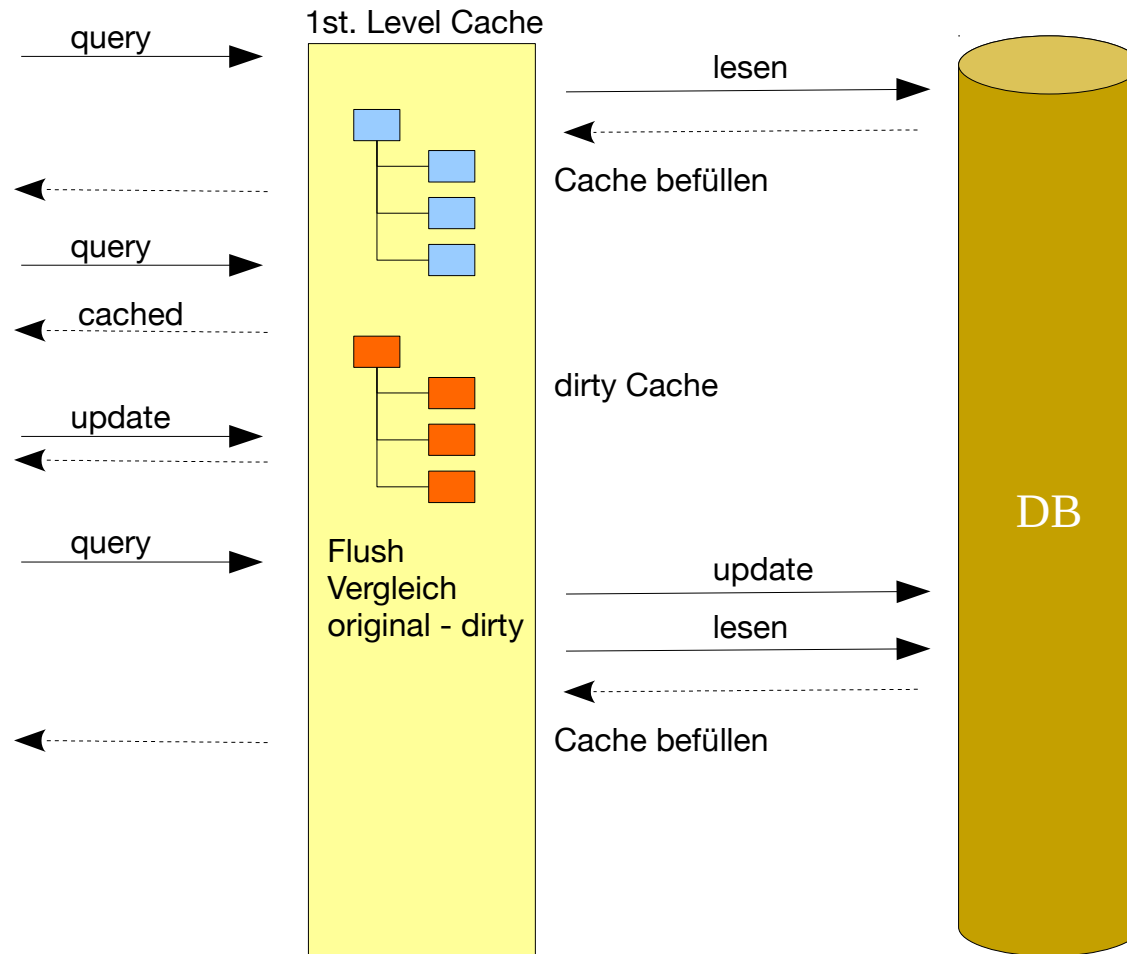


Analyse Datenbank Zugriffe



Datenbankzugriffe mit Hibernate



JDBC Zugriffe Loggen

■ Spy aktivieren

```
/subsystem=datasources/data-source=ticketMonsterDataSource:write-attribute(name=spy, value=true)
/subsystem=logging/logger=jboss.jdbc.spy/:add(level=DEBUG)
```

■ Logfile Auswerten (eigne Skripts)

- perfSql.pl
- perfStat.pl

```
perfSql.pl -h
```

Funktion:

Auswerten von SQL-Log bezueglich Performance.

usage: perfSql.pl [-q] [-h] [-d n] logfiles

Parameter:

logfiles: Namen der abzuarbeitenden Logfiles.

Optionen:

-q: Selektiert Queries

-s size: Wenn bei -q Queries länger als "size" sind, wird der Teil vor dem FROM entfernt.
Default: 200

-r: Ersetzt ? in den Queries mit den Werten. Es wird automatisch -q aktiviert.

-h: Aufrufsyntax anzeigen

-d debuglevel: Debuglevel angeben.

Perf-Skripts

```
perfSql.pl -q standalone/log/server.log | perfStat.pl > queries.csv
```

Service	Durchschn	Anz	Gesamt
insert into Ticket (id, price, number, rowNumber, section_id, ticketCategory_id) values (null, ?, ?, ?, ?, ?), 1	0	2357	1986
select ... from TicketPrice ticketpric0_ inner join Section section1_ on ticketpric0_.section_id=section1_.id inner join Venue venue2_ on section1_.venue_id=venue2_.id left outer join Mediaitem mediaitem3_ on venue2_.mediaitem_id=mediaitem3_.id inner join TicketCategory ticketcate4_ on ticketpric0_.ticketCategory_id=ticketcate4_.id where ticketpric0_.show_id=?	2	126	261
select ... from Performance performanc0_ where performanc0_.show_id=? order by performanc0_.date	1	126	182
select ... from SectionAllocation sectionall0_ where sectionall0_.performance_id=? and sectionall0_.section_id=?	1	102	173

Oracle: Queries Auswerten

```
/subsystem=datasources/xa-data-source=myDatasource:write-attribute(name=new-connection-sql, value="alter session set SQL_TRACE=TRUE")
```

- Mit Oracle Trace auswerten