AutoDOP and Resource Manager

Parallelism , Auto DOP and Resource manager
(and how they can work together).

Introduction

- Rick van Ek
- Oracle Senior DBA
- Involved with Oracle since 1992
- Independent since 1996
- https://nl.linkedin.com/in/rickvek
- Married, two children

Agenda

- Parallelism
  - Very short introduction.
- Auto DOP (Degree Of Parallel)
- Auto Dop and Resource manager

Parallelism

- Activate
  - Database level
  - Session level
  - Object level
  - Query level

Parallelism

- Database level
  - Alter system set
  - Choose with Care
- Session level (our preferred method)
  - Alter session ;
    - ALTER SESSION SET parallel_degree_policy = limited;
    - Use a logon trigger.
- Object level
  - Alter table …. parallel…(not recommended)
- Query level, use hint.
  - Needs a change in your application, you not always can or want that.

Parallelism

- Max parallel processes on > 11.2.0.2 from Oracle Documentation.
  - "*parallel_max_servers = PARALLEL_THREADS_PER_CPU * CPU_COUNT * concurrent_parallel_users * 5*
  - *In the formula, the value assigned to concurrent_parallel_users running at the default degree of parallelism on an instance is dependent on the memory management setting. If automatic memory management is disabled (manual mode), then the value of concurrent_parallel_users is 1.*
  - *If PGA automatic memory management is enabled, then the value of concurrent_parallel_users is 2.*
  - *If global memory management or SGA memory target is used in addition to PGA automatic memory management, then the value of concurrent_parallel_users is 4.*
  - *The value is capped by processes -15.*"

Parallelism

- Memory usage;
- Is hard to tell upfront. But it can be measured.
- Look into v$px_buffer_advice.
- Monitor you usage.
  - SELECT POOL, NAME, SUM(BYTES) FROM V$SGASTAT WHERE POOL LIKE '%pool%'
  - GROUP BY ROLLUP (POOL, NAME);
  - Do not forget to measure your PGA usage.

Parallelism

```
SQL> select * from v$px_buffer_advice ;

STATISTIC

VALUE

------------------------------ ----------

Servers Highwater                   4

Buffers HWM                         49

Estimated Buffers HWM               24

Servers Max                         640

Estimated Buffers Max               309120

Buffers Current Free                499

Buffers Current Total               500

7 rows selected
```

Parallelism

- Query Coordinator – QC foreground process for driving sessions. Receives data from query slaves.
- Slaves
  - Producer slaves- get their work from QC, returns data in queues. For QC or Consumer Slaves.
  - Consumer slaves (only generated when needed). Reads from queue and returns to the QC

Parallelism

- Skew Of Data Distribution Across Parallel Slaves.
  - Some slaves have more work to do then others.
  - Observed with partitions.
  - Use sql monitor, easier to find.

http://allthingsoracle.com/parallel-execution-skew-demonstrating-skew/

Parallelism

- Application can build in parallelism
- rather let the database decide this….

Parallelism

- Pros
  - Simple setup
  - Resource usage is limited, easy overview
- Cons
  - Needs maintenance
  - Downgrade when resources are exhausted
  - Downgrades makes run time unpredictable.
  - All queries gets same parallelism, even if they don't need it.

Auto DOP

- "*The AutoDOP is not a feature to use more parallelism. It is a feature that restricts the parallel execution using a balance of cost-benefit. so it is expected that with AutoDOP not all queries will run in parallel and the ones that do run in parallel may not run with full parallelism, as this is the technical specifications of the feature.*"
- Quote from Oracle Documentation.

Auto DOP

- Is dependend on :
  - IO Calibration.
  - System statistics.
  - Object statistics.
  - Resource Manager default plan active.

Auto DOP

- Check configuration before use.
- MOS script pxhcdr.sql
  - Use output as input for your setup.

Auto DOP

- IO Calibration:
  - Runs for 15 minutes on a database
  - Database must be in restricted mode
  - Parameters can be added later manually
  - For SAN, take large number of disks (> 200)
  - There are different opinions about the usage of these parameters.
  - Exadata : use metric_iorm.pl (Doc id 1337265.1)

Auto DOP

- System Statistics
  - Workload
  - No-Workload
    - Make up your mind…….

Auto DOP

- Object statistics:
  - Make sure they are up to date
  - Determine strategy
    - All including histograms
    - All excluding histograms, only histograms were needed.
  - Tip large database with lots of partitions :
    - Use concurrent=true parameter, also on > 11.2.0.3 see documentation on 12C.

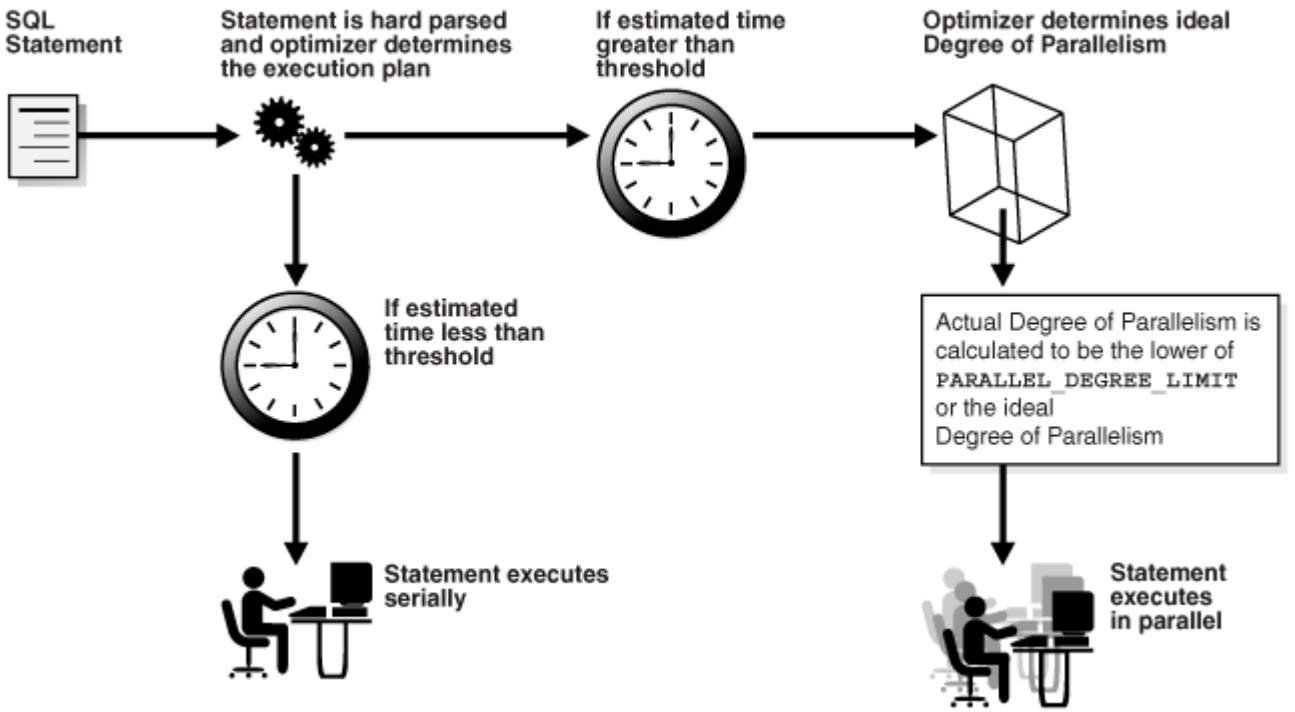
Auto DOP

- Parameters:
  - parallel_degree_policy = auto
    - Can be set on database level
    - Can be set on session level*
  - parallel_force_local= true/false
    - Can profit from cluster, more resources.
  - parallel_degree_limit= CPU
  - parallel_min_time_threshold
    - Default = 10 sec. , if elapse time is > 10 sec. then go parallel.

* We use this a lot for testing

Auto DOP

- From the oracle documentation.
-



Auto DOP

- Pros
  - Fluctuations in number DOP.
  - Some efficiency in usage of resources.
- Cons
  - Resources can still be exhausted.
  - Downgrades still possible
  - Some queries can take to much DOP
    - Seen sessions which take up 64 parallel processes.
  - Chaos can happen as they are out of control.

Monitoring

- Use sql-monitor from EM.
- There are views to use.
- Keep observing behavior, now it is (looking) under control. But it can be chaos with the next change.

Conclusion

- This setup is not suitable for a production enviroment with lots of changes.
- More companies use DevOps, so more changes.
- If you need it for a batch, use it on session level for the batch user.

Next Step

- Handle the cons, get in control.
- The resource manager gives you the tools to get in control.

Resource Manager

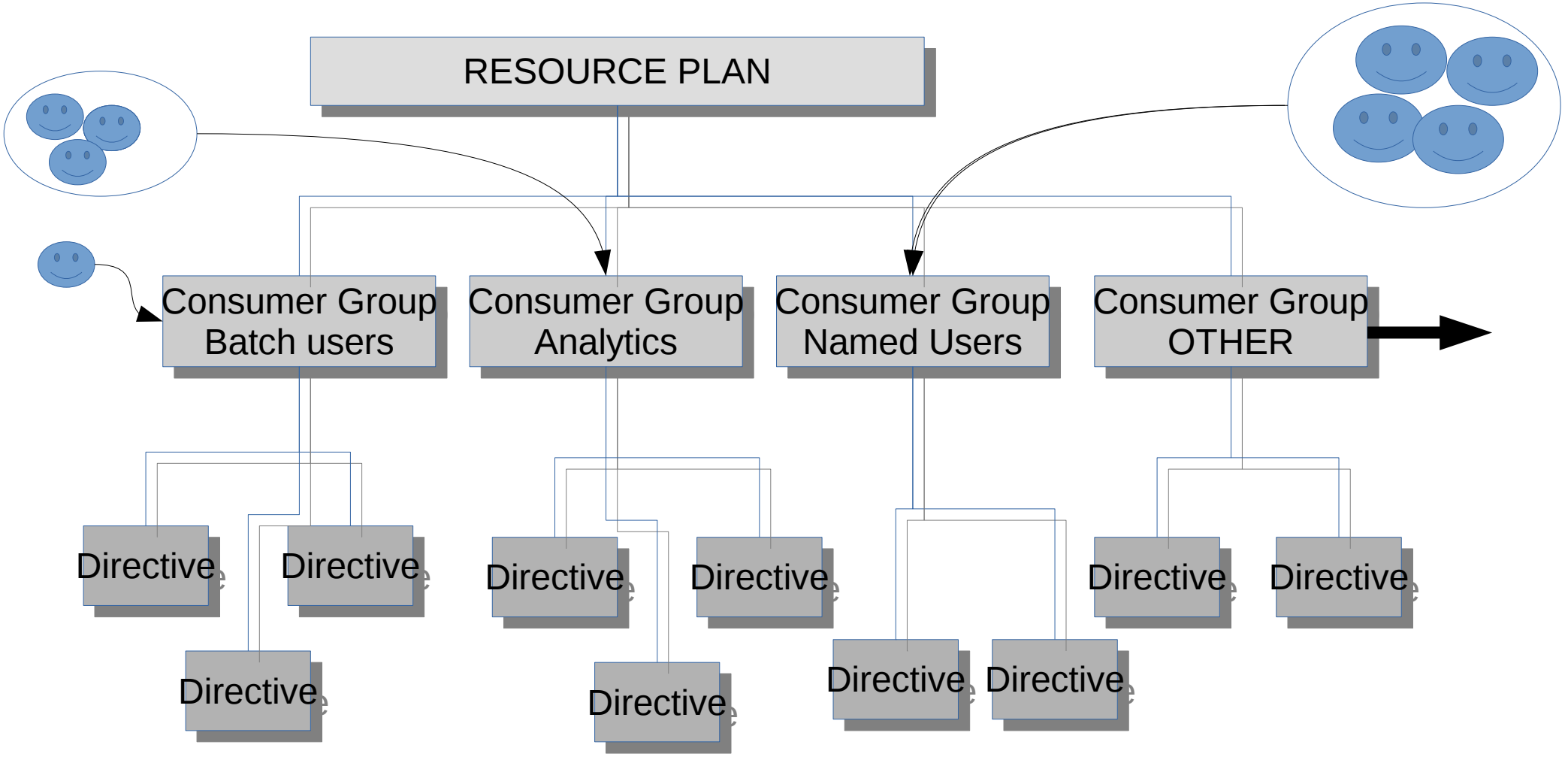
- Setup Resource Manager
  - User set limit for Auto DOP.
  - Activates queuing for parallel query.
  - Can stop/abort long runners.

- Basically getting into control.

Resource Manager

- Setting up the resource manager
  - Create resource plan
  - Create consumer group(s)
  - Add the consumer group(s) to resource plan.
  - Add “other group” to resource plan (default)
  - Create directives for consumer group(s)
  - Assign users to consumer group(s)
    - Default consumer group.
    - Logon trigger.
  - Activate resource plan
    - Example on MOS Doc id 1359043.1

Resource Manager



Resource Manager

- Resource Plan
  - Make a new one save the default for fall back
  - Do not make it more complicated then needed.
  - Divide the amount of resources between your consumer groups and other group.

Resource plan

```
begin
  dbms_resource_manager.create_pending_area();
  dbms_output.put_line('Created pending area.') ;
---1. Create resource plans
end ;
/
begin
  dbms_resource_manager.create_plan(
    plan                      => '&&rplan',
    comment                   => 'Resource plan/method for DOP');
    dbms_output.put_line('Resource plan &&rplan created') ;
  dbms_resource_manager.validate_pending_area();
  dbms_output.put_line('Validated pending area.');
end ;
/
Prompt DO NOT SUBMIT AN AREA WHICH IS NOT VALIDATED CORRECTLY!!!
set verify off ;
pause Submit this plan if  no errors occur.
---submit pending area for plan, consumer group and directives
begin
 dbms_resource_manager.submit_pending_area();
 dbms_output.put_line('Submited pending area.') ;
end;
/
```

Resource Manager

- Consumer group.
  - This is what you assign to your users.
    - Batch job.
    - Tactical queries.
    - Other application users.
  - Capabilities of group is determine by directives.
    - dba_rsrc_consumer_groups

Resource Manager

```
-- create pending area…...
begin
 dbms_resource_manager.create_consumer_group(
    consumer_group            => '&&group1',                --named users
    comment                   => '&&comment1');
 dbms_resource_manager.create_consumer_group(
  consumer_group              => '&&group2',                --batch users
  comment                     => '&&comment2');
 dbms_resource_manager.create_consumer_group(
  consumer_group              => '&&group3',                --adhoc user
  comment                     => '&&comment3');
  dbms_output.put_line('Consumer groups created') ;
end ;
/
-- validate pending area
-- submit pending area when validation is positive.
-- consumer group other already exists but must take part in your setup.
```

Resource Manager

- Directives
  - dbms_resource_manager.create_plan_directive
    - mgmt_pX Resource allocation value for level X (replaces cpu_pX):
      - EMPHASIS - specifies the resource percentage at the Xst level
    - parallel_degree_limit_p1
      - Specifies a limit on the degree of parallelism for any operation.
      - Default is NULL, which means unlimited.

Resource Manager

- The limit is very important, it set the max. level of DOP to be used by Auto DOP.
- Example :
  - Normal parallel we used 8
  - With this setup we used 14,
  - Tested with 8, 10, 12, 14, 16, 18. But 14 seemed to be optimal.
  - So gradually increase this value.

Resource Manager

- Directives
  - dbms_resource_manager.create_plan_directive
    - parallel_target_percentage
      - Specifies the maximum percentage of the target number of parallel servers in an Oracle RAC environment that a consumer group can use. Any additional parallel statements that are launched from this consumer group will be queued.
      - The default is NULL, which means that the limit is 100% of the target number. Valid values for queuing are in the range of 0 to 100 (%). For updates to the plan directive, the value of -1 will reset the value to NULL.

Resource Manager

- Queuing
  - This is the most important part. It prevents the database from downgrading the DOP.
  - Better to wait for 5 or 10 minutes then running a query which will take 8 hours. Instead of 30 minutes.
  - Makes your throughput better predictable.

Resource Manager

- Directives
  - dbms_resource_manager.create_plan_directive
    - parallel_queue_timeout
      - Specifies the time (in seconds) that a query may remain in its Consumer Group's parallel statement queue before it is removed and terminated with an error (ORA- 07454).

Resource Manager

- Directives
  - dbms_resource_manager.create_plan_directive
    - max_est_exec_time
      - Specifies the maximum execution time (in CPU seconds) allowed for a session. If the optimizer estimates that an operation will take longer than MAX_EST_EXEC_TIME, the operation is not started and ORA-07455 is issued. If the optimizer does not provide an estimate, this directive has no effect. Default is NULL, which means unlimited.

Resource Manager

```
dbms_resource_manager.create_plan_directive(
   plan                      => '&&rplan',                 --named users
   group_or_subplan          => '&&group1',
   comment                   => 'Named users sessions at level 1',
   mgmt_p1                   => 5 ,
   parallel_degree_limit_p1  => '&&degree1' );

 dbms_resource_manager.create_plan_directive(
   plan                      => '&&rplan',                 --batch users
   group_or_subplan          => '&&group2',
   comment                   => 'batch users sessions at level 1',
   mgmt_p1                   => 60 ,
   parallel_degree_limit_p1  => '&&degree2' );                     -- increased from 70% to 90%

 dbms_resource_manager.create_plan_directive(
   plan                      => '&&rplan',                 -- adhoc users
   group_or_subplan          => '&&group3',
   comment                   => 'Repos users sessions at level 1',
   mgmt_p1                   => 10 ,
   parallel_degree_limit_p1  => '&&degree3' );

 dbms_resource_manager.create_plan_directive(
   plan                      => '&&rplan',                 -- not defined users .... mandatory group
   group_or_subplan          => '&&group4',
   comment                   => 'OTHER_GROUPS at level 1',
   mgmt_p1                   => 5,
   parallel_degree_limit_p1  => '&&degree4' );

 dbms_resource_manager.create_plan_directive(
   plan                      => '&&rplan',                 -- not defined users .... mandatory group
   group_or_subplan          => '&&group5',
   comment                   => 'SYS_GROUP at level 1',
   mgmt_p1                   => 20 ,
   parallel_degree_limit_p1  => '&&degree5' );
```

Resource Manager

- Review Directives
  - dba_rsrc_plan_directives

Resource Manager

- Assign user to consumer group.
- We placed every user in its appointed consumer group.
- With login they are assigned to their initial Consumer Group.

Resource Manager

```
for people in
  (select username usr
  from dba_users
  where username    like 'OWNER%'
  and username not  like '%PC1'
)
loop
  dbms_output.put_line('Processing user : '|| people.usr ) ;
  dbms_resource_manager_privs.grant_switch_consumer_group( grantee_name => people.usr
      , consumer_group  => '&&group2'
      , grant_option    => FALSE ) ;
  dbms_resource_manager.set_initial_consumer_group(  user => people.usr
      , consumer_group  => '&&group2'  ) ;
end loop ;
```

Resource Manager

- Modify Maintenance plan
  - Make a new level add consumers group
  - If the user runs queries while maintenance plan is active. Then they are out of control.

Resource Manager

- What to monitor
  - Limits on consumer group.
  - Amount of queuing
  - Skew
  - Px buffer
    - MOS Doc id 1359043.1 has some nice queries to start with.
    - EM – SQL monitor also give some insights.

Resource Manager

- Remarks
  - 11.2.0.2 < your version >11.2.0.4 => check your patches MOS Doc id 1340180.1

What have we achieved?

- Finer tuning of Auto DOP.
- DOP only used when needed.
- Limiting on DOP while running Auto DOP.
- Queuing on PQ, preventing downgrades.
- Group users in Consumer Groups, different levels of DOP limit/resources.
- All is arranged inside the database.

Finally the end.

- There is more to explore.
- If you know the involved parts, then the search is easier.
- Plenty to test.