Abakus

ARBITER

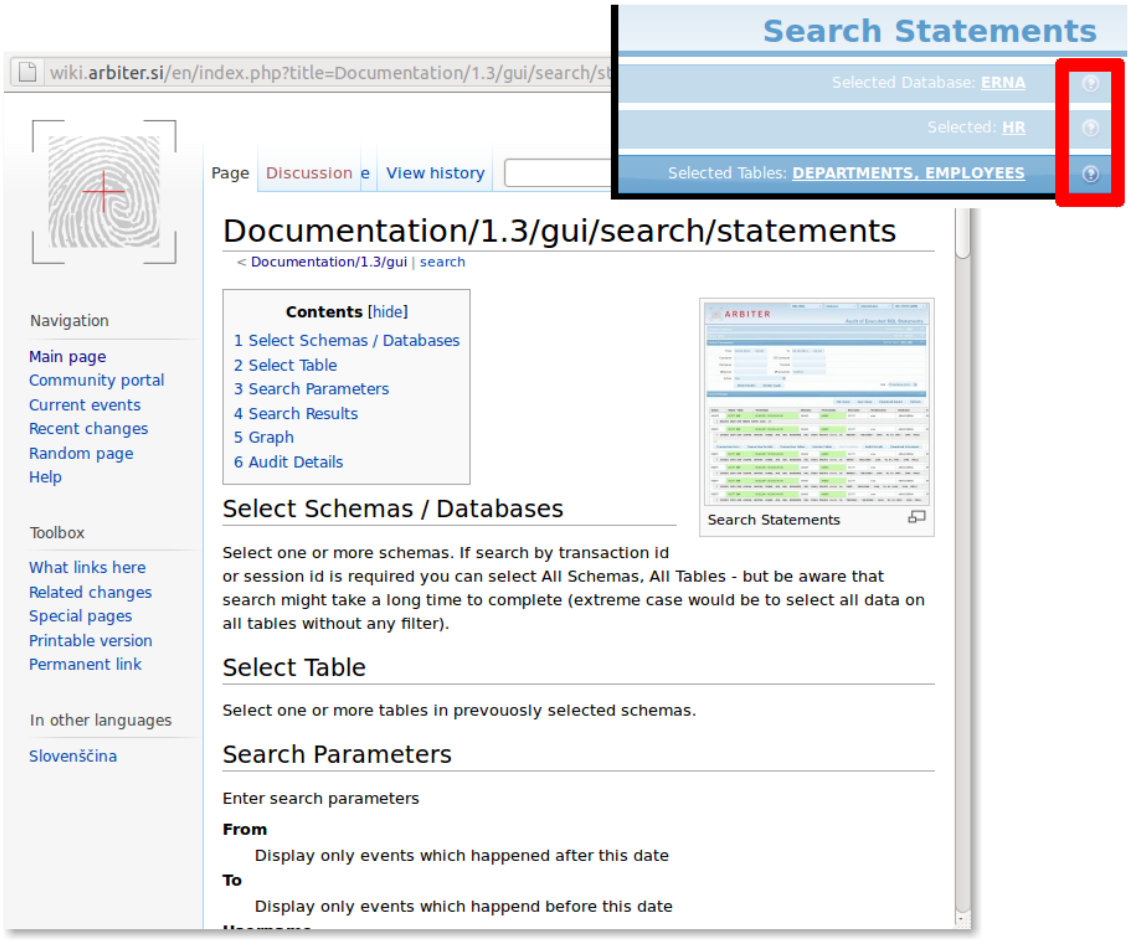
Sistem vodenja revizijskih sledi

hiter začetek uporabe za revizorje

Kazalo

Na naslednjih straneh je opisana uporaba Arbitra za revizorje oz za končne uporabnike. Za področje administracije si lahko preberete dokument "Arbiter za administratorje".

Uradna dokumentacija

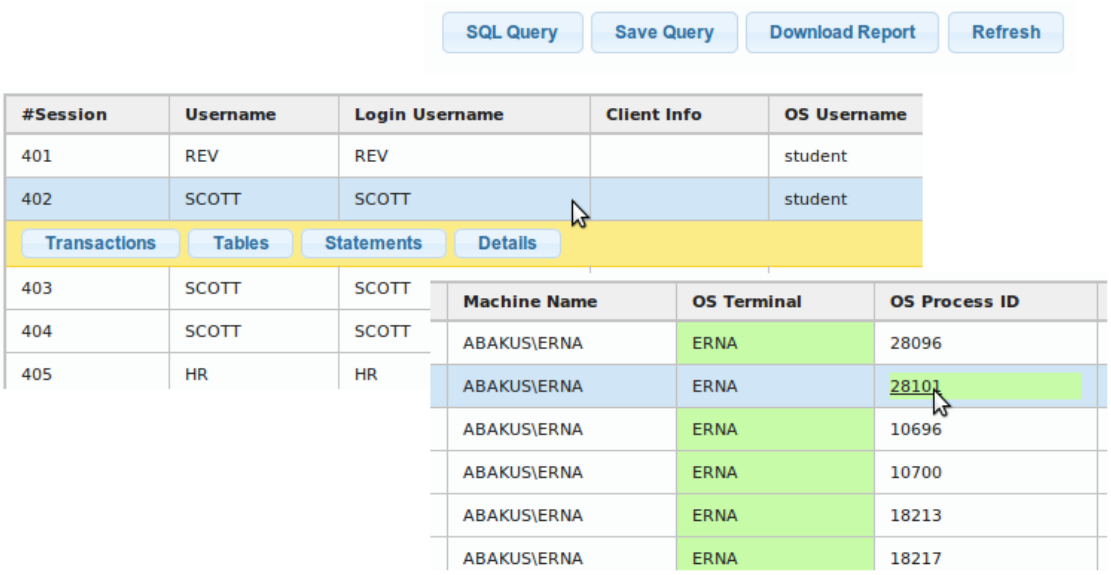


V tem vodiču so opisane glavne lastnosti grafičnega vmesnik. Vendar pa boste ob podrobnem pregledu najbrž potrebovali podrobno razlago posameznih vnosnih polj ali prikazanih kolon. Zato ima Arbiter **pomoč vgrajeno v uporabniški vmesnik**. Le-ta je dostopna s klikom na ikono v podobi vprašaja, ki se vedno nahaja na desni strani (kot prikazuje slika desno zgoraj). Ob kliku se odpre spletna stran wiki.arbiter.si na kateri se prikaže dokumentacija za trenutni zaslonski prikaz.

Opozorila

- Za delovanje povezave do dokumentacije odjemalec potrebuje dostop do interneta, ki sicer za samo delovanje Arbitra in njegovega uporabniškega vmesnika ni potreben.
- Dokumentacija je napisana v angleščini in je precej podrobna. Zato smo za splošen pregled funkcionalnosti in uporabe so pripravili tale priročnik. Podoben priročnik obstaja tudi za administracijo.

Opis osnovnih gradnikov



Zgornji tabeli prikazujeta primer interaktivnega poročila. Arbiter podpira nekaj vrst različnih poročil, ki si jih bomo ogledali na naslednjih straneh, vendar pa je uporabniška izkušnja ne glede na vrsto poročila enaka. Na vsakem poročilo so omogočeni sledeče možnosti:

Gumbi

- **SQL Query:** Izpiše SQL stavek z izvedbo katerega lahko pridete do enakih rezultatov ročno iz okolja kot je npr. SQL Developer ali SQL*Plus.
- **Save Query:** Omogoča shranjevanje parametrov poizvedbe, tako da lahko v bodoče enostavno ponovite trenutno iskanje z enakimi parametri.
- **Download Report:** Omogoča prenos (oz. izvoz) rezultatov v oblikah PDF, HTML ali CSV.
- **Refresh:** Omogoča osvežitev rezultatov (ponovno požene poizvedbo z enakimi parametri).

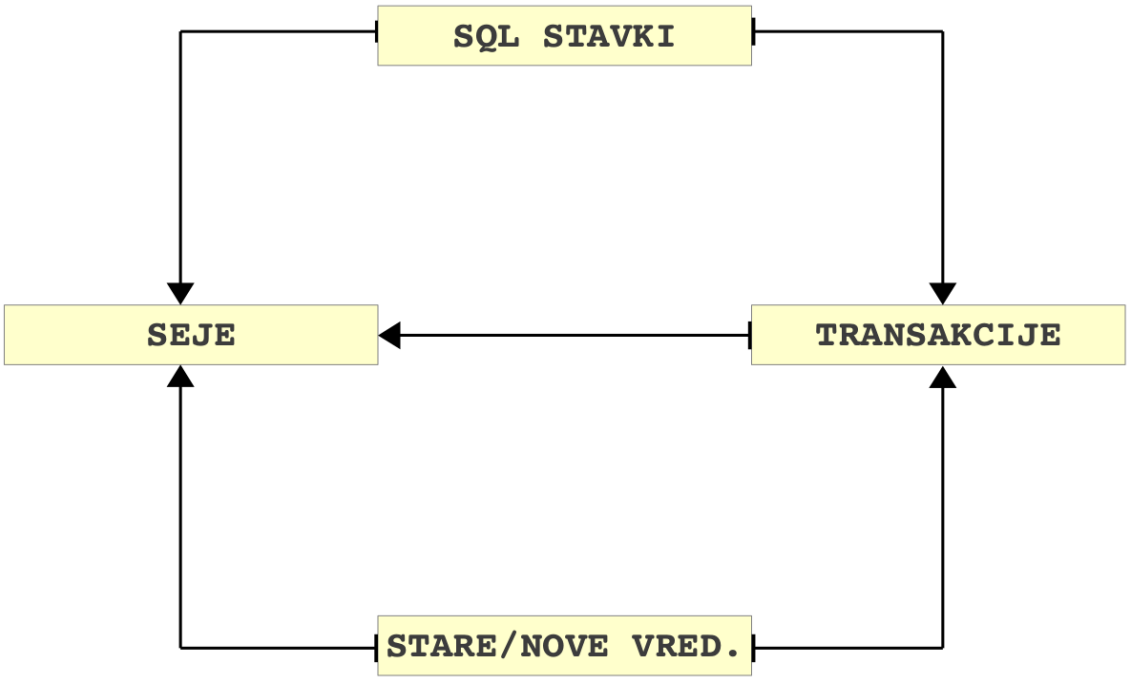
Tabele

Tabele z rezultati so narejene tako, da se **ob kliku na vrstico prikaže nekaj dodatnih gumbov** – le ti so odvisni od vrste poročila. Na ta način lahko pridemo do podrobnejših podatkov o zapisu (ker včasih ekran enostavno ni dovolj širok, da bi lahko prikazali vse podatke, ki so na voljo, prikažemo le najpomembnejše, ostali pa so na voljo s klikom na gumb Details) ali pa povezave na druga poročila v povezavi s trenutnim.

Zeleno obarvane kolone pomenijo, da so to kolone na katere ste postavili filter. V zgornjem primeru je bil nastavljen filter, da naj prikaže vse zapise, ki so nastali za `OS Terminal = 'ERNA'`.

Nekateri rezultati pa se bodo obarvali zeleno, če se z miško zapeljemo čeznje – v zgornjem primeru je kazalec miške na vrstici kolone OS Process ID. To pomeni, da se bo ob kliku na to vrednost filter posodobil. Tako kot kaže slika, bi ob kliku na 28101 filter postal `OS Terminal = 'ERNA' AND OS Process ID ='28101'.`

Podatkovni model



Na sliki je prikazan podatkovni model na osnovi katerega je zgrajena tudi navigacija in način dostop do poročil. Izhodišča za iskanje so torej lahko:

- **SQL Stavki** (dejanski ukazi, ki so jih uporabnik izvajali skupaj s podatki kdo jih je izvajal).
- **Stare in nove vrednosti podatkov** (ce je nekdo naprimer nekdo popravil kolono `placa`, je tukaj zapisano kdo je popravil vrednost, kakšna je bila višina plače pred spremembo in kakšna je bila po spremembi)
- **Seje** (vsaka prijava na podatkovno bazo ustvari sejo, Arbiter beleži kdaj se je nekdo prijavil in kdaj odjavil).

Transakcije niso možne kot izhodišče, saj so vezane na določeno sejo (transakcija brez seje ne more obstajati), zato je treba najprej najti sejo, nato pa lahko dobimo spisek transakcij, ki so se zgodile v tej seji.

Na sliki so na enem koncu povezave narisane črte na drugem pa trikotniki. Notacija predstavlja relacije. Primer za A → B: "*B ima [lahko] določen A. V sklopu A lahko obstaja nič ali več B*".

SQL Stavki

Vir podatkov je ponavadi AUDIT TRAIL (za Oracle, sicer odvisno od tipa podatkovne baze). Glavne stvari, ki se beležijo za vsak SQL stavek so:

- **Action:** Vrsta akcije, npr INSERT ali SELECT.
- **Object/Table:** Ime objekta nad katerim je bila akcija izvedena.
- **Timestamp:** Čas izvedebe akcije.
- **Username:** Uporabniško ime uporabnika, ki je izvede akcijo.
- **Hostname:** Naslov računalnika iz katerega je bil ukaz (sql stavek) prejet
- **SQL Text:** Točen SQL stavek.
- **Bind Variables:** Parametri s katerimi je bil stavek zagnan.

Stare in nove vrednosti

Vir podatkov so ponavadi `ARCHIVED REDO LOG` datoteke (za Oracle, sicer odvisno od tipa podatkovne baze). Glavne stvari, ki jih beležimo:

- **Operation:** `INSERT`, `UPDATE` ali `DELETE`
- **Timestamp:** Čas spremembe
- **User:** Uporabnik, ki je spremembo naredil.
- **Stara in nova vrednost za vsako vrstico in vsako* kolono.** (ne vedno vsako kolono – to je odvisno od konfiguracije, lahko se nastavi belezenje samo spremenjenih kolon namesto vseh)

Seje

Seje se krirajo na podlagi iz enakega vira kot SQL Stavki, točneje, na podlagi zabeleženih `LOGIN` in `LOGOFF` dogodkov. Arbiter vodi sledeče podatke za vsako sejo:

- **#Session:** Unikatna številčna oznaka seje.
- **Username:** Uporabniško ime uporabnika, ki je sejo ustvaril (s tem da se je prijavil na sistem)
- **Logon & Logoff time:** Čas odjave in prijave. Iz tega sledi tudi trajanje seje.

- **Machine Name:** Ime računalnika iz katerega se je uporabnik prijavil.
- **OS Username:** Uporabniško ime s katerim je uporabnik lokalno prijavljen (npr. Windows Username)
- **OS Program name:** Ime programa, s katerim se je uporabnik povezal na sistem.

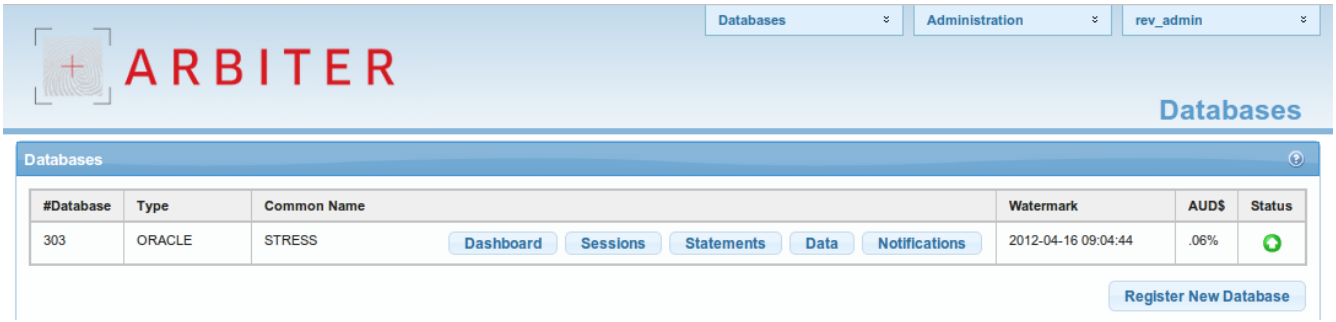
Transakcije

Transakcije se kreirajo na podlagi starih/novih podatkov in se zaključijo glede na `COMMIT/ROLLBACK` dogodke. Za vsako transakcijo se vodi:

- **#Transaction:** Unikatna številčna oznaka transakcije
- **Start & End time:** Čas začetka in konca transakcije.
- **Commited:** Ali je bila transakcija potrjena ali ne. Tudi `rollback to savepoint` postavi to vrednost na `TRUE`.

Sprehod skozi uporabniški vmesnik

Prva stran



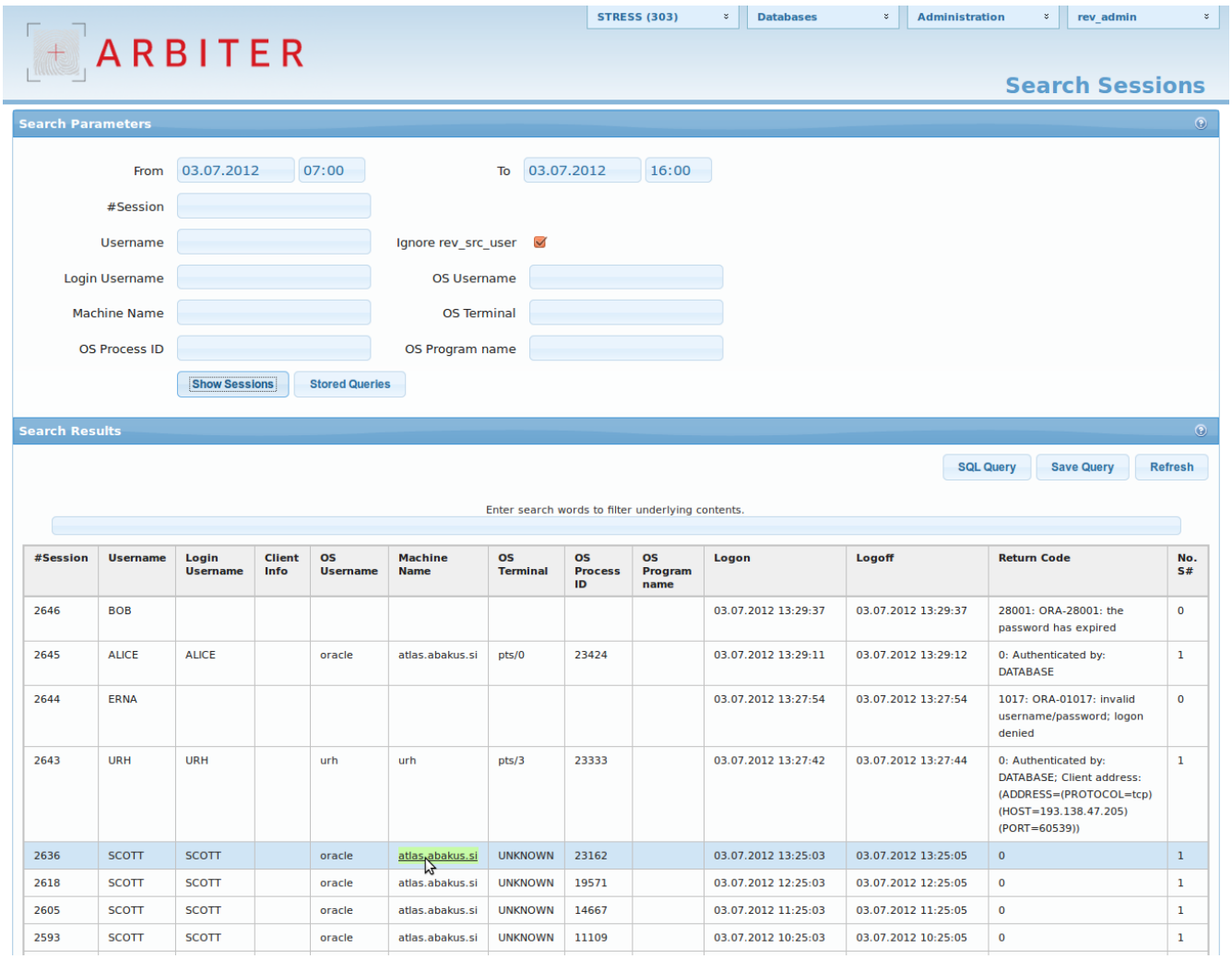
Na prvi strani so prikazane vse registrirane baze za katere ima uporabnik dovoljenje za brskanje po njenih revizijskih sledeh. Na sliki vidimo, da je registrirana samo ena podatkovna baza (ime baze je `STRESS`, njena unikatna šifra pa je `303`).

Watermark je čas, do katerega so podatki že sprocesirani in na voljo za prikaz preko grafičnega vmesnika. Podatki pred tem datumom so lahko prikazani nepopolno ali pa jih sploh še ni.

AUD$ prikazuje zasedenost prostora namenjenga začasnemu shranjevanju revizijske sledi na izvorni bazi ("tablespace usage"). Številka je pomembna, ker v primeru 100% zasedenosti izvorna baza preneha delovati ker nima več prostora za beleženje novih akcij.

Status je zelene barve ko Arbiter vsa svoja opravila opravlja nemoteno. Lahko je rumene ali rdeče barve v primeru, da je pri kateremu od opravil (ki se izvajajo v ozadju) prišlo do napake in se ne izvaja pravilno. V takem primeru je potrebno obvestiti adminstratorja da preveri v čem je problem.

Seje



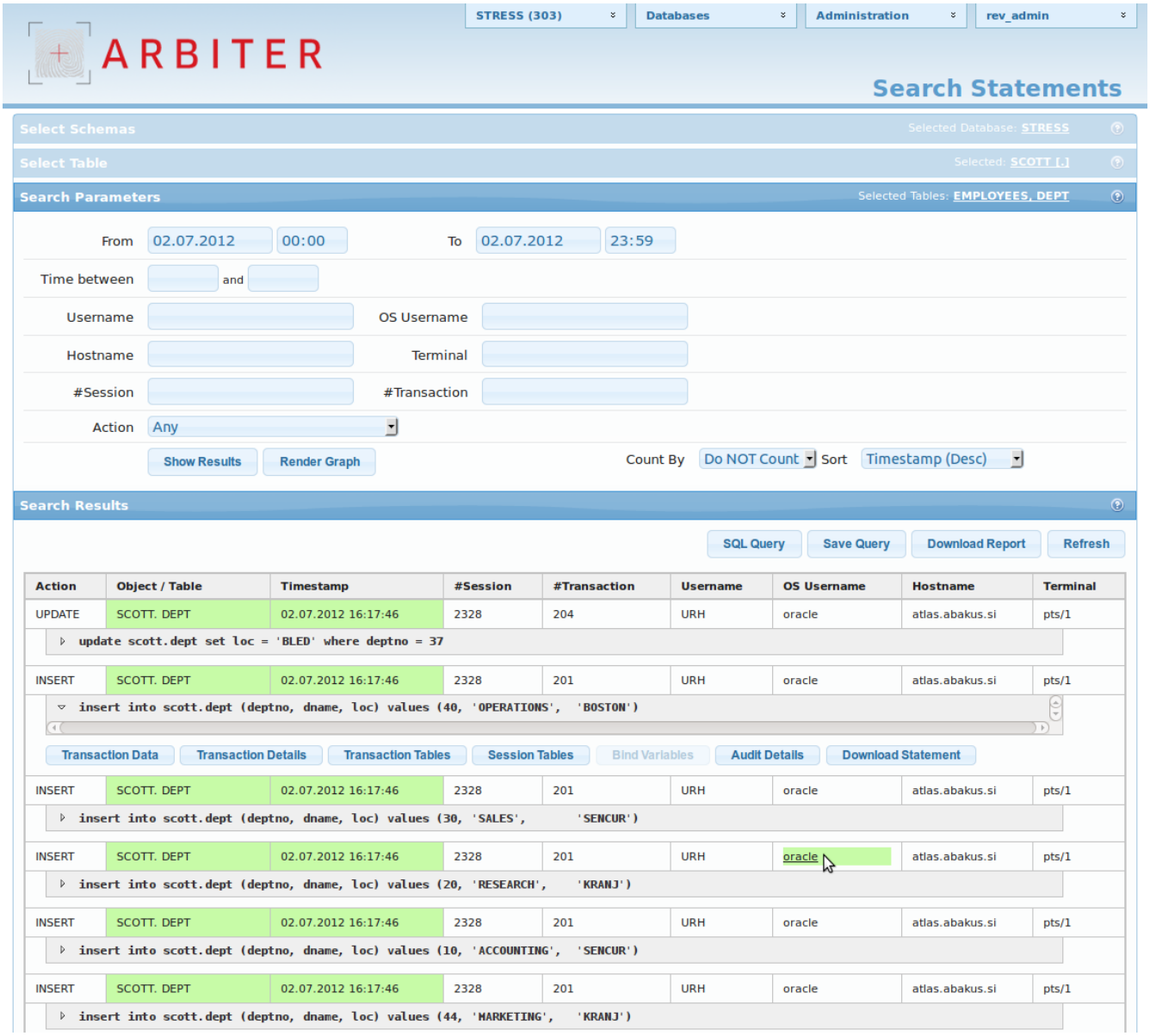
Na sliki so prikazane seje (spisek prijav na bazo) za obdobje `03.07.2012` med `7:00` in `16:00` uro.

Ignore rev_src_user opcija pomeni, da v izpisu ignorira (ne izpiše) sej, ki jih je kreiral uporabnik `REV_SRC_USER`. To je sistemski uporabnik na izvorni podatkovni bazi s katerim se na izvorno bazo prijavlja Arbiter, da lahko od tam prenaša podatke. Takih prijav je lahko (odvisno od konfiguracije) zelo veliko (reda nekaj 10 prijav na uro).

#Session je unikatna šifra seje. Če seja traja dlje kot pa dovoljuje Arbitr-ov pomnilnik (privzeto 14 dni, odvisno od konfiguracije), lahko Arbiter tako sejo zabeleži pod večimi šiframi kot več različnih sej.

Pomembno polje je še **Username**, ki prikazuje uporabniško ime s katerim se je uporabnik (poskušal) prijaviti na bazo. Ali je prijava uspela ali ne pa nakazuje kolona **Return Code**. **Machine name** prikazuje izvor (ime računalnika) od kod seja izvira.

SQL Stavki

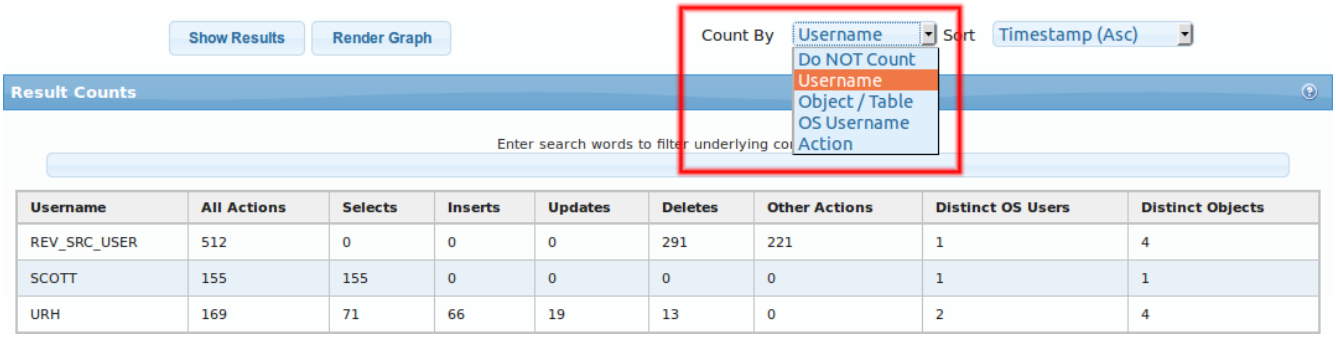


Od vrha navzdol: Najprej je prikazana izbrana baza `STRESS`, izbrana shema `SCOTT` ter dve izbrani tabeli `EMPLOYEES` in `DEPT`. Prikazani podatki so za dan `02.07.2012` (od `00:00` do `23:59`).

Razvidni so točni SQL ukazi ter podatki o uporabniku (`username`, `OS username`, `hostname`, ...), ki jih je izvedel. Gumb `Audit Details` prikaže še več podrobnosti o uporabniku.

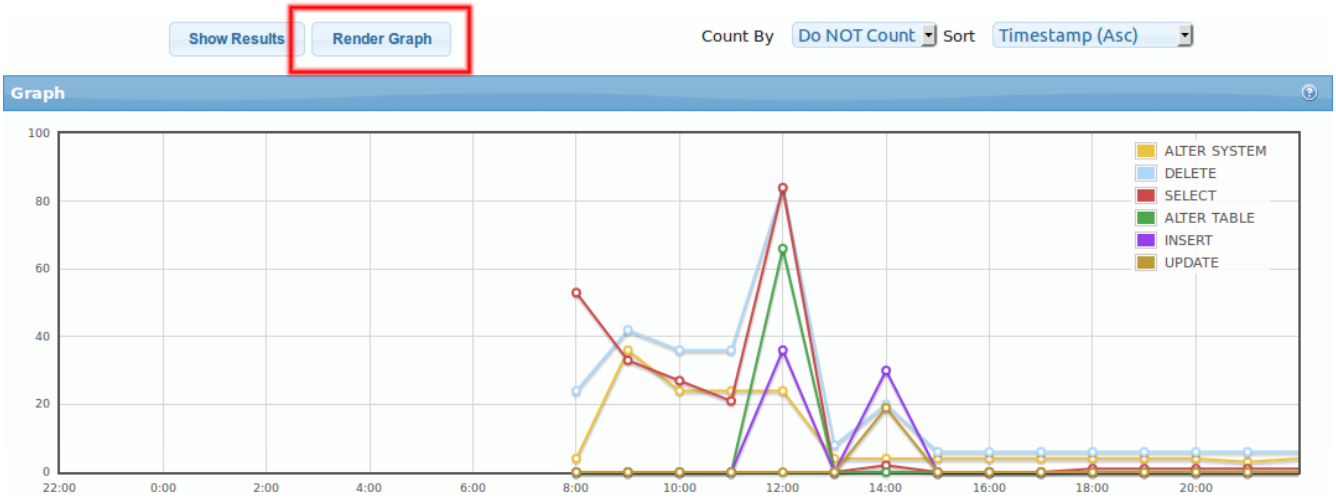
Poleg točnih ukazov lahko pogledamo povzetek v obliki seštevkov ali v obliki grafa, kot prikazujeta spodnji dve sliki.

Na sledeči sliki so razvidne možnosti štetja, lahko nas zanimajo akcije po uporabnikih, po objektih ali po sistemskih uporabnikih. Tako je glede na izbor prikazano število posameznih ukazov (kolone `Selects`, `Inserts`, `Updates`, `Deletes` prikazujejo število ukazov za branje, dodajanje, spreminjanje in brisanje podatkov).

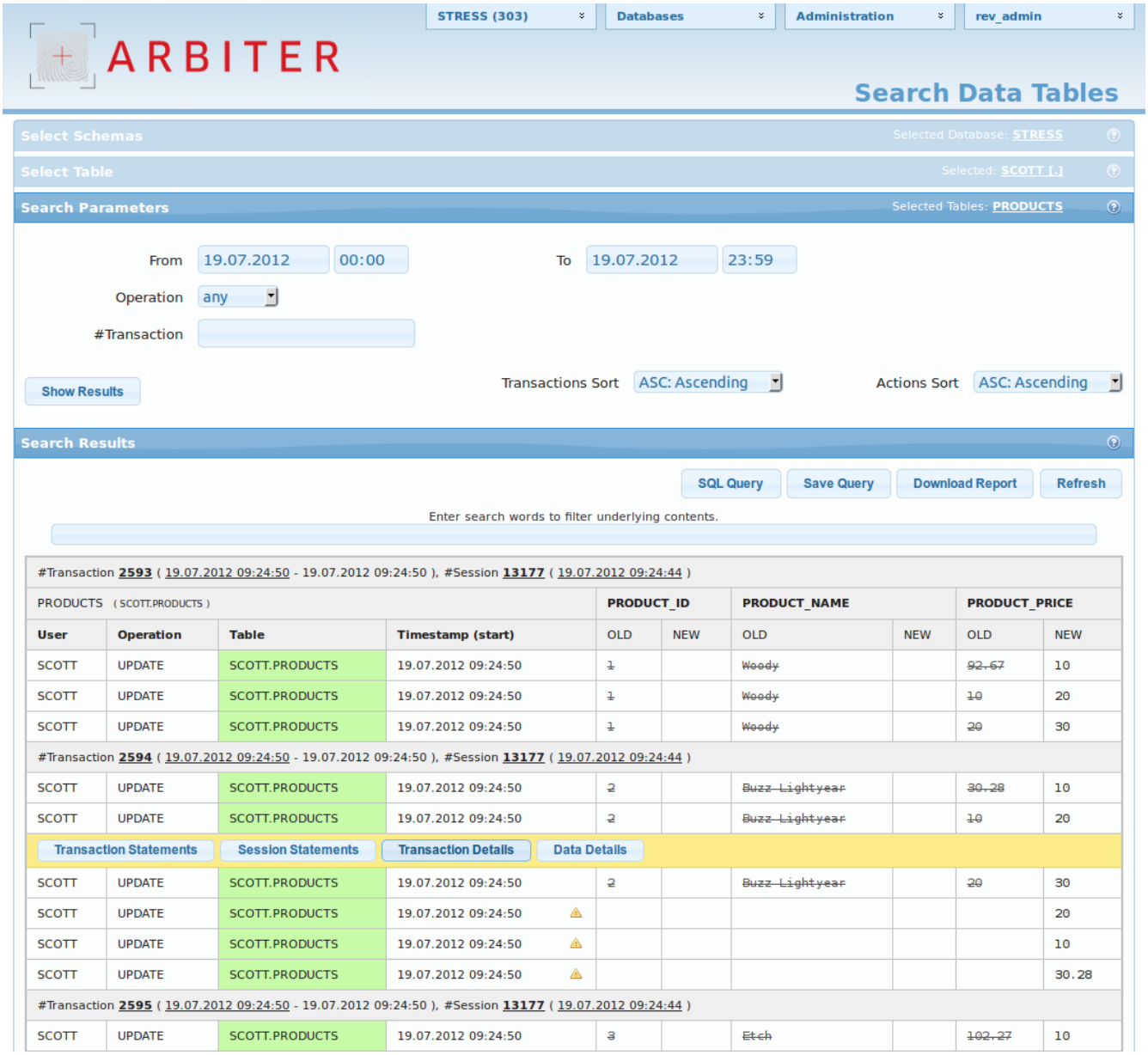


Username	All Actions	Selects	Inserts	Updates	Deletes	Other Actions	Distinct OS Users	Distinct Objects
REV_SRC_USER	512	0	0	0	291	221	1	4
SCOTT	155	155	0	0	0	0	1	1
URH	169	71	66	19	13	0	2	4

Lahko pa podatke vidimo na še višjem nivoju in prikažemo graf glede na prej izbran kriterij. Graf sam po sebi sicer ni razdeljen na posamezne uporabike/objekte, lahko pa ga na tak kriterij omejimo preden kliknemo gumb `Render Graph`.



Stare in nove vrednosti



Prikazan je potek spreminjanja podatkov za tabelo `PRODUCTS` v shemi `STRESS`.

V tabeli so s sivo barvo ozadja označene transakcije, znotraj katerih so vidne akcije. Za vsako akcijo vemo kdaj so se zgodile (kolona `Timestamp`), ter kdo je za to spremembo odgovoren (kolona `User`). Poleg tega so seveda vidne stare in nove vrednosti za vsako kolono izvorne tabele (`OLD`/`NEW`).

Nekatere vrstice imajo rumen trikotnik. To so vrstice ki so bile generirane kot posledica `ROLLBACK` operacije.

Kontakt in tehnična podpora

Na voljo smo za vsakršna vprašanja:

- po e-pošti: arbiter@abakus.si
- po telefonu: 04 287 11 00

Spletne povezave:

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

http://wiki.arbiter.si/

http://www.abakus.si/