**[derby SQL 分页问题](http://blog.csdn.net/zhulx_sz/article/details/51719029)**

标签： [JavaDB](http://www.csdn.net/tag/JavaDB)[derby](http://www.csdn.net/tag/derby)[sql](http://www.csdn.net/tag/sql)[database](http://www.csdn.net/tag/database)[java](http://www.csdn.net/tag/java)

2016-06-20 10:32 646人阅读 [评论](http://blog.csdn.net/zhulx_sz/article/details/51719029#comments)(0) [收藏](javascript:void(0);) [举报](http://blog.csdn.net/zhulx_sz/article/details/51719029#report)

http://static.blog.csdn.net/images/category_icon.jpg 分类：

JavaDB derby sql（3） http://static.blog.csdn.net/images/arrow_triangle%20_down.jpg database（1） http://static.blog.csdn.net/images/arrow_triangle%20_down.jpg java（13） http://static.blog.csdn.net/images/arrow_triangle%20_down.jpg

版权声明：本文为博主原创文章，未经博主允许不得转载。

之前在网上看到有人问 Derby SQL 分页实现的问题，网上有人给出这样的解决方案，SQL 如下：  
  
  
SELECT \* FROM (  
    SELECT ROW\_NUMBER() OVER() AS rownum, myLargeTable.\*  
    FROM myLargeTable  
)  
AS tmp  
WHERE rownum > 200000 AND rownum <= 200005;  
  
  
其实，这样的分页查询，性能不理想，我试过在 300W 数据量中采用这种分页方式，需要 20~30秒之久；  
其实 Derby 10.6 以上版本有更好的分页支持，直接给出 SQL 实现如下：  
  
  
SELECT \* FROM YOUR\_TABLE  
OFFSET ? ROWS  
FETCH NEXT ? ROWS ONLY;  
  
  
  
稍微解释一下：  
OFFSET ? ROWS - 是指跳过 ? 条记录  
FETCH NEXT ? ROWS ONLY - 是指抓取下一个 ? 条记录  
   
  
这样分页性能可以提升至毫秒级速度，  
希望对遇到同样需求的朋友有帮助···  
  
  
其实 Derby 是很棒的纯 Java 实现的开源数据库，据说现在最大能支持到 700GB 的数据量，而且正在向 TB 级别进发···