Oracle 12c Database New Features

Geert De Paep
Join the buzz:

- Wifi available
- Twitter
  - #oracleopenxperience
  - @oopenxperience
Disclaimer

• Oracle 12c is planned for 2013
  – No documentation / presentations available yet
• Based on OOW sessions
• Based on OOW demogrounds
• Based on MLC (My Little Camera)
• Nothing is guaranteed to be included in Oracle 12c production
• Nothing is guaranteed to be implemented as explained here
Room for movement

Oracle 12c

Large data volumes

Performance

Datafile

ASM

Db

Resync

standby

Monitoring and management

RMAN

Backup

Data Guard

Failover

RAC Failover transparency

Db restore
RMAN - Recovery manager

- Restoring a datafile
  - May require tapes+heads to be available
  - May be slow

- 12c
  - Recover from physical standby database
    - Transfer files from physical sby to primary

- Advantages:
  - Easier and faster restore
  - Less dependent from backup infrastructure
Recovery manager

- Accidental ‘drop table’, ‘truncate table’, wrong script, human error, ...
- Pre 12c
  - Import from older export file (if available)
  - Restore older backup to extract table (+++work)
  - Flashback physical standby (+work)
  - ...
- 12c: Table recovery
  - Restore + recover individual tables from rman backup
  - To any point in time
  - Fast and easy
Recovery manager

• Further RMAN improvements
  – Automate the use of incremental backup to bring a standby back in sync
    • Now a complex and manual procedure
  – Auto conversion for cross-platform backup/restore
    • Cfr “convert database” statement

• Rman 12c
  – Makes a lot of tasks easier
Application continuity

• Traditional RAC failover
  – TAF: queries are transparant, ins/upd/del not
  – FCF: no tcpip timeouts
    • Any sql gets “connection closed” and needs to handle this (incl. select)
• 12c: “Real Transparant Failover Mechanism”
  – For queries and transactions
  – Currently only for java connection pool or plain jdbc
  – 100% transparant failover
Data Guard - Global data services

- Failover: standby db becomes primary
  - Clients have to be redirected
  - No ‘out of the box’ solution until now
    - Dns change, on role-change trigger, ldap, tnsnames.ora, ...

- 12c: global data services
  - Purpose: no client reconfiguration in case of failover or switchover
  - + other advantages
Global data services

• Features
  – Awareness which site is primary
    • Can connect clients always to primary site
  – Rule based
    • Can direct applications to active data guard
    • If active DG 1 not available go to other active DG or primary
  – Affinity
    • Prefer local databases
• Any replication technology
  – “Global load balancing and failover for replicated databases”
Data Guard – Far sync standby

- Far sync standby
  - Instance with only standby and archive logs
  - Acts as a ‘dispatcher’ for multiple standby’s
    - Reduced WAN traffic
- Easier failover
  - “validate database” before switchover
Grid Infrastructure

• Pre 12c, every node has
  – Cluster software
  – ASM software
  – One or more database instances
  – Optionally ACFS (ASM cluster file system)
  – Optionally applications running

• New concepts
  – Flex Cluster
  – Flex ASM
Grid Infrastructure

• Flex cluster
  – Group database + application servers in one cluster
  – However, application servers
    • Do not need ASM instance
    • Longer timeouts
  – Light-weight stack
    – No inter-node traffic
    – Local storage or NFS
  – Cloud!
    • Integrated cluster solution
Grid Infrastructure

• **Flex ASM**
  - 5 node-cluster
  - Less than 5 nodes run ASM instance
  • Database requests file mapping from remote ASM
ILM

• Scope
  – Historical data – archiving - compression
  – Applications usually work on recent data
  – Older data may take a lot of place

• Solution pre 12c
  – Move data to other tablespaces on other disks
  – Change compression level
  – Set tablespaces read only
  – → Manually, scripts, 3rd party tools, ...
ILM – Oracle 12c

• “In-database archive”
  – Archive infrequently used data within the database
    • By marking data as archived
    • Making rows ‘invisible’
  – Query can choose to (not) see archived data

• Advanced data optimization
  – Automatically move data based on policies
    • Other tablespace, read only, compression level
  – Online

  alter table ... compress for query after 3 months of no modification
ILM

• Advanced compression
  – Faster and smaller
  – In-memory scan
    • Data not expanded in memory during scan
    • Up to 3x faster for low cardinality data

• Heat maps
  – Track access (read and write) to tables/partitions/rows
  – Information used for ILM
SQL Enhancements

• Duplicate indexes
  – “ORA-01408: such column list already indexed”
  – Can have both B-tree and bitmap index on same column(s)

• WITH-plsql function
  – 4-8x faster execution

WITH
  function is_number(n varchar2)
  return char is
  begin
    <check if n is a number>
  end
select * from <table>
where is_number(sal) = ‘NO’;
SQL Enhancements

• Varchar2(32K)
  – Currently max is 4000 bytes

• Auto-populate column from a sequence
  – Using the DEFAULT clause for a column
SQL Enhancements

• In-memory global temporary tables
  – Useful during reports
  – 12c: purely in memory
    • No IO for redo and undo
    • Can be used on Active Data Guard db

• In-memory LOB queries and updates
  – Speed up LOB operations
    • Concatenate, substring, length, instr, ...
Other useful enhancements

• Data pump
  – Impdp can be done in NOLOGGING
    • Must faster imports
• Move datafiles online
  – While read and write activity are going on
  – Allows easy migration to other storage
• SQL*Net
  – Larger buffers, data compression
Performance

• Very often caused by bad execution plans
  – Usually due to bad statistics
    • Old or missing statistics
    • Hard to predict number of returned rows
      – Complex predicates
        • where substr(to_char(edate,’YYMMDD’),2,2) > 8
      – Join cardinalities
      – Data skew, correlation

• Solutions
  – Adaptive cursor sharing (11g)
  – Adaptive statistics (12c)
  – Adaptive execution plans (12c)
Performance

• 12c: Adaptive statistics
  – Actual number of rows <> estimates
  – Statistics marked as ‘incorrect’ / ‘unreliable’
  – Next query will do ‘dynamic sampling’
    • Results in much better estimates
    • Better execution plans
Performance

• Adaptive execution plans
  – E.g. 2 options in execution plan
    • Join using nested loops
      – Best when few rows need to be joined
    • Join using hash join
      – Best when a lot of rows need to be joined
  – “Inflection point”
    • Rows are buffered during execution of query
    • Inflection point reached or not?: take plan 1 or 2
  – Result: “deferred execution plan”
EM Express

- Replaces Oracle 11g DbConsole
- Embedded in 12c database
- Preconfigured & installed with the database
- Uses less disk space and memory
  - +/- 20Mb footprint
- Subset of OEM12c features
  - Similar interface
EM Express

- Enhanced Real Time ADDM
  - Proactive problem detection and analysis
    - Lightweight check runs every 3 sec
    - On detection of bad performance, analysis is triggered
      - High cpu, io spikes, memory, hangs, ...
    - Collects rich set of data for analysis
    - Stores reports in AWR (persistent - purged)
  - Can be triggered manually
EM Express

• Monitor composite operations
  – ‘label’ a unit of work
    • E.g. SQL*Plus script, batch job, dpump job, ...
    – View top SQL and performance metrics

• “Database Performance Hub”
  – Single view of all performance related info
    • ADDM, Top SQL, ASH analytics, ...
    • Switch easily between sql monitoring, ash, addm, ...
OEM 12c

• **Database Instant Cloning**
  - Using copy-on-write
    • Initial clone takes no space
    • Only modified blocks take space
  - Functional testing with minimal space consumption

• **Integrated subsetting and masking**
  - One-step masking + subsetting as data leaves the source db
Oracle12c + OEM12c = extreme database management
Pluggable databases

- Cloud -> shared infrastructure -> multi-tenancy
  - One application for multiple customers
  - Customers may not see each other's data
    - Solution?
      - Add extra column + where condition (+++work)
      - Create multiple databases
        - High resource usage
          - Memory (1 SGA per database)
          - Processes (a lot per database)
Pluggable db

- **Solution 12c**
  - One ‘container database’
    - Background processes
    - Memory allocation
  - Multiple ‘pluggable databases’
    - The union of
      - Tables, views, procedures, ... all application objects
      - User definitions
      - Privileges
  - Can be plugged in a container database
  - Can easily be moved to another container database
Pluggable databases

• Resource usage
  – 6x less H/W resource, 5x more scalable
Pluggable databases

- Use cases
  - Consolidation
    - Typical 11.2 database has +/- 30 background processes
    - x15 database = 450 processes
      - + user processes
  - “Application as a Service”
    - Multiple customers for same appl
    - Each a separate PDB
Pluggable databases

• Advantages
  – Less resources
  – Security
  – Role separation
    • Administrator of PDB can <> administrator of CDB
  – Less applications in one database
  – Faster upgrades
    • Unplug from 12.1, plug into 12.2
• Use of shared infrastructure
• OEM 12c for provisioning, management, metering, ...
Conclusion

- 12c features
  - Data Guard
    - Get more from Active Data Guard
    - Making Data Guard easier and better
  - RMAN
    - Making backup and recovery easier and better
      - Table recovery, cross platform, incr bup for standby, ...
  - Performance
    - Making the database faster
      - Optimizer, SQL, ...
  - ...

Conclusion

• **Awareness**
  - Management of large volumes of data
    • Compression
    • ILM
    • Automation
  - Management of the environment
    • OEM 12c
      - Get more out of your Diagnostic and Tuning Pack
    • More than just the database
    • Lifecycle management
Conclusion

• Architecture
  – Grid infrastructure
    • Larger RAC clusters
  – Cloud infrastructure
    • Provisioning and management of database resources
    • Private cloud
  – Pluggable databases
    • Change database management
    • Very likely to become a ‘natural thing’
Want to know more?

Contact: geert.depaep@uptime.be