

Exadata Database Machine Resource Management – teile und herrsche!

DOAG Conference 2011

Konrad Häfeli

Senior Technology Manager
Trivadis AG

17.11.2011

BASEL BERN LAUSANNE ZÜRICH DÜSSELDORF FRANKFURT A.M. FREIBURG I.BR. HAMBURG MÜNCHEN STUTTGART WIEN

1

2011 © Trivadis

Exadata Database Machine Resource Management - teile und herrsche!
17.11.2011

trivadis
makes IT easier. ■ ■ ■

Trivadis facts & figures



11 Trivadis locations with more than 550 employees

Financially independent and sustainably profitable

Key figures 2010

- Revenue CHF 101 / EUR 73 Mio.
- Services for more than 700 clients in over 1,800 projects
- Over 170 Service Level Agreements
- More than 5,000 training participants
- Research and development budget: CHF 5.0 / EUR 3.6 Mio.

Why we are special

Customer-specific solution competence and vendor independence

- offers substantiated techniques and skills as well as self-developed approaches
- guarantees repeatable quality and a safe execution

Technology competence

- offers more than 17 years of expertise in Oracle and Microsoft
- has its own Technology Center and strives for technological excellence

Solution and integration expertise

- has a wide and cross-sectoral customer basis and more than 1800 projects every year
- Combines technological expertise with an understanding of the specific business needs of the client

Support for the entire IT project lifecycle

- has a modular portfolio of services for the entire IT project lifecycle
- provides the appropriate combination of solutions and services for every „level of maturity“

AGENDA

1. Introduction
2. Database Resource Manager
3. Exadata IO Resource Manager
4. Tests
5. Tooling
6. Conclusion

Introduction

- Exadata is known for its enormous performance
- This is optimal for consolidation
- Concurrency forces bottlenecks
- ... but the resource cake is not endless



- How can I guarantee everybody's piece?

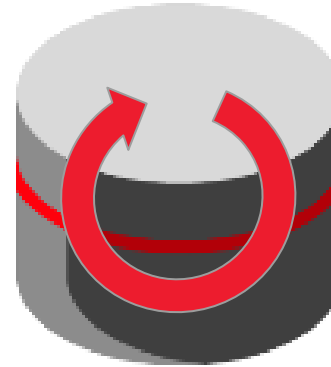
→ Resource Management

Database Resource Management

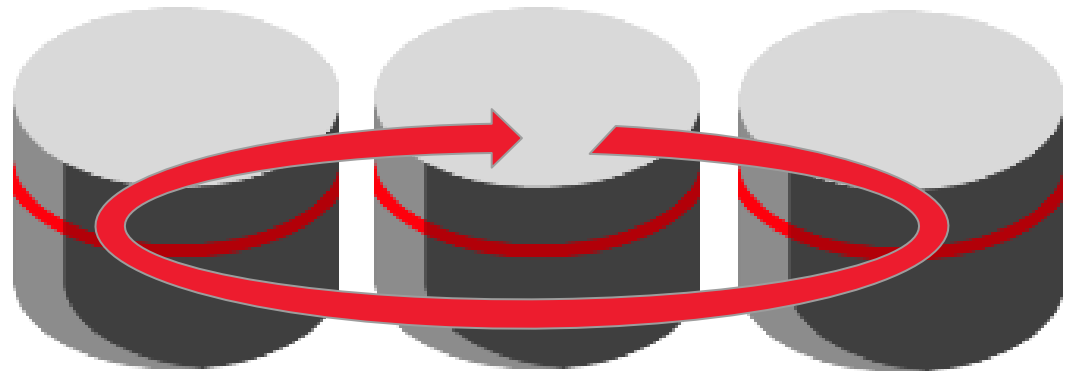
- The goal of resource management is to optimize resources allocation among **different users / groups or programs**
 - Example: Online users get more power than batch users
 - Example: „batch“ gets much resources during the night and little during the day
- Sounds familiar!?!
 - DBRM was introduced in Oracle 8i
- It's not just about CPU (more introduced over the coming versions)
 - Also helps to manage runaway queries, parallel queries, idle sessions, and undo space
 - Instance caging
- Limits, and takes automatic actions if a limit is exceeded

Exadata Resource Management

- Intra-Database is done
 - Via DBRM



- Inter-Database is introduced
 - with IORM on Exadata



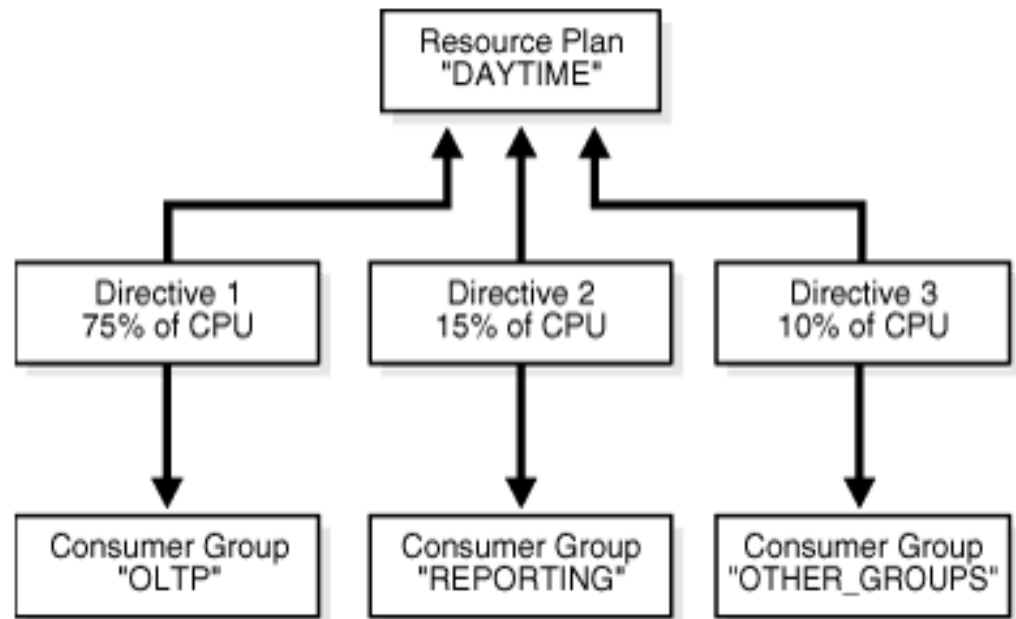
AGENDA

1. Introduction
2. Database Resource Manager
3. Exadata IO Resource Manager
4. Tests
5. Tooling
6. Conclusion

Overview DBRM

Oracle Resource Manager is organized with

- Resource Plans
- Directives
- Consumer Groups



Ref: Oracle 11gR2 Database Administrator's guide

DBRM – Resource Plans

- A resource plan regulates, how specific resources are shared among the consumer groups. It also regulates how unused resources are redistributed
 - Resource Plans get activated on Oversubscription
 - Only one resource plan is active at a specific time
 - It can be enabled manually by parameter
 - Or it can be enabled by a scheduler window

DBRM – Directives

- A resource plan directive assigns values/rules to the resource plan. There are many parameters available, specifying values for such as
 - cpu consumption up to 8 levels
 - number of concurrently active sessions
 - parallel degree limit
 - amount of undo information in KB for a session
 - Maximum idle time for a blocking session
 - Amount of MB transferred (Read/Write) by a session before action specified in SWITCH_GROUP ist taken

DBRM – Consumer Groups

- A consumer group is a collection of sessions that have similar requirements
- Consumer groups can also be created to represent certain kinds of workload
- There exist predefined consumer groups in the oracle server, for example SYS_GROUP and the default group for sessions OTHER_GROUPS
- A consumer group mapping rule defines, how sessions are mapped to consumer groups, for ex:
 - service_name
 - oracle_user
 - client_program
 - client_machine

DBRM – Manipulation (1)

- The Resource Manager can be switched on- or off online

```
SQL> alter system set resource_manager_plan = 'DEFAULT_PLAN'
                                             sid='*' `;
```

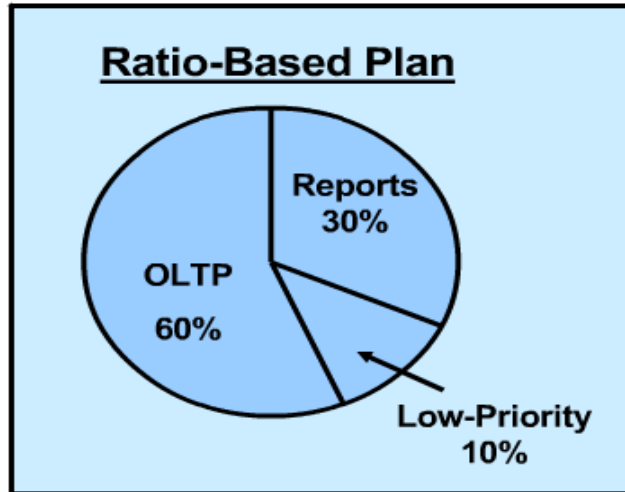
```
SQL> select plan from dba_rsrc_plans order by 1;
PLAN
-----
APPQOS_PLAN
DAYTIME_PLAN
DEFAULT_MAINTENANCE_PLAN
DEFAULT_PLAN
DSS_PLAN
ETL_CRITICAL_PLAN
INTERNAL_PLAN
INTERNAL QUIESCE
MIXED_WORKLOAD_PLAN
ORA$AUTOTASK_HIGH_SUB_PLAN
ORA$AUTOTASK_SUB_PLAN
```

DBRM – Manipulation (2)

```
BEGIN
DBMS_RESOURCE_MANAGER.CREATE_PENDING_AREA();
DBMS_RESOURCE_MANAGER.CREATE_PLAN('DAYTIME_PLAN', 'Resource plan for managing
all applications between 9 am and 5 pm');
DBMS_RESOURCE_MANAGER.CREATE_CONSUMER_GROUP('CRITICAL_APPS', 'Critically
important Apps');
DBMS_RESOURCE_MANAGER.CREATE_CONSUMER_GROUP('ONLINE_APPS', 'Online Apps');
DBMS_RESOURCE_MANAGER.CREATE_CONSUMER_GROUP('DEVELOPMENT', 'Developpers');
DBMS_RESOURCE_MANAGER.CREATE_PLAN_DIRECTIVE('DAYTIME_PLAN', 'CRITICAL_APPS',
'Allocation for CRITICAL_APPS', MGMT_P1 => 60);
DBMS_RESOURCE_MANAGER.CREATE_PLAN_DIRECTIVE('DAYTIME_PLAN', 'ONLINE_APPS',
'Allocation for ONLINE_APPS', MGMT_P1 => 25);
DBMS_RESOURCE_MANAGER.CREATE_PLAN_DIRECTIVE('DAYTIME_PLAN', 'DEVELOPMENT',
'Allocation for DEVELOPMENT', MGMT_P1 => 10);
DBMS_RESOURCE_MANAGER.CREATE_PLAN_DIRECTIVE('DAYTIME_PLAN', 'OTHER_GROUPS',
'Allocation for default group', MGMT_P1 => 5);
DBMS_RESOURCE_MANAGER.VALIDATE_PENDING_AREA();
DBMS_RESOURCE_MANAGER.SUBMIT_PENDING_AREA();
END;
/
```

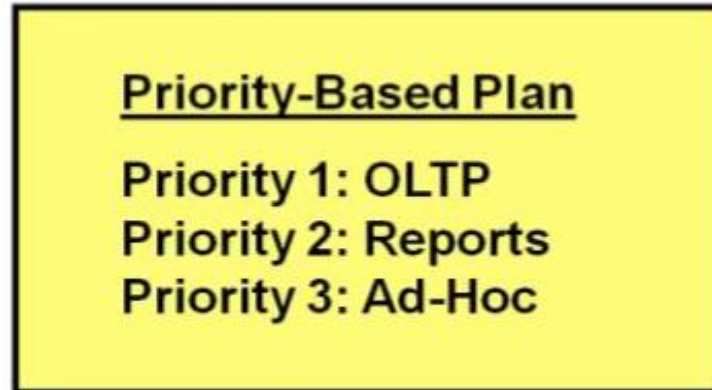
DBRM – Components

- Ratio Based



- Reports wants 40% -> Okay?
 - Yes, if not oversubscribed

- Priority Based



Hybrid Plan

	Level 1	Level 2
OLTP	90%	
Reports		60%
Low-Priority		40%

DBRM – Instance Caging

- Small but useful new feature of Oracle 11gR2
- Limits (only) the amount of CPU's a Instance can use
- One Instance will not be able to allocate all the CPU resources of a server
- Instance Caging has no impact to the CPU's that has to be licensed
- Two different approaches:
 - Over Provisioning
 - Partitioning

```
SQL> alter system set resource_manager_plan='some plan';
```

```
SQL> alter system set cpu_count=5;
```


AGENDA

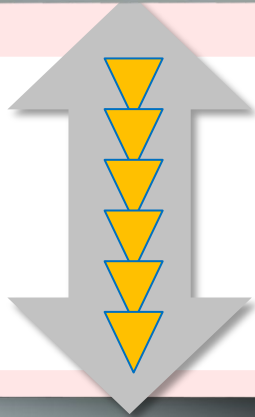
1. Introduction
2. Database Resource Manager
3. Exadata IO Resource Manager
4. Tests
5. Tooling
6. Conclusion

Exadata I/O Resource Management (1)

ORACLE
DATABASE



Consumer Groups /
Categories



I/O Metadata:
DB Name, Request Type, Consumer Group

ORACLE
EXADATA



Resource Manager

Overview Exadata IORM (1)

- Adds the possibility to prioritize I/O's
- Can be used on cell only (**dbPlan**, Inter Database IORM) or in combination with Resource Manager within the DB (**catPlan**, Intra Database Plan)
- Hard limits (**limit**) or soft limits (**allocation**) possible
- Limits the I/O requests only if there is I/O pressure on the cell (soft limits)
- Without pressure, every database or consumer group is able to use 100% of the I/O throughput of a cell (soft limits)
- IORM Plans must be implemented and activated on every cell. Use **dcli** for that.

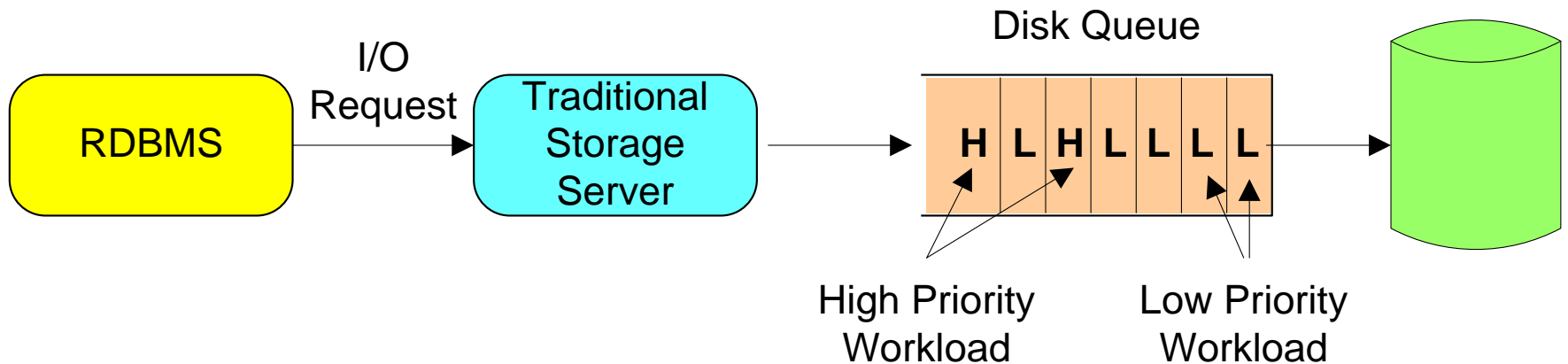
Overview Exadata IORM (2)

- IORM Plans can have multiple levels
 - Sublevels get what is left over from the upper levels
- DB Plans and Category Plans always need a "OTHER" group
- Database Role specific settings possible
- IORM can be activated, deactivated or altered online

```
oracle@exadb01:bin/ [rdbms1120] ./dcli -g mycells cellcli -e 'alter
iormpplan active'
exacel01.trivadis.com: IORMPLAN successfully altered
exacel02.trivadis.com: IORMPLAN successfully altered
exacel03.trivadis.com: IORMPLAN successfully altered
```

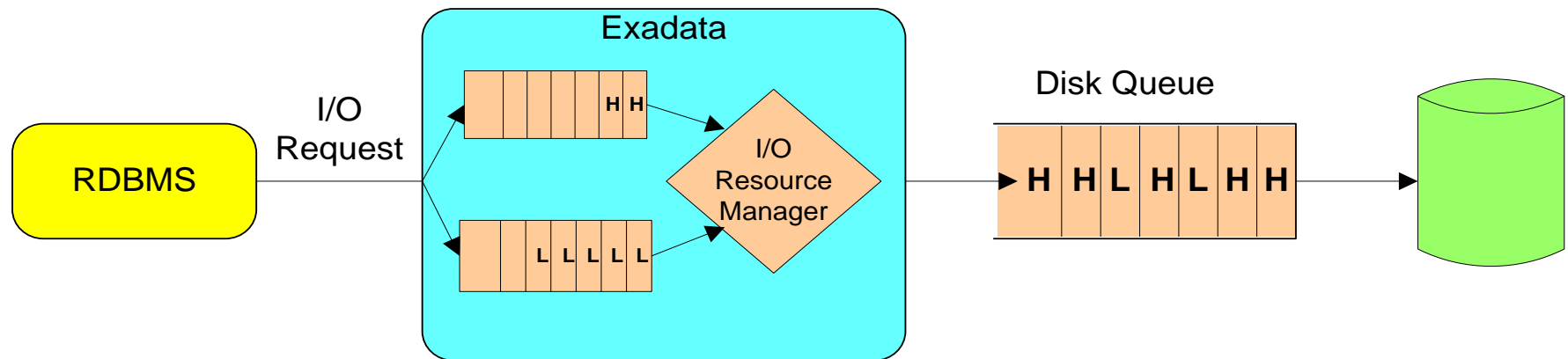
IORM – I/O Scheduling Traditional Way

- Traditional I/O schedulers are black boxes
- No possibility to influence
- I/O requests are processed in FIFO order



IORM – I/O Scheduling Exadata Way

- I/O Requests are executed based on the prioritisation
- Low priority Requests can be queued if needed



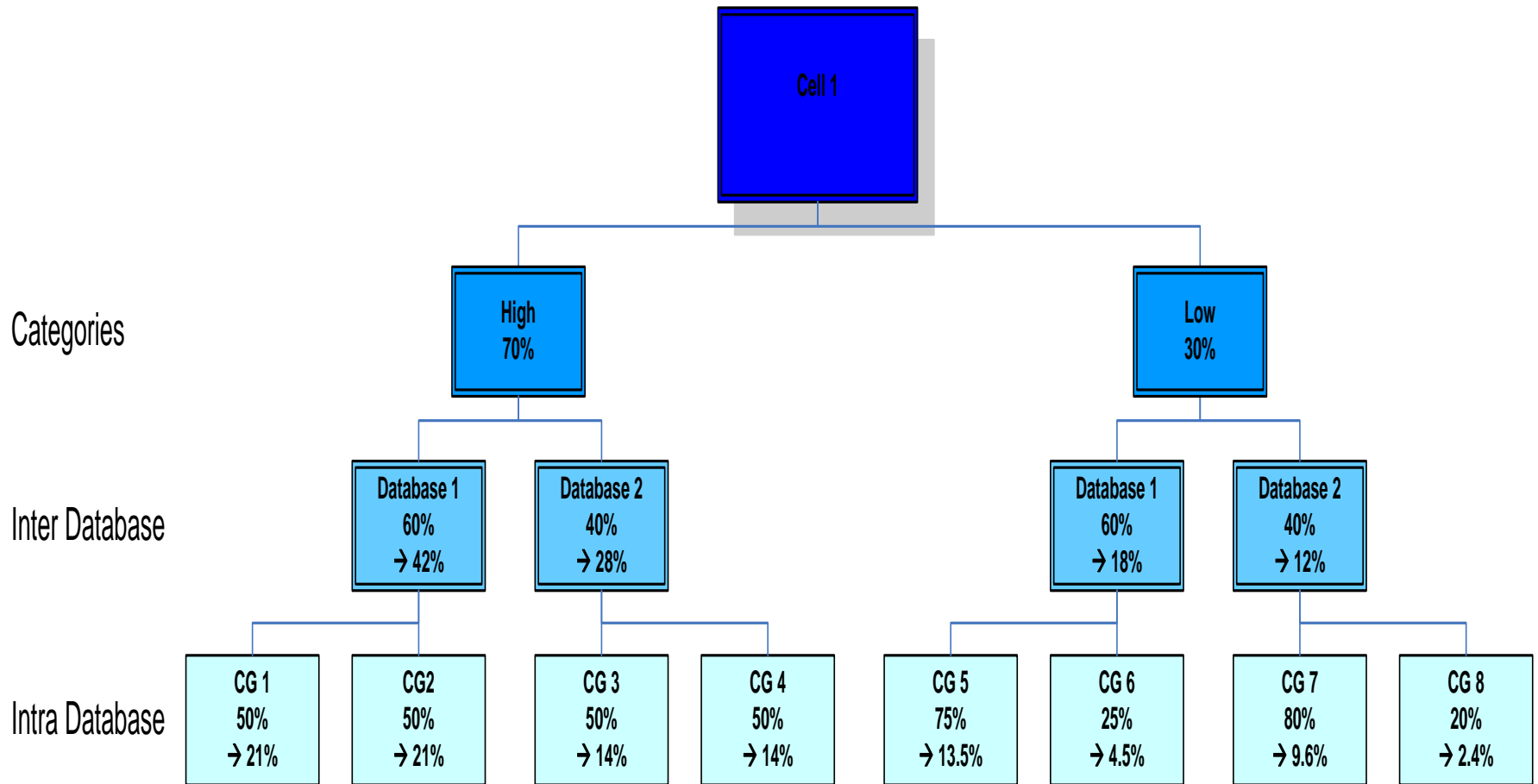
Inter- / Intra Database Resource Management

- **Intra** DB RM controls the resources **within** one database
 - Tools to use:
 - dbms_resource_manager
 - Consumer Groups and Plan Directives: CPU and I/O resources
 - (Consumer groups represent collections of users within a database)
- **Inter** DB RM controls the resources **between** multiple databases
 - Tools to use:
 - (Instance Caging: CPU Resources)
 - IORM: I/O Resources (**dbPlan** and/or **catPlan** on cell)
 - dbms_resource_manager Categories: I/O resources
 - The category's must be present in the DB and on each cell
 - (categories represent collections of consumer groups across all databases)

IORMPLAN on Cells

```
dcli -g mycells cellcli -e \
  "ALTER IORMPLAN catPlan=\ ( \
    \(name=administrative, level=1, allocation=80\) , \
    \(name=prio1, level=2, allocation=60\) , \
    \(name=prio2, level=2, allocation=30\) , \
    \(name=prio3, level=2, allocation=10\) , \
    \(name=other, level=3, allocation=100\) \
      \) , \
      dbPlan=\ ( \
        \(name=rmtest1, level=1, allocation=60\) , \
        \(name=rmtest11, level=1, allocation=10\) , \
        \(name=other, level=2, allocation=100\) \
          \) "
```


IORM – Priorities

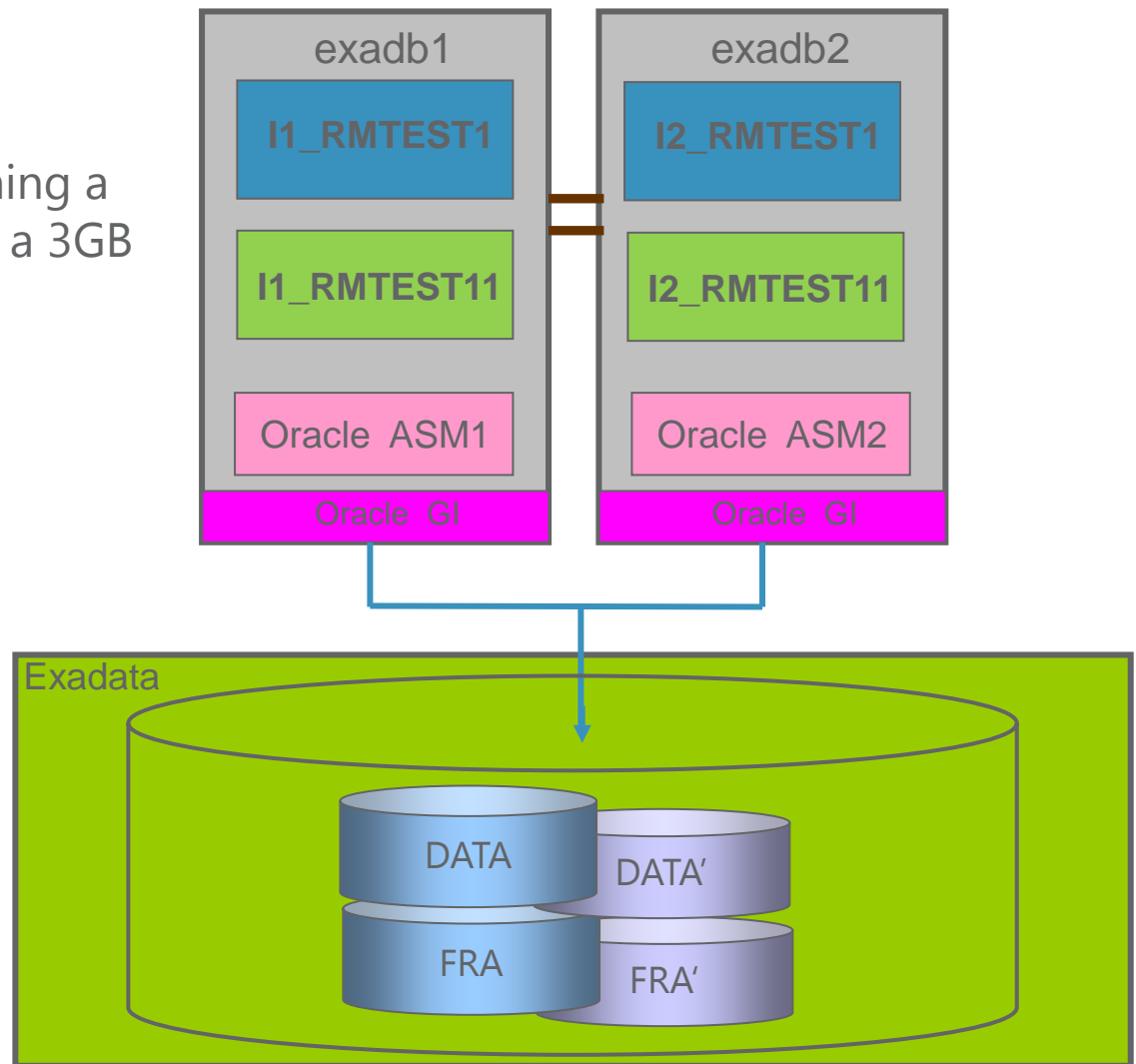


AGENDA

1. Introduction
2. Database Resource Manager
3. Exadata IO Resource Manager
4. Tests
5. Tooling
6. Conclusion

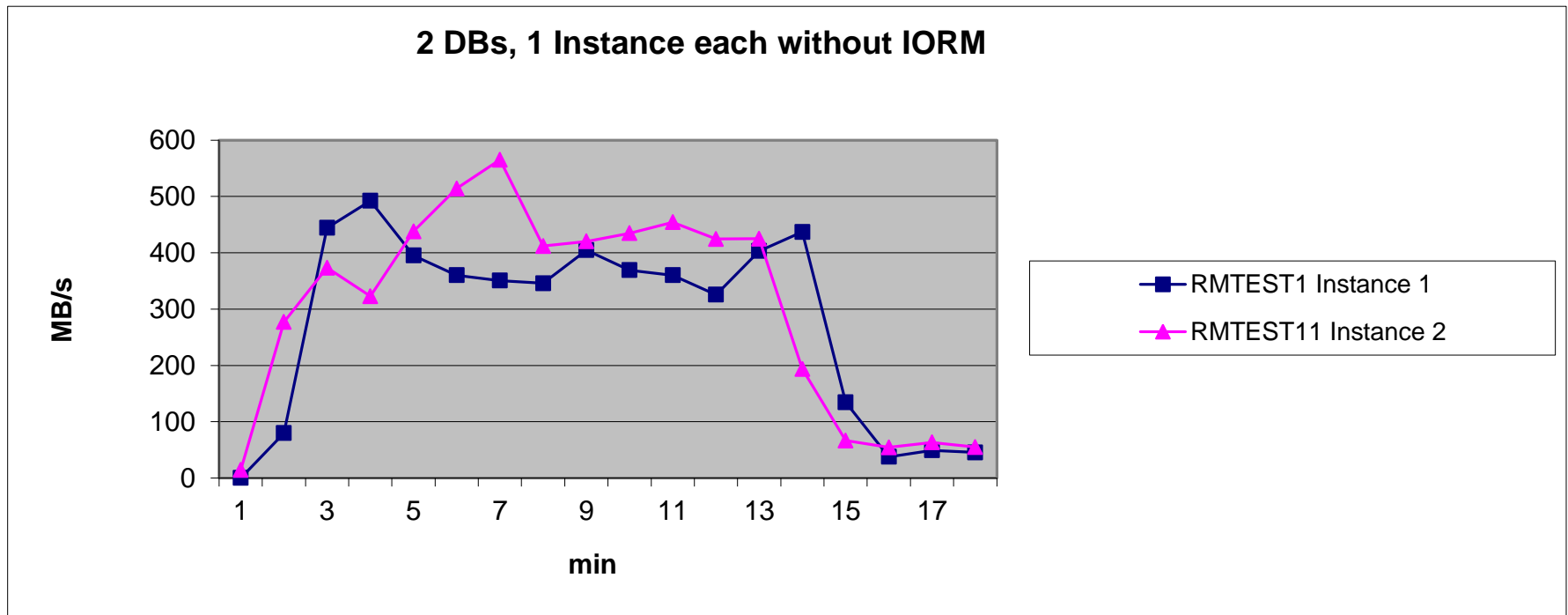
DBRM and IORM Tests – Test Environment

- Test runs
 - 80 parallel sessions running a create table as select on a 3GB table



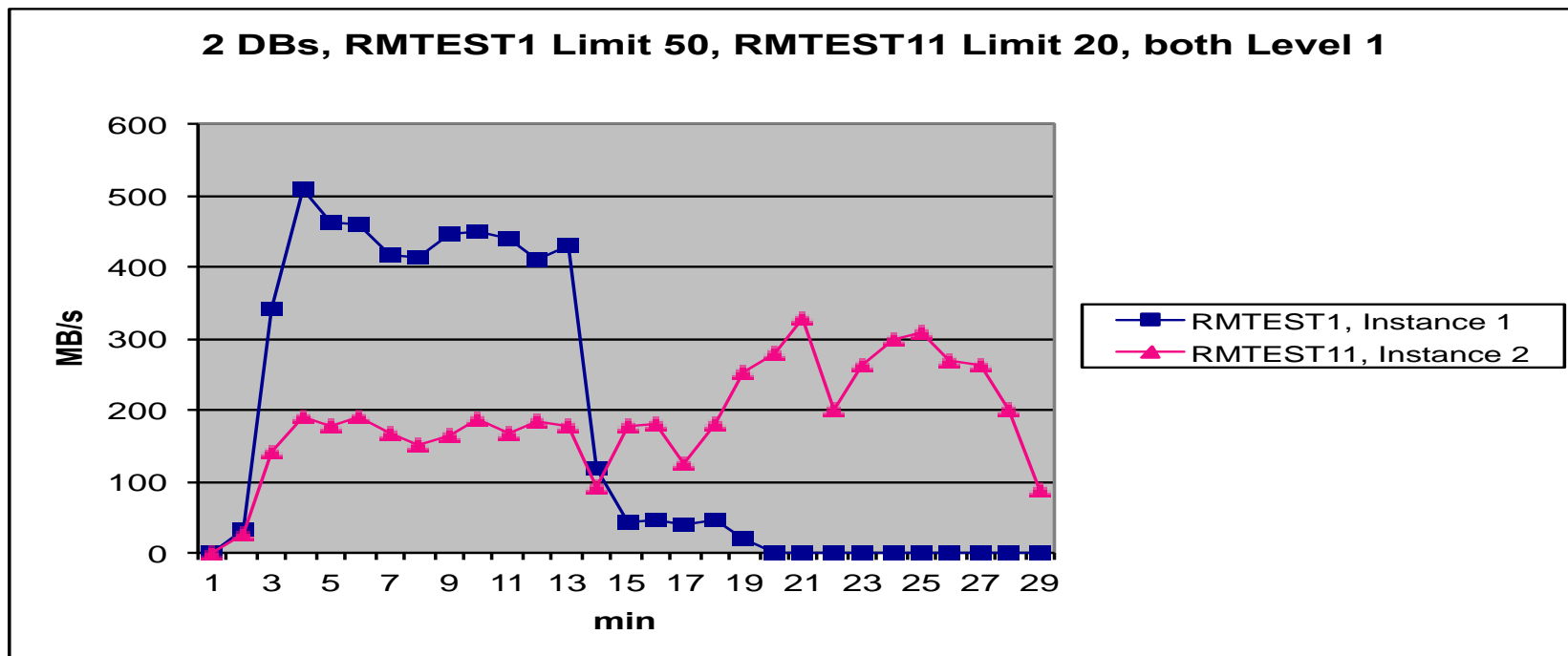
DBRM and IORM Tests – Results Inter-Database IORM (1)

- 2 Databases, 1 Instance each, **no IORM**
 - Rivalry is visible nicely



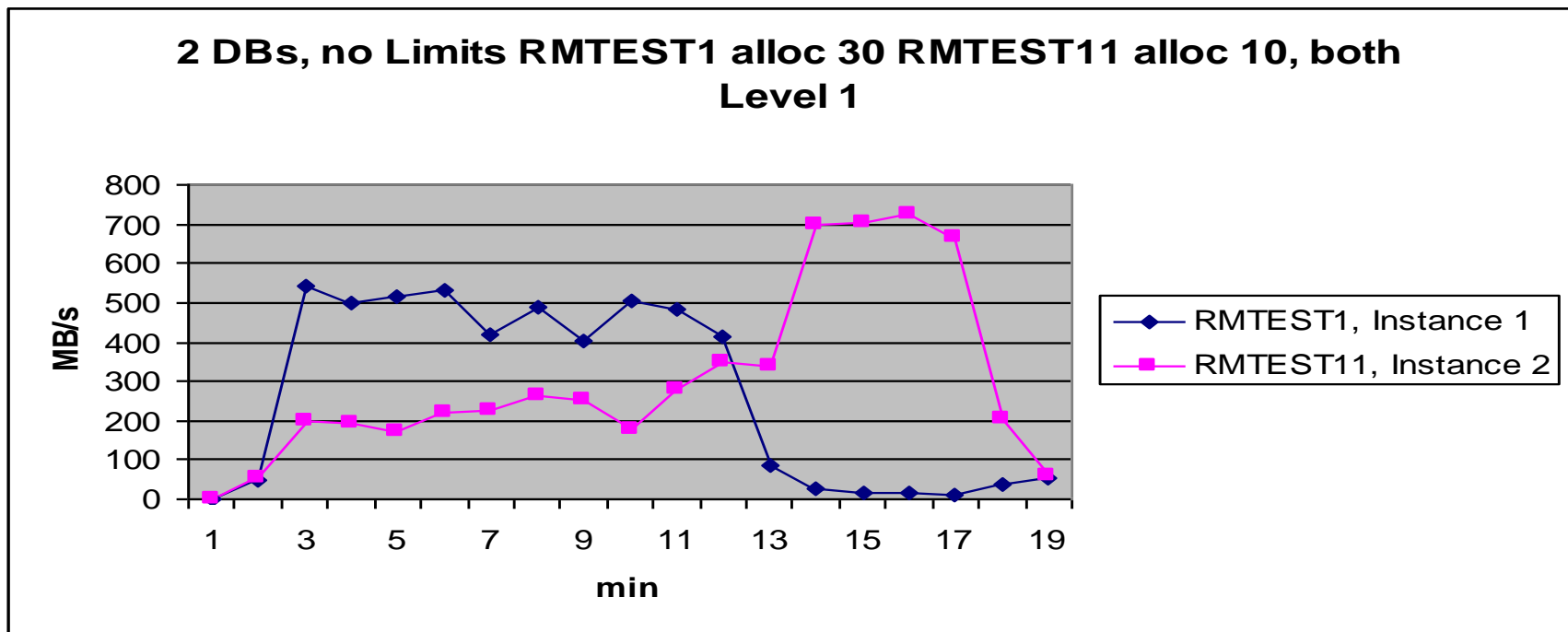
DBRM and IORM Tests – Results Inter-Database IORM (2)

- 2 Databases, 1 Instance each,
 - IORM Hard limit (limit=50 and limit=20, level 1 of max possible throughput)
 - Hard limit: RMTEST11 does not get the full throughput even if RMTEST1 is idle



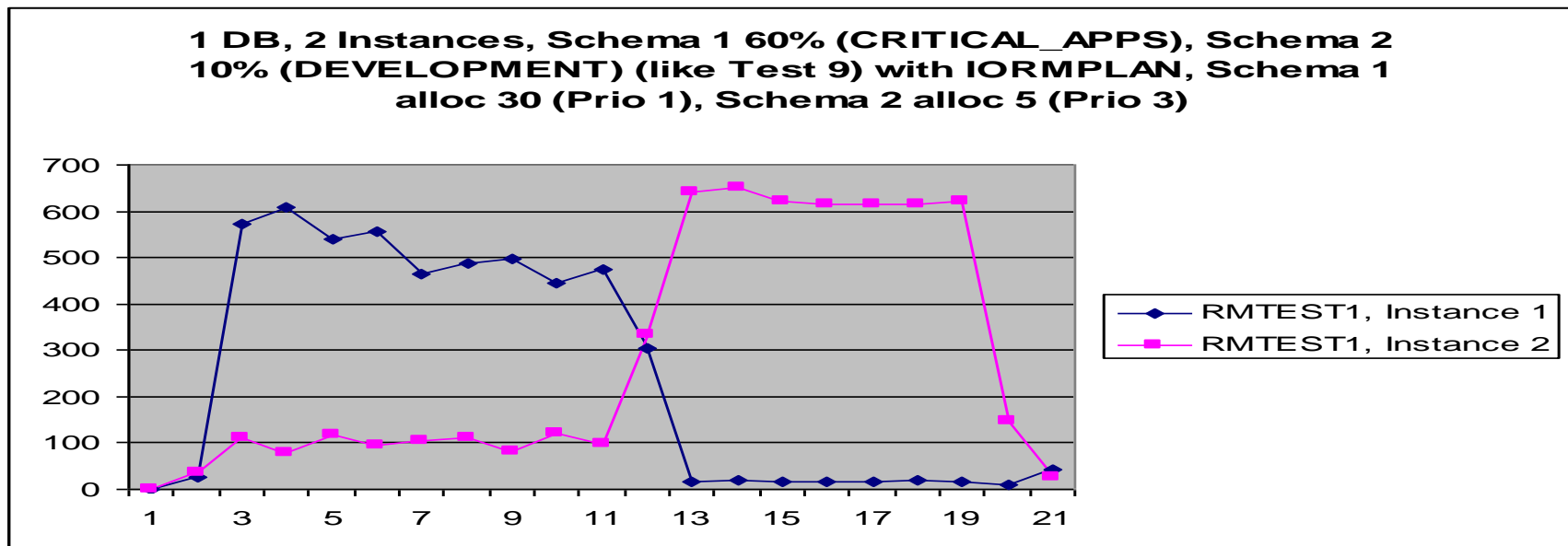
DBRM and IORM Tests – Results Inter-Database IORM (3)

- 2 Databases, 1 Instance each, IORM Soft limit
 - Soft limit: RMTEST1 has 3x more resources (alloc=30 and alloc=10)
 - As soon as RMTEST1 is finish, RMTEST11 gets all the resources



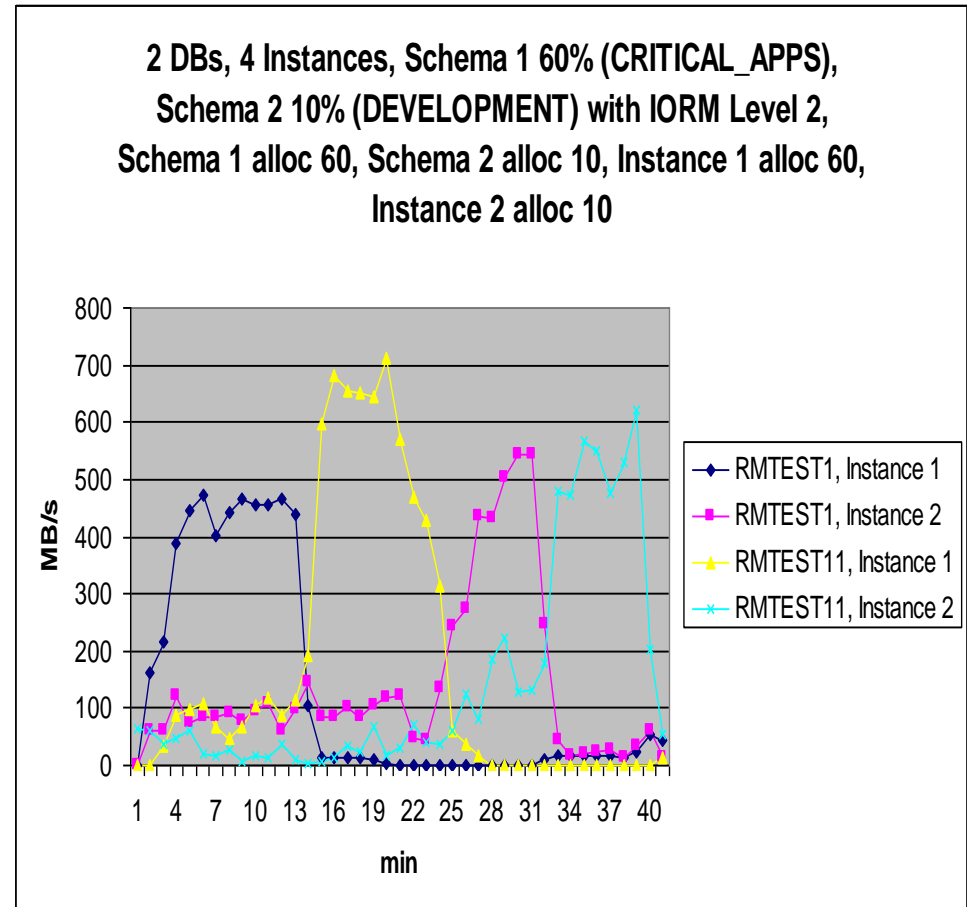
DBRM and IORM Tests – Results Intra-Database IORM (1)

- 1 Database, 2 Instances
 - consumer group "CRITICAL_APPS" (category "prio 1"): 60% on DB
 - consumer group "DEVELOPMENT" (category "prio 3"): 10% on DB
 - Category "prio 1": alloc 30
 - Category "prio 3": alloc 5



DBRM and IORM Tests – Results Intra-Database IORM (2)

- 2 Databases, 4 Instances
 - consumer group "CRITICAL_APPS" (category "prio 1"): 60% on DB
 - consumer group "DEVELOPMENT" (category "prio 3"): 10% on DB
 - Category "prio 1" Level 2: alloc 60 on cell
 - Category "prio 2" Level 2: alloc 10 on cell
 - RMTEST1: alloc 60% on cell
 - RMTEST11: alloc 10% on cell
- RMTEST11, Instance 1 gets more resources because of the prioritisation
- CATEGORY is evaluated first



AGENDA

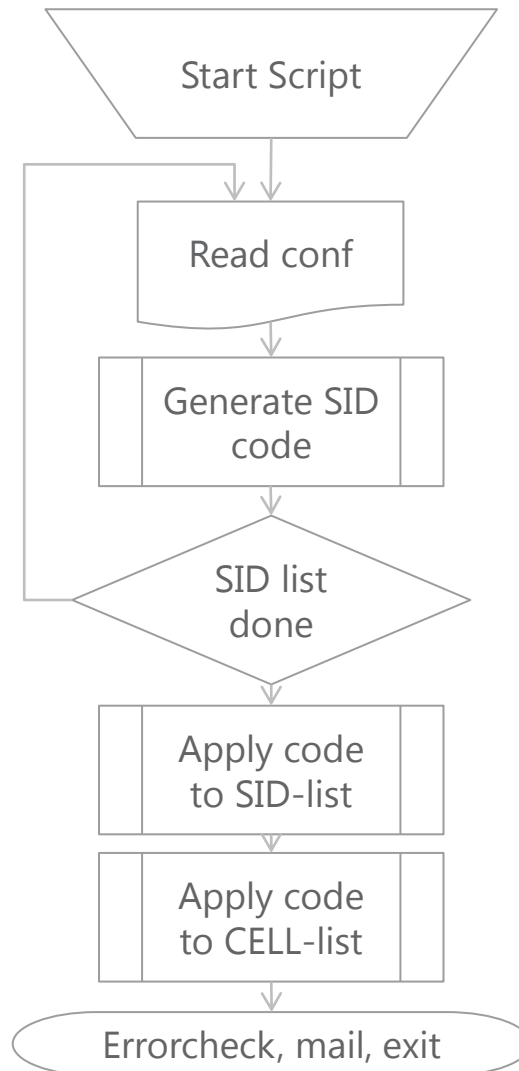
1. Introduction
2. Database Resource Manager
3. Exadata IO Resource Manager
4. Tests
5. Tooling
6. Conclusion

Resource Management Challenges

- Kind of “complicated”
- Has to be changed according the systems needs
 - New Database
 - New Applications
 - New User
- Flexible and easy to use
- Has to be reproducible
- Monitorable
- Resilient

→ Write a tool with a configuration file

Resource Management Tool



Resource Management Tool – Configuration Sections

- Intra
 - Database Category's
 - Intra Database ConsumerGroups
 - Intra Database IORM catPlan
 - Intra Database Resource Plan
 - Intra Database Resource Plan Directives
 - Intra Database Consumer Group Mapping
 - Intra Database Consumer Group Grants
- Inter
 - Database IORM dbPlan

```
# Primary databases
dbplan name=RMTEST1, level=1, allocation=60, role=primary
dbplan name=RMTEST11, level=1, allocation=10, role=primary
dbplan name=ECAZT, level=1, allocation=10, role=primary
# Standby databases
dbplan name=RMTEST1, level=2, allocation=60, role=standby
```

Resource Management Tool – Generate script (1)

```
# ***** *
# * M A I N *
# ***** *
# *-----*
clear

# -----
# Exadata IORM
# -----

getCatPlans
getDBPlans
genCellCommand
getRMPlans
getRMCategorys
getRMConsGroups
getRMPlanDirectives
getRMCGMappings
getRMCGGrants
```

Resource Management Tool – Generate script (2)

```
.  
.br/>else  
    genRMScript  
    applyRMScript  
fi  
done  
fi  
  
checklog  
  
# ***** *  
# * E N D *  
# ***** *  
  
sendMail
```

AGENDA

1. Introduction
2. Database Resource Manager
3. Exadata IO Resource Manager
4. Tests
5. Tooling
6. Conclusion

Oracle Exadata Resource Management – Conclusion (1)

- Resource Management in Consolidation Environments is a MUST
- DBRM and IORM work very well
- KISS! Keep it ... simple!
- First implement a good CATEGORY Plan
 - CRITICAL_APPS, BATCH_LOADING, DEVELOPMENT, REPORTS etc.
- Second implement a Database IORM Plan
- Only limit the Consumer Groups within a database if needed, it's easier to use category's within a database only
- A generator Script helps to manage the needed PL/SQL Code

Oracle Exadata Resource Management – Conclusion (2)

- How can you keep the smile on your Database?



→ With Resource Management!

THANK YOU.

VISIT US AT THE
TRIVADIS-STAND:

Floor 3, No. 304

Trivadis AG

Konrad Häfeli

Papiermühlestrasse 73
CH-3014 Bern

Tel. +41-31-928 09 60

Fax +41-31-928 09 64

konrad.haefeli@trivadis.com

www.trivadis.com

BASEL BERN LAUSANNE ZÜRICH DÜSSELDORF FRANKFURT A.M. FREIBURG I.BR. HAMBURG MÜNCHEN STUTTGART WIEN