

How to Trace Trigger Actions Using Oradebug

Tracing trigger actions is easy to do in oradebug, you simply set the 10309 event using a level 1 trace.

```
$ sqlplus -s /nolog
SQL> connect / as sysdba
Connected.
SQL> oradebug event 10309 trace name context forever, level 1
Statement Porcessed.
```

Don't forget to turn it off using context off when you are done.

Checking on Temporary Segment Usage with Oradebug

Temporary segments are created by sorts, hashes, temporary table usage and DDL statements like "create table as select .." or "alter index ... rebuild;".

When a temporary segment involves a DDL statement, the new object is originally created as a TEMPORARY segment in the target tablespace and is changed to the appropriate object type (table, index) when DDL action is finished.

You can check whether such action is active using ORADEBUG in the following manner:

1. connect internal or in Oracle9i sys as sysdba
2. SELECT owner FROM dba_segments WHERE segment_name=<temporary_segment>;
3. SELECT pid FROM v\$process WHERE username=<owner of segment>;
4. oradebug setorapid <pid>
5. oradebug dump errorstack 3
6. This will generate trace files, which contain the "current sql statement".
7. If this is a CTAS or alter index ... rebuild, then you will have to wait for the DDL action to finish.