

# 1. Art des Events



CREATE TRIGGER beispielTrigger AFTER INSERT ON R  
FOR EACH ROW ...



INSERT OR DELETE

INSERT

UPDATE [ OF column\_name [, ... ] ]

DELETE

## 2. Granularität des Events

UPDATE

CREATE TRIGGER beispielTrigger AFTER INSERT ON R

FOR EACH ROW ...



FOR EACH ROW

→ UPDATE R  
SET a = 4?

FOR EACH STATEMENT

### 3. Zeitpunkt des Events



```
CREATE TRIGGER beispielTrigger AFTER INSERT ON R  
FOR EACH ROW ...
```

BEFORE

AFTER

Insgesamt

UPDATE R SET a=42;

BEFORE UPDATE ON R FOR EACH STATEMENT ...

einmal

BEFORE UPDATE ON R FOR EACH ROW ...

AFTER UPDATE ON R FOR EACH ROW ...

für jedes  
geänderte  
Tupel in R

AFTER UPDATE ON R FOR EACH STATEMENT ...

einmal

# Anwendungsbeispiele für Trigger

Nachbestellen, wenn Lagerbestände leerlaufen

# Anwendungsbeispiele für Trigger

Nachbestellen, wenn Lagerbestände leerlaufen

Anschreiben von Studenten, wenn CEP < vordefinierte Summe

# Anwendungsbeispiele für Trigger

Nachbestellen, wenn Lagerbestände leerlaufen

Anschreiben von Studenten, wenn CEP < vordefinierte Summe

Überwachung von Kontobeständen

# Anwendungsbeispiele für Trigger

Nachbestellen, wenn Lagerbestände leerlaufen

Anschreiben von Studenten, wenn CEP < vordefinierte Summe

Überwachung von Kontobeständen

Rechnungen, automatische Mahnung



# Anwendungsbeispiele für Trigger

Nachbestellen, wenn Lagerbestände leerlaufen

Anschreiben von Studenten, wenn CEP < vordefinierte Summe

Überwachung von Kontobeständen

Rechnungen, automatische Mahnung

Tabellenübergreifende Datenkonsistenz

CHECK

# Nachteile von Triggern

Übergang von deklarativ zu prozedural

# Nachteile von Triggern

Übergang von deklarativ zu prozedural

Was passiert wirklich bei insert/update/delete?

# Nachteile von Triggern

Übergang von deklarativ zu prozedural

Was passiert wirklich bei insert/update/delete?

terminiert die Rekursion?

# Nachteile von Triggern

Übergang von deklarativ zu prozedural

Was passiert wirklich bei insert/update/delete?

terminiert die Rekursion?

Laufzeit? FOR EACH ROW

# Nachteile von Triggern

Übergang von deklarativ zu prozedural

Was passiert wirklich bei insert/update/delete?

terminiert die Rekursion?

Laufzeit? FOR EACH ROW

zusätzliche Daten**I**nkonsistenz



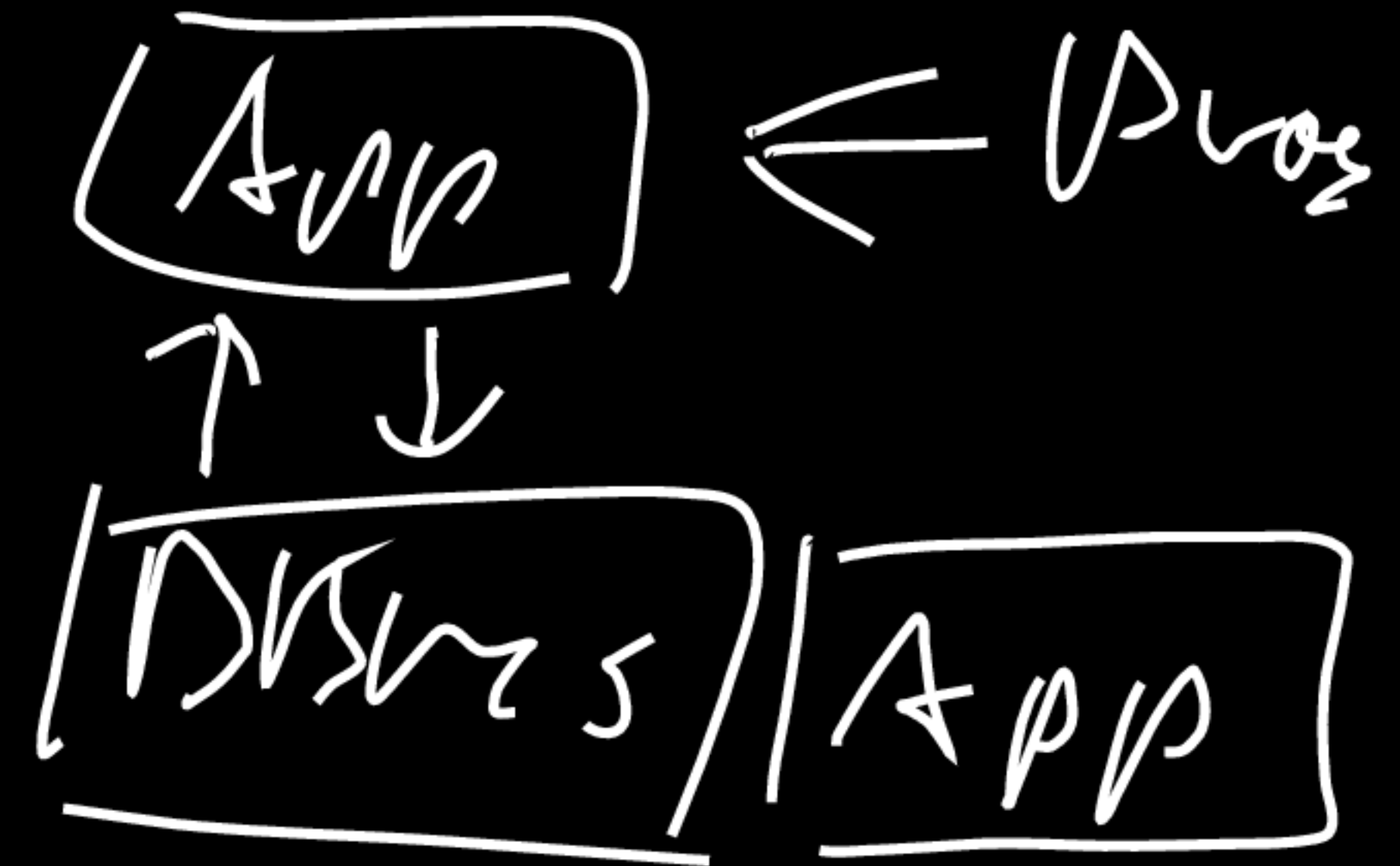
# Vorteile von Triggern

Datenbestand steuert Applikation

# Vorteile von Triggern

Datenbestand steuert Applikation

keine unnötige Auslagerung von prozeduralem Code





# Vorteile von Triggern

Datenbestand steuert Applikation

keine unnötige Auslagerung von prozeduralen Code

Rekursion oft beherrschbar

# Vorteile von Triggern

Datenbestand steuert Applikation

keine unnötige Auslagerung von prozeduralen Code

Rekursion oft beherrschbar

Laufzeit kein Problem, wenn vorsichtig eingesetzt

# Vorteile von Triggern

Datenbestand steuert Applikation

keine unnötige Auslagerung von prozeduralen Code

Rekursion oft beherrschbar

Laufzeit kein Problem, wenn vorsichtig eingesetzt

zusätzliche Datenkonsistenz