

# Grundlagen von Datenbanken

---

Probeklausur

Normalisierung, XPath



# Normalisierung

## Änderungsanomalien

Bücherei ( KID, Signatur, KName, ISBN, Autor, Titel, Auflage, Bestand, Ausleihdatum, Leihdauer )

*KID, Signatur → Ausleihdatum, Leihdauer*

*KID → KName*

*Signatur → ISBN*

*ISBN → Autor, Titel*

*Autor, Titel → ISBN*

*ISBN, Auflage → Bestand*

*Autor, Titel, Auflage → Bestand*

KID	Signatur	KNname	ISBN	Autor	Titel	Auflage	Bestand	Ausleihdatum	Leihdauer
14	1234-35	Meier	3540421335	Härder	Datenbanksysteme	1	5	29.01.07	30
2	4711-12	Müller	1558601902	Gray	Transaction Processing	1	12	01.07.06	365
2	0815-13	Müller	3540421335	Härder	Datenbanksysteme	2	10	01.07.06	365
7	9999-99	Schmidt	0130384887	Tanenbaum	Computer Networks	1	10	02.01.07	30
	...	...	...	...	...	...	...	...	...

# Normalisierung

---

## Änderungsanomalien

**Einfüge-Anomalie:**

Es ist nicht möglich ein neues Buch einzufügen,  
wenn dieses nicht von einem Kunden ausgeliehen wird.

ODER

Es ist nicht möglich, einen Kunden anzulegen ohne  
dass dieser ein Buch ausleiht.

# Normalisierung

---

## Änderungsanomalien

**Lösch-Anomalie:**

ODER

Wenn der Kunde das letzte Buch zurückgibt, gehen auch die Kundeninformationen verloren.

Mit dem Löschen eines Kunden gehen auch die Informationen über ein Buch verloren, wenn dies von keinem anderen Kunden ausgeliehen ist.

# Normalisierung

---

## Änderungsanomalien

**Modifikations-Anomalie:** Um den Titel eines Buches zu korrigieren, müssen die Tupel für alle Leihverhältnisse desselben Buches aktualisiert werden.

ODER:

Ändert sich der Name eines Kunden, müssen die Tupel für alle Ausleihverhältnisse des Kunden aktualisiert werden.

# Normalisierung

## Normalformen

- Funktionale Abhängigkeiten:  
(die Werte auf X bestimmen die Werte auf Y eindeutig)

$$X \rightarrow Y$$

$$\forall t_1, t_2 \in R: t_1[X] = t_2[X] \Rightarrow t_1[Y] = t_2[Y]$$

*KID, Signatur  $\rightarrow$  Ausleihdatum, Leihdauer*

*KID  $\rightarrow$  KName*

*Signatur  $\rightarrow$  ISBN*

*ISBN  $\rightarrow$  Autor, Titel*

*Autor, Titel  $\rightarrow$  ISBN*

*ISBN, Auflage  $\rightarrow$  Bestand*

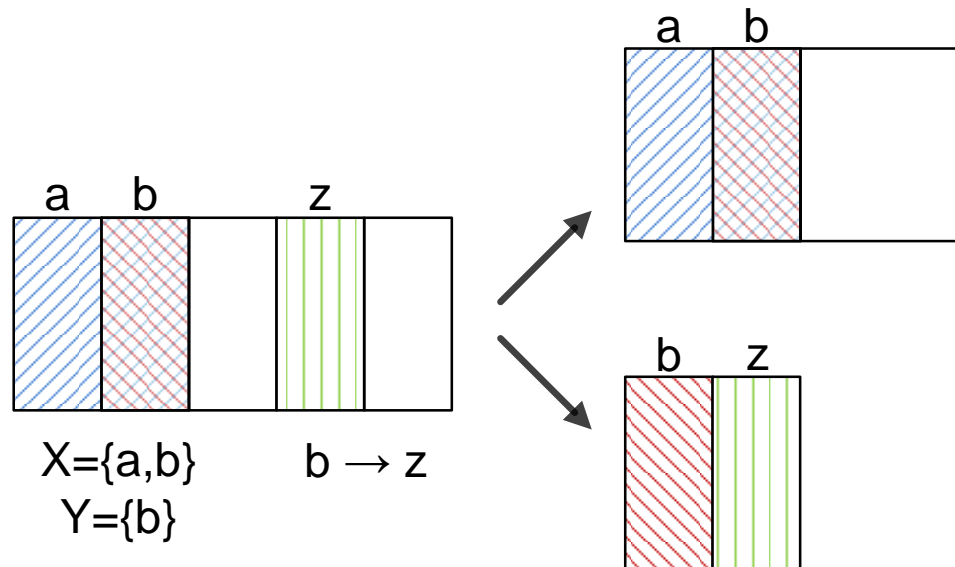
*Autor, Titel, Auflage  $\rightarrow$  Bestand*

# Normalisierung

## Normalformen

2. NF    kein Nicht-Primärattribut ist partiell von einem Schlüsselkandidaten abhängig.

*NPA*  $z$ , Schlüsselkandidat  $X$                        $\nexists Y : (Y \subset X \wedge Y \rightarrow z)$



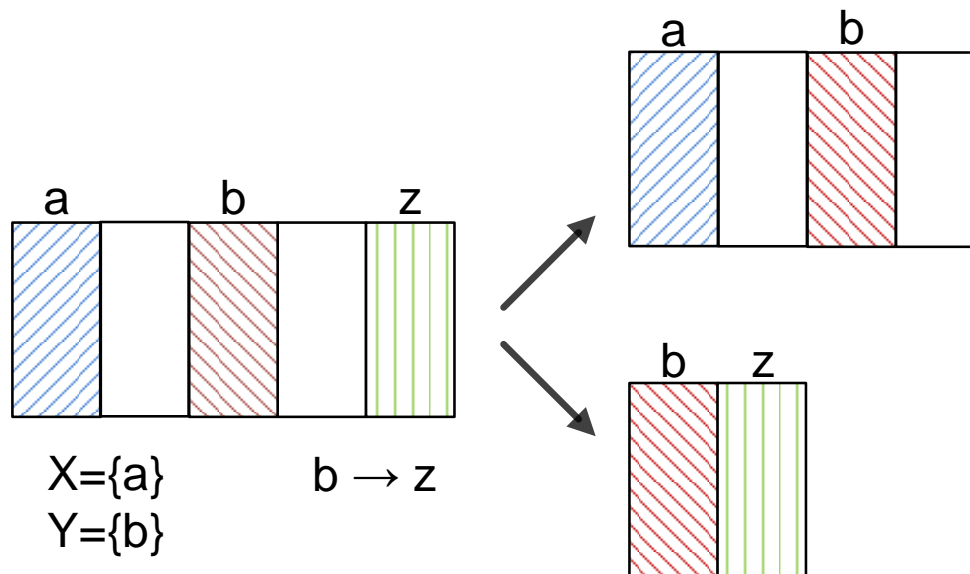
# Normalisierung

## Normalformen

3. NF kein Nicht-Primärattribut ist transitiv von einem Schlüsselkandidaten abhängig.

*NPA z, Schlüsselkandidat X*

$$\nexists Y : (X \rightarrow Y \wedge \neg(Y \rightarrow X) \wedge Y \rightarrow z \wedge z \notin Y)$$





# Normalisierung

---

**R1(Name, Straße, Datum, Treffpunkt)**

**a) Fkt. Abhängigkeiten**

**FA1:** Name  $\rightarrow$  Straße

**FA2:** Name, Datum  $\rightarrow$  Treffpunkt

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

a) Fkt. Abhängigkeiten

FA1: Name  $\rightarrow$  Straße

FA2: Name, Datum  $\rightarrow$  Treffpunkt

Schlüsselkandidaten:

Name, Datum

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

a) Fkt. Abhängigkeiten

FA1: Name  $\rightarrow$  Straße

FA2: Name, Datum  $\rightarrow$  Treffpunkt

Schlüsselkandidaten:

Name, Datum

Nicht-Primärattribute:

Straße, Treffpunkt

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

a) Fkt. Abhängigkeiten

FA1: Name → Straße

FA2: Name, Datum → Treffpunkt

Schlüsselkandidaten:

Name, Datum

Nicht-Primärattribute:

Straße, Treffpunkt

Normalformen:

1NF, aber da Straße partiell von Name, Datum abhängt ist die Relation bzgl. dieser Menge an funktionellen Abhängigkeiten nicht in 2NF. Folglich auch nicht in 3NF.

# Normalisierung

---

**R1(Name, Straße, Datum, Treffpunkt)**

**b) Fkt. Abhängigkeiten**

**FA1:** Name, Straße, Datum → Treffpunkt

# Normalisierung

---

**R1(Name, Straße, Datum, Treffpunkt)**

**b) Fkt. Abhängigkeiten**

**FA1:** Name, Straße, Datum → Treffpunkt

**Schlüsselkandidaten:**

Name, Straße, Datum

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

b) Fkt. Abhängigkeiten

FA1: Name, Straße, Datum → Treffpunkt

Schlüsselkandidaten:

Name, Straße, Datum

Nicht-Primärattribute:

Treffpunkt

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

**b) Fkt. Abhängigkeiten**

**FA1:** Name, Straße, Datum  $\rightarrow$  Treffpunkt

**Schlüsselkandidaten:**

Name, Straße, Datum

**Nicht-Primärattribute:**

Treffpunkt

**Normalformen:**

Da nur eine funktionale Abhängigkeit existiert, kann es weder partielle noch transitive Abhängigkeiten geben. Die Relation ist bzgl. dieser Menge an funktionellen Abhängigkeiten somit in 3NF.



# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

**c) Fkt. Abhängigkeiten**

**FA1:** Name, Straße  $\rightarrow$  Datum, Treffpunkt

**FA2:** Datum, Treffpunkt  $\rightarrow$  Name, Straße

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

c) Fkt. Abhängigkeiten

FA1: Name, Straße → Datum, Treffpunkt

FA2: Datum, Treffpunkt → Name, Straße

Schlüsselkandidaten:

Name, Straße oder Datum, Treffpunkt

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

**c) Fkt. Abhängigkeiten**

**FA1:** Name, Straße  $\rightarrow$  Datum, Treffpunkt

**FA2:** Datum, Treffpunkt  $\rightarrow$  Name, Straße

**Schlüsselkandidaten:**

Name, Straße oder Datum, Treffpunkt

**Nicht-Primärattribute:**



# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

**c) Fkt. Abhängigkeiten**

**FA1:** Name, Straße  $\rightarrow$  Datum, Treffpunkt

**FA2:** Datum, Treffpunkt  $\rightarrow$  Name, Straße

**Schlüsselkandidaten:**

Name, Straße oder Datum, Treffpunkt

**Nicht-Primärattribute:**

∅

**Normalformen:**

Da es keine Nicht-Primärattribute gibt, muss die Relation bzgl. dieser Menge an funktionellen Abhängigkeiten in 3NF sein.

# Normalisierung

---

**R1(Name, Straße, Datum, Treffpunkt)**

**d) Fkt. Abhängigkeiten**

**FA1:** Name, Straße  $\rightarrow$  Datum, Treffpunkt

**FA2:** Datum  $\rightarrow$  Treffpunkt

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

d) Fkt. Abhängigkeiten

FA1: Name, Straße → Datum, Treffpunkt

FA2: Datum → Treffpunkt

Schlüsselkandidaten:

Name, Straße

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

d) Fkt. Abhängigkeiten

FA1: Name, Straße → Datum, Treffpunkt

FA2: Datum → Treffpunkt

Schlüsselkandidaten:

Name, Straße

Nicht-Primärattribute:

Datum, Treffpunkt

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

d) Fkt. Abhängigkeiten

FA1: Name, Straße  $\rightarrow$  Datum, Treffpunkt

FA2: Datum  $\rightarrow$  Treffpunkt

Schlüsselkandidaten:

Name, Straße

Nicht-Primärattribute:

Datum, Treffpunkt

Normalformen:

Es treten keine partiellen Abhängigkeiten auf. Die Relation ist somit in 2NF. Da **Treffpunkt** aber über **Datum** transitiv von **Name, Straße** abhängt, ist die Relation bzgl. dieser Menge an funktionellen Abhängigkeiten nicht in 3NF.



# Normalisierung

---

**R1(Name, Straße, Datum, Treffpunkt)**

**e) Fkt. Abhängigkeiten**

**FA1:** Name  $\rightarrow$  Datum

**FA2:** Datum, Treffpunkt  $\rightarrow$  Straße

**FA3:** Straße  $\rightarrow$  Name

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

### e) Fkt. Abhängigkeiten

FA1: Name  $\rightarrow$  Datum

FA2: Datum, Treffpunkt  $\rightarrow$  Straße

FA3: Straße  $\rightarrow$  Name

### Schlüsselkandidaten:

Datum, Treffpunkt

oder Name, Treffpunkt

oder Straße, Treffpunkt

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

### e) Fkt. Abhängigkeiten

FA1: Name → Datum

FA2: Datum, Treffpunkt → Straße

FA3: Straße → Name

### Schlüsselkandidaten:

Datum, Treffpunkt

oder Name, Treffpunkt

oder Straße, Treffpunkt

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

### e) Fkt. Abhängigkeiten

FA1: Name  $\rightarrow$  Datum

FA2: Datum, Treffpunkt  $\rightarrow$  Straße

FA3: Straße  $\rightarrow$  Name

### Schlüsselkandidaten:

Datum, Treffpunkt  
oder Name, Treffpunkt  
oder Straße, Treffpunkt

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

### e) Fkt. Abhängigkeiten

**FA1:** Name  $\rightarrow$  Datum

**FA2:** Datum, Treffpunkt  $\rightarrow$  Straße

**FA3:** Straße  $\rightarrow$  Name

### Schlüsselkandidaten:

Datum, Treffpunkt  
oder Name, Treffpunkt  
oder Straße, Treffpunkt

### Nicht-Primärattribute:

∅

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

**e) Fkt. Abhängigkeiten**

**FA1:** Name  $\rightarrow$  Datum

**FA2:** Datum, Treffpunkt  $\rightarrow$  Straße

**FA3:** Straße  $\rightarrow$  Name

**Schlüsselkandidaten:**

Datum, Treffpunkt  
oder Name, Treffpunkt  
oder Straße, Treffpunkt

**Nicht-Primärattribute:**

∅

**Normalformen:**

Da es keine Nicht-Primärattribute gibt, muss die Relation bzgl. dieser Menge an funktionellen Abhängigkeiten in 3NF sein.

# Normalisierung

---

**R1(Name, Straße, Datum, Treffpunkt)**

**f) Fkt. Abhängigkeiten**

**FA1:** Name  $\rightarrow$  Straße

**FA2:** Datum  $\rightarrow$  Treffpunkt

**FA3:** Straße, Datum  $\rightarrow$  Name

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

f) Fkt. Abhängigkeiten

FA1: Name → Straße

FA2: Datum → Treffpunkt

FA3: Straße, Datum → Name

Schlüsselkandidaten:

Name, Datum oder Straße, Datum



# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

f) Fkt. Abhängigkeiten

FA1: Name  $\rightarrow$  Straße

FA2: Datum  $\rightarrow$  Treffpunkt

FA3: Straße, Datum  $\rightarrow$  Name

Schlüsselkandidaten:

Name, Datum oder Straße, Datum

Nicht-Primärattribute:

Treffpunkt

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

f) Fkt. Abhängigkeiten

FA1: Name  $\rightarrow$  Straße

FA2: Datum  $\rightarrow$  Treffpunkt

FA3: Straße, Datum  $\rightarrow$  Name

**Schlüsselkandidaten:**

Name, Datum oder Straße, Datum

**Nicht-Primärattribute:**

Treffpunkt

**Normalformen:**

1NF, aber da **Treffpunkt** partiell von **Straße, Datum** und von **Name, Datum** abhängt ist die Relation bzgl. dieser Menge an funktionellen Abhängigkeiten nicht in 2NF. Folglich auch nicht in 3NF.

# Normalisierung

---

## **R1(Name, Straße, Datum, Treffpunkt)**

**g) Fkt. Abhängigkeiten**

**FA1:** Name, Straße  $\rightarrow$  Treffpunkt

**FA2:** Datum  $\rightarrow$  Treffpunkt

**FA3:** Treffpunkt  $\rightarrow$  Datum

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

g) Fkt. Abhängigkeiten

FA1: Name, Straße → Treffpunkt

FA2: Datum → Treffpunkt

FA3: Treffpunkt → Datum

Schlüsselkandidaten:

Name, Straße

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

g) Fkt. Abhängigkeiten

FA1: Name, Straße → Treffpunkt

FA2: Datum → Treffpunkt

FA3: Treffpunkt → Datum

Schlüsselkandidaten:

Name, Straße

Nicht-Primärattribute:

Datum, Treffpunkt

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

g) Fkt. Abhängigkeiten

FA1: Name, Straße  $\rightarrow$  Treffpunkt

FA2: Datum  $\rightarrow$  Treffpunkt

FA3: Treffpunkt  $\rightarrow$  Datum

Schlüsselkandidaten:

Name, Straße

Nicht-Primärattribute:

Datum, Treffpunkt

Normalformen:

Es treten keine partiellen Abhängigkeiten auf. Die Relation ist bzgl. dieser funktionellen Abhängigkeiten somit in 2NF. Da Datum aber über Treffpunkt transitiv von Name, Straße abhängt, ist die Relation bzgl. dieser Menge an funktionellen Abhängigkeiten nicht in 3NF.

# Normalisierung

---

**R1(Name, Straße, Datum, Treffpunkt)**

**h) Fkt. Abhängigkeiten**

**FA1:** Treffpunkt  $\rightarrow$  Name

**FA2:** Name  $\rightarrow$  Datum, Straße, Treffpunkt

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

h) Fkt. Abhängigkeiten

FA1: Treffpunkt  $\rightarrow$  Name

FA2: Name  $\rightarrow$  Datum, Straße, Treffpunkt

Schlüsselkandidaten:

Treffpunkt oder Name



# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

**h) Fkt. Abhängigkeiten**

**FA1:** Treffpunkt  $\rightarrow$  Name

**FA2:** Name  $\rightarrow$  Datum, Straße, Treffpunkt

**Schlüsselkandidaten:**

Treffpunkt oder Name

**Nicht-Primärattribute:**

Straße, Datum

# Normalisierung

---

## R1(Name, Straße, Datum, Treffpunkt)

h) Fkt. Abhängigkeiten

FA1: Treffpunkt  $\rightarrow$  Name

FA2: Name  $\rightarrow$  Datum, Straße, Treffpunkt

Schlüsselkandidaten:

Treffpunkt oder Name

Nicht-Primärattribute:

Straße, Datum

Normalformen:

Da beide Schlüsselkandidaten einattributig sind, kann es keine partiellen Abhängigkeiten geben. Die Relation ist bzgl. dieser funktionellen Abhängigkeiten somit in 2NF. Da es sich bei **Treffpunkt** und **Name** jeweils um einen Schlüsselkandidaten handelt und somit **Treffpunkt  $\leftrightarrow$  Name** gilt, existieren auch keine transitiven Abhängigkeiten. Die Relation ist bzgl. dieser Menge an funktionellen Abhängigkeiten daher sogar in 3NF.

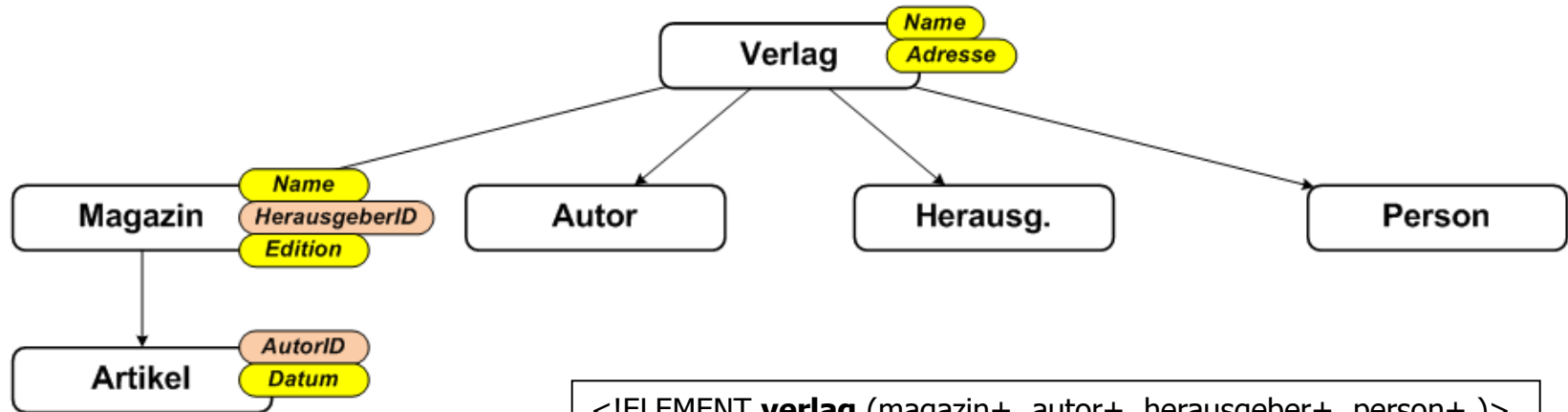
# XML und XPath

---

**Gegeben: DTD eines XML-Dokuments.**

**Tipp: Zunächst einen *Knotenbaum* mit der Struktur des XML-Dokuments erstellen**

# XML und XPath



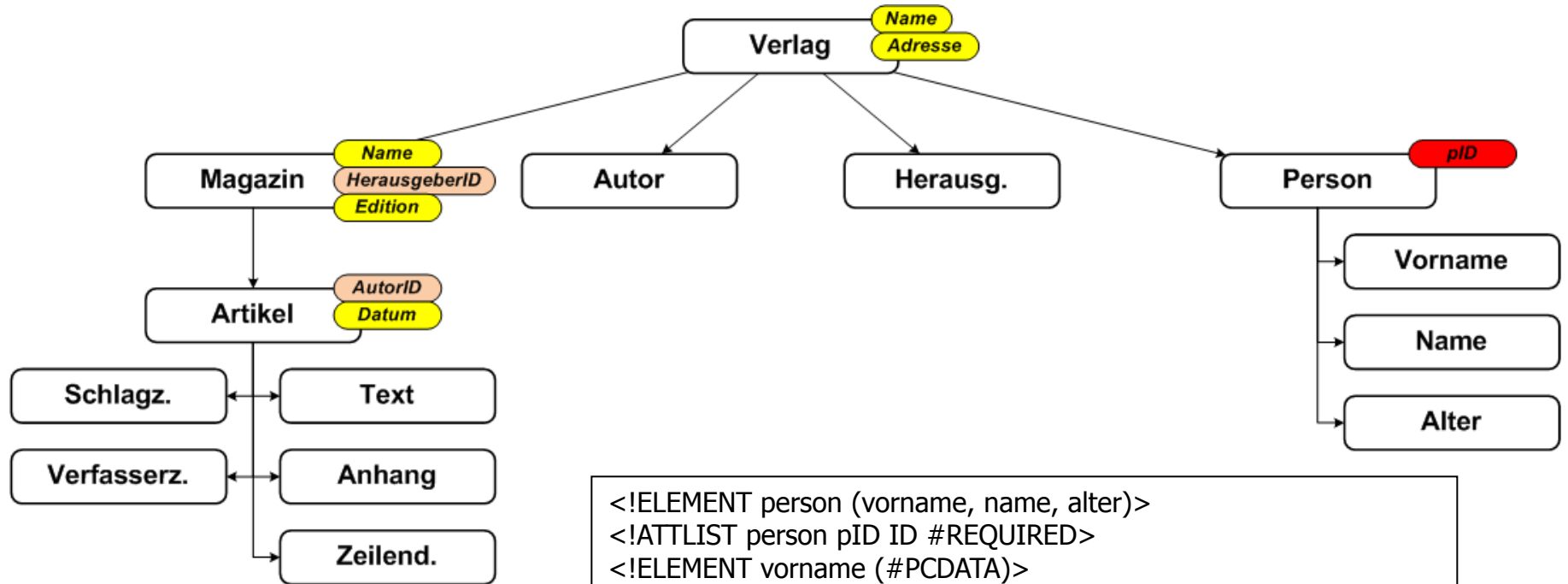
```
<!ELEMENT verlag (magazin+, autor+, herausgeber+, person+ )>
<!ATTLIST verlag name CDATA #REQUIRED>
<!ATTLIST verlag adresse CDATA #REQUIRED>

<!ELEMENT magazin (artikel+)>
<!ATTLIST magazin name CDATA #REQUIRED>
<!ATTLIST magazin herausgeberID IDREF #IMPLIED>
<!ATTLIST magazin edition CDATA #IMPLIED>

<!ELEMENT artikel (schlagzeile, verfasserzeile,
                    zeilendurchschuss, text, anhang)>
<!ATTLIST artikel autorID IDREF #REQUIRED>
<!ATTLIST artikel datum CDATA #IMPLIED>
```

Red oval = Id    Orange oval = IdRef    Yellow oval = CData

# XML und XPath



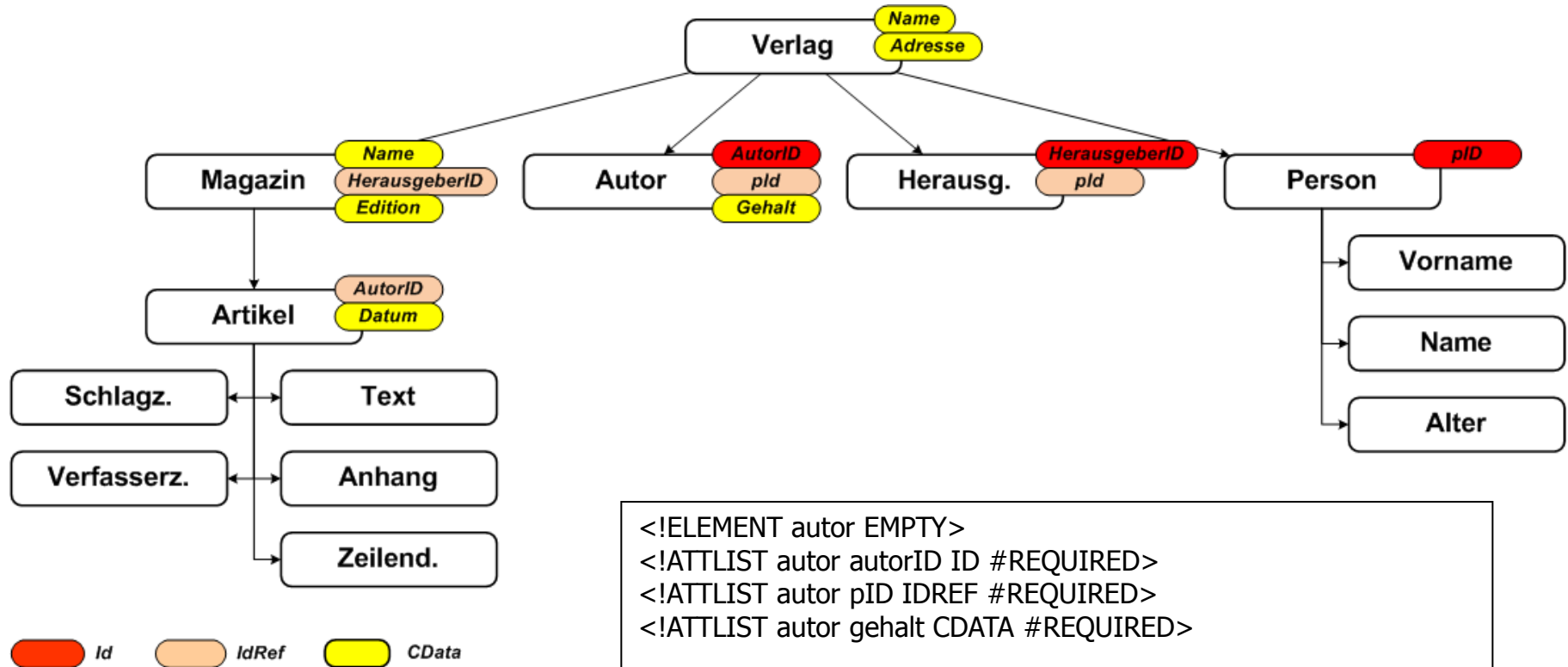
```

<!ELEMENT person (vorname, name, alter)>
<!-- ATTLIST person pID ID #REQUIRED -->
<!ELEMENT vorname (#PCDATA)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT alter (#PCDATA)>
    
```

```

<!ELEMENT schlagzeile (#PCDATA)>
<!ELEMENT verfasserzeile (#PCDATA)>
<!-- ELEMENT zeilendurchschuss (#PCDATA) -->
<!ELEMENT text (#PCDATA)>
<!ELEMENT anhang (#PCDATA)>
    
```

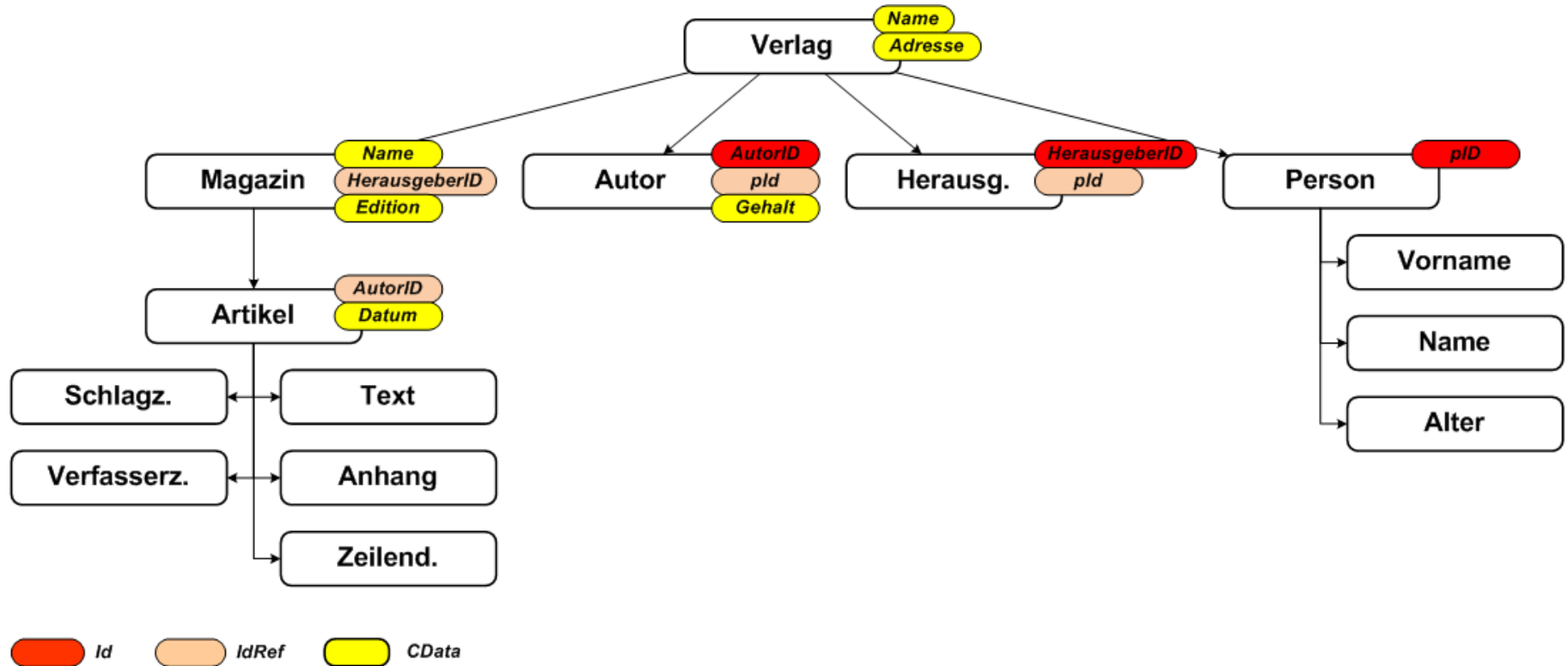
# XML und XPath



```
<!ELEMENT autor EMPTY>
<!ATTLIST autor autorID ID #REQUIRED>
<!ATTLIST autor pID IDREF #REQUIRED>
<!ATTLIST autor gehalt CDATA #REQUIRED>
```

```
<!ELEMENT herausgeber EMPTY>
<!ATTLIST herausgeber herausgeberID ID #REQUIRED>
<!ATTLIST herausgeber pID IDREF #REQUIRED>
```

# XML und XPath



# XML und XPath

- Pfadausdruck adressiert (selektiert) eine Sequenz von Knoten in einem Dokument
  - besteht aus Schritten, durch "/" voneinander getrennt
    - Bsp.: Namen aller Personen  
`/child::Personen/child::Person[Alter>18]/attribute::Name`
  - Verkürzte Schreibweisen für häufig genutzte Konstrukte  
( "@" = attribute::, "/" = /descendant-or-self::node()/, Achse fehlt = child::, etc.)
    - Bsp.: Namen aller Personen  
`//Person[Alter>18]/@Name`
  - wird sukzessive, von links nach rechts ausgewertet
    - Pfadanfang: Dokumentwurzel oder von außen vorgegebener Kontext
    - Jeder Schritt geht von Knotensequenz aus
      - Sucht für jeden Knoten in der Sequenz weitere Knoten auf
      - Prädikate (z.B. [Alter>18]) sorgen dabei für eine Knotenselektion
    - Leere Resultate führen nicht zu Fehler

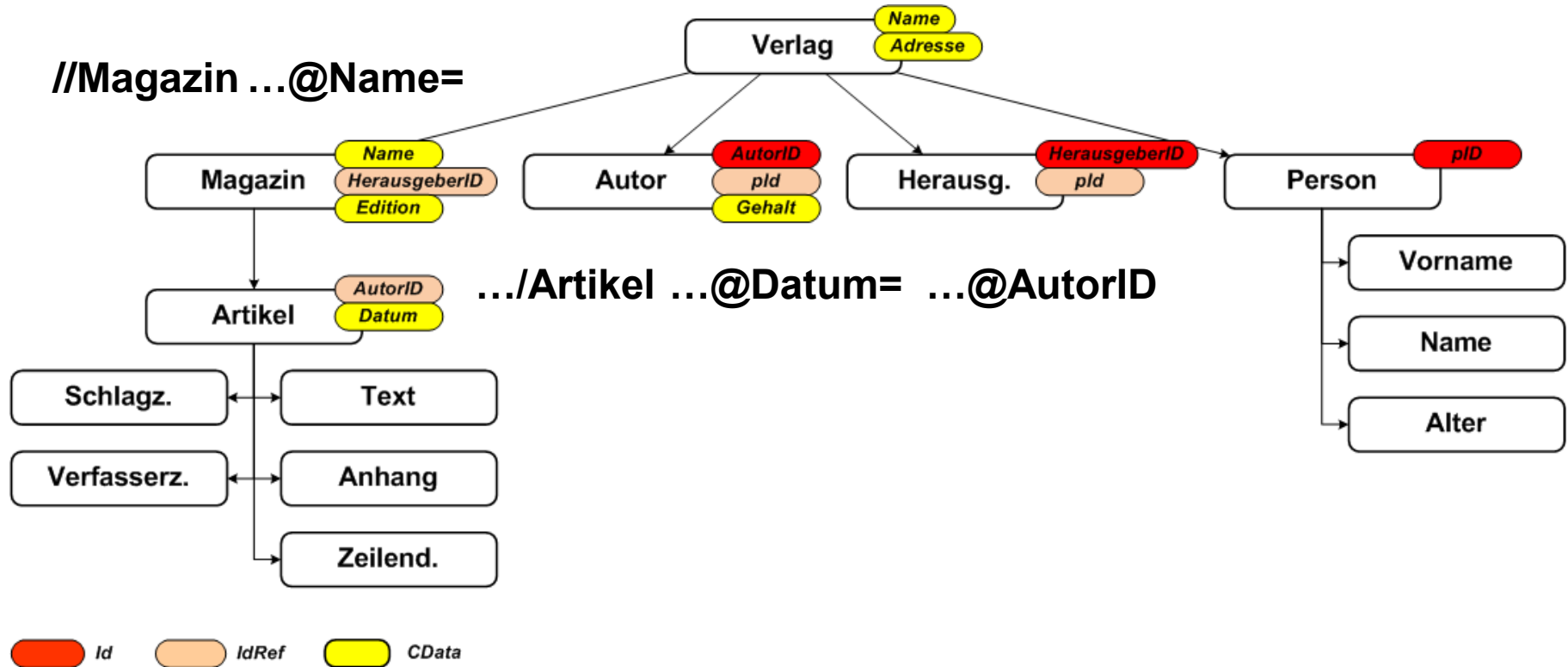


# XML und XPath

- Nutzung von Funktionen
  - fn:id = löst IDREF-Referenzen auf und gibt referenzierte Elemente zurück
  - fn:count = zählt die selektierten Elemente
- Kurze Beispiele für XPath mit und ohne Funktionen:
  - `//person[anschrift/ort/text()=,Hamburg`]/Nachname`  
„Die Nachnamen aller Personen, die in Hamburg wohnen.“
  - `fn:id(//verhandlung[@datum='18.10.2010']/zeuge/@person)/Nachname`  
„Die Nachnamen aller Zeugen, die an Verhandlungen am 18.10.2010 teilgenommen haben.“
  - `fn:count(//person[@pID = //zeuge/@person and @alter < 18])`  
„Die Anzahl der Zeugen, die jünger als 18 sind.“

**Bsp1:** //Magazin[ @Name = 'Spiegel' ]/Artikel[ @Datum = '2010-03-22' ]/@AutorID

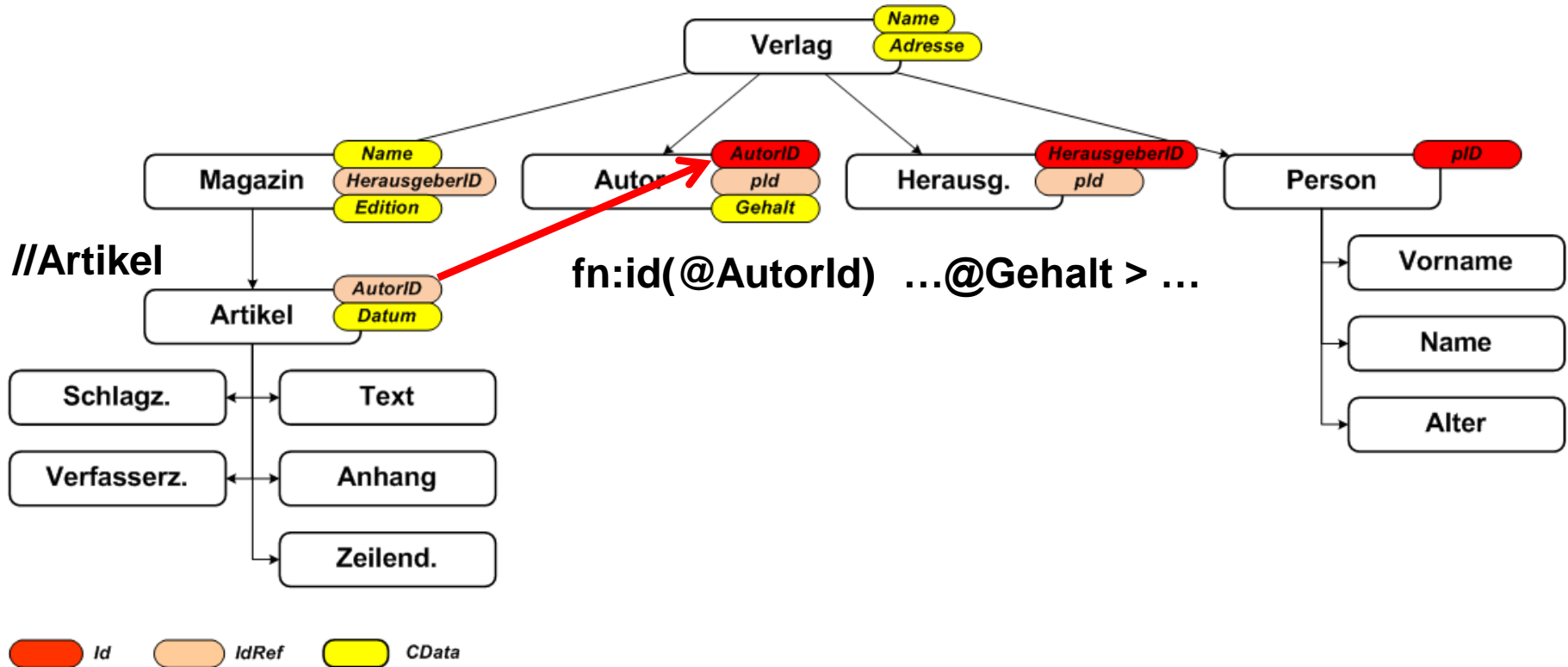
//Magazin ...@Name=



.../Artikel ...@Datum= ...@AutorID

Die IDs aller Autoren, die für das Magazin 'Spiegel' am 22. März einen Artikel verfasst haben.

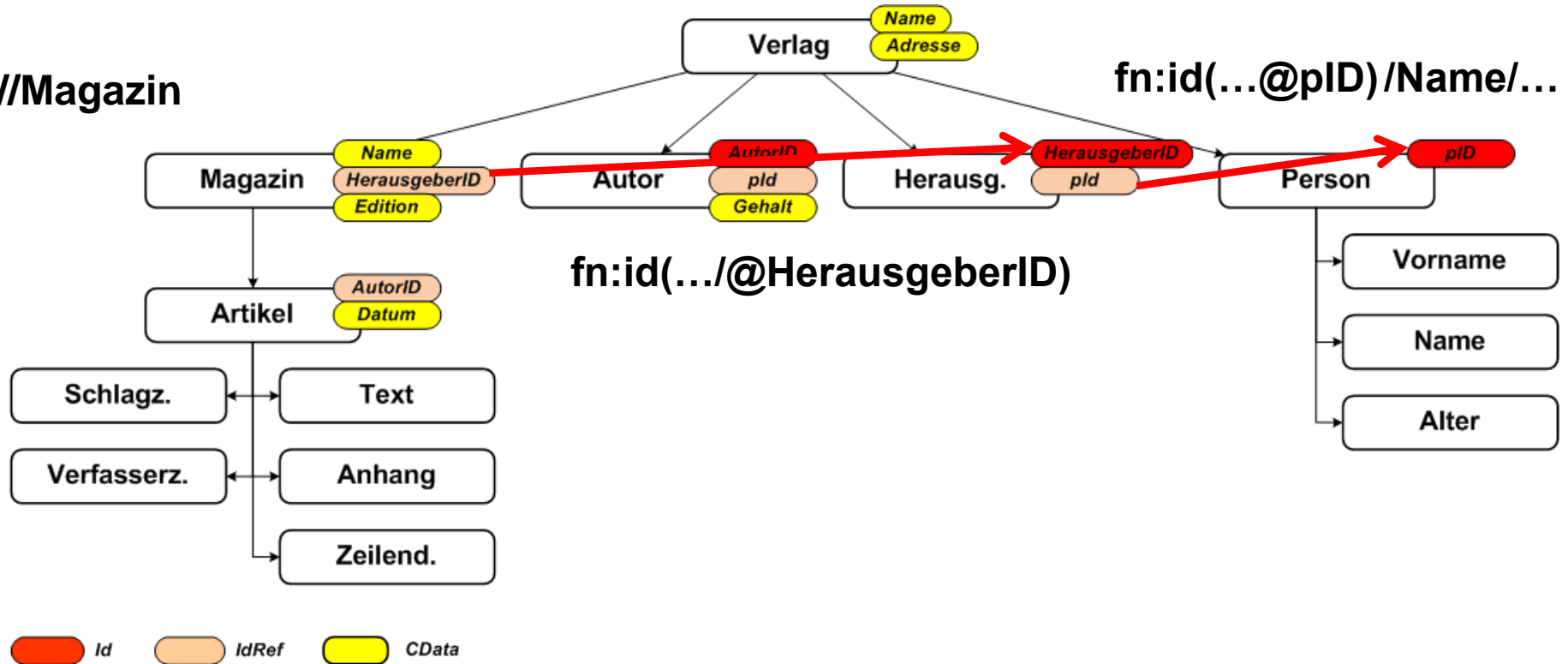
## Bsp2: `fn:count( //Artikel[ fn:id(@AutorID)/@Gehalt > 100000 ] )`



Die Anzahl der Artikel, die von Autoren mit mehr als 10000 Euro Gehalt geschrieben wurden.

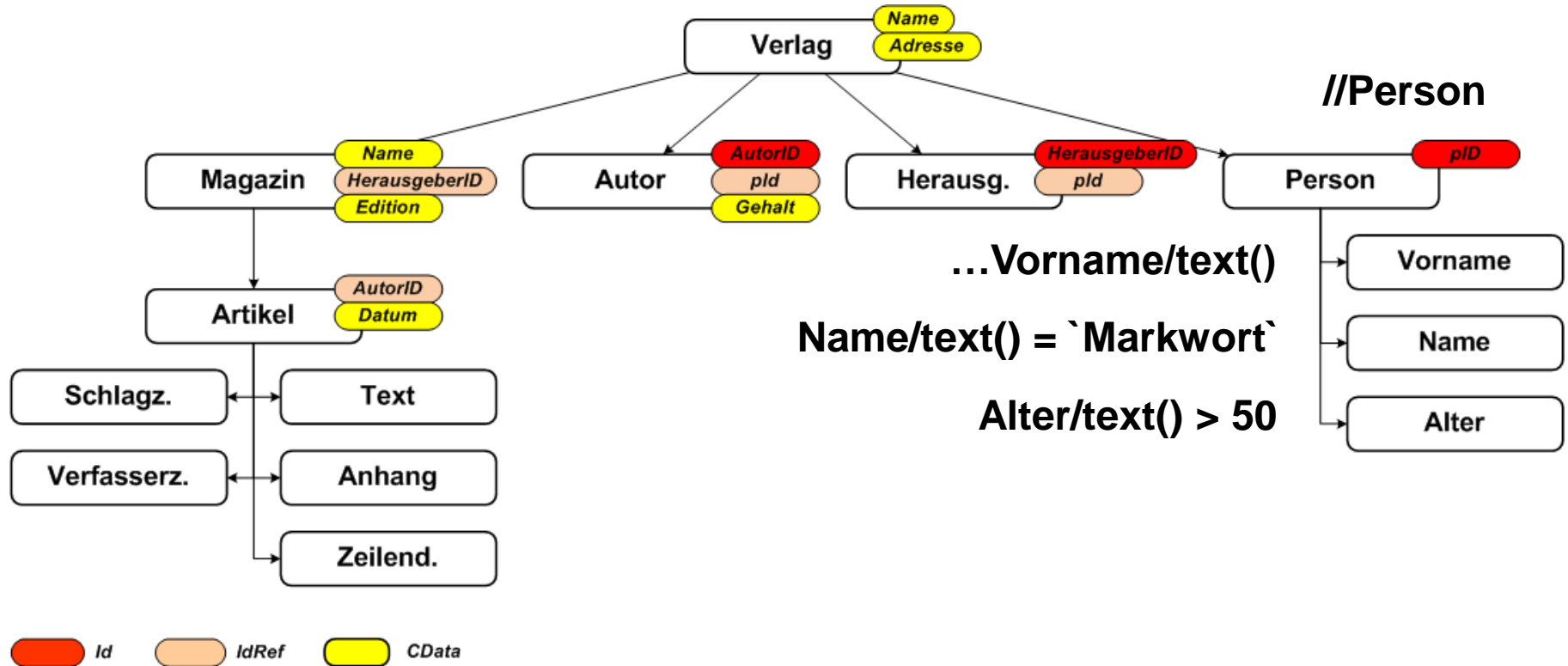
# Bsp3: `fn:id(fn:id(//Magazin/@HerausgeberID)/@pID)/Name/text()`

//Magazin



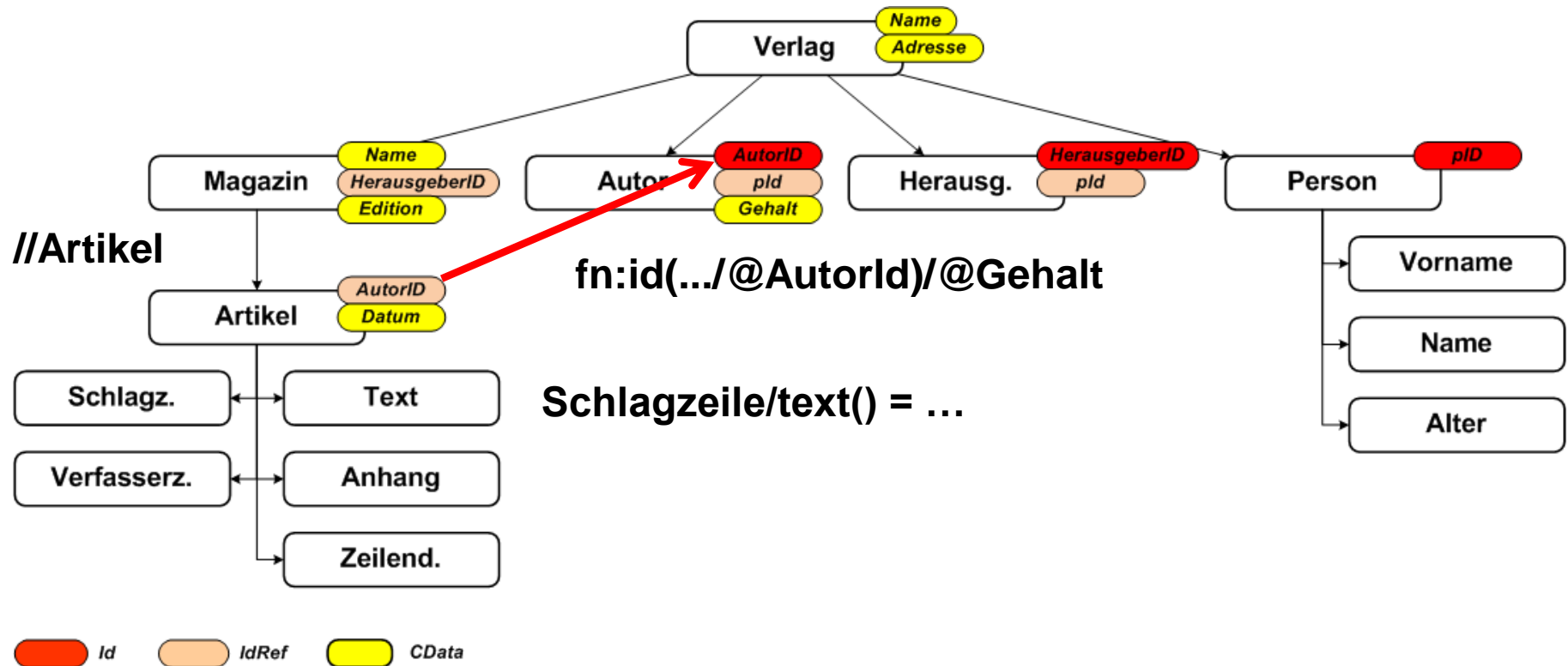
Die Namen der Herausgeber von Magazinen.

# Bsp4: Die Vornamen aller Personen mit dem Namen 'Markwort' älter als 50 Jahre.



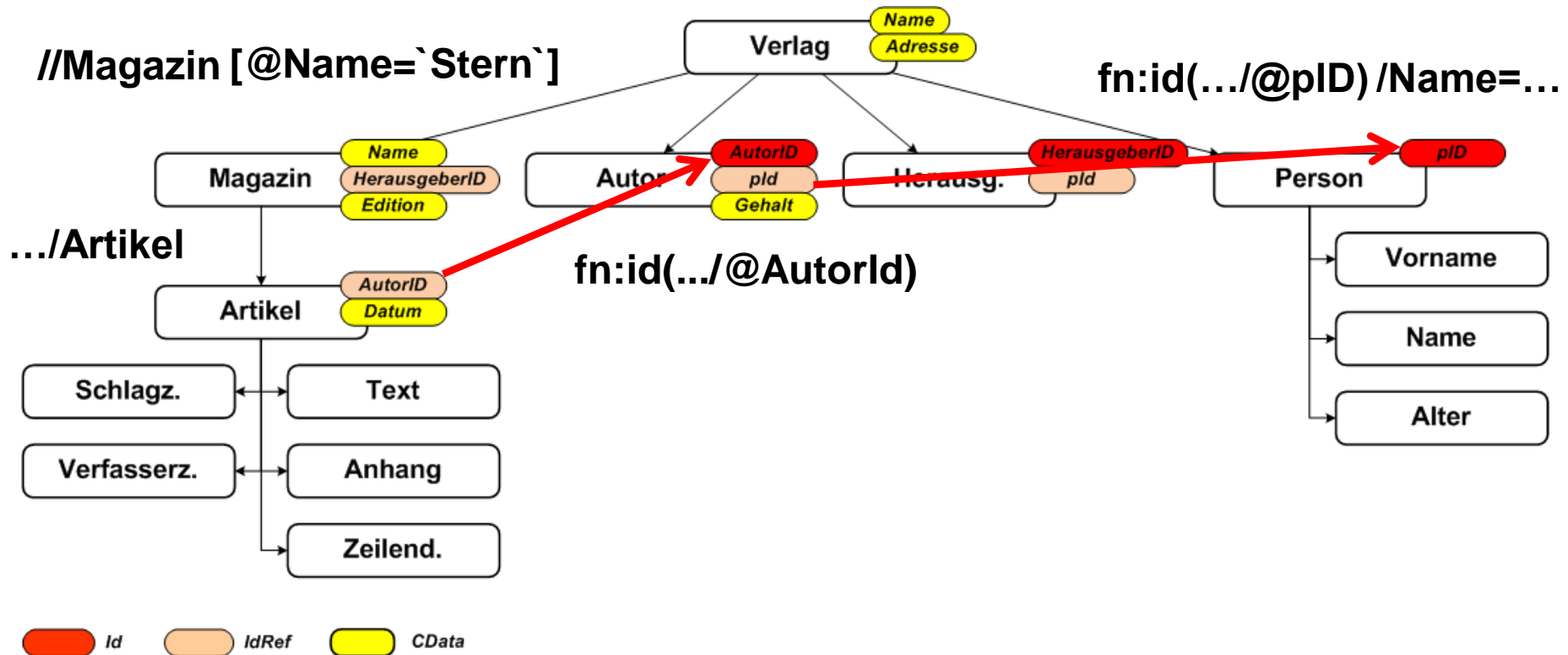
```
//Person[ Alter/text() > 50 and Name/text() = `Markwort` ]/Vorname/text()
```

# Bsp5: Die Gehälter von Autoren, die einen Artikel mit der Schlagzeile 'Spätrömische Dekadenz!' geschrieben haben.



fn:id(//Artikel[Schlagzeile/text() = 'Spätrömische Dekadenz']/@AutorID)/@Gehalt

# Bsp6: Alle Artikel im Magazin „Stern“, die vom Autor mit dem Namen `Thomsen` geschrieben wurden



```
//Magazin[@Name='Stern']/Artikel[ fn:id(fn:id(@AutorID)/@pID)/Name/text() = 'Thomsen' ]
```

# Organisatorisches

---

## **Besprechung Probeklausur**

**Erzw.H, 28.01.2013, 14.00-16.00 Uhr**