



Abakus

ARBITER

Audit trail management

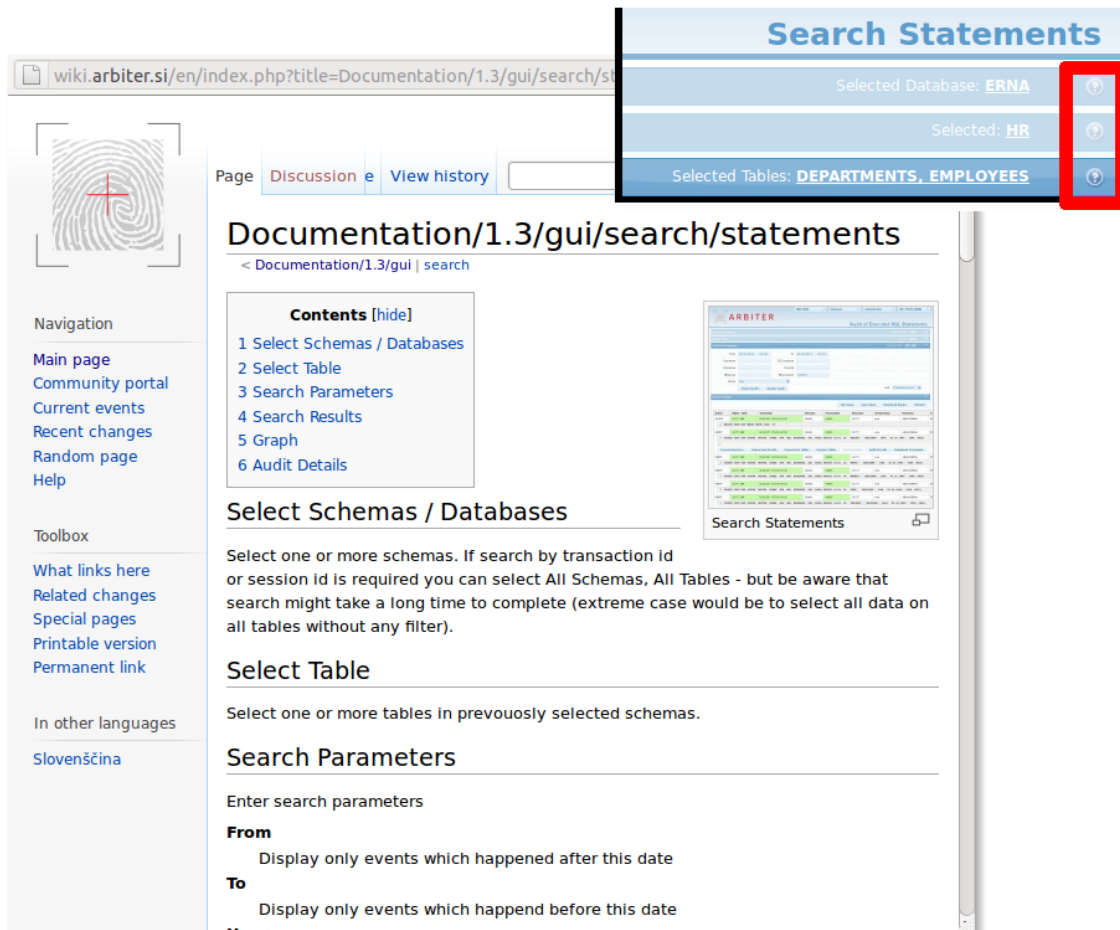
Quick-Start Guide

Table of Contents

| | |
|---|----|
| Reference Manual..... | 1 |
| GUI Features Overview..... | 2 |
| Buttons..... | 2 |
| Tables..... | 2 |
| Data Model..... | 4 |
| SQL Statements..... | 5 |
| Changed Data..... | 5 |
| Sessions..... | 5 |
| Transactions..... | 6 |
| Walkthrough the Graphical User Interface..... | 7 |
| First Page..... | 7 |
| Sessions..... | 8 |
| SQL Statements..... | 9 |
| Changed Data..... | 11 |
| Contact and Technical Support..... | 12 |

This is quick start guide for end-users, auditors. Administration is covered in separate guide "Arbiter for administrators".

Reference Manual

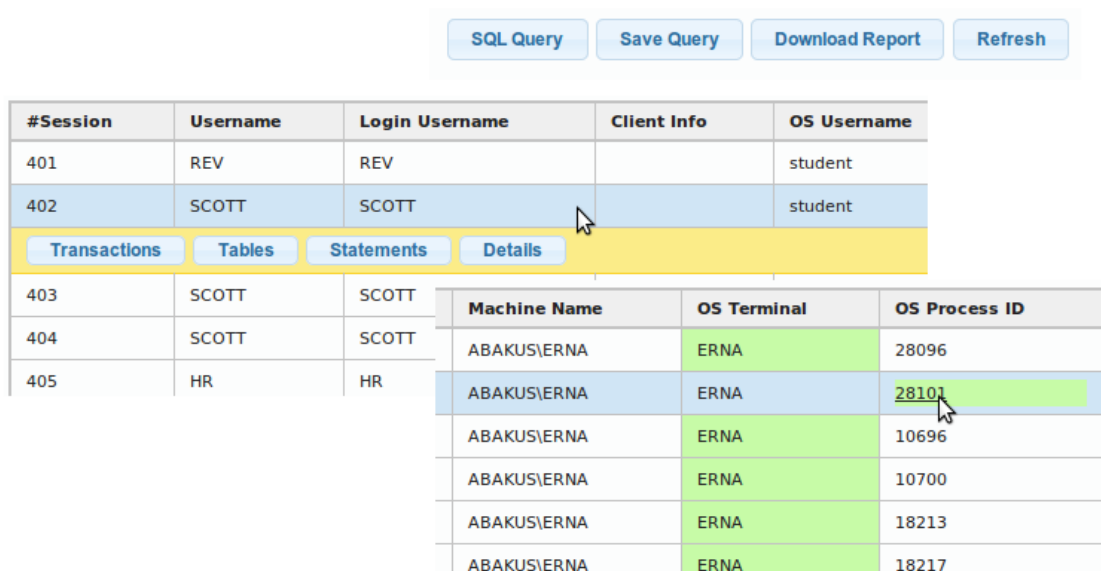


This guide explains main features of graphical user interface. However, any detailed examination will require exact column definitions to completely understand the data presented. This is why Arbiter has **integrated help links** on every page. Help page is accessible by clicking blue question-mark icon in the top right corner (as outlined in picture above). Click should open www.arbiter.si with relevant help content.

Warning

- Documentation is published on internet, which is why client will require working internet connection in order to access help pages. Otherwise, internet connection is not a requirement.
- Documentation is written in english - this guide is the only part accessible in other languages.

GUI Features Overview



The screenshot shows a GUI interface with four buttons at the top: "SQL Query", "Save Query", "Download Report", and "Refresh". Below the buttons are two tables. The first table has columns: #Session, Username, Login Username, Client Info, and OS Username. The second table has columns: Machine Name, OS Terminal, and OS Process ID. A yellow bar with tabs "Transactions", "Tables", "Statements", and "Details" is positioned between the two tables.

| #Session | Username | Login Username | Client Info | OS Username |
|----------|----------|----------------|-------------|-------------|
| 401 | REV | REV | | student |
| 402 | SCOTT | SCOTT | | student |

| Machine Name | OS Terminal | OS Process ID |
|--------------|-------------|---------------|
| ABAKUS\ERNA | ERNA | 28096 |
| ABAKUS\ERNA | ERNA | 28101 |
| ABAKUS\ERNA | ERNA | 10696 |
| ABAKUS\ERNA | ERNA | 10700 |
| ABAKUS\ERNA | ERNA | 18213 |
| ABAKUS\ERNA | ERNA | 18217 |

The two tables above show example of interactive report. Arbitrator supports a few different reports which we are going to examine in next sections. User experience is the same regardless of the exact report. Every report has following features:

Buttons

- **SQL Query:** Displays SQL statement which was used to produce the report currently displayed by the GUI. You can use that to get results from tools like SQL Developer or SQL*Plus.
- **Save Query:** Saves parameters of the search so you can simply repeat the search with the same parameters anytime later.
- **Download Report:** Exports results as PDF, HTML or CSV format. Some reports support choosing of user-defined column list.
- **Refresh:** Refresh results displayed. Actually, it runs the query again – any new data which match current parameters is displayed.

Tables

Most tables are support **clicking on individual rows** to display additional options. Those options depend on report type. This is how you can get to the detailed data of a record (sometimes screen is simply not wide enough to display everything) or you can display

another report with parameters based on currently selected row.

Green columns are those which have filters defined. In previous example, there is filter defined to display only rows which have `OS Terminal = 'ERNA'`.

Some results are colored green if you roll over them with mouse – in previous example mouse pointer is on column `OS Process ID`. This is a shortcut to add a filter based on this column/row, eg, by clicking on 28101 in above screenshot, filter will be set to `Terminal = 'ERNA' AND OS Process ID = '28101'`

Data Model

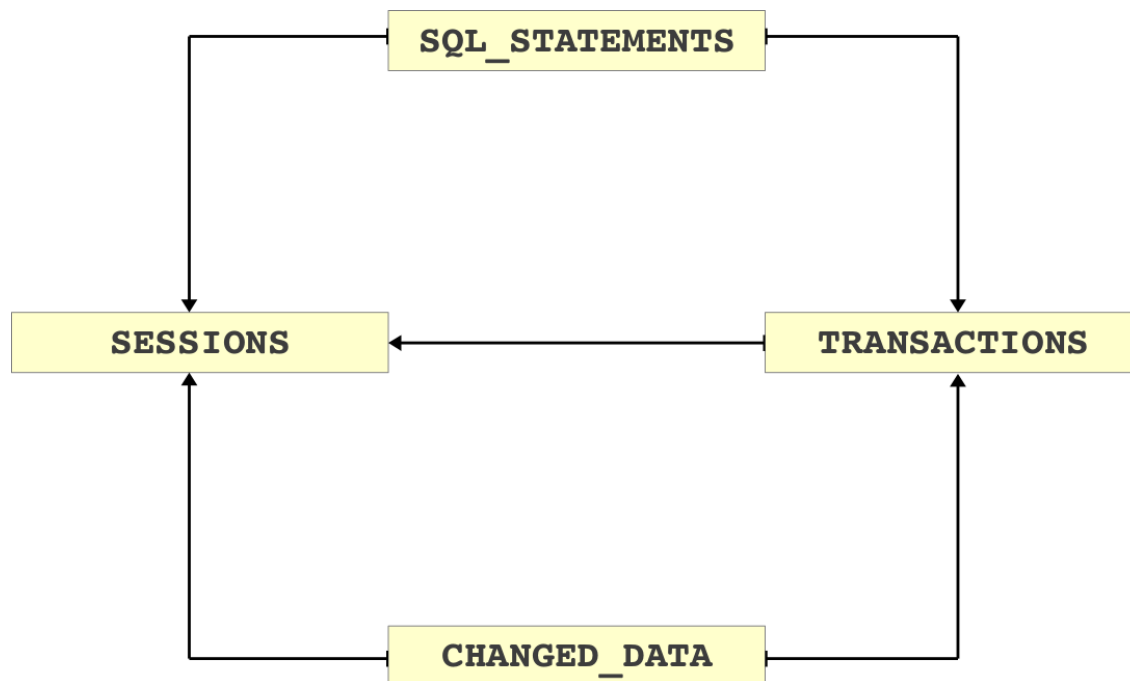


Image above depicts data model on which the navigation and report relations in GUI are built.

Starting point could be

- **SQL Statements** – actual SQL queries sent to database by the user.
- **Changed Data** – old and new values for data in tables. Eg, if someone has changed the salary column, this is where you can see who changed it, when and what was the salary before the change was made.
- **Sessions** – every connection to database creates a session. And Arbiter has a list of all the sessions: how was logged on from where and for how long.

Transactions are not the best starting point because they're part of a session and you would have to find the session of interest before you can find a transaction within.

SQL Statements

Data source is usually AUDIT TRAIL (at least for Oracle, depends on source database type). Main data available for each SQL statement:

- **Action:** Type of statement, eg INSERT or SELECT.
- **Object/Table:** Name of object on which action was executed
- **Timestamp:** Time when the statement was executed
- **Username:** Database username of user who executed the query.
- **Hostname:** Address or domain name of client computer from which the session originated.
- **SQL Text:** Exact SQL statement
- **Bind Variables:** Parameter with which the SQL statement was run.

Changed Data

Data source are usually ARCHIVED REDO LOG files (at least for Oracle, depends on source data type). Main data available for each record:

- **Operation:** INSERT, UPDATE or DELETE
- **Timestamp:** Time when the change occurred
- **User:** Database username of user who executed the query
- **Old and new value for each row and each column:** well.. Not necessarily every column, this is configurable.

Sessions

Sessions are created based on LOGIN and LOGOFF events from previously mentioned AUDIT TRAIL. Main data available for each session:

- **#Session:** Unique session identifier (assigned by Arbiter)
- **Username:** Database username of a user who created the session (by logging on the database)
- **Logon & Logoff time:** Time when the logon and logoff was made. This is also the basis for how long the session lasted.

- **Machine Name:** Name of client computer from which user logged on.
- **OS Username:** Local operating system username of a user who created the session (eg. Windows Username)
- **OS Program name:** Name of program used by client to connect to database.

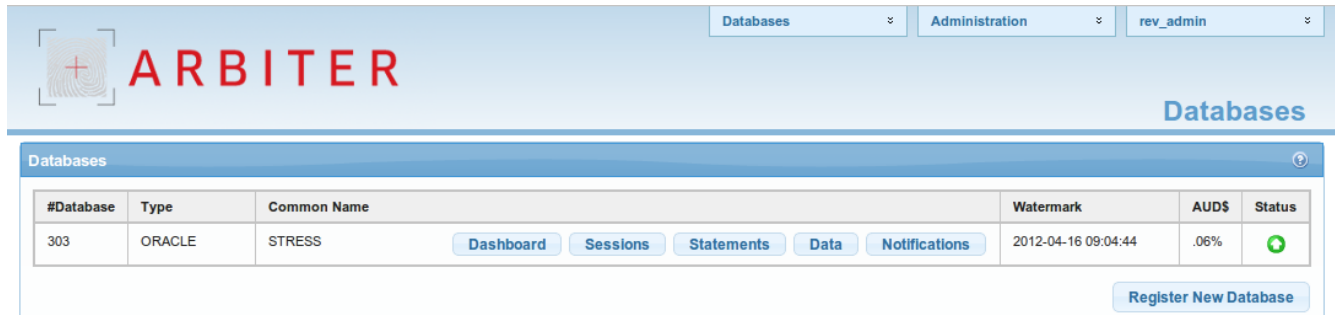
Transactions

Transactions are created based on old/new data which also contains COMMIT/ROLLBACK operations. Main data available for each transaction:

- **#Transaction:** Unique number identifier for each transaction (assigned by Arbiter)
- **Start & End time:** Time when transaction started(first change) and ended (commit/rollback)
- **Committed:** TRUE if transaction was committed or not. Careful rollback to savepoint set this column to TRUE .

Walkthrough the Graphical User Interface

First Page



The screenshot shows the ARBITER GUI interface. At the top, there is a navigation bar with three dropdown menus: 'Databases', 'Administration', and 'rev_admin'. The main header area contains the ARBITER logo on the left and the word 'Databases' on the right. Below the header, there is a table titled 'Databases' with the following columns: '#Database', 'Type', 'Common Name', 'Watermark', 'AUD\$', and 'Status'. A single row is visible in the table for database ID 303, which is of type ORACLE and named STRESS. The Watermark is 2012-04-16 09:04:44, and the AUD\$ is .06%. The Status column shows a green plus icon. Below the table, there are several buttons: 'Dashboard', 'Sessions', 'Statements', 'Data', 'Notifications', and a 'Register New Database' button.

| #Database | Type | Common Name | Watermark | AUD\$ | Status |
|-----------|--------|-------------|---------------------|-------|--------|
| 303 | ORACLE | STRESS | 2012-04-16 09:04:44 | .06% | + |

First page contains list of all registered databases for which audit has access to search their audit trail. Image depicts single registered database (named `STRESS`, with unique database id of 303).

Watermark is time up to which all data is processed and available to be displayed via GUI. Data after this date can be incomplete or not available at all (yet).

AUD\$ displays space usage on source database where the audit trail is temporarily stored until it is copied to Arbiter. This number is important, because source database may refuse to perform any more work if it cannot store audit to this space. This should never reach 100%!

Status displays green icon if all background jobs are running correctly. It can be yellow or red if any of the background jobs is disabled or encountered an error/warning. You should notify administrator about such state.

Sessions

The screenshot shows the ARBITER web interface. At the top, there are navigation tabs for 'STRESS (303)', 'Databases', 'Administration', and 'rev_admin'. The main header includes the ARBITER logo and a 'Search Sessions' button. Below this is a 'Search Parameters' section with various input fields: 'From' (03.07.2012 07:00), 'To' (03.07.2012 16:00), '#Session', 'Username', 'Login Username', 'Machine Name', 'OS Process ID', 'Ignore rev_src_user' (checked), 'OS Username', 'OS Terminal', and 'OS Program name'. There are 'Show Sessions' and 'Stored Queries' buttons. The 'Search Results' section features a search filter bar and buttons for 'SQL Query', 'Save Query', and 'Refresh'. Below the filter is a table with 13 columns: #Session, Username, Login Username, Client Info, OS Username, Machine Name, OS Terminal, OS Process ID, OS Program name, Logon, Logoff, Return Code, and No. S#. The table contains 10 rows of session data.

| #Session | Username | Login Username | Client Info | OS Username | Machine Name | OS Terminal | OS Process ID | OS Program name | Logon | Logoff | Return Code | No. S# |
|----------|----------|----------------|-------------|-------------|-----------------|-------------|---------------|-----------------|---------------------|---------------------|--|--------|
| 2646 | BOB | | | | | | | | 03.07.2012 13:29:37 | 03.07.2012 13:29:37 | 28001: ORA-28001: the password has expired | 0 |
| 2645 | ALICE | ALICE | | oracle | atlas.abakus.si | pts/0 | 23424 | | 03.07.2012 13:29:11 | 03.07.2012 13:29:12 | 0: Authenticated by: DATABASE | 1 |
| 2644 | ERNA | | | | | | | | 03.07.2012 13:27:54 | 03.07.2012 13:27:54 | 1017: ORA-01017: invalid username/password; logon denied | 0 |
| 2643 | URH | URH | | urh | urh | pts/3 | 23333 | | 03.07.2012 13:27:42 | 03.07.2012 13:27:44 | 0: Authenticated by: DATABASE; Client address: (ADDRESS=(PROTOCOL=tcp) (HOST=193.138.47.205) (PORT=60539)) | 1 |
| 2636 | SCOTT | SCOTT | | oracle | atlas.abakus.si | UNKNOWN | 23162 | | 03.07.2012 13:25:03 | 03.07.2012 13:25:05 | 0 | 1 |
| 2618 | SCOTT | SCOTT | | oracle | atlas.abakus.si | UNKNOWN | 19571 | | 03.07.2012 12:25:03 | 03.07.2012 12:25:05 | 0 | 1 |
| 2605 | SCOTT | SCOTT | | oracle | atlas.abakus.si | UNKNOWN | 14667 | | 03.07.2012 11:25:03 | 03.07.2012 11:25:05 | 0 | 1 |
| 2593 | SCOTT | SCOTT | | oracle | atlas.abakus.si | UNKNOWN | 11109 | | 03.07.2012 10:25:03 | 03.07.2012 10:25:05 | 0 | 1 |

Screenshot displays sessions (list of logins on database) for at 03.07.2012 between 7:00 and 16:00.

Ignore rev_src_user option means that output ignores sessions created by REV_SRC_USER on source database. This is Arbiter database account used to fetch audit trail data. Number of such sessions can be quite big (cca 10 connections each hour).

#Session is unique session ID. If session lasts longer that Arbiter open-sessions-cache allows (default is 14 days, depends on configuration), then such session can be represented as two separate sessions.

Another important field is **Username**, which displays username of database user account to which login was made. **Return Code** shows whether or not login was successful, **Machine Name** displays domain name or address of client computer.

SQL Statements

The screenshot shows the ARBITER web application interface. At the top, there are navigation tabs for 'STRESS (303)', 'Databases', 'Administration', and 'rev_admin'. The main header includes the ARBITER logo and a 'Search Statements' button. Below the header, there are sections for 'Select Schemas', 'Select Table', and 'Search Parameters'. The search parameters section includes fields for 'From' (02.07.2012 00:00) and 'To' (02.07.2012 23:59), 'Username', 'OS Username', 'Hostname', 'Terminal', '#Session', and '#Transaction'. There are also buttons for 'Show Results' and 'Render Graph', and a 'Count By' dropdown set to 'Do NOT Count' and a 'Sort' dropdown set to 'Timestamp (Desc)'. The 'Search Results' section shows a table of SQL statements with columns for Action, Object / Table, Timestamp, #Session, #Transaction, Username, OS Username, Hostname, and Terminal. The table contains several rows of data, including an UPDATE statement and several INSERT statements. Below the table, there are buttons for 'SQL Query', 'Save Query', 'Download Report', and 'Refresh'. A 'Transaction Data' button is also visible.

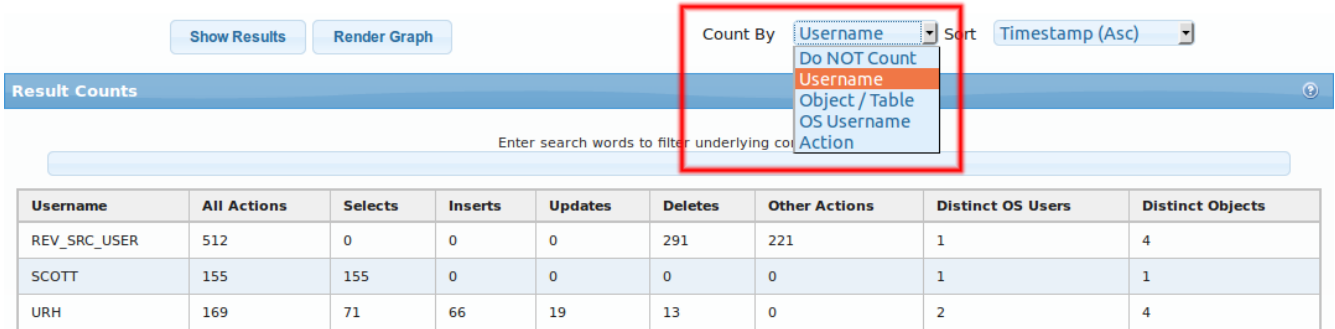
| Action | Object / Table | Timestamp | #Session | #Transaction | Username | OS Username | Hostname | Terminal |
|---|----------------|---------------------|----------|--------------|----------|-------------|-----------------|----------|
| UPDATE | SCOTT.DEPT | 02.07.2012 16:17:46 | 2328 | 204 | URH | oracle | atlas.abakus.si | pts/1 |
| update scott.dept set loc = 'BLED' where deptno = 37 | | | | | | | | |
| INSERT | SCOTT.DEPT | 02.07.2012 16:17:46 | 2328 | 201 | URH | oracle | atlas.abakus.si | pts/1 |
| insert into scott.dept (deptno, dname, loc) values (40, 'OPERATIONS', 'BOSTON') | | | | | | | | |
| INSERT | SCOTT.DEPT | 02.07.2012 16:17:46 | 2328 | 201 | URH | oracle | atlas.abakus.si | pts/1 |
| insert into scott.dept (deptno, dname, loc) values (30, 'SALES', 'SENCUR') | | | | | | | | |
| INSERT | SCOTT.DEPT | 02.07.2012 16:17:46 | 2328 | 201 | URH | oracle | atlas.abakus.si | pts/1 |
| insert into scott.dept (deptno, dname, loc) values (20, 'RESEARCH', 'KRANJ') | | | | | | | | |
| INSERT | SCOTT.DEPT | 02.07.2012 16:17:46 | 2328 | 201 | URH | oracle | atlas.abakus.si | pts/1 |
| insert into scott.dept (deptno, dname, loc) values (10, 'ACCOUNTING', 'SENCUR') | | | | | | | | |
| INSERT | SCOTT.DEPT | 02.07.2012 16:17:46 | 2328 | 201 | URH | oracle | atlas.abakus.si | pts/1 |
| insert into scott.dept (deptno, dname, loc) values (44, 'MARKETING', 'KRANJ') | | | | | | | | |

From top to bottom: Selected database is STRESS, selected schema is SCOTT and the two selected tables are EMPLOYEES and DEPT. Data is filtered by timestamp for date 02.07.2012, from 00:00 to 23:59.

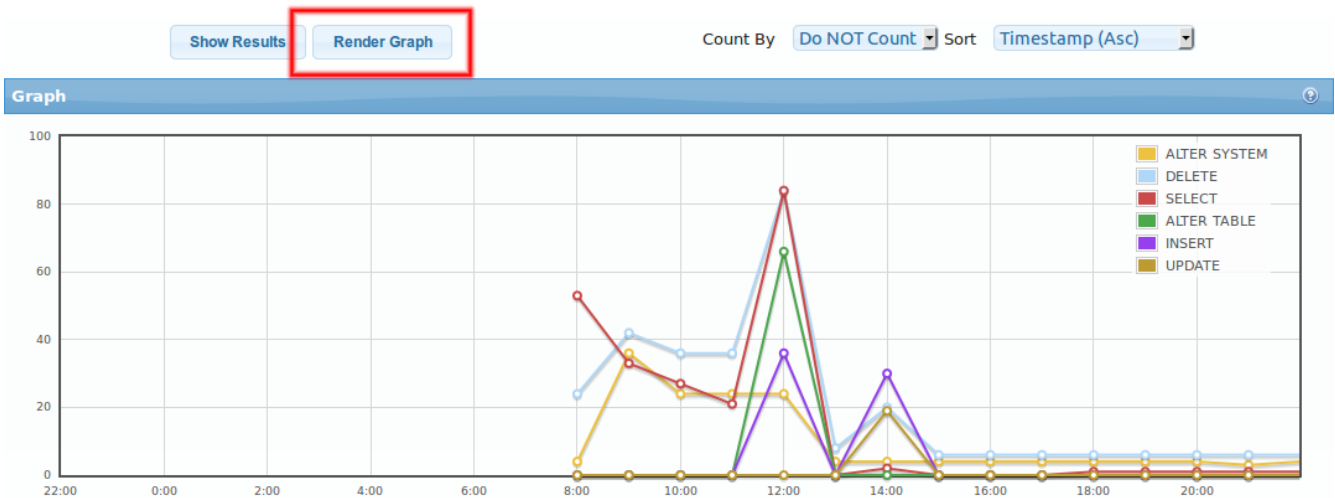
There are exact SQL statements (gray rows) and data about the user who executed them (username, OS username, hostname, ...). Button Audit Details displays even more details about the user.

Besides exact SQL statements, there are counts and graphs available as displayed in following screenshots.


Here are displayed counting options. We can get counts by username, by object or by operating system username. Based on that selection, number of actions by action type (columns Selects, Inserts, Updates, Deletes) are displayed.



Another way to see high-level data is to display a graph based on specified filter. Graph by itself is not partitioned by users/object but you can specify such filter that only values for single user or object are displayed. To render the graph, click the Render Graph button.



Changed Data



ARBITER

STRESS (303) ▾
Databases ▾
Administration ▾
rev_admin ▾

Search Data Tables

Select Schemas Selected Database: **STRESS**

Select Table Selected: **SCOTT [.]**

Select Schemas Selected Tables: **PRODUCTS**

From
To

Operation

#Transaction

Transactions Sort
Actions Sort

Search Results

| #Transaction 2593 (19.07.2012 09:24:50 - 19.07.2012 09:24:50), #Session 13177 (19.07.2012 09:24:44) | | | | PRODUCT_ID | | PRODUCT_NAME | | PRODUCT_PRICE | | |
|---|--------------------|---------------------|----------------|---------------------|-----|--------------|----------------|---------------|--------|--------|
| PRODUCTS (SCOTT.PRODUCTS) | User | Operation | Table | Timestamp (start) | OLD | NEW | OLD | NEW | OLD | NEW |
| | SCOTT | UPDATE | SCOTT.PRODUCTS | 19.07.2012 09:24:50 | 1 | | Woody | | 92-67 | 10 |
| | SCOTT | UPDATE | SCOTT.PRODUCTS | 19.07.2012 09:24:50 | 1 | | Woody | | 10 | 20 |
| | SCOTT | UPDATE | SCOTT.PRODUCTS | 19.07.2012 09:24:50 | 1 | | Woody | | 20 | 30 |
| #Transaction 2594 (19.07.2012 09:24:50 - 19.07.2012 09:24:50), #Session 13177 (19.07.2012 09:24:44) | | | | PRODUCT_ID | | PRODUCT_NAME | | PRODUCT_PRICE | | |
| | SCOTT | UPDATE | SCOTT.PRODUCTS | 19.07.2012 09:24:50 | 2 | | Buzz-Lightyear | | 30-28 | 10 |
| | SCOTT | UPDATE | SCOTT.PRODUCTS | 19.07.2012 09:24:50 | 2 | | Buzz-Lightyear | | 10 | 20 |
| Transaction Statements | Session Statements | Transaction Details | Data Details | | | | | | | |
| | SCOTT | UPDATE | SCOTT.PRODUCTS | 19.07.2012 09:24:50 | 2 | | Buzz-Lightyear | | 20 | 30 |
| | SCOTT | UPDATE | SCOTT.PRODUCTS | 19.07.2012 09:24:50 | | | | | | 20 |
| | SCOTT | UPDATE | SCOTT.PRODUCTS | 19.07.2012 09:24:50 | | | | | | 10 |
| | SCOTT | UPDATE | SCOTT.PRODUCTS | 19.07.2012 09:24:50 | | | | | | 30, 28 |
| #Transaction 2595 (19.07.2012 09:24:50 - 19.07.2012 09:24:50), #Session 13177 (19.07.2012 09:24:44) | | | | PRODUCT_ID | | PRODUCT_NAME | | PRODUCT_PRICE | | |
| | SCOTT | UPDATE | SCOTT.PRODUCTS | 19.07.2012 09:24:50 | 3 | | Etch | | 102-27 | 10 |

This screenshot displays order of data changes for table PRODUCTS in schema STRESS.

Table has gray headers, those are transactions within which are single actions (updates, inserts and deletes). For each action, there are columns to show which user has done what on which table when. And the main thing: There is list of all columns from PRODUCTS table and all previous values before the current one!

Some rows have yellow warning icon – those rows are generated by ROLLBACK operation.

Contact and Technical Support

We are available for any questions you may have:

- by e-mail: arbiter@abakus.si
- by phone: 04 287 11 00

Web Links:

<http://www.arbiter.si/en/documentation>

<http://wiki.arbiter.si/>

<http://www.abakus.si/>

