



ciber
committed to the limit.

ORACLE Platinum
Partner

SecureFiles

dbfs - DataBase FileSystem

14 september 2010

Ronald Rood

Even voorstellen

- Ronald Rood
- IT sinds 1987
- Oracle sinds 1991
- Ciber Technical Solutions
- Auteur van Mastering Oracle Scheduler
- Reviewer van Oracle 11g RAC Handbook
- Actief in otn forums en blog

ORACLE®
10g Certified Master



ORACLE®
Certified Master
Oracle Database 10g
Administrator

ORACLE®
Certified Professional
Oracle Database 10g
Administrator

ORACLE®
Certified Professional



plan

- large objects in de database
- SecureFiles
- dbfs
 - administratie
 - gebruik
 - toekomst?

Large objects in de database

- unstructured data
- overal op filesystems
 - lastig consistent te houden
 - lastig te onderhouden
 - lastig te backuppen en restoren
 - lastig te doorzoeken
- begin in v7 met LONG
- vanaf v8 LOB (CLOB en BLOB)
- nog steeds lastig om files de database in te krijgen
- iFS was zeer complex en onderdeel van collaboration suite geworden

Large objects in de database

- in v9 xmlDB
- nog steeds veel coderen, lastig sturen
- nu 11gR2 => dbfs
 - lijkt nog het meest op nfs
 - op linux een filesystem in userspace, transparant voor applicaties
 - andere platforms
 - vanaf het OS alleen met `dbfs_client` commando's te 'gebruiken'.

• SecureFiles

- heeft niets met security te maken
- meer met ALLE data kan veilig in de database terecht
- een ander code path onder LOBS
- kan advanced compressie gebruiken
 - de-duplication
 - compressie
- instance parameter: db_securefile
- eenmaal gebruikt: geen downgrade meer mogelijk

dbfs

- **administratie**

- tikje rommelig
- problemen ‘opgelost’ in de documentatie
- voorbeeld scripts in `?/rdbms/admin/`
 - `dbfs_create_filesystem.sql` (roept `dbfs_create_filesystem_advanced` aan met `@`)
 - `dbfs_create_filesystem_advanced.sql`
 - `dbfs_drop_filesystem.sql`
- `dbfs_role` vs `dba` rol
- `dbfs_client` staat in de client server installatie

- **documentatie**

- insecure by default - als je de aanbevelingen en of defaults volgt
- veel info in de **Large Objects Developers Guide**

dbfs - terminologie

- **data stores**
- **mounts**
- **filesystems**
- **volumes**
- **main**
- **exports**

dbfs - implementatie

- Volledig in pl/sql, niet in de rdbms kernel!
 - OOK niet in *_OBJECTS !
- packages
 - dbms_dbfs_sfs
 - FileSystem akties (maak,export,register)
 - dbms_dbfs_content
 - mount, stores
 - dbms_fuse
 - files, directories
 - dbms_lob
 - inhoud van files

dbfs - implementatie

• tabellen

– gelukkig wel in SYS

- dbfs\$_mounts
- DBFS\$_STORES
- DBFS_SFSS\$_FS
- DBFS_SFSS\$_VOL
- DBFS_SFSS\$_TAB

dbfs - gebruik

- in het gebruik verassend goed - niet foutloos
- geen cluster FileSystem dus geen concurrency control
- vergelijkbaar met nfs
- draait op fuse
 - `/etc/fuse.conf`
- schaalbaarheid komt uit - hash - partitioning
 - key kan zijn fileid, file_name of path - let op consequentie!
- posix property kolommen; updates hebben echt effect
- chown zonder root via tabel updates

dbfs - gebruik

- **cross schema gebruik lastig maar mogelijk**
 - aanmaken filesystem
 - exporteren het filesystem (effectief vnl grant)
 - onder het andere schema aanmaken mount
- **export filesystem grant default aan dbfs_role!**
- **drop user met een dbfs?**
 - dbfs wordt niet opgeruimt
 - probleem in documentatie opgelost: eerst dbfs droppen (zucht)
 - oplossen door direct in sys tabellen te muteren (!?)
 - delete from dbfs\$_mounts;
 - delete from DBFS\$_STORES;
 - delete from DBFS_SFSS\$_FS;
 - delete from DBFS_SFSS\$_VOL;
 - delete from DBFS_SFSS\$_TAB;

conclusie

- **Large Objects**
 - **SecureFiles**
 - **dbfs**
-
- **op Linux clients bruikbaar**
 - **pas op met drop user**
 - **performance is trager dan die van nfs**
 - **data is nu veilig, consistent te houden**
 - **dbfs heeft waarde maar is nog niet af**
 - **goed genoeg voor 'create database'**



vragen



ciber
committed to the limit.

ORACLE Platinum
Partner

- <http://ronr.blogspot.com>
- <mailto:ronald.rood@ciber.nl>