

MUNIQSOFT

DOAG 2015

Reproduction of fee-based options

Marco Patzwahl

MuniQSoft GmbH

◆ Tätigkeitsbereiche:

- ▶ Oracle Support
Hotline: Mo-Fr 8.00 – 18.00 Uhr
Erweiterung um Rufbereitschaft auch am Wochenende möglich
- ▶ Oracle IT-Consulting & Services, Schwerpunkt auch RAC
- ▶ Oracle Schulungen (SQL, PL/SQL, DBA, APEX, B&R, MySQL...51 verschiedene Schulungen, gerne auch Inhouse)
- ▶ Software-Lösungen
- ▶ Oracle Lizenzen

MuniQSoft GmbH
Schulungszentrum
Grünwalder Weg 13a
82008 Unterhaching
Tel.: 089 / 679090 40

MUNIQSOFT

MuniQSoft GmbH
**IT-Consulting, Support,
& Software-Lösungen**
Witneyst. 1
82008 Unterhaching
Tel.: 089 / 6228 6789 0

Would you like to save up to 100K€??

- ◆ Why you should stay here and listen....
- ◆ If you own a Enterprise Edition with 100 Named User Plus:
- ◆ Partitioning: $200\text{€} * 100 = 20,000\text{€}$
- ◆ Database Vault: $200\text{€} * 100 = 20,000\text{€}$
- ◆ Advanced Security: $260\text{€} * 100 = 26,000\text{€}$
- ◆ Data Masking and Sub. Pack $200\text{€} * 100 = 20,000\text{€}$

86,000€

What's not possible with STD?

Function	Prio	Alternatives in STD
Active Data Guard (Opt)	3	Log Miner
Advanced Compression (Opt)	2	utl_compress Package
Advanced Replication	2	individual replication with AQ
Application Role	3	None
Advanced Security	2	Individual encoding (dbms_crypt)
Automatic Storage Management	2	SE included in RAC 2 Node
Backup Encryption (Opt)	3	Individual encoding after RMAN Backup
Basic Compression	2	None
Bit-mapped indexes	1	Use Non Unique Index with Option compressed
Block Change Tracking	3	None
Block Media Recovery	2	Keine
Change Data Capture	3	None
Data Mining (Opt)	3	???

Note: In Version 12.1 it is not allowed to use some Pluggable Database container as Standard and others as Enterprise Edition

What's not possible with STD? (ff)

Function	Prio	Alternatives in STD
DBMS_PRIVILEGE_CAPTURE	1	Use Unified Auditing to check which Privs are used
Database Resource Manager	2	non, but DEFAULT_MAINTENANCE_PLAN in STD available
Deferred Segment Creation	2	Non
Duplexed backups	3	two backups at different Mount Points
Enterprise User Security	3	Write you own Package
Export transportable tablespaces	2	Non
Fast-Start Fault Recovery	2	???
File Mapping	3	???
Fine-grained access control	2	Context with Logon Trigger
Fine-grained Auditing	3	Normal audit for complete table

What's not possible with STD? (ff)

Function	Prio	Alternatives in STD
Flashback Data Archive (Opt)	3	trigger und support tables (included since 12)
Flashback Database	2	time-based recovery
Flashback Table	2	Flashback Query
Join index	3	Non
Managed Standby	2	individual standby solution
Materialized view rewrite	1	Non
OLAP (Opt)	3	Non
Online Index Build	1	create second index, then drop the first
Online Redefinition	1	Materialized View on prebuild Table
Oracle Data Guard (Redo Apply)	2	Non / third-party software
Oracle Data Guard (SQL Apply)	2	Non / third-party software, Log Miner
Oracle Database Vault (Opt)	3	DDL Trigger

What's not possible with STD? (ff)

Function	Prio	Alternatives in STD
Oracle Label Security (Opt)	3	since 12c: Invisible Columns
Parallel backup and recovery	3	start several backups in parallel
Partitioning (Opt)	2	view-concept with UNION ALL on tables
Pluggable Database (ab 12c)	2	consolidate several DBs into one
Point-in-time tablespace recovery	3	Recovery on second computer
Real Application Clusters	possible	2 Node Cluster possible
Real Application Testing (Opt)	3	Log Miner (dbms_logmnr)
Result Cache	2	Non
Rolling Upgrade	2	Non
SecureFiles Encryption (Opt)	2	use dbms_crypto Package

What's not possible with STD? (ff)

Function	Prio	Alternatives in STD
Server Flash Cache (Opt)	3	since 12c: Tablespace on Ram Disk
Spatial (Opt)	3	Third-party Supplier
SQL Plan Management (Opt)	3	Stored Outlines
Streams Capture	3	Log Miner
Transparent Data Encryption (Opt)	3	Operating System Disk Encryption
Trial Recovery	3	Non
Unused Block Compression	3	Non

Common misconceptions (about what isn't possible)

- ◆ **available for free in the Standard Edition:**
 - ▶ **Index (Standard-)Compression (Advanced Compression starting from 12c fee-based)**
 - ▶ **Basis Table Compression**
 - ▶ **Segment and Undo Advisor**
 - ▶ **Audit (including Unified Auditing starting from 12c)**
 - ▶ **Query Flashback**
 - ▶ **NOLOGGING on Tables and Indices**

Activate Partitioning/Row Level Security

- ◆ in the Standard and Express Edition you cannot create partitioned tables
- ◆ but with the following (illegal?) trick you can do it anyway:
 - ▶ ALTER SESSION SET EVENTS
'14524 trace name context forever, level 1';
- ◆ you can even activate Row Level Security:
 - ▶ ALTER SESSION SET EVENTS
'28131 trace name context forever, level 1';

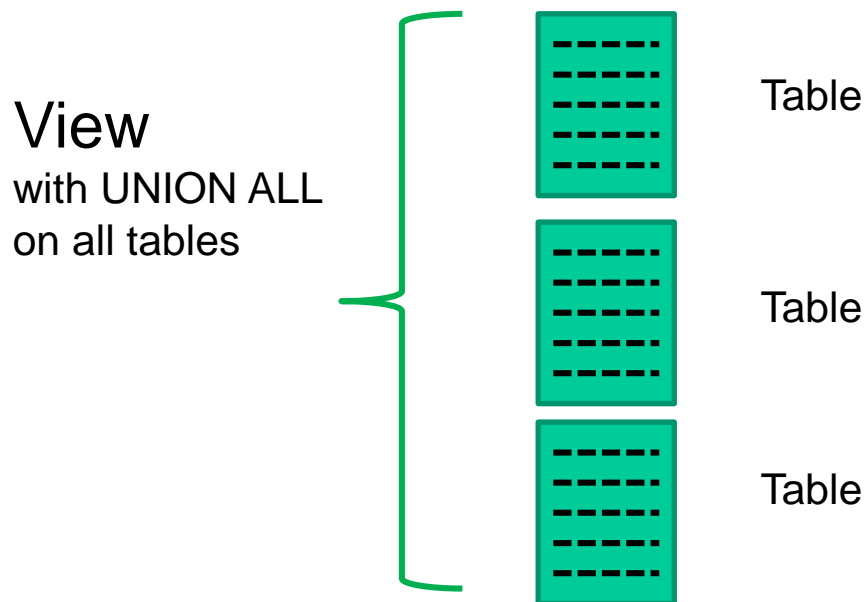
Maybe illegal!

Partitioning on SE

- ◆ in our example we only simulate a Range Partitioning
- ◆ Hash, List, System and Subpartitions are not discussed (these are rarely used in practice though)
- ◆ disadvantages for our Solution compared with original Partitioning in the EE:
 - ▶ PL/SQL Code instead of C++ in the Kernel
 - ▶ **Optimizer is optimized for Partitioning**
 - ▶ individual Package for Partition Maintenance => customizations are possibly required in every new version
 - ▶ there is no global Index

Partitioning on SE: Structure

- ◆ Partitions are built like pearls (i.e. tables) threaded on a string, and thus appear to be an object
- ◆ You can recreate this behavior with a view:



Partitioning on SE: Structure

◆ Please note:

- ▶ the Optimizer has to know the tables in which the data is found
- ▶ only then it **won't perform a FULL TABLE** scan on all tables
- ▶ all data recorded in the view has to be forwarded by Instead of Trigger
- ▶ the view has to know all tables (partitions) that exist in total

Partitioning (example)

◆ Create several tables and link them with a view

- ▶ CREATE TABLE `scott.emp_10`
AS SELECT * FROM `scott.emp`
WHERE `deptno=10`;
- ▶ CREATE TABLE `scott.emp_20`
AS SELECT * FROM `scott.emp`
WHERE `deptno=20`;
- ▶ CREATE VIEW `scott.empv` AS
SELECT * FROM `scott.emp_10`
UNION ALL
SELECT * FROM `scott.emp_20`;

Partitioning (alternative) (f)

- ◆ **Tables need a check-constraint for the optimizer to skip unnecessary tables**
 - ▶ `ALTER TABLE scott.emp_10
ADD CONSTRAINT check_deptno_10
CHECK (deptno=10);`
 - ▶ `ALTER TABLE scott.emp_20
ADD CONSTRAINT check_deptno_20
CHECK (deptno=20);`
- ◆ **you also need another Instead of Trigger that distributes the data into the correct table**

What should your Package do?

- ◆ **Create a View over all tables (with union all)**
- ◆ **Create a Trigger on the View that decides which insert is done in which table**
- ◆ **The Same for Update and Delete**
- ◆ **Take Care of moving rows in another "partition" (table)
Delete in orig. Table and Insert in new Table !**
- ◆ **For a new "partition" create a new Table with a check constraint (on the partition col)**

Implementations (as of June 2015)

- ◆ **The following functions are currently in my Package implemented:**
 - ▶ **CONVERT_TO_PARTITION (table in part. table)**
 - ▶ **ADD_PARTITION (add new Partition)**
 - ▶ **DROP_PARTITION (drop Partition)**
 - ▶ **TRUNCATE_PARTITION (truncate Partition)**
 - ▶ **SPLIT_PARTITION (split Partition in two Partitions)**
 - ▶ **CREATE_INDEX (create Index for every Partition)**
 - ▶ **GATHER_STATS (create statistics for Partitions)**

Code Snippet of the Trigger

```
◆ ...
◆ s:='CREATE OR REPLACE TRIGGER '||owner||'.'||table_name||'_PART_TRG
◆ INSTEAD OF INSERT OR UPDATE OR DELETE
◆ ON '||owner||'.'||table_name||'
◆ FOR EACH ROW
◆ DECLARE
◆ stmt VARCHAR2(32000);
◆ BEGIN' ||chr(13);
◆ for R in (select PARTITION_NAME,max_value,part_id from SE_TABLE_PARTITIONS
◆         where OWNER=create_trigger.OWNER
◆         and TABLE_NAME=CREATE_TRIGGER.TABLE_NAME) LOOP
◆     S:=S||'--Tabelle '||R.PARTITION_NAME||
◆     ' Max Value '||R.MAX_VALUE||CHR(13);
◆     IF R.PART_ID=0 THEN
◆         S:=S||'IF nvl(:new.'||v_part_col||',:old.'||v_part_col||') <= '||r.max_value||' THEN '||lf;
◆         s:=s||' IF UPDATING THEN '||lf;
◆         s:=s||'     '||REPLACE(upd_str,'#TAB#',r.partition_name)||';' ||lf||' END IF;' ||lf;
◆         s:=s||' IF INSERTING THEN '||lf;
◆         s:=s||'     '||REPLACE(ins_str,'#TAB#',r.partition_name)||';' ||lf||' END IF;' ||lf;
◆         s:=s||' IF DELETING THEN '||lf;
◆         s:=s||'     '||DELETE FROM '||owner||'.'||r.partition_name||
◆         lf||' WHERE '||V_PART_COL||' =:old.'||V_PART_COL||
◆         ';' ||lf||' END IF;' ||lf;
◆     ...
◆
```

Standby DB in SE

- ◆ please consider, that two computers have to be licenced both computers count!!
 - ▶ Named User License => "Standby DB" is included
- ◆ the parameter `log_archive_dest_1-31` can only be set on **LOCATION**, not on one **SERVICE**
- ◆ the transmission of Archivelogs has to be done manually
 - ▶ Windows: Robocopy, Powershell, ...
 - ▶ Unix: scp, sftp, rsync...
 - ▶ `rsync -e ssh -Pazv /ora/oracle/arch/
oracle@remote:/export/home/oracle/arch/`

Standby DB (f)

◆ Control-File for STBY DB:

- ▶ `ALTER DATABASE CREATE STANDBY CONTROLFILE AS 'c:\temp\standby.ctl';`

◆ the target database runs in Mount Phase

- ▶ `STARTUP MOUNT`

◆ start the script every 5 minutes with these lines:

- ▶ `WHenever SQLERROR EXIT`
- ▶ `RECOVER AUTOMATIC STANDBY DATABASE`

Example with Robocopy (Win)

- ◆ **set up an optional network drive**

- ▶ `net use o: \\oracle_backup_server\archivelog`

- ◆ **robocopy c:\oracle\archive**

- `\\oracle_backup_server\archivelog /MIR /FFT /Z /W:5`

- ◆ **Note:**

- ▶ **the Fast Recovery Area can be placed completely on the Remote Computer**
 - ▶ **in Windows: the service has to run as administrator though**

Alternative Commercial Products

◆ DBVISIT Standby

- ▶ Standby database also for Standard Edition
- ▶ Easy installation and configuration
- ▶ Graphical administration possible
- ▶ www.dbvisit.com

XOR Verschlüsselung

◆ A better way (compared to pure RAW conversion) is XOR encryption

◆ For decryption the key must be known

▶ `SELECT
utl_raw.bit_xor(utl_raw.cast_to_raw('ABCDEF'),
utl_raw.cast_to_raw('MARCO_KEY'))
FROM dual;`

▶ => **0C0311070A194B4559**

▶ `SELECT utl_raw.cast_to_varchar2(
utl_raw.bit_xor('0C0311070A194B4559',
utl_raw.cast_to_raw('MARCO_KEY'))
FROM dual;`

DBMS_CRYPTO Package

- ◆ **Replaces / completes the previous package `dbms_obfuscation`**
- ◆ **Strings do not have to be a multiple of 8 byte and longer but must be converted to RAW initially**
- ◆ **Possesses a higher amount of encryption routines:**
 - ▶ **`dbms_crypto.des_cbc_pkcs5` (Schlüssellänge 8 Byte), `dbms_crypto.des3_cbc_pkcs5` (24 Byte)**
 - ▶ **`dbms_crypto.encrypt_rc4`**
 - ▶ **`dbms_crypto.encrypt_aes128`**

DBMS_CRYPTO encryption

◆ Encryption:

- ▶ `p_encrypted_raw := dbms_crypto.encrypt(
src => p_text_raw,
typ => p_crypto_typ,
key => p_key);`

◆ Decryption:

- ▶ `p_decrypted_raw := dbms_crypto.decrypt(
src => p_encrypted_raw,
typ => p_crypto_typ,
key => p_key);`

Improved Hash Function

- ◆ **Even in EE the pbkdf2 Hash Function is currently not implemented (yet)**
- ◆ **A Hash values is run 1000 times over a Hash Function to to complicate a Brut Force Attack**
- ◆ **This function is easily implementable.**
- ◆ **Or copy & paste from:**
- ◆ **<https://mikepargeter.wordpress.com/2012/11/26/pbkdf2-in-oracle/>**

VPD reproduced

- ◆ **VPD stands for Virtual Private Database and allows the multi-tenancy of individual tables**
- ◆ **so that different users only can see certain data of a table**
- ◆ **The idea:**
 - ▶ **Context parameter which can be read out with `sys_context` can be set**
 - ▶ **View set on table with additional filter:
`<vpd_col>=sys_context(...)`**
 - ▶ **Moreover individual columns can be encrypted and have to be decrypted by a Session Key**

VPD reproduced (f)

◆ Create Context in Session:

```
▶ CREATE OR REPLACE CONTEXT my_context  
  USING my_sec_check  
  /
```

◆ Create Security Package:

```
▶ CREATE OR REPLACE PACKAGE my_sec_check IS  
  PROCEDURE set_context(p_key IN varchar2);  
  FUNCTION crypt (  
    text    IN VARCHAR2,  
    key     IN VARCHAR2 DEFAULT 'MuniQSoft_Key',  
    cryptmode    IN VARCHAR2 DEFAULT 'E'  
  ) RETURN VARCHAR2;  
END;  
/
```

VPD reproduced (ff)

- ◆ **The View should look like:**


- ▶ `CREATE OR REPLACE VIEW emp_sec
AS select * FROM emp
WHERE '298C5C173F48AEC60B06B8671FC67CDB'
=sys_context('my_context', 'key');`

- ◆ **Don't forget to revoke the privs from the Table and grant privs on the view !**

DBMS_PRIVILEGE_CAPTURE (DV Option)

◆ Use Unified Audit Trail for monitoring of privileges:

```
WITH s AS (select 'SCOTT' as uname from dual)
select /* Direkt-Rechte */ 'CREATE AUDIT POLICY ' || s.uname || '_PRIV_CAPT ' FROM s
UNION
SELECT 'PRIVILEGES ' || listagg(privilege, ',' || chr(10)) within group (order by privilege)
from s, dba_sys_privs where grantee=s.uname
UNION ALL -- Objektrechte
select 'ACTIONS ' || listagg(privilege || ' ON ' || owner || '.' || table_name, ',' || chr(10) )
within group (order by 1)
from s, dba_tab_privs where grantee=s.uname
UNION ALL -- Rechte über Rollen (nur erste Hierarchieform)
select 'ROLES ' || listagg(granted_role, ',' || chr(10)) within group (order by granted_role)
from s, dba_role_privs where grantee=s.uname
UNION ALL
SELECT ';' FROM dual;
```



```
CREATE AUDIT POLICY SCOTT_PRIV_CAPT
PRIVILEGES CREATE VIEW,
SELECT ANY TABLE,
UNLIMITED TABLESPACE
ACTIONS EXECUTE ON SYS.DBMS_CRYPTO
ROLES CONNECT, RESOURCE
;
```

Audit evaluation

◆ For detailed information:

```
▶ SELECT event_timestamp,action_name,sql_text,
system_privilege_used
FROM unified_audit_trail
WHERE unified_audit_policies='SCOTT_PRIV_CAPTURE';
```

◆ Used Privs:

```
▶ SELECT distinct
CASE WHEN SYSTEM_PRIVILEGE_USED IS NULL THEN
replace(action_name,'PL/SQL ','')||
' ||object_schema||'. ' ||object_name
ELSE
SYSTEM_PRIVILEGE_USED END as priv
FROM unified_audit_trail
WHERE unified_audit_policies='SCOTT_PRIV_CAPTURE';
```

Audit evaluation (f)

◆ Unused Privs:

```
◆ WITH s AS (select 'SCOTT' as uname from dual)
  (SELECT privilege from s,dba_sys_privs
  where grantee=s.uname
  UNION ALL -- Objektrechte
  select privilege||' '||owner||'.'||table_name from s,dba_tab_privs where
  grantee=s.uname
  UNION ALL -- Rechte über Rollen (nur erste Hierarchieform)
  select granted_role from s,dba_role_privs where grantee=s.uname)
  MINUS
  select distinct
  CASE WHEN SYSTEM_PRIVILEGE_USED IS NULL THEN replace(action_name,'PL/SQL
  ','')||' '||object_schema||'.'||object_name
  ELSE
  SYSTEM_PRIVILEGE_USED END as priv
  from s,unified_audit_trail
  where unified_audit_policies=s.uname||'_PRIV_CAPT';
```


DATA MASKING / Redaction (f)

- ◆ You can easily write your own Data Masking Function
- ◆ Example Code Snippet for VARCHAR2 Values:

```
...
SELECT listagg(ch,'') within GROUP (ORDER BY dbms_random.value) INTO v_tralat
FROM
  (SELECT * FROM
    ( SELECT 1 AS z,chr(rownum) ch,rownum r FROM dual CONNECT BY level<256)
    WHERE r BETWEEN v_startchar AND v_endchar)
GROUP BY z;
SELECT listagg(ch,'') within GROUP (ORDER BY r) INTO v_tralat_o
FROM
  (SELECT * FROM
    ( SELECT chr(rownum) ch,rownum r FROM dual CONNECT BY level<256)
    WHERE r BETWEEN v_startchar AND v_endchar);
RETURN TRANSLATE(i_string,v_tralat_o,v_tralat); ...
```

DATA MASKING / Redaction (f)

```

◆ SELECT ename,my_redact.text(ename,'T') as
   translate,my_redact.text(ename,'H') as
   hash,my_redact.text(ename,'R') as
   random,hiredate,my_redact.text(hiredate+1/24,'R') as
   random_hiredate,my_redact.text(hiredate,'H') as
   hash_hiredate,sal,my_redact.text(sal,'R') as
   random_sal,my_redact.text(sal,'H') as hash_sal
FROM scott.emp;

```

ENAME	TRANSLATE	HASH	RANDOM	HIREDATE	RANDOM_HIREDATE	HASH_HIREDATE	SAL	RANDOM_SAL	HASH_SAL
SMITH	j[hoY	BC70B	RHQWX	17.12.1980 00:00:00	04.06.1958 11:54:23	20.04.1948 00:00:00	800	604	290
ALLEN	t x[6E1CA	G36IG	20.02.1981 00:00:00	22.04.1951 00:45:44	21.11.1954 00:00:00	1600	70	774
WARD	h{Bn	7C7F	ZC3V	22.02.1981 00:00:00	01.09.1940 06:04:59	06.01.2011 00:00:00	1250	660	6766
JONES	rUnCu	A1787	US1X7	02.04.1981 00:00:00	08.05.1944 16:04:02	07.01.1982 00:00:00	2975	1754	5032
MARTIN	rW`CSo	F3B9B8	499EQW	28.09.1981 00:00:00	15.03.1964 18:49:23	21.03.1958 00:00:00	1250	152	6766
BLAKE	\ikS]	7CDE5	0FN1P	01.05.1981 00:00:00	26.06.1949 04:12:03	10.11.1957 00:00:00	2850	981	9708
CLARK	LPmBW	4F90E	S03L6	09.06.1981 00:00:00	30.04.1963 17:30:26	24.10.1992 00:00:00	2450	49	3524
SCOTT	\D__	7634F	6FL31	19.04.1987 00:00:00	18.04.1944 14:22:39	08.08.2006 00:00:00	3000	868	7507
KING	h_ms	50B8	EXSA	17.11.1981 00:00:00	09.06.1963 13:40:38	02.06.1986 00:00:00	5000	173	8237
TURNER	WqjIq	E043E3	IU5YYX	08.09.1981 00:00:00	23.11.1955 23:50:49	29.12.2015 00:00:00	1500	1466	7841
ADAMS	X@Xvp	2D2EB	6K8WQ	23.05.1987 00:00:00	20.05.1953 14:03:16	31.05.1984 00:00:00	1100	949	1245
JAMES	€^Ihu	6AFC4	RC7KZ	03.12.1981 00:00:00	13.09.1944 13:18:54	02.05.1981 00:00:00	950	658	636

Any questions?



Officer: Since when
does honey-mead
number
among alcohol?

Wie? Honig-Met
zählt als Alkohol?

MuniQSoft GmbH

◆ Field of business activity:

- ▶ Oracle support
Hotline: Mo-Fr 8.00 a.m. to 6.00
On-call-duty possible on weekends
- ▶ Oracle IT consulting & services, focus also RAC
- ▶ Oracle trainings (SQL, PL/SQL, DBA, APEX (inclusive APEX 5), Backup & Recovery..., 51 various trainings, als inhouse)
- ▶ Software solutions
- ▶ Oracle licences

MuniQSoft GmbH
training center
Grünwalder Weg 13a
82008 Unterhaching
phone 089 / 679090 40

MUNIQSOFT

MuniQSoft GmbH
**IT consulting, support,
& Software solutions**
Witneystr. 1
82008 Unterhaching
phone 089 / 6228 6789 0