

漫谈Java数据库存取技术Java认证考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/607/2021_2022__E6_BC_AB_E8_B0_88Java_c104_607780.htm 首先为大家列出Java数据库英雄榜 1.JDBC直接访问数据库 2.EJB entity bean. 3.JDO技术。 4. 第三方O/R工具，如目前大红大紫的Hibernate，其它如Castor，Toplink. 先说说这个历史最为悠久的JDBC吧。从Java诞生的那天起，这位仁兄就开始登上历史舞台了。Java能有今天这么风光，JDBC可以说是功不可没。一路走来，如今已是JDBC3.0了。在没有JDBC的时候，访问Java数据库那是八仙过海，各显神通，各家数据库厂商都有自己的一套API，苦就苦了开发人员了。换了个Java数据库，那个程序要改是面目全非。JDBC规范的出台，向世界宣告从此有了访问关系数据库的标准通用接口了。JDBC标准获得了几乎所有数据库厂商的支持，好像还真难找到这么一个数据库，它是没有JDBC支持的。JDBC规范一经发布，获得了空前成功，很快成为java访问数据库的标准。JDBC的成功在于它的规范统一标准的接口，只需要掌握标准的SQL语言就可以访问各种不同的数据库了。这种数据库间的可移植性和Java一直高喊的口号Compile Once，Run everywhere遥相呼应。JDBC今天还是java访问数据库的基石，CMP，JDO，Hibernate说到底只是更好的封装了JDBC，提供了更为上层的更为强大的接口而已。然后说说JDBC直接访问数据库的方式来实现java持久性。这种方式相对于CMP来说比较简单直接，特别是对于小型应用十分方便。比如，我要写一个简单的留言版程序，就没有必要session bean，entity bean，又是home接口又是远程接口，

一层层调了吧。直接JDBC，写SQL语句了事。和其它持久化技术相比，JDBC直接访问数据库的方式需要程序员操心的事情多了一些，你得自己关心transaction，自己关心连接池，你得写大量的get set方法，把SQL Oselect出来的值一个一个塞到你的java object中，或者把java object的值一个一个给取出来，用SQL insert 到数据库，完全手动进行O/R mapping.为了克服这些缺点，CMP，JDO等等开始陆续登上历史舞台。下面EJB登场，EJB作为Sun J2EE体系的核心部分，是Sun所力推的企业级开发的首选，而EJB entity目前仍然是Sun j2EE白皮书所最为推荐的java持久化技术。Entity Bean作为EJB规范的一部分，也是EJB规范里面最备受争议的一种技术，它伴随着EJB规范走过了风风雨雨几个春秋。目前EJB3.0规范草案已经出台，<http://jcp.org/en/jsr/detail?id=220>.从家庭出生来看，EJB可谓根正苗红，规范处于JCP管理之下，拥有超级豪华的专家组成员，Sun，IBM，Oracle，Borland，Bea，SAP，Jboss，Apache软件基金组织等等。单从这一点来看，选它作为企业级开发，技术支持应该就无需担心了。当然向IBM，Bea等寻求项目咨询价格当然也不菲。从提供功能上来看，EJB entity经历了EJB1.0，EJB1.1，EJB2.0，功能也越来越完善了。包括了完善的事务支持，EJBQL查询语言，透明的分布式访问等等。不过作为一个重量级技术，entity bean的性能不太尽人意，这成为它备受争议的一个焦点，不知在3.0以后这个状况会不会有所改进。再有一个，它功能虽然强大，可是对于易用性来说，实在不敢恭维，写一个最简单的bean，也非得home接口，远程接口，要再加上2.0以后加入的本地接口，这么林林总总一大堆，足以让Java初学者望而却步了。但是这

一点在一段时间内竟然也成了EJB 功能强大，技术高深的“佐证”。记得多年以前刚毕业那阵，EJB应用在国内还比较少，公司里也没有人研究Why EJB这个问题，反正凡是用EJB的项目就是牛项目，用EJB的人就是牛人，分到EJB项目组的兄弟们走路都是抬头挺胸的，说话都比我等还在JDBC，SQL的人要高两嗓门。EJB 技术目前盘踞着企业级应用的大部分江山，老大地位短时间内很难撼动。下面新生代代表JDO隆重登场，JDO绝对属于超年轻选手，JDO1.0也不过是2002四月份才发布。2003五月份出台1.0.1，目前最新2.0草案已经发布，<http://www.jcp.org/en/jsr/detail?id=243>.就为这2.0，江湖上展开的讨论可以说是“血雨腥风”，两大兵团，JDO兵团和EJB兵团争得是不可开交。有兴趣的不妨到这里瞧瞧，http://theserverside.com/news/thread.tss?thread_id=25695，里面也不乏重量级人物。单从这一点来看，它能对EJB产生这么大的冲击，足以说明了这个初生牛犊确有过人之处。JDO的诞生给java数据持久性带来很多新特性，特别是它弥补了EJB对OO编程的先天不足，JDO提供了完全的OO支持，继承，多态。JDO和EJB比属于轻量级工具，无需容器支持。不像EJB，要用你就非得整一个Weblogic，webSphere之类的。JDO的简单易用是最为人们所称道的，不需要你写大量无用的接口，不需要你继承什么特殊的类，唯一所要做的就是对你的class文件做一下 enhance.用了JDO，可以说我们的java程序这下真正OO了，我们无需再理会数据库里面有啥表格了，存取都是以java object为对象了，所有数据库表格都是自动生成的。这一点可以说也是一个革命了。在此之前，项目设计阶段，Database Schema设计可以说是个重头戏。而现在用JDO

开发，完全不需要数据库设计了。那你的Database Schema呢？就是你的Class啊，JDO会根据你的Class自动生成相应的数据库表格。一个字，爽！从数据库可移植性来看，JDO也是优势明显，就我使用过的Kodo和Genie来看，几个简单应用程序换数据库时候除了换一个JDBC driver，换一下数据库URL，无需对程序做任何改动。这一点对EJB来说又是处于劣势。从家庭出身来看，JDO也是出生名门，从一开始就处于JCP管理之下。从企业级支持来看，它可以很好的和Session bean协同工作，对于企业级开发，Session bean JDO的方式是Session bean entity方式的一个强有力竞争对手。虽然有这么多优点，不过它的发展之路也非一帆风顺，这不，今年五月份JDO2.0的投票，IBM，Oracle，Bea三大巨头同时投了反对票。不过稍微一想，就可以理解，这并不是JDO本身技术有什么重大缺陷，而是JDO动到这些巨头们的奶酪了。Bea，IBM做着业界最为著名应用服务器，weblogic和WebSphere，在EJB上面是投下了血本了，他们不能眼睁睁看着JDO来蚕食EJB市场。而Oracle，还在卖着自己自己的O/R工具Toplink，看着JDO日渐强大，他能不着急么。不过呢，公司再牛，他也挡不住历史前进的车轮吧，最终JDO2.0的投票还是以绝对的票数（12：3）通过了。还有其它散落江湖的Java持久化技术，如Hibernate，Castor，Toplink，他们虽然没有皇家血统，不过实力也是不容小视。就拿Hibernate来说，是javaworld评选出来的2003年度最佳java数据存取工具，目前可以说是大红大紫。而Castor和Toplink也算是历史悠久了，在JDO没有出世之前，它们就在江湖上混着了。目前也占据着一定的市场。这些第三方的工具从功能上来说很类似

于JDO，只是各自的API互不相同。这也是后来JDO规范的呼声越来越高的一个原因吧。这些第三方O/R mapping工具能在江湖上立足，也确实都有各自过人之处。如Hibernate金字招牌就是Open Source，支持几乎市面上所能看到得绝大部分数据库，并且文档也非常齐全。Toplink么，可谓历史悠久，又榜着Oracle这棵大树。目前来看，这些工具也占据着java数据库存取的不小市场。个人觉得，随着JDO规范的不段完善，JDO产品的普及，这一部分人员可能会在以后渐渐退出历史舞台。不过从Hibernate目前如日中天的气势来看，好像说这句话还为时过早。关于这些技术优劣之争从它们刚刚出生那天起从来就没有停止过，而各家各派也从来没有能够说服过对方。对于我们应用开发者而言，撇开应用纯粹来争论技术优劣并没有多大意义。还是俗话说的好，没有最好的，只有最合适的。我们能够在做开发的时候能够选择一个最合适于自己应用的技术，那就足够了。总的来说，JDBC面向RDBMS，比较适合关系数据库模式驱动的应用，例如统计表格数据，生成报表之类的应用。EJB技术以J2EE应用服务器为中心，如果你的应用确实需要灵活的可声明的事务边界，需要支持大容量的访问和不间断的服务，需要应用服务器的集群，那么选EJB吧。JDO则面向对象，对于以域对象为中心的应用，包含图，树模型的应用，JDO是首选。更多优质资料尽在百考试题论坛 百考试题在线题库 java认证更多详细资料 100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com