Abakus

ARBITER

Sistem vodenja revizijskih sledi

Arbiter za administratorje

Kazalo

Sistemske zahteve in Instalacija

Podatkovna Baza

Podprti so vsi operacijski sistemi na katerih je certificirana podatkovna baza **Oracle Database 11.2.0.3**, Abakus Plus priporoča **Oracle Enterprise Linux**, saj na sistemih, ki niso Linux sicer združljivost z ostalimi Abakus rešitvami (nagios checks, APPM) ni vedno mogoča.

Sistemske zahteve so odvisne od količine zajetih podatkov. Več podatkov kot se hrani, več prostora na disku se potrebuje. CPU in RAM sta tudi odvisna od količine podatkov, saj je večino podatkov pred zapisov v končne tabele potrebno obdelat – to velja predvsem za Oracle LogMiner, ki v času svojega delovanja načeloma porabi en celoten CPU.

Arbiter vsebuje tudi nekaj skript, ki približno ocenijo količino generiranih podatkov za posamezno bazo. Na podlagi teh rezultatov in glede na zahteve uporabnika (spisek tabel za katere je potrebno revizijsko sled voditi) pa lahko ocenimo kakšne so hardware zahteve.

Če je zahtevana visoka zanesljivost se lahko postavi gručo podatkovnih strežnikov – Arbitrove procedure so pisane tako, da se zavedajo take postavitve tako da se obdelave optimalno razdelijo po posameznih node-ih kar ugodno vpliva na performance.

Grafični Vmesnik

Podprte so vse platforme na katerih teče **PHP 5.3** ali novejši, predlagamo Oracle Enteprise Linux. Lahko teče na ločenem ali na istem strežniku kot podatkovna baza, zahteve pa so minimalne (512 MB ram, 20GB HDD, 1 CPU), lahko je tudi virtualni strežnik.

Instalacija

Privzeto Arbiter pride predinstaliran na strežniku kot black-box, **stranka z instalacijo nima dela**. Za *1st level support* partnerje in stranke, ki same želijo izvest instalacijo pa so točni koraki opisani v Arbiter Reference Manual-u http://www.arbiter.si/en/documentation/install.

Za namen razumevanja delovanja sledi opis instalacije.

1. Kreira se nova, "prazna" Oracle podatkovna baza 11.2.0.3 (patchset je pomemben zaradi znanih bug-ov). Arbiter si baze ne more deliti z drugimi produkti, za tako postavitev ni podpore.

2. Požene se `install.sql`, ki kreira vse potreben objekte. Od tu naprej je Arbiter pripravljen za uporabo. (za nadgradnje obstaja `upgrade.sql`)

3. Grafični vmesnik se instalira z `yum install aba-arbiter-gui`. To deluje če ima GUI strežnik dostop do interneta in če na njem teče Oracle Enteprise Linux 6. Na drugih sistemih je nekaj dela s prevajanjem in instalacijo dependency-jev. Sledi editiranje `Settings.php` datoteke. Od tu naprej je Arbiter GUI pripravljen za uporabo.

Registracija podatkovne baze

Po instalaciji Arbiter še ne ve za katere baze naj vodi sled. Zato je potrebno vsako izvorno bazo posebej registrirat. Postopek se za vsako bazo nekoliko razlikuje, točna navodila so objavljena v **Arbiter Reference Manual: Installation Guide** (prosto dostopen na Arbitrovi spletni strani)

Za postopke registracije nudimo podporo, ni potrebno, da se stranka sama ukvarja z registracijo. V splošnem je postopek sledeč:

1. V grafičnem vmesniku se klikne gumb `Register New Database`. Nato se izpolnijo osnovni podatki o bazi (ime baze, naslov strežnika, pristopni podatki, ...)

2. Ko so podatki vnešeni se prenese (download) .sql skripta, ki jo je potrebno na izvorni bazi pognat z administratorskimi privilegiji. DBA lahko to skripto preveri, da se seznani z novimi prametri podpornimi objekti, ki bodo na izvorni bazi kreirani.

3. Klikne se gumb `Finish` s čimer se na Arbitrovi bazi ustvarijo obdelave, ki bodo tekle v ozadju in skrbele za prenos podatkov iz izvorne baze v Arbitrovo bazo.

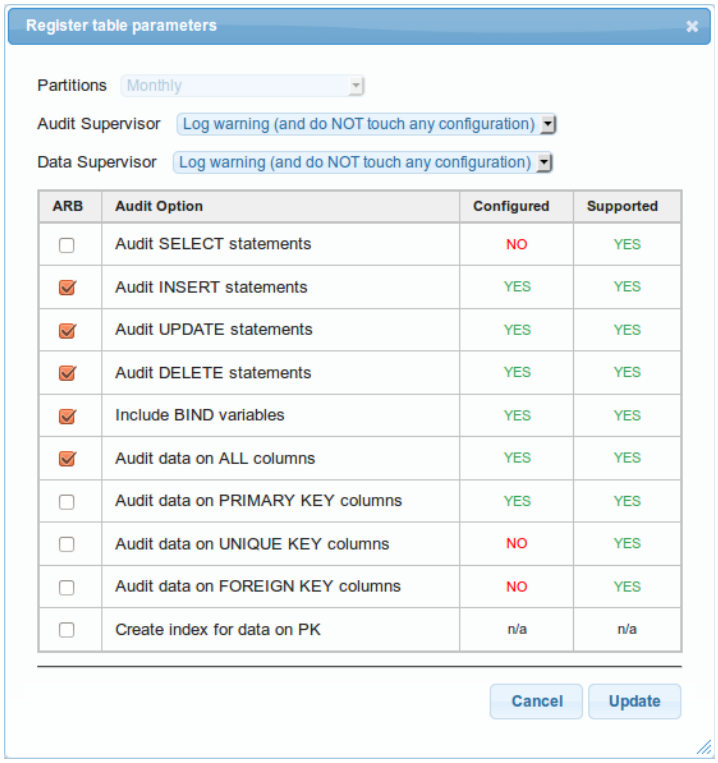
Sheme in Tabele

Potem ko je baza registrirana je potrebno registrirati še vse tabele, za katero naj se vodi reviziska sled. To pomeni, da si Arbiter v svoje metapodatke shrani osnovne podatke o registrirani tabeli ter na izvorni bazi popravi konfiguracijo, tako da izvorna baza začne beležiti revizijsko sled za to tabelo. Konfiguracijo popravi le v primeru, da ima na izvorni bazi za to ustrezne privilegije, v nasprotnem primeru taka konfiguracija postane breme administratorja.

Arbiter periodično (privzeto 1x dnevno) preveri konfiguracijo izvorne baze in v primeru, da je administrator izklopil beleženje za določeno tabelo lahko Arbiter ponovno popravi konfiguracijo in jo omogoči nazaj ali pa le zapiše opozorilo in obvesti administratorja Arbitrove podatkovne zbirke. Alternativno se lahko tako preverjanje tudi izklopi.

Ker je tabel lahko veliko, je mogoče registirati tudi vse tabele v določeni shemi na enkrat. Za nekater vire podatkov (Oracle Database) je mogoče tudi sestaviti poljuben SELECT, ki vrne spisek tabel, ki jih je potrebno registrirati.

Slika prikazuje možnosti registracije tabele v primeru Oracle-ve izvorne baze.



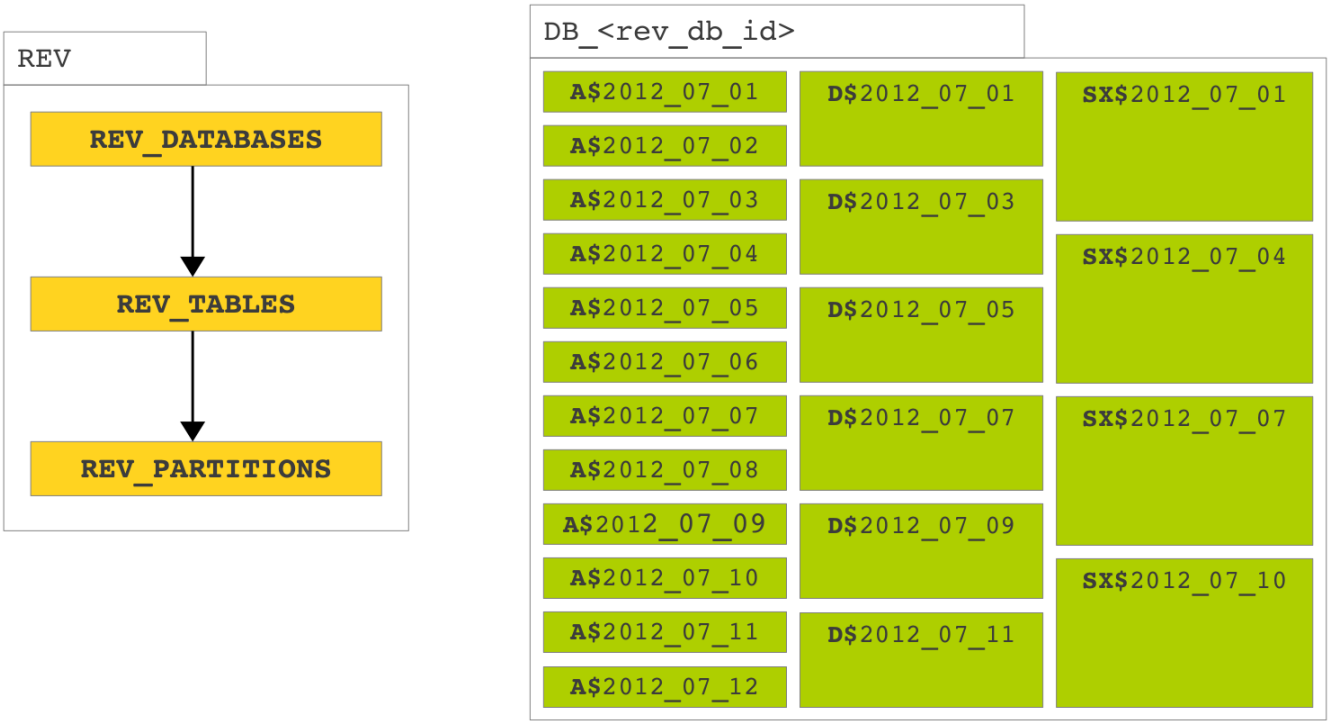
Particioniranje v SE

Ker Standard Edition ne podpira particioniranja je problem rešen tako, da se na vsakih n dni kreira nova tabela. Spisek takih tabel (particij) se nahaja v shemi z metapodatki (`REV`) v tabeli `REV_PARTITIONS`. Na koliko dni se kreira nova tabela je odvisno od konfiguracije, leto pa se določi glede na količino podatkov. Na sliki so idejno prikazane `A$` particije, ki se kreirajo dnevno, `D$` ki se kreirajo na vsake dva dni in `SX$` ki se kreirajo na vsake tri dni.

Imena particij so sestavljena z oznako `XY$yyyy_mm_dd[_hh]`. XY predstavlja ime strukture tabele, datum za znakom $ pa je čas, ko je bila particija kreirana.

Končnemu uporabniku ni potrebno vedeti ničesar o tej interni strukturi, grafični vmesnik poskrbi za abstrakcijo. V `REV_GUI` shemi je tudi paket `REV_UI`, ki nad takimi particijami(tabelami) zna dinamično sestavljat poizvedbe. Za administratorja je pomembno, da se zaveda načina delovanja, saj je hitrost poizvedb odvisna od količine podatkov v tabelah.

V primeru velikih količin podatkov se morajo tudi uporabniki zavedati, da preširok izbor obdobja za iskanje lahko traja nekaj časa preden pride do rezultatov.



Vir podatkov: Oracle Database

Database Audit Trail

Oracle Security Guide: Verifying Security Access with Auditing

Baza zna beležiti dogodke, ki se dogajajo na bazi. Dogodki so posamezni SQL stavki, ki jih uporabnik izvede ter dogodki kot so LOGON, LOGOFF, ...). Beležijo se lahko na različne načine (glede na parameter `AUDIT_TRAIL`):

- `SYS.AUD$` tabelo (database)
- plaintext `.aud` datoteke (filesystem)
- strukturirane `.xml` datoteke (filesystem)
- `syslog` (network/filesystem)

Zajem vezanih spremnljivk je možen le preko `SYS.AUD$` (tudi če se sled zapisuje v .xml datoteke, se vezane spremenljivke vseeno beležijo v tabelo na bazi).

Operacije, ki jih izvede SYSDBA (administrator) se v vsakem primeru beležijo v `.aud` datoteke (v primeru da je audit_sys_operation nastavljen na `TRUE`).

Arbiter zna brat vse omenjene formate, sledi primer konfiguracije:

```
SQL> show parameter audit

NAME                 TYPE     VALUE
-------------------------------------------------
audit_file_dest      string   /oradmin/ARBAUX/adump
audit_syslog_level   string
audit_sys_operations boolean  TRUE
audit_trail          string   DB, EXTENDED
```

Arhivske log datoteke

Database Administrator's Guide: Running a Database in ARCHIVELOG Mode

Database Utilities: Using LogMiner to Analyze Redo Log Files

V arhivskih log-ih so zajete vse spremembe podatkov. Arbiter arhivse log datoteke preko db

link-a prenese k sebi na datotečni sistem ali v ASM, kjer jih nato avtomatsko obdelava prebere in iz njih izlušči spremembe na posameznih registriranih tabelah.

Avtomatska obdelava privzeto uporablja Oracle-ov LogMiner za parsiranje arhivskih logov, alternativno pa smo razvili tudi Abakusovo alternativo miner-ja (kar sicer ni uradno podprta rešitev s strani Oracle-a, je pa veliko hitrejša in za razliko od Oracle-ove rešitve omogoča paralelno procesiranje logov iz različnih baz).

Da pa ima obdelava vse potrebne meta podatke, pa je na bazi potrebno vklopit MINIMAL SUPPLEMENTAL LOGGING. Na ta način se v log-e zapiše malenkost več podatkov, vendar glede na naše teste performančni pribitek ni bil niti opazen.

Vir podatkov: rsyslog

http://www.rsyslog.com/

rsyslog je syslog daemon preko katerega se beležijo sistemski log-i v Linux okolju. Lahko se ga nastavi tako, da poleg tekstovnih log-ov piše tudi direktno v Oracle podatkovno bazo od koder Arbiter podatke prebere in zapiše v svoje tabele. Konfiguracija je opisana v Arbitr-ovem reference manual-u: http://www.arbiter.si/en/documentation/install/src/rsyslog

Iz syslog-a se beležijo vsi prejeti dogodki, filter se lahko nastavi v rsyslog konfiguraciji. Strukturirana sprocila ki jih v syslog beleži Oracle Database zna Arbiter prebrat in zapisat v audit trail tabele.

Vir podatkov: Microsoft SQL Server

Arbiter ima definiran API s katerim se lahko razvije t.i. Collector-je za katerokoli bazo. Trenutno je Collector implementiran le za Micorsoft SQL Server.

MSSQL Collector Install Guide: http://www.arbiter.si/en/documentation/install/src/mssql

Collector API: http://www.arbiter.si/en/documentation/collector