

hibernate从性能上来说，它比jdbc运行的效率低，因此需要对其进行一个优化。

## 本章学习目标

- Hibernate 对象的三种状态
- Hibernate 的一级缓存与快照机制
- 一级缓存的管理
- 延迟加载策略
- 抓取策略
- 整合 [durid](#) 连接池
- Hibernate 的二级缓存

在hibernate程序运行过程中，一个对象会存在以下的三种状态，分别为：

- 1) 临时状态：没有对象的OID值，没有session管理。
- 2) 持久化状态：有对象的OID值，并且被session管理。
- 3) 游离状态：有对象的OID值，但是已经脱离session管理。

其中OID值是用来标识内存中对象和数据库中表的记录的对应关系，对象的OID值和数据库表的主键对应。

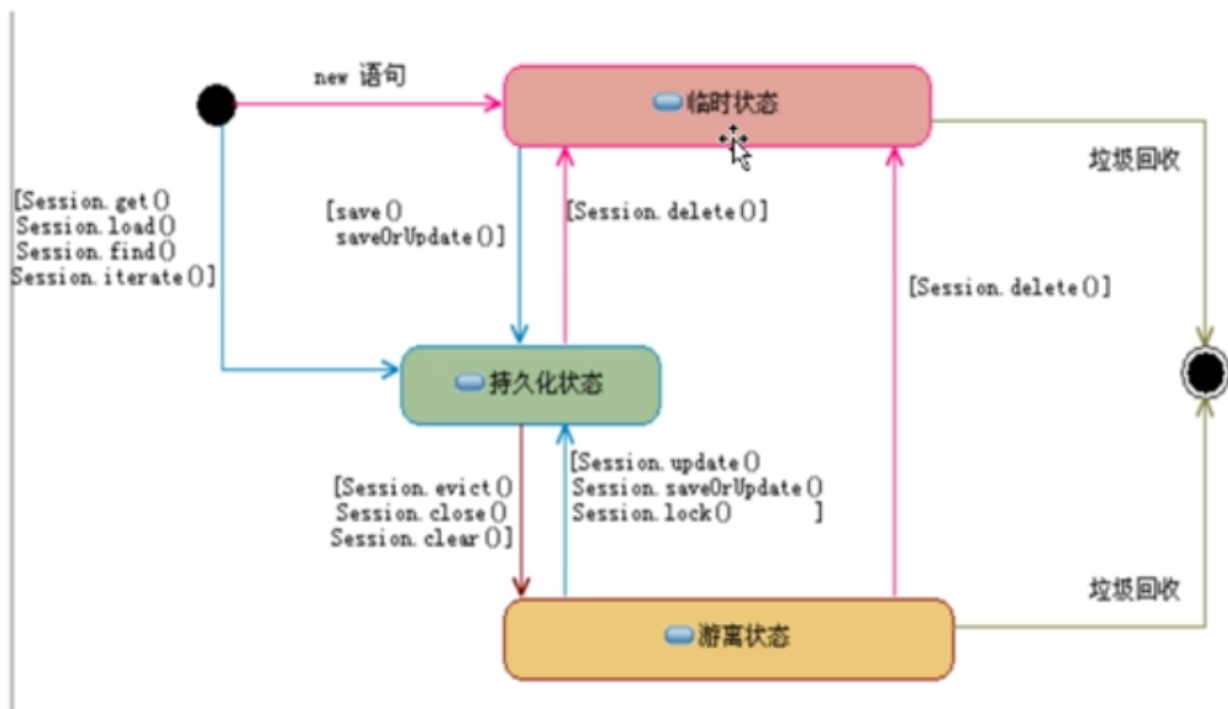
## hibernate三种状态之初认识：

// 这时p1对象没有OID值，没有被session管理，这时的p1是临时状态

```
person p1 = new person();  
p1.setName("qnli");
```

```
Session session = hibernateUtils.getSession();  
Transaction ts = session.beginTransaction();  
// 此时p1有OID值，并且被session管理，这时p1是一个持久态对象。  
session.save(p1);  
System.out.println(p1.getId());  
  
ts.commit();  
session.close();  
// 此时p1对象有OID值，没有被session管理，此时是一个游离状态。  
System.out.println(p1.getId());
```

## hibernate三种状态之间的转换：



重点：在hibernate的三种状态中，最为重要的状态是持久态！！

持久态对象可以直接影响数据库中的数据。