

Abbildung – *Spezialisierung* kann auf mehrere Arten in Tabellen implementiert werden.

Bekannt – Dabei sind die *Superklasse* und die *Subklasse* vollständig getrennte Tabellen

Eine Tabelle – Alle *Subklassen* sind in die *Superklasse* integriert, sowie ein weiteres Attribut, welches angibt welche *Subklassen* „Region“ verwendet wird.

Eine Tabelle (mehrere Flags) – Anstelle eines Attributs wird ein Flagattribut für jede *Subklasse* verwendet. So kann ein Datensatz mehrere *Subklassen* verwenden.

Keine Superklasse – Die Attribute der *Superklasse* sind alle jeweils in den *Subklassen* vorhanden.

Attribut – Eigenschaft einer Entität.

Abgeleitetes Attribut – Attribut, welches von einem oder mehreren anderen berechnet wurde. Ähnlich zu: *Zusammengesetztes Attribut*

Mehrwertiges Attribut – Attribut welches mehrere Werte gleichzeitig beinhalten kann.

Schlüsselattribut – Attribut welches als Primärschlüssel verwendet werden kann. Muss Einzigartig in Tabelle sein.

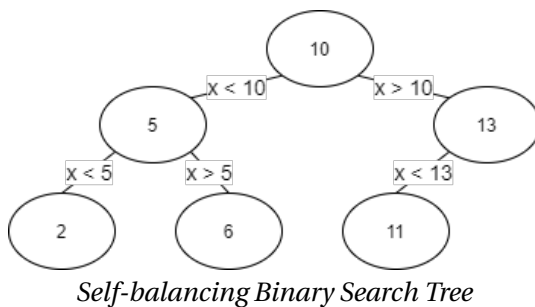
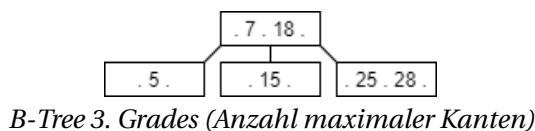
Teilattribut – Teil eines *Zusammengesetzten Attribut*

Zusammengesetztes Attribut – Attribut bestehend aus mehreren *Teilattributen*. Ähnlich zu: *Abgeleitetes Attribut*



Ausgeglichene Bäume – siehe *Balanced Trees*

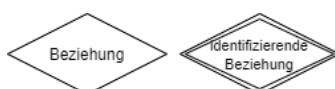
Balanced Trees – *Search trees* verschiedener Ausführungen, deren Sucheffektivität durch eine ausgeglichene Kantenlänge. Sammelbegriff für mehrere Arten von *Trees*.



Beziehung – Beziehung zwischen zwei Tabellen.

Identifizierende Beziehung – Beziehung einer Tabelle, dessen Einträge von einem Eintrag in einer anderen Tabelle abhängen.

Nicht-identifizierende Beziehung – Beziehungen zwischen 2 Tabellen, welche jedoch nicht voneinander abhängen.



Crow's Foot Notation – Kardinalitätsschreibweise, siehe *Kardinalität*

Datenmodell – Datenmodelle sind Darstellungen welche verschiedene Aspekte von Daten und dazugehörigen Datenbanken visualisieren. Dies entweder zur Dokumentation oder Planung.

Konzeptionelles Datenmodell – Grobe Darstellung einer Datenbank, unabhängig der Implementierung. Meist als *ER-Modell*, teilweise als *UML-Diagramm* dargestellt. Beinhaltet den Namen und Beziehungen von *Entitäten*, sowie die Namen der *Attribute* und die Primärschlüssel.

Logisches Datenmodell – Konzeptionelle Abbildung der systemabhängigen Datenbankstruktur. Beinhaltet den Namen der *Tabellen*, *Spalten*, den Primär- und Fremdschlüssel, sowie die Datentypen der Spalten.

Physisches Datenmodell – Genauere, systemabhängige Darstellung mit weiteren Informationen zur Implementation.

Datentypen – Datentypen von Datenbanken. (n) bezeichnet die definierbare Bytelänge. (m) die Anzahl Nachkommastellen.

blob (n) – Binary Large Object

boolean – Wahrheitswert (TRUE | FALSE)

char (n) – Zeichenkette von n-Länge

clob (n) – Character Large Object der Länge n

date – Datum ohne Zeit

decimal (n, m) – Festkommazahl mit n Stellen und m Nachkommastellen

double – Gleitkommazahl

float (m) – Gleitkommazahl mit m Nachkommastellen

integer – Ganze Zahl

numeric (n, m) – Festkommazahl mit n Stellen und m Nachkommastellen

real – Gleitkommazahl

text – Zeichenkette

time – Zeitangabe (evtl. mit Zeitzone)

timestamp – Datum und Uhrzeit (evtl. mit Zeitzone), kann bis Mikrosekundengenauigkeit aufweisen.

varchar (n) – auch character varying (n). Zeichenkette von maximaler Länge (n)

Disjunktheit – siehe *Einschränkungen*

Disjoint – siehe *Disjunktheit*

Einschränkungen – Der Grad der *Spezialisierung* kann in vier Kategorien eingeteilt werden. Die Vererbung wird im *konzeptionellen Datenmodell* durch eine Verbindung der *Superklasse* mit den *Subklassen* zu einem Kreis dargestellt. Im Kreis ist ein Buchstabe für die Kategorisierung: ○: *Overlap*, ⊙: *Disjoint*.

Abbildung – siehe *Abbildung*

Partiell disjunkt – Partielle *Spezialisierung*. Einigen *Superklassen* ist keine *Subklasse* zugeordnet.

Partiell überlappend – Ein *Superklassen*-Wert kann gleichzeitig von mehreren *Subklassen* verwendet werden.

Total disjunkt – Totale *Spezialisierung*. Jeder *Superklasse* ist eine *Subklasse* zugewiesen. Bei der Darstellung

ist eine doppelte Linie zwischen \textcircled{d} und der *Superklassen*-Entität verwendet.

Total überlappend – *Superklassen*-Werte werden gleichzeitig von mehreren *Subklassen* verwendet.

Entität – Eine Einheit, analog zu einem Objekt oder einer Klasse.

Schwache Entität – Entität, welche nicht allein durch dessen Attribute identifizierbar sind. (Benötigt verbundene Tabellen für die Identifikation.)



ER-Modell – *Entity-Relationship-Model*. Darstellung von *Datenmodellen* mit *Entitäten*, *Attributen*, *Beziehungen* und *Kardinalitäten*. (siehe jeweilige Einträge)

Generalisierung – siehe *Spezialisierung / Generalisierung*

Kardinalität – Einordnung der Vielzahligkeit der Verbindungsseite.

1 – Die gegenüberliegende *Entität* bezieht sich auf genau eine *Entität* der Seite der Kardinalitätsangabe. (*UML*: 1, *Crow's Foot*: horizontale Linie mit einem oder zwei vertikalen Strichen)

c – Die gegenüberliegende *Entität* bezieht sich auf genau eine oder gar keine *Entität* der Seite der Kardinalitätsangabe. (*UML*: 0..1, *Crow's Foot*: [innen] gefüllter Kreis und [ausen] vertikaler Strich auf horizontaler Linie)

m – Die gegenüberliegende *Entität* bezieht sich auf eine oder mehrere *Entitäten* der Seite der Kardinalitätsangabe. (*UML*: 1..*, *Crow's Foot*: (falls explizit auch einfach: [innen] vertikaler Strich und) [ausen] „<“ (falls *Entität* rechts) auf horizontaler Linie)

mc – Die gegenüberliegende *Entität* bezieht sich auf keine, eine oder mehrere *Entitäten* der Seite der Kardinalitätsangabe. (*UML*: 0..*, *Crow's Foot*: [innen] gefüllter Kreis und [ausen] „<“ (falls *Entität* rechts) auf horizontaler Linie)

Klassendiagramm – siehe *UML*

Overlap – siehe *Überlappend*

Planung – Dies ist ein vorgeschlagener Ablauf für die Implementation einer Datenbank.

1. Analyse aus Anwendersicht – Erste Skizzen, Gespräch mit Kunden. Enthält die Namen und Beziehungen von *Entitäten*.

2. Entwurf eines Datenmodells – Entwurf des *konzeptionellen Datenmodells* (*ER-Modell*). Definition der *Entitäten*, *Attribute*, *Beziehungen*, *Schlüssel*, *Kardinalitäten*

3. Entwurf des physischen Datenmodells – Systemabhängige Definition der *Tabellen*, *Attribute*, *Datentypen*, *Einschränkungen*, *Schlüssel*, *Indexe*, (*UML*)

4. Realisierung der Datenbank – Programmierung der DB und Views, Datenimport, Tests

Spezialisierung / Generalisierung – Aufteilen einer *Superklasse* um besser auf die jeweiligen *Subklassen* zu passen, respektive umgekehrter Prozess. Dargestellt in Verbindung: $Sub \in Super$

Attributdefiniert – Anstelle *prädikatdefiniert* wird ein Attribut in der *Superklasse* verwendet um die Kategorie eines Eintrages zu definieren. Dies umgeht allfällige Probleme mit ausgeschöpften Kategorien.

Benutzerdefiniert – Alle Entitätentypen werden in eigener Tabelle gespeichert. Der aufgeteilte Datensatz verwendet in der *Sub*- und *Superklasse* denselben Primärschlüssel. So sind in der *Superklasse* alle Schlüssel vorhanden. Ein Primärschlüssel ist in der Gesamtmenge der *Subklassen* nur in jeweils einer vorhanden, respektive in mehreren, falls mehrere Eigenschaften auf das Tupel zutreffen. (z.B. Bahnhof ist auch Bushaltestelle)

Effektive Abbildung – siehe *Abbildung*

Einschränkungen – siehe *Einschränkungen*

Prädikatdefiniert – Die Einteilung der Kategorisierung erfolgt über Wertbereiche des Primärschlüssels. (z.B. 0-999: Typ A, 1'000-1'999: Typ B)

Subklasse – siehe *Vererbung*

Superklasse – siehe *Vererbung*

UML – *Unified Modeling Language*.

Klassendiagramm – Darstellung welche für *logische* und *physische Datenmodelle* verwendet werden kann. Darstellung der *Kardinalitäten* siehe denselben.

Tabellenname
Primärschlüssel (Datentyp)
Fremdschlüssel (Datentyp)
Attribut (Datentyp)

Überlappend – siehe *Einschränkungen*

Vererbung – Datensätze, respektive Tabellen können Teilweise auf mehrere andere Datensätze angewendet werden. (z.B. Adresse) Diese können entsprechend vererbt werden.

Subklasse – Die Klasse, welche auf die *Superklasse* zugeht. (z.B. Lieferant mit zugeordneter Adresse)

Superklasse – Klasse, welche auf mehrere *Subklassen* angewendet wird. (z.B. Adresse)

[This space is intentionally left blank.]

Dieser Spick wurde ursprünglich im Februar 2021 verfasst. Im September 2021 wurde das Dokument leicht überarbeitet um auf flueck.weebly.com/gibb hochzuladen. Es wurden keine fachlichen Fehler korrigiert.