

MyBatis缓存

2021年10月7日 0:14

MyBatis缓存在应用服务器这一级，它可以把前面的数据库查询的结果放在缓存中，下次就不用再去查数据库，从而降低数据库的负载

MyBatis缓存

一级缓存

是在同一个SqlSession中，它在一个sqlSession会话里面的所有查询操作都会保存到缓存中，一般来说一个请求中的所有增删改查操作都是在同一个sqlSession里面的，所以我们可以认为每个请求都有自己的一级缓存，如果同一个sqlSession会话中2个查询中间有一个 insert、update或delete语句，那么之前查询的所有缓存都会清空。

二级缓存

是SqlSessionFactory级别（或者说namespace级别）的，通过同一个SqlSessionFactory创建的SqlSession查询的结果是会被缓存的，此后如果再次执行相同的查询语句，结果就会从缓存中获取。简单来说，多个请求可以共享二级缓存。

二级缓存需要手动开启。

所有的更新操作(insert,delete,update)都会触发缓存的刷新，从而导致二级缓存失效，所以二级缓存适合在读多写少的场景中开启。

MyBatis如何判断两次查询相同

1. Statement的id相同
即在xml中的id、接口中的方法名
2. 预编译的SQL相同
statement中会产生动态的SQL语句，比如用#标注的那些变量（#{name}）
3. 查询的参数相同

一级缓存具体实现

MyBatis是靠SQLSession来把对于接口的调用转换成SQL语句的

在SQLSession中，有一个Executor，这个Executor是具体负责把接口的调用变成在XML中的SQL语句的

Executor中有一个LocalCache本地缓存

当用户发起查询时，SQLSession会根据XML中的定义生成一个SQL语句，然后根据判断查询是否相同的三个条件到LocalCache中去找这样的查询是否有执行过，如果有则直接从LocalCache中拿，没有则要到数据库中去查，并且要把查到的结果缓存到LocalCache中这样的缓存只存在一次事务中，或者说存在一个SQLSession中



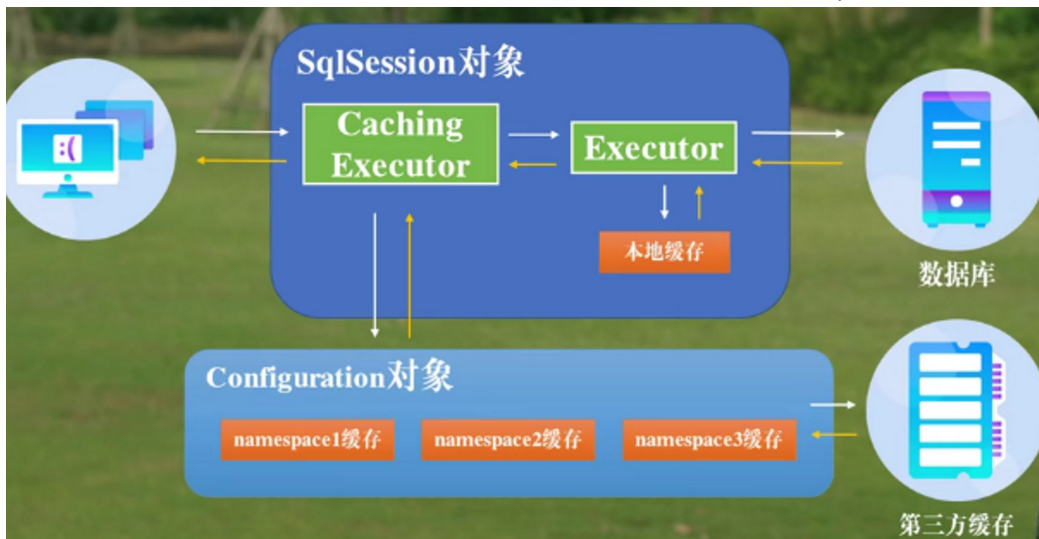
二级缓存具体实现

二级缓存是在一级缓存的基础上来实现的，我们称之为mapper级的缓存，它的作用域是一个namespace（XML中定义的namespace）

在每个namespace中，如果不同的SQLSession执行的查询相同，就会把上次查询的结果拿出来直接返回给前端

具体是，二级缓存也是通过SQLSession来完成对于接口的调用和SQL语句的映射。在Executor之前有一个CachingExecutor

当一个查询由SQLSession来执行的时候，首先会通过CachingExecutor，CachingExecutor会从namespace的缓存中去查询是否有相同查询，如果没有则交给Executor去执行后面的步骤，并且把查到的数据放在namespace缓存中



二级缓存对应用程序的意义更大，因为在一个事务中，同一个查询很少执行两次
但是二级缓存存在一定的缺陷

二级缓存的缺陷

1. 难以支持应用服务器集群

因为二级缓存是在应用服务器级别来做的缓存，不同的应用服务器会有不同的二级缓存

2. 不能高效利用内存

二级缓存会不加区分的进行缓存，会占用大量内存空间，同时要删除一些重复数据也会占用一定内存