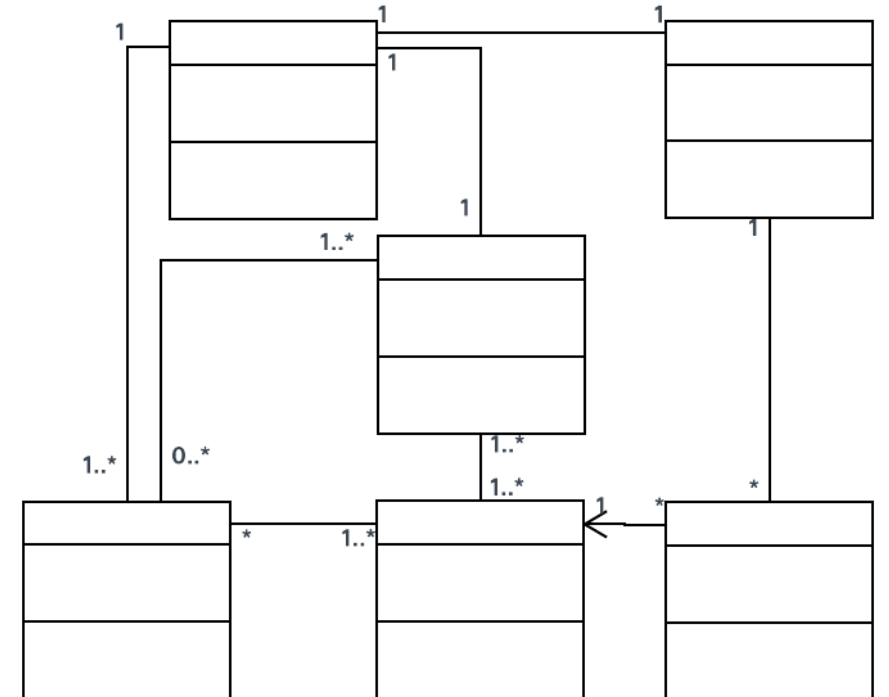


UML-Klassendiagramme – Eine Einführung



Was ist UML?

- UML steht für *Unified Modeling Language*
- Standardisierte grafische Sprache zur Modellierung von Software
- Verwendet zur Visualisierung, Spezifikation, Konstruktion und Dokumentation
- Enthält 14 verschiedene Diagrammtypen (z. B. Anwendungsfall-, Sequenz- und Klassendiagramme)

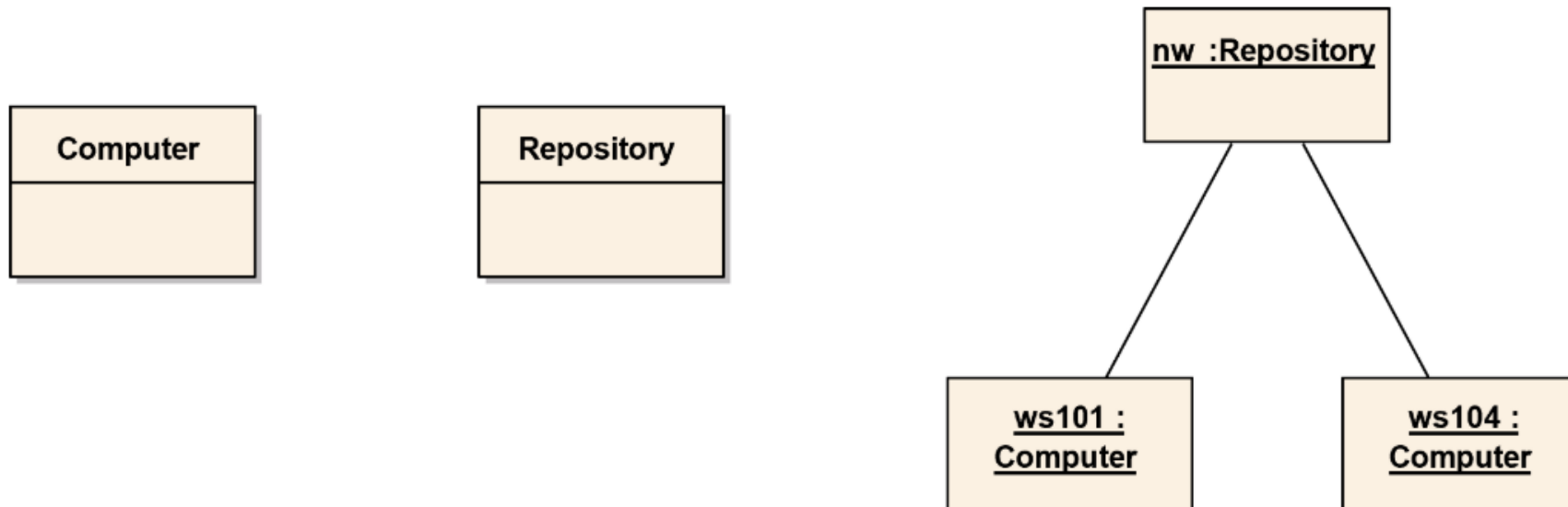
Warum UML?

- Klare Kommunikation zwischen Entwicklern, Kunden und Stakeholdern
- Bessere Strukturierung von Software-Projekten
- Erleichtert Wartung und Erweiterung von Software
- Unabhängig von Programmiersprachen nutzbar

UML-Diagrammtypen (Kurzübersicht)

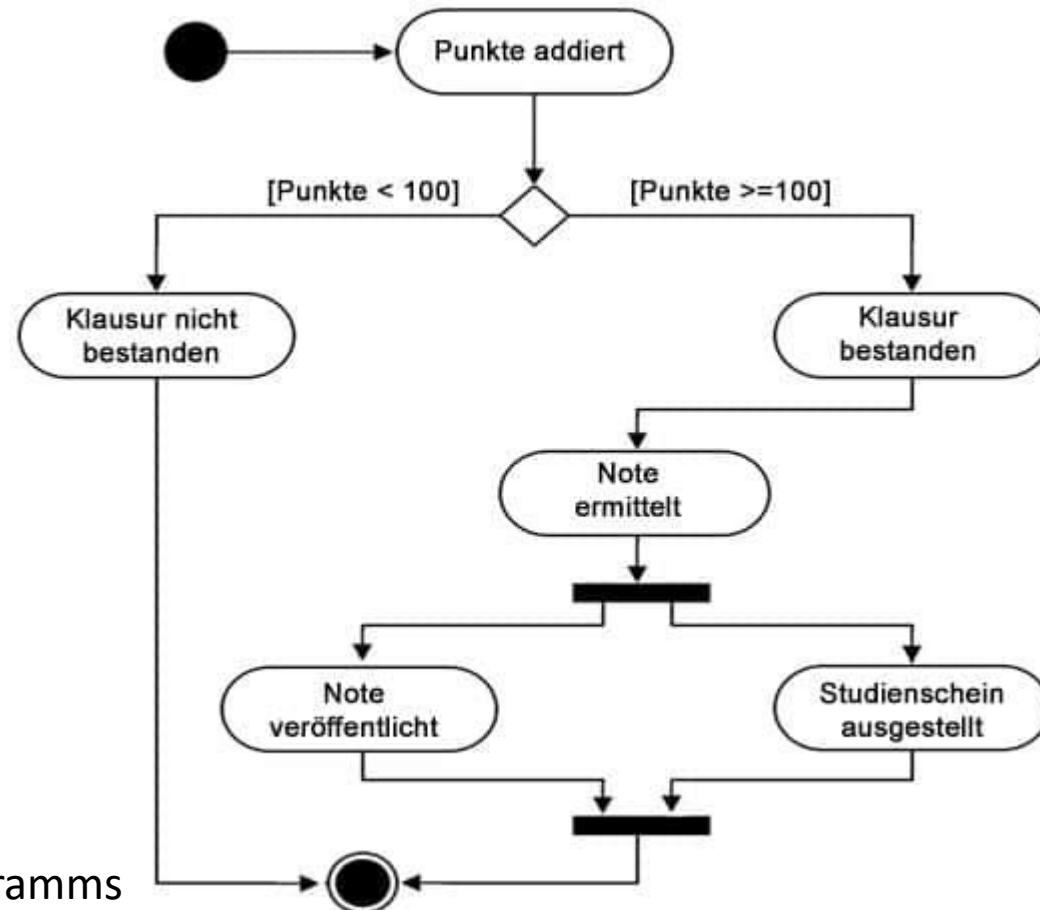
Strukturdiagramme: Klassendiagramm, Objekt-, Paket-, Komponentendiagramme

Im folgenden **Beispiel** ist links ein Klassendiagramm abgebildet und rechts das zugehörige Objektdiagramm.



UML-Diagrammtypen (Kurzübersicht)

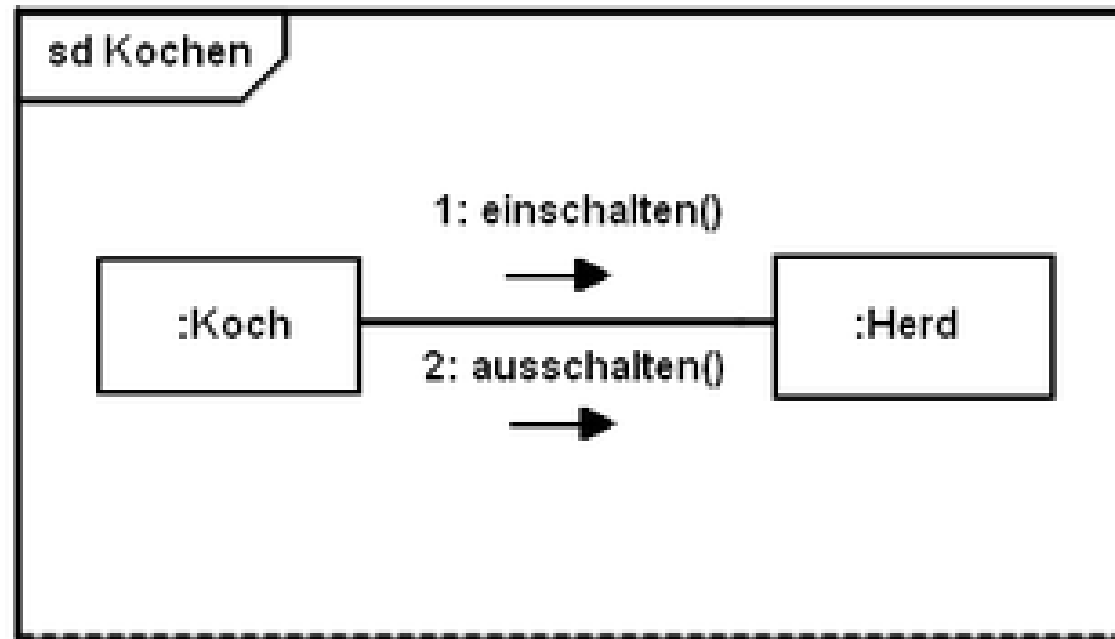
Verhaltensdiagramme: Anwendungsfall-, Aktivitäts-, Zustandsdiagramme



Beispiel eines Verhaltensdiagramms

UML-Diagrammtypen (Kurzübersicht)

Interaktionsdiagramme: Sequenz-, Kommunikations-, Zeitverlaufdiagramme

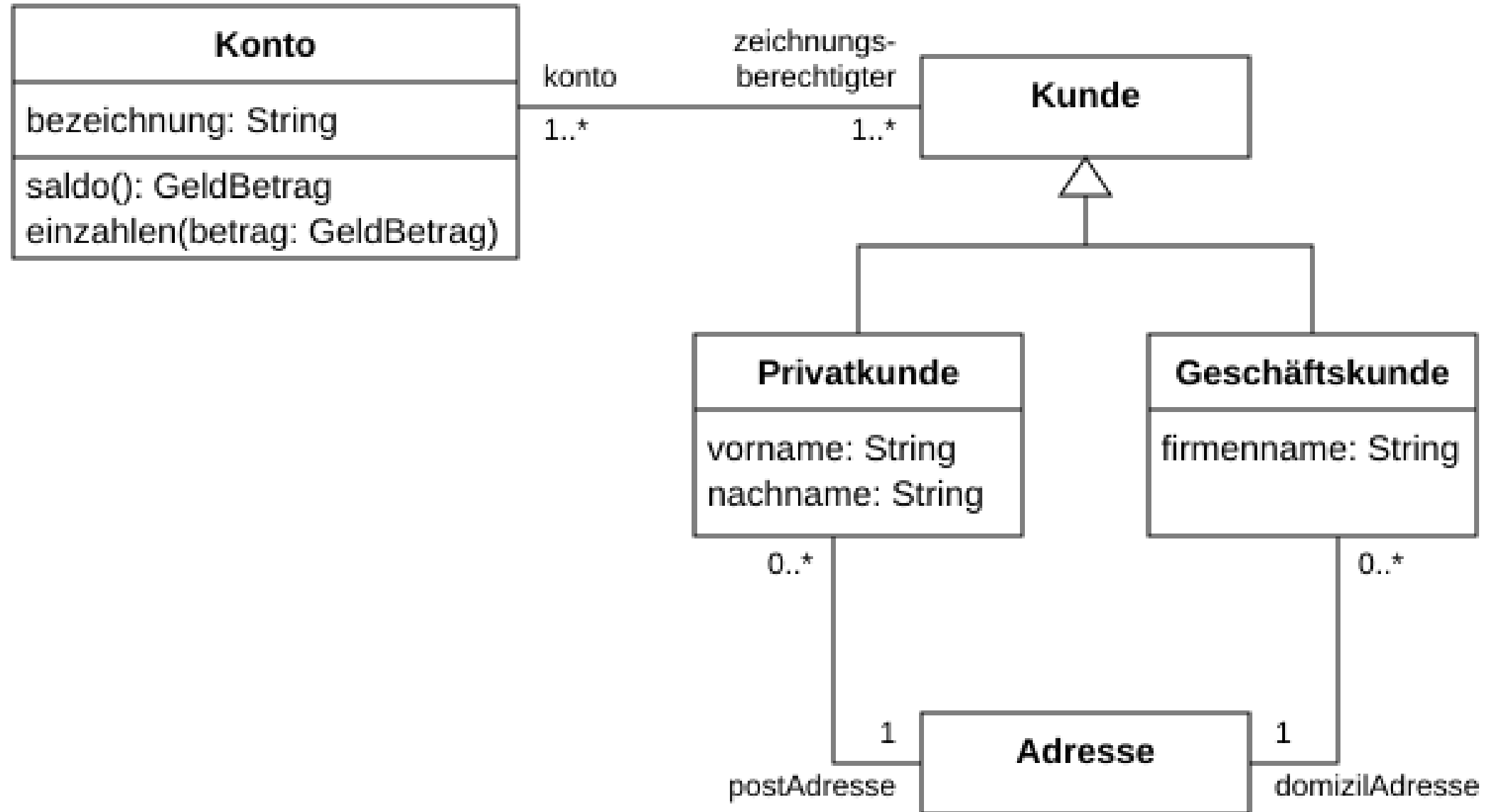


Beispiel eines Kommunikationsdiagrammes

Klassendiagramme

Klassendiagramme

Beispiel eines Klassendiagramms

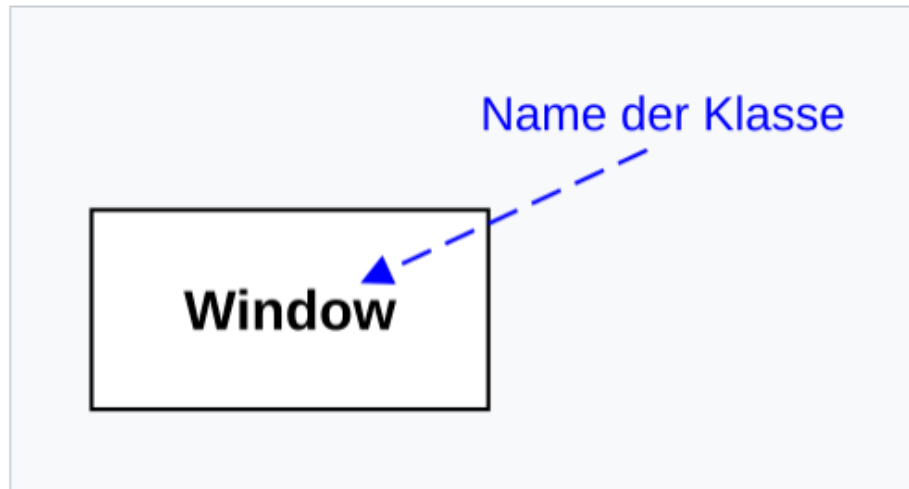


Klassendiagramme

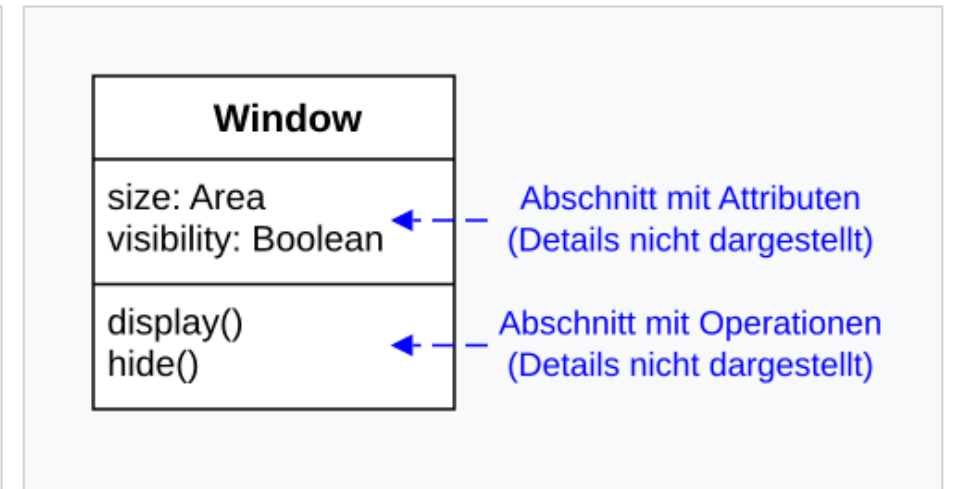
- Modellieren die statische Struktur einer Software
- Bestehen aus Klassen, Attributen, Methoden und Beziehungen
- Grundlage für objektorientierte Programmierung

Klassendiagramme: Aufbau einer Klasse

- Eine Klasse besteht aus:
 - **Namen** (z. B. *Person*)
 - **Attribute** (Eigenschaften, z. B. *name*, *alter*)
 - **Operationen** (Funktionalität, z. B. *sprechen()*, *laufen()*)
- Notation:



Einfachste Form der Darstellung für eine Klasse



Zusätzliche Darstellung von Attributen und Operationen [4]

Klassendiagramme: Sichtbarkeiten

- Die Sichtbarkeit von Attributen und Methoden/Operationen wird wie folgt gekennzeichnet:
 - „+“ für public – (engl. öffentlich), unbeschränkter Zugriff
 - „#“ für protected – (engl. geschützt), Zugriff nur von der Klasse sowie von Unterklassen (Klassen, die erben)
 - „-“ für private – (engl. privat), nur die Klasse selbst kann es sehen
 - „~“ für package – (engl. Paket), innerhalb des Pakets sichtbar (nur in wenigen Programmiersprachen, etwa Java und C#, implementierbar)

Klassendiagramme: Erste Übung

Aufgabe: Identifiziere Klassen aus folgender Aufgabenstellung:
"Ein Hotelverwaltungssystem soll Buchungen, Zimmer und Kunden verwalten."

- Welche Klassen lassen sich daraus ableiten?
- Welche Attribute und Methoden könnte jede Klasse haben?
- Skizziere ein einfaches Klassendiagramm(Ohne Beziehungen!)

Quellen

- [1] <https://www.sparxsystems.de/sprachen/uml/diagramme/objektdiagramm>
- [2] <https://t2informatik.de/wissen-kompakt/verhaltensdiagramm/>
- [3] [https://de.wikipedia.org/wiki/Kommunikationsdiagramm \(UML\)](https://de.wikipedia.org/wiki/Kommunikationsdiagramm_(UML))
- [4] <https://de.wikipedia.org/wiki/Klassendiagramm>