Database debugging with p6spy

Wiki Home | Deploying Grouper | Grouper Guides | Grouper Deployment Guide | Community Contributions | Internal Developer Resources

This is the new wiki for this page

p6spy is an open source project which is abandoned, but which is a very useful project. Note: the queries are not the EXACT queries, if it was a prepared statement with params, the params are substituted for the values

Example

If you turn on p6spy, each statement of db traffic, and each resultset will be printed to a log file. Here is an example:

```
 02-16-08 15:24:25:137, 0ms, statement: QueryImpl.java.iterate() line 46, Hib3GroupDAO.java._updateAttributes()
line 907, Hib3GroupDAO.java.onUpdate() line 657, Hib3MembershipDAO.java.update() line 867, Membership.java.
internal_addImmediateMembership() line 357, Group.java.addMember() line 405, Group.java.addMember() line 349,
Test_uc_WheelGroup.java.test_canAdminWhenMemberOfWheel() line 110, Test_uc_WheelGroup.java.main() line 51
   select hib3attrib0_.id as col_0_0_ from grouper_attributes hib3attrib0_ where hib3attrib0_.
group_id='8e076591-27d7-497b-84e7-ae50301b896b'
02-16-08 15:24:25:153, resultset: ALIASES: 0: id
02-16-08 15:24:25:153, resultset: 0: 402881821823e6b3011823eb3e34001f
02-16-08 15:24:25:153, resultset: 0: 402881821823e6b3011823eb3e340021
02-16-08 15:24:25:184, resultset: 0: 402881821823e6b3011823eb3e34001e
02-16-08 15:24:25:200, resultset: 0: 402881821823e6b3011823eb3e340020
02-16-08 15:24:25:246, 0ms, statement: JDBCTransaction.java.commit() line 106, Hib3MembershipDAO.java.update()
line 873, Membership.java.internal_addImmediateMembership() line 357, Group.java.addMember() line 405, Group.
java.addMember() line 349, Test_uc_WheelGroup.java.test_canAdminWhenMemberOfWheel() line 110,
Test_uc_WheelGroup.java.main() line 51
   insert into grouper_memberships (owner_id, member_id, list_name, list_type, mship_type, via_id, depth,
parent_membership, membership_uuid, creator_id, create_time, id) values ('8e076591-27d7-497b-84e7-
ae50301b896b', '3be9112b-5797-46e0-abc1-ee0a9847510b', 'members', 'list', 'immediate', null, 0, null, '8c39af0f-
7ecb-42c8-83d4-841b55bcd869', '28ed31c4-f000-4a25-91d1-e832f2c0bd14', 1203193464512,
'402881821823e6b3011823eb49020029')
02-16-08 15:24:25:262, 0ms, statement: JDBCTransaction.java.commit() line 106, Hib3MembershipDAO.java.update()
line 873, Membership.java.internal_addImmediateMembership() line 357, Group.java.addMember() line 405, Group.
java.addMember() line 349, Test_uc_WheelGroup.java.test_canAdminWhenMemberOfWheel() line 110,
Test_uc_WheelGroup.java.main() line 51
   update grouper_groups set uuid='8e076591-27d7-497b-84e7-ae50301b896b', create_time=1203193462293,
modifier_id='28ed31c4-f000-4a25-91d1-e832f2c0bd14', modify_time=1203193465090, create_source=null,
modify_source=null where id='402881821823e6b3011823eb3e25001d'
02-16-08 15:24:25:293, 15ms, statement: JDBCTransaction.java.commit() line 106, Hib3MembershipDAO.java.update()
line 873, Membership.java.internal_addImmediateMembership() line 357, Group.java.addMember() line 405, Group.
java.addMember() line 349, Test_uc_WheelGroup.java.test_canAdminWhenMemberOfWheel() line 110,
Test_uc_WheelGroup.java.main() line 51
   delete from grouper_attributes where id='402881821823e6b3011823eb3e340020'
02-16-08 15:24:25:293, 0ms, commit
```

- Each time the app is started (or app server restarted), the file will clear and overwrite. The file never rotates, so dont leave it on too long or it gets big
- Each line has a timestamp. mm-dd-yyyy
- Each line tells how long it took (e.g. 0ms, 15ms). Note this is not too accurate... its probably accurate to 10ms.
- The "statement" is the type of jdbc call. And after that is the horizontal stack of where the call came from:

JDBCTransaction.java.commit() line 106, Hib3MembershipDAO.java.update() line 873 ...

- When there is a resultset, first the columns are printed (the alias in the 0 index column is "id"):

02-16-08 15:24:25:153, resultset: ALIASES: 0: id

- Then the data is printed in the same column order as the columns (the data in the 0 column is a string 402881821823e6b3011823eb3e34001f)

02-16-08 15:24:25:153, resultset: 0: 402881821823e6b3011823eb3e34001f

Configure

- Change the DB driver wherever you want to log (e.g. grouper.hibernate.properties, sources.xml)
  to: com.p6spy.engine.spy.P6SpyDriver
- Make sure there is a spy.properties file in the classpath in the default package. Customize the conf/spy.example.properties in grouper
- Make sure the p6spy.jar is in the lib dir (its in grouper/lib)

- In the spy.properties, specify the underlying driver (e.g. for mysql): realdriver=com.mysql.jdbc.Driver
- Set the logfile in the spy.properties: logfile=f\:\\temp
  grouperSpy.log
- Other useful things in the config file: you can filter out things you dont want to see based on patterns, if you are looking for long running queries you can restrict a threshold of millis, you can print full stacktrace (vertical) for each statement (not recommend), you can set it to not delete file on start

Customizations Made to P6SPY

Again, this is a defunct project, so no need to ever upgrade it.

I customized it to work with grouper, and I can further tweak it if people have suggestions (right now the queries with bind vars are printed out with dates and timestamps in oracle format... I could detect which underlying driver is used and give output for other db's). The source and build script are in the jar, though some other jars are needed to compile it.

Some things I did are:

1. Substitute the values of prepared statement params so the query can be runable (note, might not work if not oracle... might need to tweak it)
2. Put a horizontal stacktrace on each call (ignore DB driver parts)
3. In the resultset, print the column headers in one line, then each subsequent line is the data that was read
4. Add timing for each statement
5. Improve the filtering for long running queries
6. Take some things out that didnt work (it converted columns by index to name, but that didnt work in mysql when a column name is "id"). Not sure why it existed anyway