**[db2快照分析工具-鲁晨（外）](http://wiki.primeton.com/pages/viewpage.action?pageId=10223895)**

       某客户线上系统，某天出现数据库死锁导致的宕机事故。重启生效后，一直定位不出来原因。初步分析，应为大事务，长查询和update等交互锁，导致的锁升级。

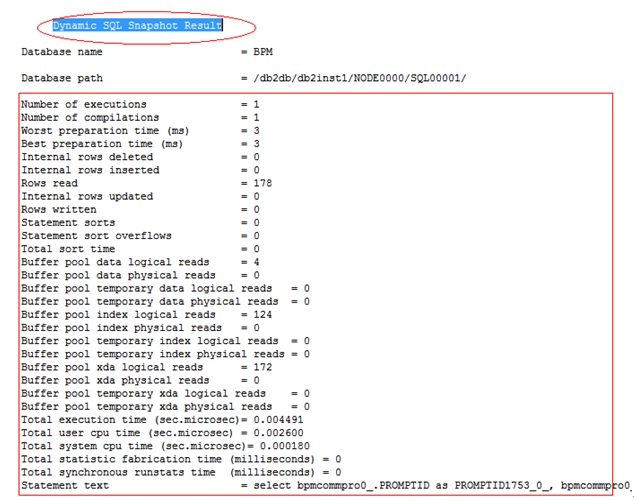
       客户没有购买DB2 Performance expert 这类的性能分析工具，只好抓快照来分析。系统管理员做了个任务，每隔20分钟抓一个DB2快照，抓8个小时的，拿下来分析。

       拿下来一看，哇靠，全是文本文件啊，完全没法分析，只是能大致看到，一些SQL执行确实出乎人的意料，看样子要首先对快照进行分析，摘出来最需要优化的sql进行优化。

       我写了一个分析db2快照文件的工具。此工具的主要功能是分析db2快照文件，读取每条SQL的执行次数、执行总时间，计算出平均执行时间排序，导出为Excel文件，以便可以最直观的获取执行时间最长、最需要优化的SQL语句。

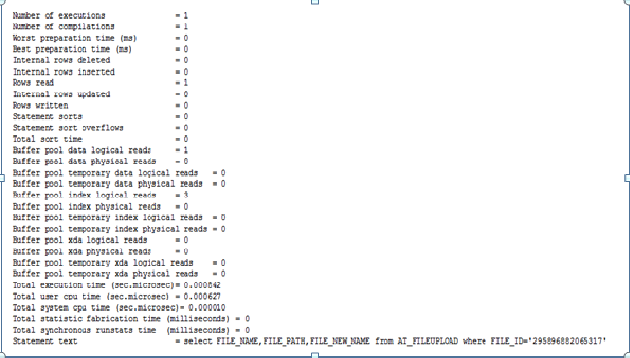
       db2快照文件都是以.out为后缀的文件，打开可以看到此文件记录着db2数据库的运行状况，我所需要的便是记录SQL语句执行情况的那部分。我的思路便是将文件读取到内存，再用字符串分割的方式，截取到有用的信息。

       首先，据观察快照文件，此文件从“Dynamic SQL Snapshot Result”关键词开始，便开始记录关于SQL执行情况的信息，并且这些信息都是以每条SQL语句，分段展示的，如下图所示：

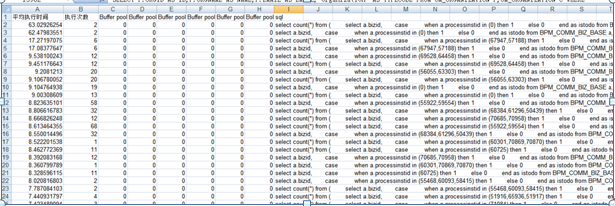


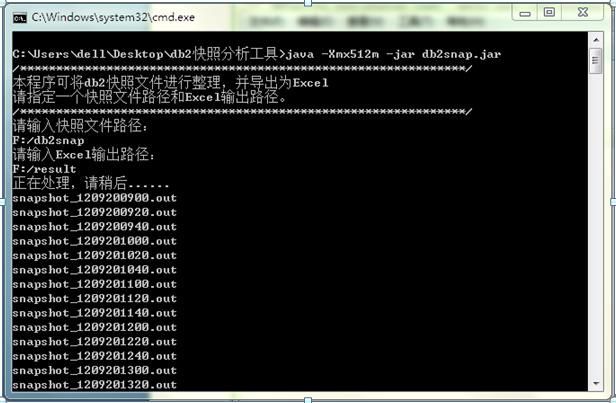
       此字符段包涵有所有关于SQL执行的信息，例如：Number of exexcutions是执行次数，Total exexcution time为执行总时间，Statement text为SQL语句。得到这些信息，就可以得出SQL的平均执行时间。

       接着，我尝试用FileInputstream将文件分段读入内存，快照文件通常都比较大，因此我设定每次读取1M，循环读入内存。 将内容读取放入byte数组中，然后转化为字符串。由于每段信息都是由“Number of exexcutions”开头，并且最后一条是“Statement text”，因此，很容易就可以把读取内容分割为下图所示一段段的字符串：



       将所需信息分割为上述段落之后，再根据“=”号之前的名称与“=”号之后的数值，进行分割读取，在此，定义一个SQLEntity 类，将所有数值作为属性保存到SQLEntity的对象中，计算出每条SQL语句的平均执行时间。拿出了Buffer pool data physical reads，Buffer pool temporary data physical reads，Buffer pool index physical reads， Buffer pool temporary index physical reads，Buffer pool xda physical reads，Buffer pool temporary xda physical reads等几个优化比较关注的物理读取次数数值。到此，已经做好了信息的提取。

         然后，将结果导出Excel文件，在此，我用的是Apache的office处理包POI，其中，HSSF提供读写Microsoft Excel XLS格式档案的功能，POI包的使用非常简单，很轻松就将结果变成想要的方式导出为Excel。结果如下图：   
  
最后，为了方便使用，将此程序导出为可执行jar包，用批处理文件Run.bat运行，运行的时候将提示输入两个路径，一是快照文件的路径，另一是Excel文件输出路径。程序运行情况如下图所示：



抓取快照，拿工具分析，这可以作为应用对应数据库优化的第一个步骤。

欢迎大家试用。