1 – Analytische Datenbanken („Normal“ transaktional)

|  |  |  |
| --- | --- | --- |
| Anfragen | Transaktional | Analytisch |
| Fokus | Read/Write/Update/Delete | Read, periodic write |
| Transaktionsdauer/Typ | Short read/write operation | Lange Leseoperationen |
| Anfragestruktur | Einfach strukturiert | Komplex |
| Datenvolumen/Anfrage | Wenige Datensätze | Viele Datensätze |
| Datenmodell | Anfragenflexible | auswertungsbezogen |

Datenqualität:Glaubwürdig, Nützlich, Interpretierbar

Datenqualitätsmängel: Dopplung, Unterschiedliche Einheiten, Inkonsistenz

Snowflakeschema(Normal[isiert]) vs Starschema (Faktentabelle + Dimensionstabellen)

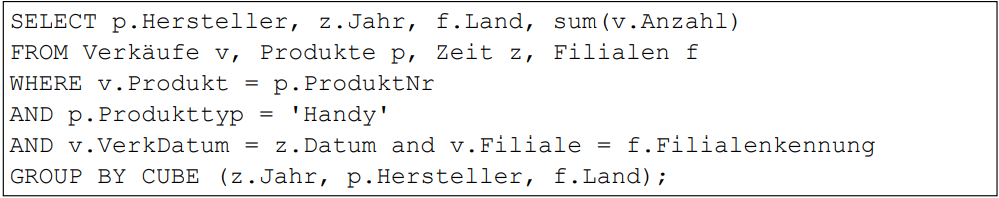
Snowflake: Langsam

Starschema: Schnell, dafür Einfüge/Änderungs/- und Löschanomalien

Roll-Up: Bei SELECT von Daten **wenige** Attribute in GROUP BY -> Daten verdichtet

Drill-Down: Bei SELECT von Daten **viele** Attribute in GROUP BY -> **hoher** detailgrad

Beispiel: Starschema, Faktentabelle Verkäufe wird mit Dimensionstabellen Produkt/Zeit/Filliale verbunden



**create** **Table** Verkauf\_2DC

**AS**

**SELECT** **COALESCE**(p.produkt\_gruppe,'alle') **as** Gruppe, **COALESCE**(TO\_CHAR(k.kunde\_alter), 'alle') **as** "ALTER", **SUM**(v.verkauf\_Anzahl) **as** anzahl

**FROM** Kunde k, Produkt p, Verkauf v

**WHERE** v.verkauf\_datum **>=** TO\_DATE('2020-01-01', 'YYYY-MM-DD')

**AND** v.produkt\_ID **=** p.Produkt\_ID

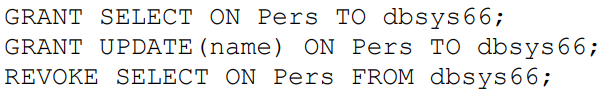
**AND** v.kunde\_ID **=** k.Kunde\_ID

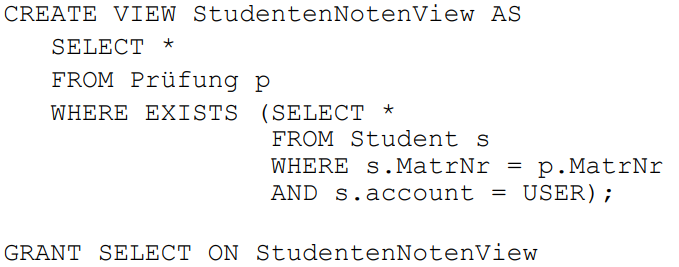
**GROUP** **BY** **CUBE** (p.produkt\_gruppe, k.kunde\_alter);

Decode als Alternative zu Coalesce

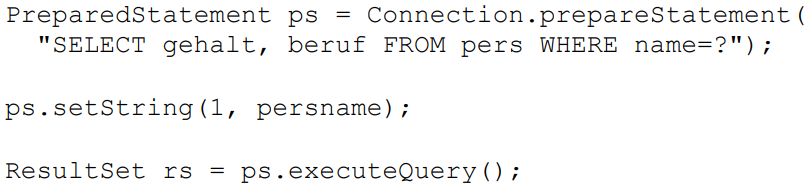
DECODE(GROUPING(region), 1, 'All Regions', 0, region)) AS Region

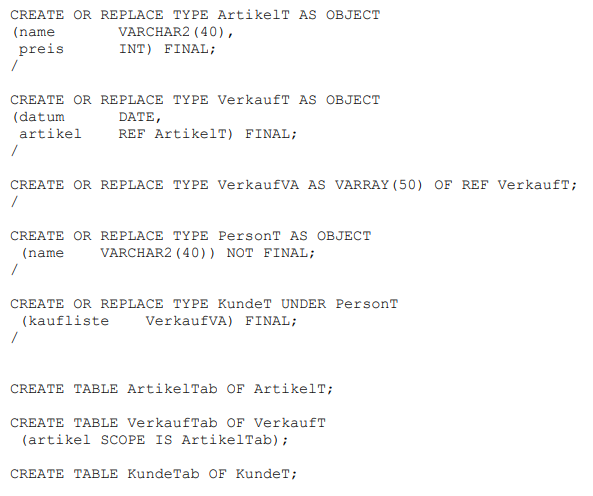
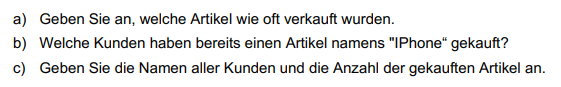
Ersteller von SQL Tabellen hat immer alle Rechte -> Software muss mit anderem Benutzer zugreifen

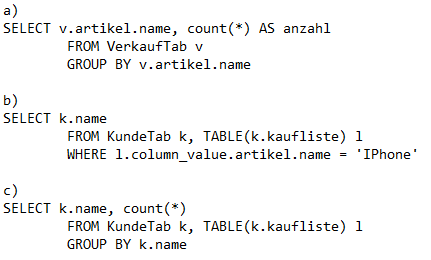


Workaround um Rechte nur auf eine Zeile zu geben:  


Prepared Statements, Input Validierung, Authentifizierung





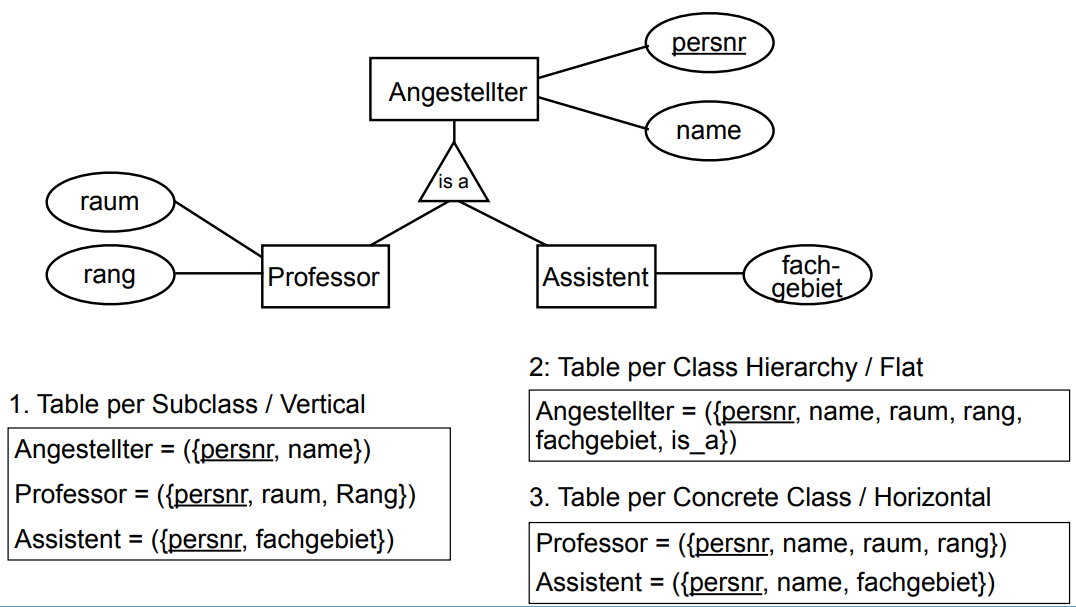


Kapitel 4

OR-Mapper: Seperation of concerns

Factory Method: Genaue Objekt Instanz durch Factory gekapselt, kann ausgetauscht werden

Abbildung der Vererbung, 3 Möglichkeiten:



Identitymap: Keeps track of Objects that have been loaded from db. If object is loaded again the object already in memory is used.

Metadaten für Mapping als xml Datei oder Annotation im Java Code

Vorgehensweise Hibernate: Topw-down/Bottom-up/Middle-out/Meet in the middle

Kapitel 5

Data Dictionary: Speicherung aller Informationen, die zur Verwaltung der Objekte in der Datenbank benötigt werden

– Schema-Informationen, also Tabellen, Attribute, etc.

– Statistiken

– Verwendeter Speicherplatzbedarf

– Name von Datenbank Benutzern, Rollen, Zugriffsrechte

TAB – SELECT \* FROM Tab; – Alle eigenen Tabellen, Views

USER\_TABLES – Tabellen eines Benutzers

USER\_TAB\_COLUMNS – Attributes einer Tabellen eines Benutzers

USER\_TRIGGERS – Trigger eines Benutzers

USER\_TAB\_PRIVS – Zugriffsrechte auf eine Tabelle eines Benutzers

USER\_VIEWS – Views eines Benutzers