

Kapitel 12: Transaktionsverwaltung

Transaktion

Bündelung mehrerer Datenbankoperationen

- Mehrbenutzersynchronisation
- Recovery

Überweisung von 50 €

```
read(A, a);  
a := a - 50;  
write(A, a);  
read(B, b);  
b := b + 50;  
write(B, b);
```

Operationen einer Transaktion

- **begin of transaction (BOT):**
Anfang einer Transaktion.
- **commit:**
Ende einer Transaktion. Alle Änderungen seit dem letzten BOT werden festgeschrieben.
- **abort:**
Abbruch einer Transaktion. Die Datenbasis wird in den Zustand vor Beginn der Transaktion zurückgeführt.
- **define savepoint:**
zusätzlicher Sicherungspunkt.
- **backup transaction:**
Setzt die Datenbasis auf den jüngsten Sicherungspunkt zurück.

Abschluss einer Transaktion

BOT $op_1; op_2; \dots op_n; COMMIT$

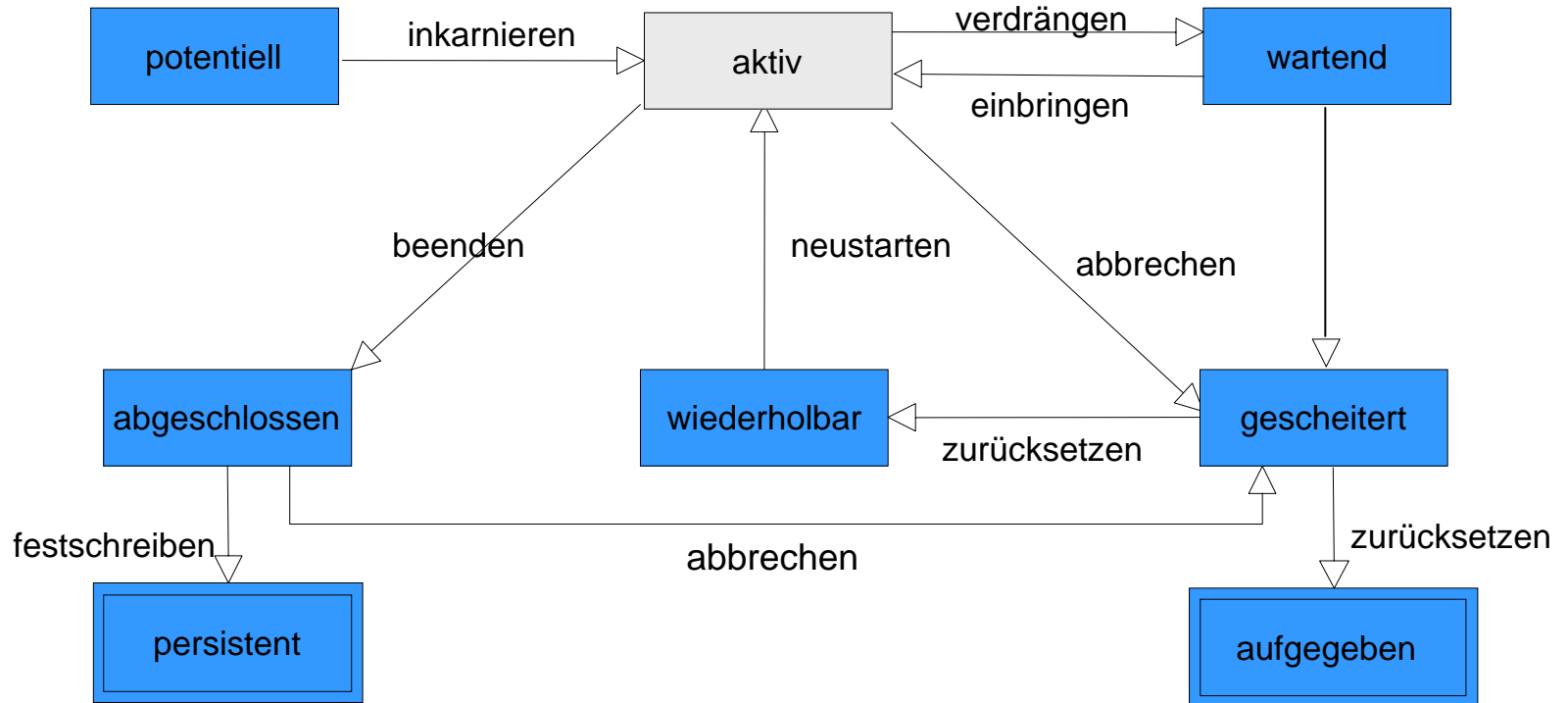
BOT $op_1; op_2; \dots op_n; ABORT$

BOT $op_1; op_2; \dots op_n; <FEHLER>$

Eigenschaften von Transaktionen: ACID

- Atomicity:
nicht weiter zerlegbare Einheit (*alles-oder-nichts*)
- Consistency:
Nach Abschluß: Konsistenz,
während der Transaktion: ggf. Inkonsistenzen.
- Isolation:
keine Beeinflussung durch nebenläufig ausgeführte
Transaktionen
- Durability:
Erfolgreich abgeschlossene Transaktion bleibt dauerhaft in
der Datenbank (auch nach einem späteren Systemfehler)

Zustandsübergänge



Transaktionsverwaltung beim SQL-Server

```
begin transaction
insert into professoren
values(55555, 'Erika', 'C4', 333, 1950-12-24)
select * from professoren where name='Erika'
commit work
```

```
begin transaction
select * from studenten where name like 'F%'
delete from studenten where name='Fichte'
select * from studenten where name like 'F%'
rollback transaction
select * from studenten where name like 'F%'
```

[Query Analyzer](#)