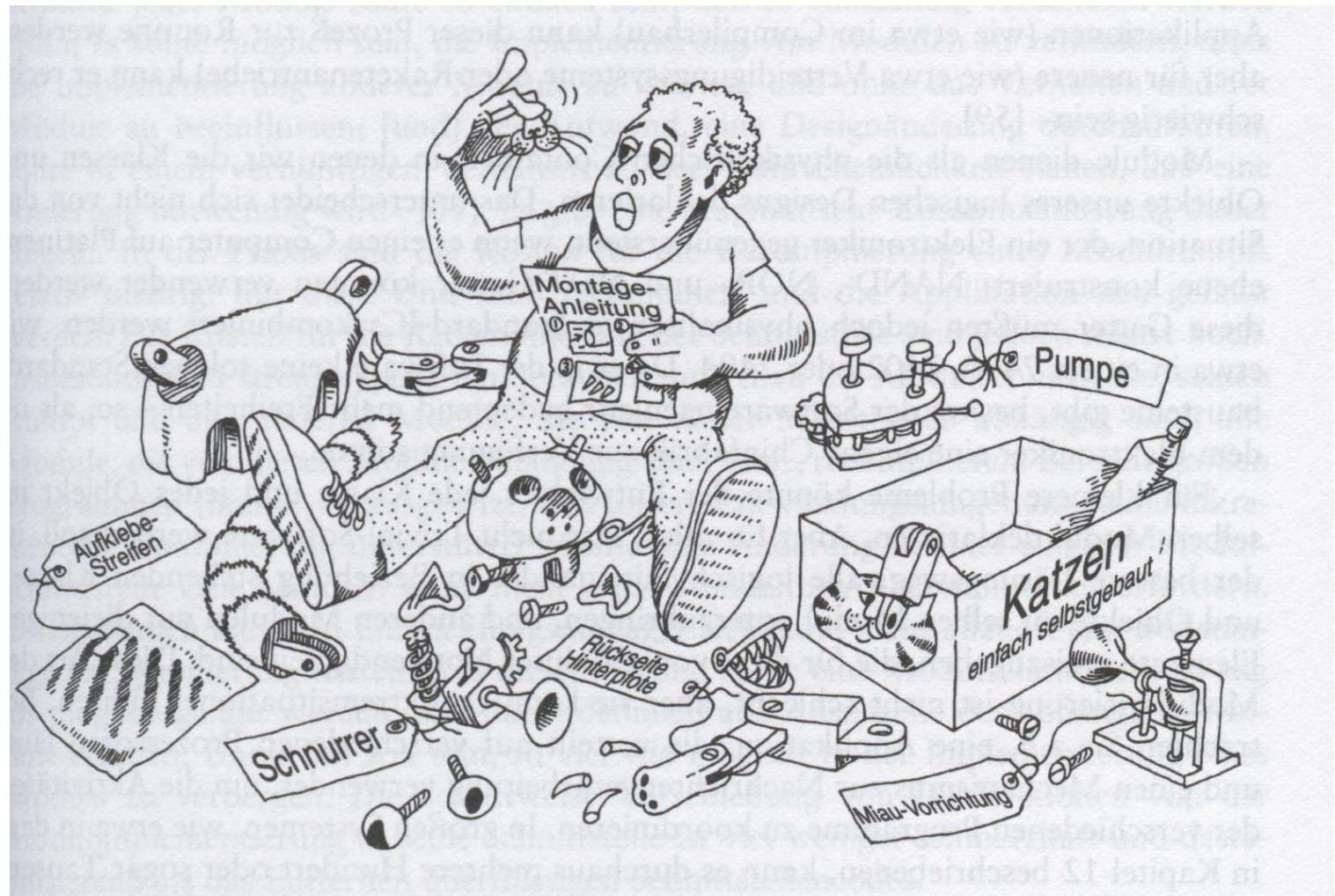


Booch, Grady „Objektorientierte Analyse und Design“ Addison Wesley, S. 84



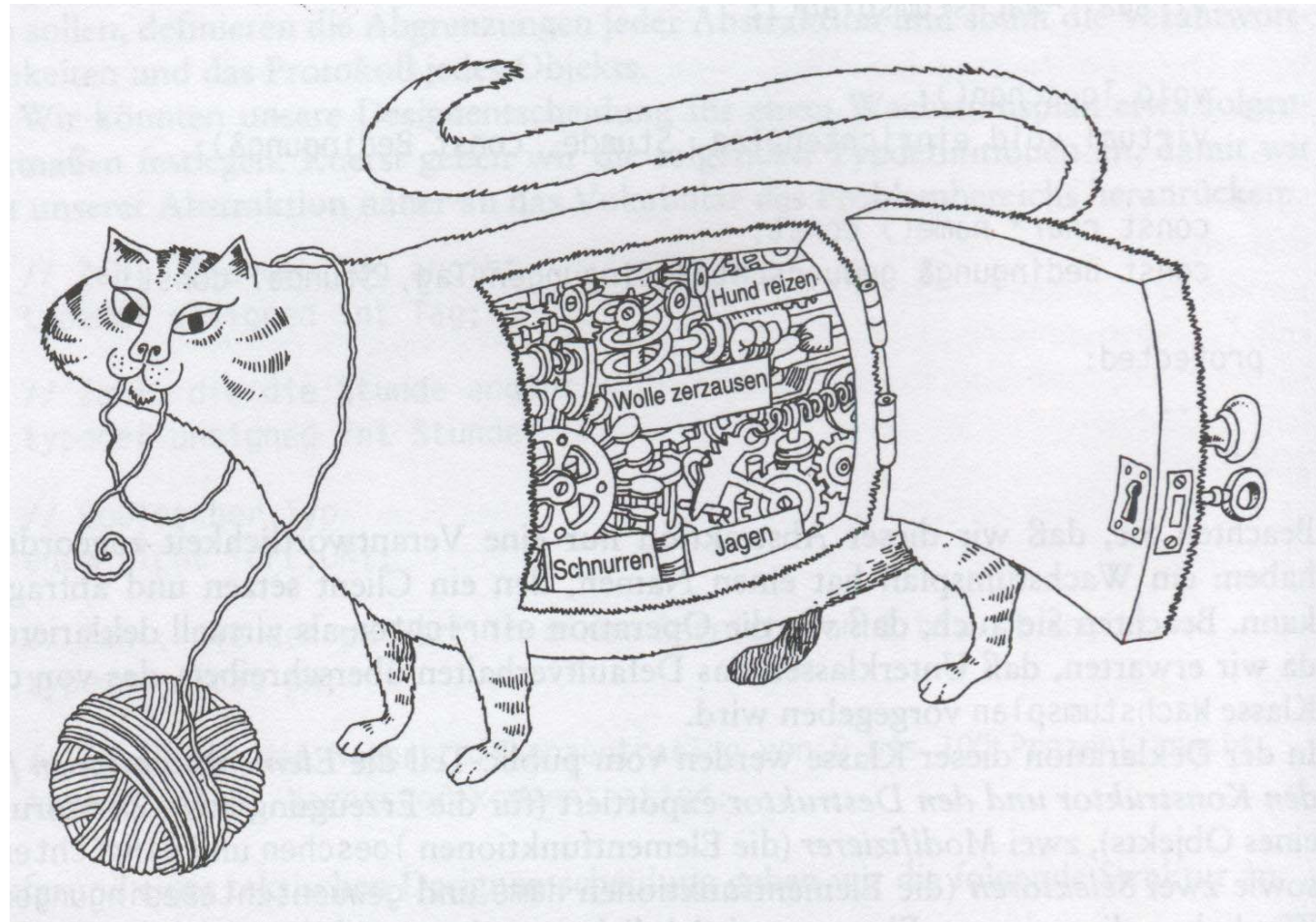
Zerlegung

Zusammen-  
setzung



Booch, Grady „Objektorientierte Analyse und Design“ Addison Wesley, S. 77

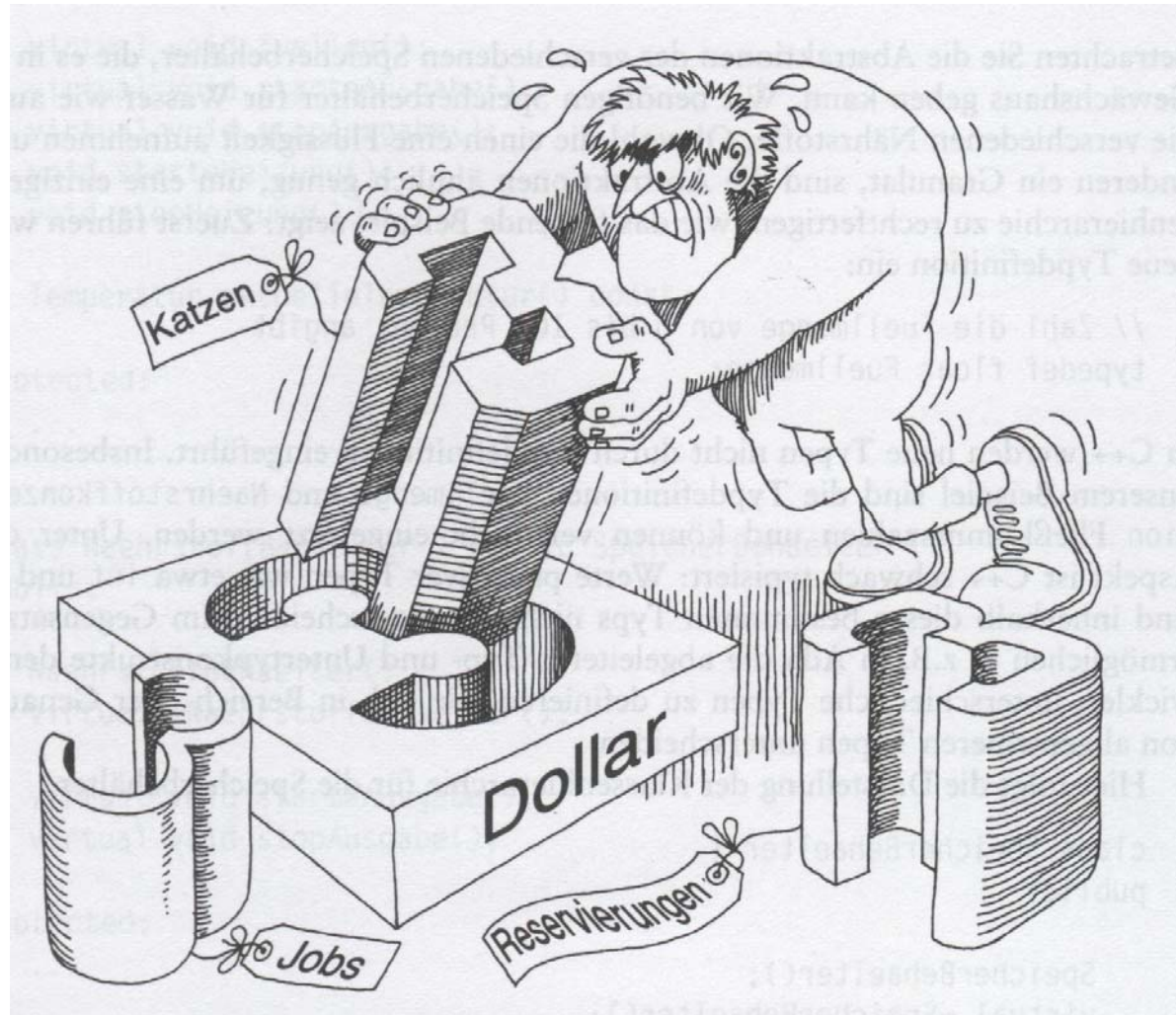
## Kapselung



Booch, Grady „Objektorientierte Analyse und Design“ Addison Wesley, S. 7

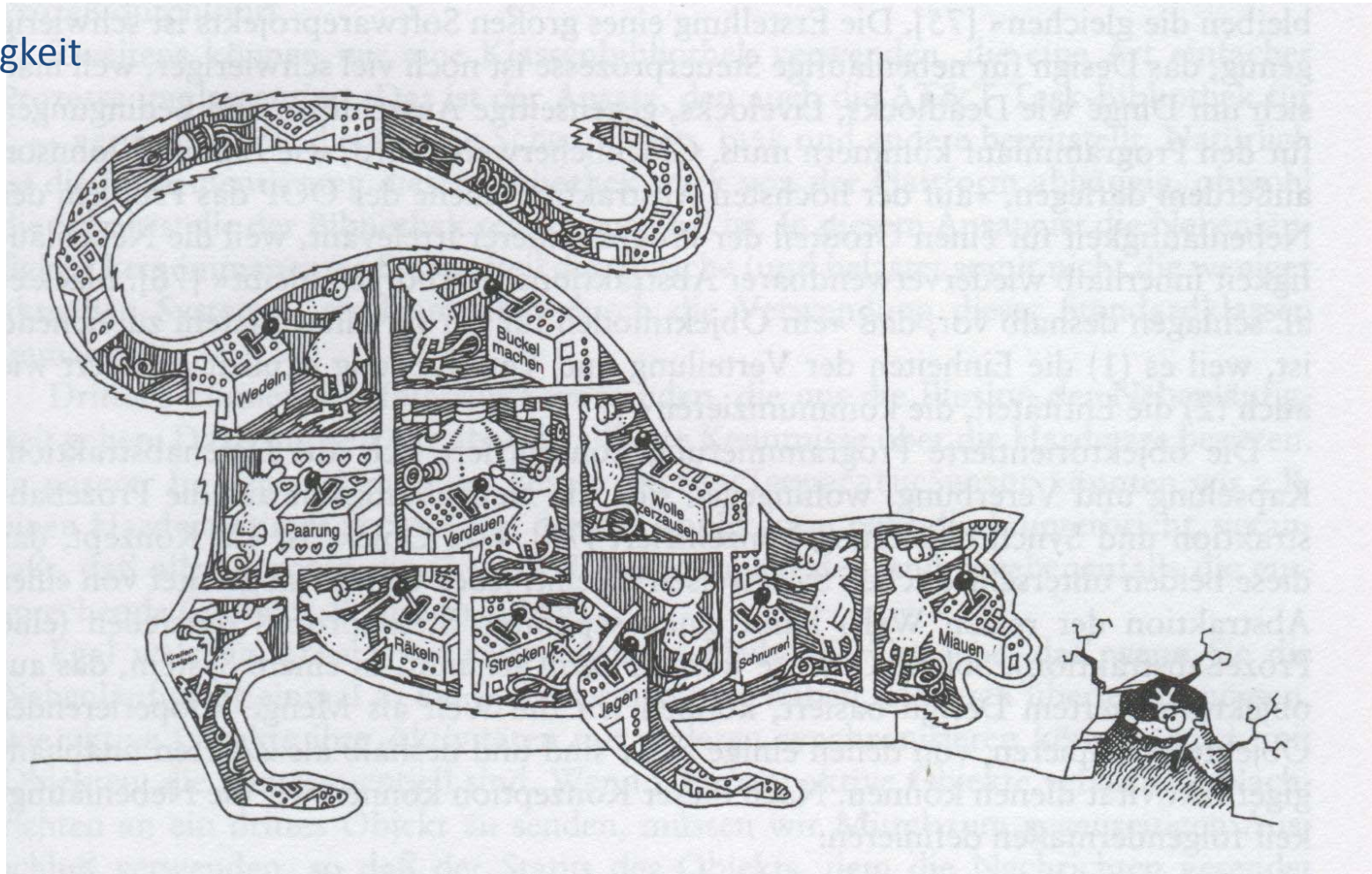


## Strenge Typisierung



Booch, Grady „Objektorientierte Analyse und Design“ Addison Wesley, S. 91

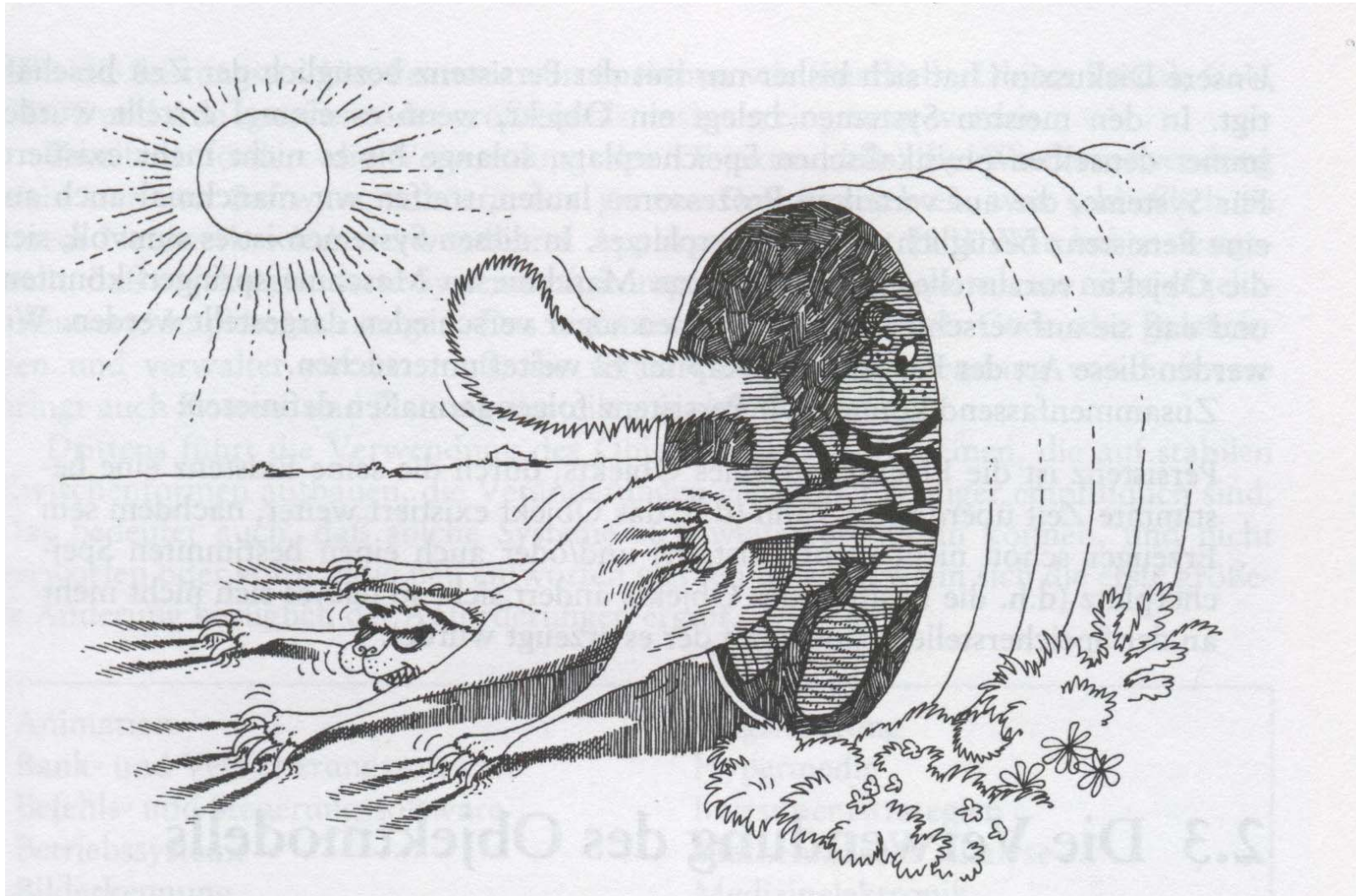
## Nebenläufigkeit



Booch, Grady „Objektorientierte Analyse und Design“ Addison Wesley, S. 99

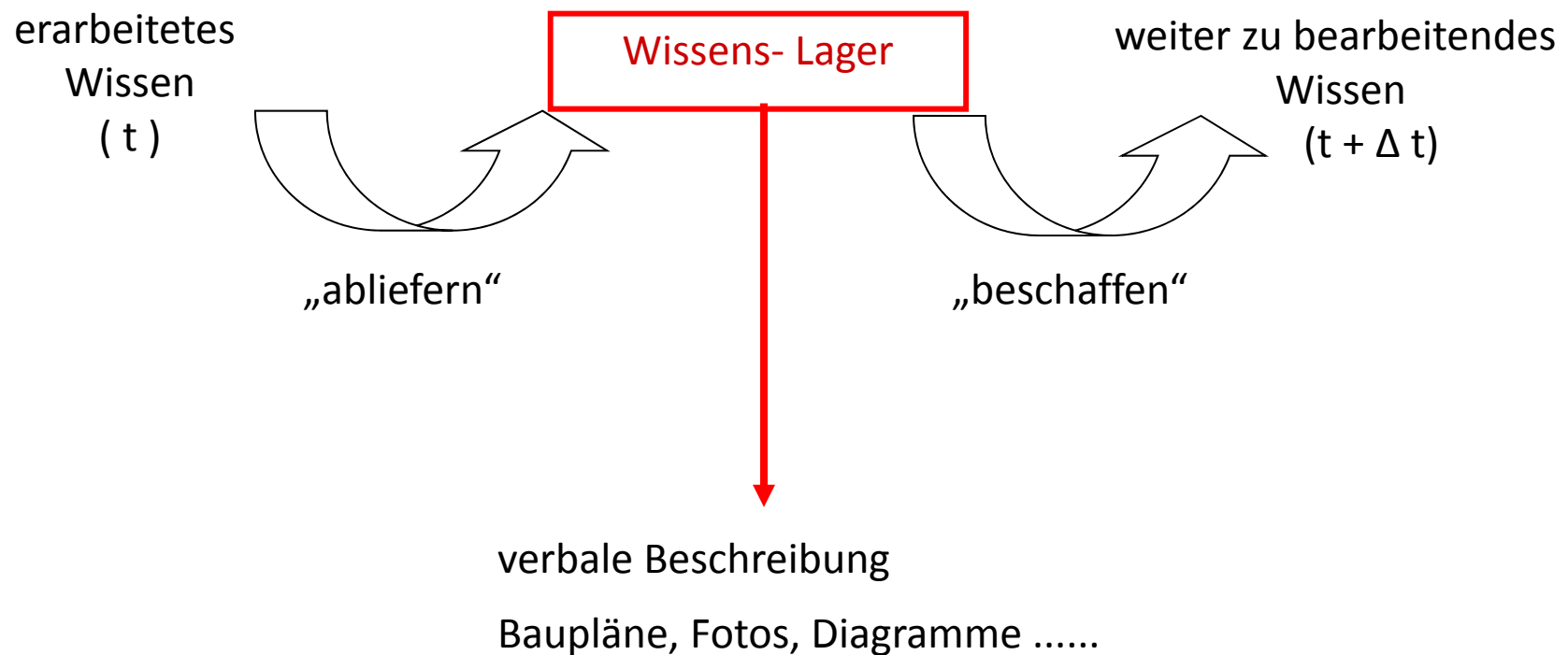


## Persistenz



Booch, Grady „Objektorientierte Analyse und Design“ Addison Wesley, S. 103

# integrierte Dokumentation → Weitergabe von Wissens



Was kennzeichnet ein System?

Ein System besteht aus Komponenten, die miteinander in Beziehung stehen.  
Eine Grenze trennt das System von seiner Umgebung (auch Kontext genannt).  
Schnittstellen (Verbindungsstellen) verbinden das System mit seiner Umwelt.  
Ein System kann Subsysteme enthalten. Ein System befindet sich zum Zeitpunkt  $t$  in einem def. Zustand.

Warum sind Software-Systeme ganz spezielle Systeme?

SW-Systeme sind „immateriell“ und komplex; sie haben keine natürliche Lokalität.  
SW-Systeme sind aus einem Werkstoff hergestellt, der von sich aus keine  
Strukturierung im „Großen“ erfordert.

Welchen Unterschied gibt es zwischen Modellen und Prototypen bei der Software-Entwicklung?

Modelle bilden ab; Prototypen sind lauffähig.

Welche Grundprinzipien finden in der SW-Entwicklung Anwendung?

Abstraktion, Strukturierung, Zerlegung, Kapselung, Hierarchisierung,  
Standardisierung, integrierte Dokumentation

Danke für Ihre Aufmerksamkeit und Ihre Mitarbeit.