**sqlautoreview程序文档**

赵林/丹臣

2011/12/22

版本记录

| 版本编号 | 版本日期 | 修改者 | 说明 |
| --- | --- | --- | --- |
| 1.0 | 2011/12/22 | 赵林 | sqlautoreview文档正式第一版 |
| 1.1 | 2012/1/17 | 赵林 | 修改架构图,现在支持多表 |
| 1.2 | 2012/1/18 | 赵林 | 增加扩展性，以及与前台程序的结合章节 |
| 1.3 | 2012/2/7 | 赵林 | 增加merge index模块的内容 |

目录

[版本记录 2](#_Toc317152086)

[第一部份: sqlautoreview程序的安装 4](#_Toc317152087)

[1. 安装jdk 1.6 4](#_Toc317152088)

[2. 配置环境变量 4](#_Toc317152089)

[3. sqlautoreview程序上传,以及解压 4](#_Toc317152090)

[4. 创建存放日志的目录 4](#_Toc317152091)

[5. 配置sqlautoreview程序的配置文件 5](#_Toc317152092)

[6. 运行sqlautoreview程序 7](#_Toc317152093)

[第二部份:sqlautoreview的依赖 8](#_Toc317152094)

[第三部份:sqlautoreview的整体架构 10](#_Toc317152095)

[第四部份:sqlautoreview自动化审核的一些规则 11](#_Toc317152096)

[第五部份:sqlautoreview的扩展性 12](#_Toc317152097)

[第六部份：sqlautoreview后台程序与前台程序的配合 12](#_Toc317152098)

[6.1 sqlmapfile的解析 12](#_Toc317152099)

[6.2 create index方法的使用 13](#_Toc317152100)

[6.3 merge Index方法的使用 13](#_Toc317152101)

[第七部份:sqlautoreview自动化审核当前面临的一些困难 14](#_Toc317152102)

[第八部份:sqlautoreview展望 14](#_Toc317152103)

# 第一部份: sqlautoreview程序的安装

1. 安装jdk 1.6

网上有很多资料，google一下

1. 配置环境变量

环境变量只需要配置JAVA\_HOME,PATH两个环境变量,示例如下:

$cat .bash\_profile

# .bash\_profile

# Get the aliases and functions

if [ -f ~/.bashrc ]; then

. ~/.bashrc

fi

# User specific environment and startup programs

export JAVA\_HOME= /opt/install/jdk-1.6.0\_26

export PATH=$JAVA\_HOME/bin:$HOME/bin:$PATH

1. sqlautoreview程序上传,以及解压

将程序sqlautoreview-0.0.1-SNAPSHOT-sqlautoreview.tar.gz文件上传到linux服务器上,执行如下的解压命令:

$tar -xzvf sqlautoreview-0.0.1-SNAPSHOT-sqlautoreview.tar.gz

sqlautoreview-0.0.1-SNAPSHOT/lib/dom4j-1.6.1.jar

sqlautoreview-0.0.1-SNAPSHOT/lib/xml-apis-1.0.b2.jar

sqlautoreview-0.0.1-SNAPSHOT/lib/mysql-connector-java-5.1.18.jar

sqlautoreview-0.0.1-SNAPSHOT/lib/log4j-1.2.15.jar

sqlautoreview-0.0.1-SNAPSHOT/lib/mail-1.4.jar

sqlautoreview-0.0.1-SNAPSHOT/lib/activation-1.1.jar

sqlautoreview-0.0.1-SNAPSHOT/lib/jms-1.1.jar

sqlautoreview-0.0.1-SNAPSHOT/lib/jmxtools-1.2.1.jar

sqlautoreview-0.0.1-SNAPSHOT/lib/jmxri-1.2.1.jar

sqlautoreview-0.0.1-SNAPSHOT/lib/sqlautoreview-0.0.1-SNAPSHOT.jar

sqlautoreview-0.0.1-SNAPSHOT/bin/log4j.xml

sqlautoreview-0.0.1-SNAPSHOT/bin/start\_createindex.sh

sqlautoreview-0.0.1-SNAPSHOT/bin/start\_xmltosql.sh

sqlautoreview-0.0.1-SNAPSHOT/bin/tools-run-class.sh

sqlautoreview-0.0.1-SNAPSHOT/conf/productdb.xml

sqlautoreview-0.0.1-SNAPSHOT/conf/sqlmapfile.xml

sqlautoreview-0.0.1-SNAPSHOT/conf/sqlreviewdb.xml

1. 创建存放日志的目录

进入sqlautoreview解压过后的目录,创建logs目录,程序运行的所有的日志文件都放在这个地方.

$cd sqlautoreview-0.0.1-SNAPSHOT

[MM-Writable@test.machine ~/sqlautoreview-0.0.1-SNAPSHOT]

$ls

bin conf lib

[MM-Writable@test.machine ~/sqlautoreview-0.0.1-SNAPSHOT]

$mkdir logs

[MM-Writable@test.machine ~/sqlautoreview-0.0.1-SNAPSHOT]

$ls

bin conf lib logs

[MM-Writable@test.machine ~/ sqlautoreview-0.0.1-SNAPSHOT /logs]

$pwd

/home/danchen/ sqlautoreview-0.0.1-SNAPSHOT /logs

修改/home/danchen/ sqlautoreview-0.0.1-SNAPSHOT /bin目录下的log4j.xml日志输出配置文件,将路径改成/home/danchen/ sqlautoreview-0.0.1-SNAPSHOT /logs,你可以根据自己的系统设置日志的输出路径.

1. 配置sqlautoreview程序的配置文件

进入conf目录,下面会出现三个配置文件

[MM-Writable@test.machine ~/sqlautoreview-0.0.1-SNAPSHOT]

$cd conf

[MM-Writable@test.machine ~/sqlautoreview-0.0.1-SNAPSHOT/conf]

$ls

productdb.xml sqlmapfile.xml sqlreviewdb.xml

productdb.xml是这些SQL即将在哪个数据库上运行,即目标数据库的配置信息,会包括ip,port,dbname,user,password. 下面是配置文件的示例内容:

<?xml version="1.0" encoding="UTF-8"?>

<database>

<ip>192.168.168.100</ip>

<port>3306</port>

<dbname>testdb</dbname>

<user>test</user>

<password>test</password>

</database>

Sqlmapfile.xml配置文件主要是配置java sqlmapfile的两个信息,一个file\_id字段,你可以随意指定一个数字,最好不要与以前的指定相同; 另外一个字段为sqlmap的filename,这个必须要包含绝对路径,以下是一个linux下此路径的范例:

<?xml version="1.0" encoding="UTF-8"?>

<sqlmapfile>

<file\_id>-500</file\_id>

<file\_name>/home/danchen/itemonlinedetail\_sqlmap.xml</file\_name>

</sqlmapfile>

如果在windows下,此路径需要写成如下的形式,即路径中的\,要变成\\

<?xml version="1.0" encoding="UTF-8"?>

<sqlmapfile>

<file\_id>-20</file\_id>

<file\_name>D:\\itemonlinedetail\_sqlmap.xml</file\_name>

</sqlmapfile>

sqlreviewdb.xml配置文件的结构与productdb.xml相同,这个是sqlautoreview自己的数据库,所有审核的SQL,以及审核结果都会放在这个数据库里.此配置文件的示例如下:

<?xml version="1.0" encoding="UTF-8"?>

<database>

<ip>192.168.168.200</ip>

<port>3306</port>

<dbname>sqlreview</dbname>

<user>danchen</user>

<password>hellodanchen</password>

</database>

在示例的192.168.168.200 mysql上创建sqlreviewdb,以及xmltosql, mergeresult表,xmltosql, mergeresult的表结构如下:

CREATE TABLE `xmltosql` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`sqlmap\_file\_id` int(11) NOT NULL,

`java\_class\_id` varchar(200) NOT NULL,

`sql\_xml` varchar(4000) NOT NULL,

`sql\_comment` varchar(200) DEFAULT NULL,

`real\_sql` varchar(4000) NOT NULL,

`real\_sql\_hash` varchar(32) DEFAULT NULL,

`table\_name` varchar(100) DEFAULT NULL,

`status` int(11) DEFAULT NULL,

`auto\_review\_err` varchar(200) DEFAULT NULL,

`auto\_review\_tip` varchar(200) DEFAULT NULL,

`auto\_review\_time` datetime DEFAULT NULL,

`sql\_auto\_index` varchar(200) DEFAULT NULL,

`dba\_review\_time` datetime DEFAULT NULL,

`sql\_dba\_index` varchar(200) DEFAULT NULL,

`dba\_advice` varchar(200) DEFAULT NULL,

`gmt\_create` datetime NOT NULL,

`gmt\_modified` datetime NOT NULL,

PRIMARY KEY (`id`),

KEY `idx\_java\_class\_id` (`java\_class\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=gbk

CREATE TABLE `mergeresult` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`sqlmap\_file\_id` int(11) NOT NULL,

`tablename` varchar(30) DEFAULT NULL,

`real\_tablename` varchar(30) DEFAULT NULL,

`exist\_indexes` varchar(4000) DEFAULT NULL,

`new\_indexes` varchar(4000) DEFAULT NULL,

`merge\_result` varchar(4000) DEFAULT NULL,

`gmt\_create` datetime NOT NULL,

`gmt\_modified` datetime NOT NULL,

PRIMARY KEY (`id`),

KEY `idx\_result\_sqlmap\_file\_id` (`sqlmap\_file\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=gbk

1. 运行sqlautoreview程序

进入到bin目录下,对三个shell脚本,加上可执行+x的权限

[MM-Writable@test.machine ~/sqlautoreview-1.0-SNAPSHOT/bin]

$chmod +x \*.sh

[MM-Writable@test.machine ~/sqlautoreview-1.0-SNAPSHOT/bin]

$ls -l

total 20

-rw-r--r-- 1 danchen users 1900 Jan 9 09:36 log4j.xml

-rwxr-xr-x 1 danchen users 85 Jan 9 09:36 start\_createindex.sh

-rwxr-xr-x 1 danchen users 84 Feb 15 11:39 start\_mergeindex.sh

-rwxr-xr-x 1 danchen users 82 Jan 9 09:36 start\_xmltosql.sh

-rwxr-xr-x 1 danchen users 422 Jan 9 09:36 tools-run-class.sh

在上面三个.cnf文件配置好的情况下,首先需要执行start\_xmltosql.sh脚本,这个脚本完成的功能是把sqlmapfile.xml中配置的sqlmapfile从XML文件中解析出来，放入到sqlreviewdb数据库中。

[MM-Writable@test.machine ~/sqlautoreview-0.0.1-SNAPSHOT/bin]

$./start\_xmltosql.sh

如果log4j.xml配置的log4j log level为info，你会看到解析出来的SQL语句.

第二步执行创建索引的脚本，这个会读取sqlreviewdb中待审核的SQL语句,解析SQL，根据规则生成创建索引的脚本，并保存到sqlreviewdb中，整个审核结束。

[MM-Writable@test.machine ~/sqlautoreview-0.0.1-SNAPSHOT/bin]

$./start\_createindex.sh

如果log4j.xml配置的log4j log level为info，你会看到审核的一些细节,如果为debug会看到更多的细节。

Tips:

1. 支持MySQL,不支持Oracle
2. 不支持含有子查询的SQL
3. 支持普通的单表查询
4. 支持单表使用别名的查询
5. 支持单表的标准的分页
6. 支持多表的join

第三步是执行索引merge的脚本,

[MM-Writable@test.machine ~/ sqlautoreview-0.0.1-SNAPSHOT /bin]

$./start\_mergeindex.sh

这个会读取sqlreviewdb中的生成的索引,以及这次审核所涉及的表上原有的索引,应用一些规则,进行索引的自动合并,生成最后的变更脚本.

# 第二部份:sqlautoreview的依赖

Sqlautoreview是一个maven工程,它只依赖于三个jar包:

mysql-connector-java-5.1.18.jar

dom4j-1.6.1.jar

log4j-1.2.15.jar

原始的pom.xml文件如下:

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.taobao</groupId>

<artifactId>sqlautoreview</artifactId>

<version>1.0.0-SNAPSHOT</version>

<packaging>jar</packaging>

<name>sqlautoreview</name>

<properties>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

</properties>

<build>

<plugins>

<plugin>

<artifactId>maven-assembly-plugin</artifactId>

<configuration>

<descriptors>

<descriptor>release.xml</descriptor>

</descriptors>

</configuration>

</plugin>

</plugins>

<resources>

<resource>

<directory>src/main/resources</directory>

<filtering>true</filtering>

</resource>

</resources>

</build>

<dependencies>

<dependency>

<groupId>dom4j</groupId>

<artifactId>dom4j</artifactId>

<version>1.6.1</version>

</dependency>

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

<version>5.1.18</version>

</dependency>

<dependency>

<groupId>log4j</groupId>

<artifactId>log4j</artifactId>

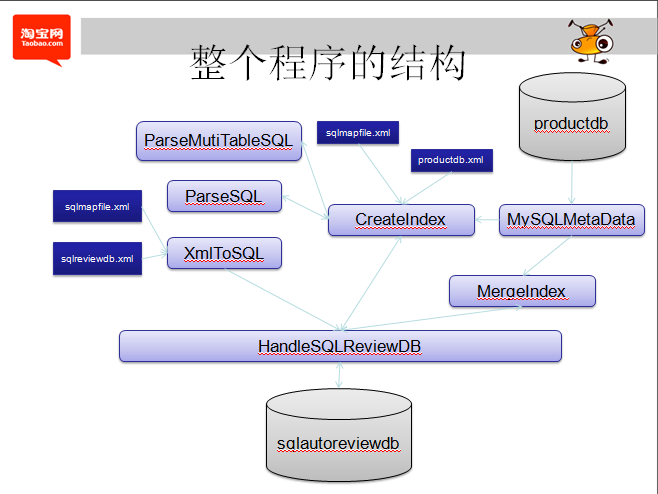
<version>1.2.15</version>

</dependency>

</dependencies>

</project>

# 第三部份:sqlautoreview的整体架构



XmlToSQL.java程序的功能: 把开发编写的sql map file中的sql从xml标签中解析出来.在这个解析的程序里,我使用了开源的dom4j遍历整个xml文件,获取到单个element节点,在单个element内部,自己编写解析程序,来处理各种特别的标签,比如include引用,动态append,CDATA转义<,>等.

XmlToSQL.java要运行正常,需要读取sqlmapfile.xml,sqlreviewdb.xml两个配置文件,以及sqlmapfile.xml里面配置的sql map file文件.

如果你的应用并不是java应用程序,你可以自己写程序,把你的程序里用到的sql写入到数据库sqlautoreviewdb的xmltosql表中.

Status字段有三个值,0代表未审核；1代表正常审核完成；2代表审核中出现错误。其它字段的含义通过字段名，应该很容易理解。这个是根据淘宝的一些特点设计的表结构，你需要根据自己的情况,调整这个表结构,比如你的系统是非java的程序,java\_class\_id字段可以删除.

XmlToSQL.java程序会把解析出来的SQL,通过HandleSQLReviewDB类的insertDB方法写入到sqlreviewdb中.

CreateIndex.java通过HandleSQLReviewDB类的getAllSQL方法,取出sqlmapfile.xml中file\_id所对应的待审核的SQL.取出每一条SQL,用ParseSQL或者ParseMutiTableSQL类进行分析处理.

在分析的过程中,会通过MySQLMataData类加载相应表的元数据,元数据主要包括表的字段元数据,索引的元数据,**以及表的所有字段的不同值个数的一个抽样统计,这个sample rows size is 10000 now.现在这个是硬编码,统计的行数越多,SQL自动化审核的结果也会更加精准. 如果你在productdb.xml配置文件中配置的是开发环境的database,表里面的数据没有,或者比较少,或者跟真实的数据分布有很大差异,那么审核的结果可能会不准**.

Tips: productdb.xml最好配置的是生产环境的数据库或者有数据的测试环境

每一条SQL审核完成后,CreateIndex类会把SQL审核的结果通过HandleSQLReviewDB类的updateSQLStatus方法更新到sqlreviewdb中.

MySQLMetaData类,主要的功能是做元数据加载,最早的一个版本是会加载productdb中配置的数据库中所有表的元数据,对于有些分库分表的系统,一个数据库中的表会非常多,加载的时间相当长.后来对这里进行了优化,对于SQL中出现的表,才加载这张表的元数据,大大提高了加载数度.现在在程序里,仍然有这个开关,默认是采用优化后的加载方式.

MergeIndex.java的功能是把属于这个sqlmap\_file\_id的所有SQL新建的索引,以及这些SQL所涉及的表上原有的索引进行合并,达到整体最优.

# 第四部份:sqlautoreview自动化审核的一些规则

1. 如果where等值查询字段中，是primary key的第一个字段，那么使用主键索引
2. 如果where等值查询字段中，是unique key的第一个字段，那么使用唯一键索引
3. 如果前两个条件不满足,则需要对所有条件字段分成三类,一类是等值操作字段,第二类是非等值操作字段,第三类是排序字段.对于前两类字段,会在这两类columns分别进行Cardinality排序,Cardinality越大,越有可能排在索引字段顺序的最前面.**索引字段的组合顺序(等值条件操作字段,排序操作字段,非等值条件操作字段)**
4. 如果SQL含有排序字段,会对排序字段的类型,及长度进行检查,如果排序字段中有一个column的type是varchar类型,并且定义的长度超过200,会出现一个Warnning
5. 现在的sqlautoreview程序,努力做到对于每一条SQL,所生成的创建索引的脚本是最优的.但局部最优,并不代表着全局最优.
6. 前面生成的单条最优的索引,会经历一个merge index的过程,综合所有新建索引,以及这张表原来的索引,自动进行索引合并,并产生最后的索引变更脚本.
7. merge index的一些规则: 1. 所有新建索引进行最简单的去重; 2. 新建索引的索引字段是否是已存在的索引字段的子集,并且索引字段顺序相同.比如一个新建的索引为ind\_test\_user(user,status),而原来表上存在一个索引ind\_test\_user(user,status,type),那么这个新建的索引就不需要建了,直接重用原来的索引.3.新建索引的内部也进行2的方式合并;4.以新建索引为标准,如果老的索引是新建索引的子集,老的索引会删除,然后创建一个新的索引.比如老的索引为ind\_test\_user(user,status),而新的索引为ind\_test\_user(user,status,type),这种情况也会产生合并:drop index ind\_test\_user on test;

Create index ind\_test\_user on test(user,status,type)

# 第五部份:sqlautoreview的扩展性

程序里设计了两个接口,一个是IMetaData接口类,一个是IHandleDB接口类.

我们的业务数据库有不同的数据库类型,比如常用的mysql,oracle,...对于不同的数据库,数据字典不同,所以元数据的获取方式有差异,在当前的sqlautoreview程序里,MySQLMetaData是IMetaData的具体实现,OracleMetaData还没有时间去实现.

对于保存sql审核结果的数据库sqlreviewdb,可以是oracle,也可以是mysql,或者是其它,所以设计了IHandleDB接口类, 对于mysql数据库，HandleSQLReviewDB是IHandleDB的具体实现

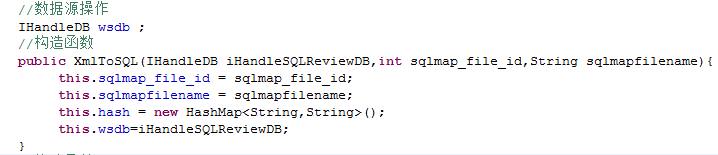
Sql parse的两个类（ParseSQL, ParseMutiTableSQL）要作少量的修改,就可以支持其它数据库了。

另外一点,sqlautoreview三大组件(xmltosql,createindex,mergeindex)可以独立运行,即如果你的程序并没有用java的sqlmap,那么你可以把这些SQL写到sqlreviewdb里面去,然后调用createindex,mergeindex类的方法,即可完成审核.

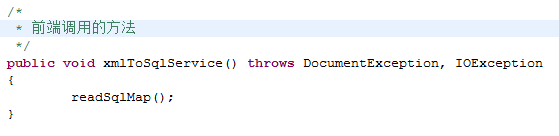
# 第六部份：sqlautoreview后台程序与前台程序的配合

## 6.1 sqlmapfile的解析

初始化构造函数：



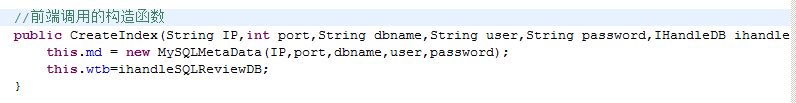
然后调用一个方法：



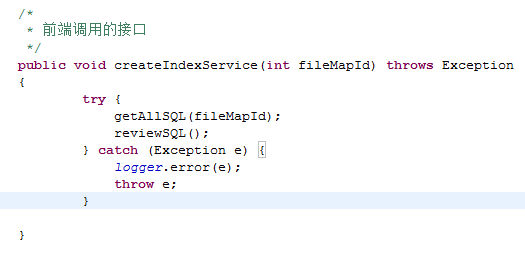
通过以上的步骤，就可以把解析出的SQL放入sqlreviewdb数据库的xmltosql表中。

6.2 create index方法的使用

初始化构造函数：



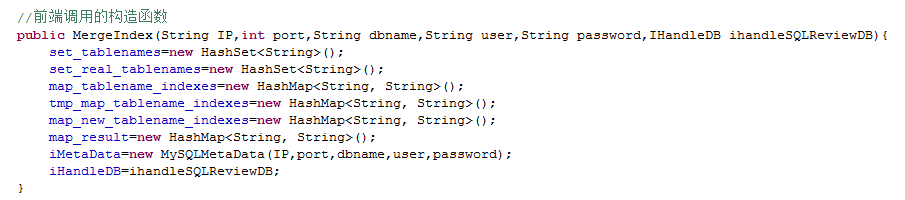
然后调用一个方法：



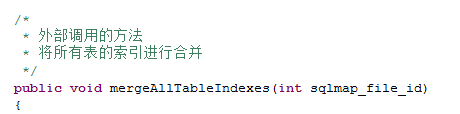
就可以把索引创建的结果保存入sqlreviewdb数据库的xmltosql表中。

6.3 merge Index方法的使用

初始化构造函数:



然后调用一个方法:



就可以合并sqlmap\_file\_id所涉及的所有SQL的新建索引,以及原来的索引,产生最后的变更脚本,并把相关的信息保存到sqlreviewdb数据库中的mergeresult表中.

与前台程序合并时,后台程序不需要conf目录下的三个配置文件了,所有配置信息都是由前端直接传递给后端程序的.

# 第七部份:sqlautoreview自动化审核当前面临的一些困难

1. 开发编写的sql map file的解析,仍然存在有些解析不对的地方,有些地方也需要去理解XML标签的语义,才能解析出一个正确的SQL语句
2. 提高SQL审核的精准度,当前主要是通过采样数据来理解业务的数据分布.
3. 当前版本只支持MySQL数据库，不支持oracle数据库.
4. 支持大部份的Mysql sql语句类型,不支持子查询,以及left join,right join这种写法的SQL
5. 此次开源的是sql自动化审核的后台程序,大家可以根据自己的情况,设计不同的前端页面,以及sqlreviewdb中的其它表,将整个功能web化,提高大家使用这个产品的用户体验.

# 第八部份:sqlautoreview展望

1. 希望越来越多的朋友,加入到这个开源软件的中来，不断完善，提高SQL审核质量，使数据库系统更稳定，解放全国的DBA朋友。
2. 希望越来越多的朋友试用这款软件，反馈bug, 这也是一种支持
3. 我也会继续在这个产品上努力，不断修正大家提出的bug，有任何使用上的问题，都可以联系淘宝的赵林/丹臣，给我发email: [danchen@taobao.com](mailto:danchen@taobao.com) [zhaolinjnu@gmail.com](mailto:zhaolinjnu@gmail.com)

或者新浪微博@me:丹臣