



Gerenciamento de Recursos,
da conexão ao I/O

Fernando Simon

Agenda

- Apresentação;
- Consolidação;
- Gerenciamento de Recursos:
 - Services;
 - Instance caging;
 - Resource Manager;
 - IORM;
- Análise do ambiente Exadata:
 - Enterprise Manager;
 - Métricas;
- Dúvidas;

Apresentação

- Fernando Simon:
 - Formado em Ciência da Computação;
 - DBA desde 2005:
 - Oracle, SQLServer, DB2, PostgreSQL;
 - Certificações:
 - OCA, OCP, OCE RAC, MCST;
 - DBA TJSC desde 2011
 - DMA Exadata;

Apresentação

- Exadata no TJSC:
 - Foi o segundo do Brasil:
 - E o primeiro OLTP;
 - Consolidação do ambiente:
 - Centralização das bases de dados do Primeiro Grau
 - + de 100 bases consolidadas em 4;
 - Projeto de 3 anos (término 12/2013 – 03/2014);
 - Bases do Segundo Grau:
 - Migração de ambiente, SPARC -> Intel;
 - Conta única:
 - \$\$\$;
 - Serviços/Bases Administrativas;
 - Total de 14 bases OLTP/DSS em um Exadata V2:
 - Dataguard para uma X2-2 (12/2013);

Consolidação

- Tema do Webinar do Rodrigo Almeida;
- Em resumo, centralização de diversos bancos em um único local;
- Palavra chave em um ambiente Exadata;
- Com a consolidação vem a necessidade por controle, pela gerência inteligente do ambiente;
 - Felizmente no Exadata temos algumas funcionalidades que podem nos ajudar;

Consolidação

- Fato, nem todas as suas bases tem a mesma prioridade:
 - Sempre temos alguma base “mais importante”;
- Em um ambiente consolidado temos que identificar e organizar os recursos necessários por cada base:
 - CPU, Memória, Disco, Rede...
- Em um ambiente como o Exadata o gerenciamento de recursos é fundamental;

Gerenciamento de Recursos

- Um bom plano de gerenciamento de recursos começa antes da implantação da base de dados
 - Capacity planning?
- A chave é levantar o máximo de informação possível;
- Em um ambiente consolidado é importante identificar pontos em comum entre as bases que rodam no Exadata;

Gerenciamento de Recursos

- Pense em serviços:
 - No Exadata temos a possibilidade do RAC, serviços podem ser fundamentais;
 - Pense que você não irá disponibilizar bases de dados, mas sim “serviços” a serem consumidos;
- Vá além:
 - Em um ambiente de consolidação bases distintas podem ter o mesmo padrão de consumo:
 - Por exemplo, carga de dados;

Gerenciamento de Recursos

- No TJSC qualquer base de dados (ou esquema) antes de ser consolidado no Exadata responde um questionário com aproximadamente 30 questões:
 - Dividido em 6 tópicos:
 - Informações básicas;
 - Conexão;
 - Ambiente;
 - Consultas;
 - Rede;
 - Prioridade;
- Exemplo:
 - Quem é o responsável pelo serviço?
 - Este serviço receberá conexões externas?
 - Este serviço de conexão utiliza POOL para gerenciamento de conexões?
 - Qual o tipo de priorização ao acesso deste serviço? Response time ou Throughout?

Gerenciamento de Recursos

- Monte uma planilha com as informações do ambiente.
 - Visualização das relações entre as demandas fica mais fácil e acessível;
- Compare as respostas do questionário com as informações colhidas da base de dados:
 - O questionário não é a sua única fonte de de dados.
 - Seja amigo do AWR:
 - Manual querying AWR for trend analysis and capacity planning[Community Document ID 6725];
 - DBA_HIST_SYSMETRIC_SUMMARY;
 - DBA_HIST_SERVICE_STAT ;

Gerenciamento de Recursos

- SERVICES:
 - Fundamental para quem usa Oracle RAC e indispensável para o Exadata;
 - Mudança de paradigma para quem não está familiarizado:
 - Pense em serviços, o Exadata provê serviços para conexão e não bases de dados;
 - É a base para qualquer plano de recursos;
 - Com base no questionário, ou na análise dos padrões da base, você pode identificar aplicações que apresentam o mesmo comportamento e podem ser agrupadas;

Gerenciamento de Recursos

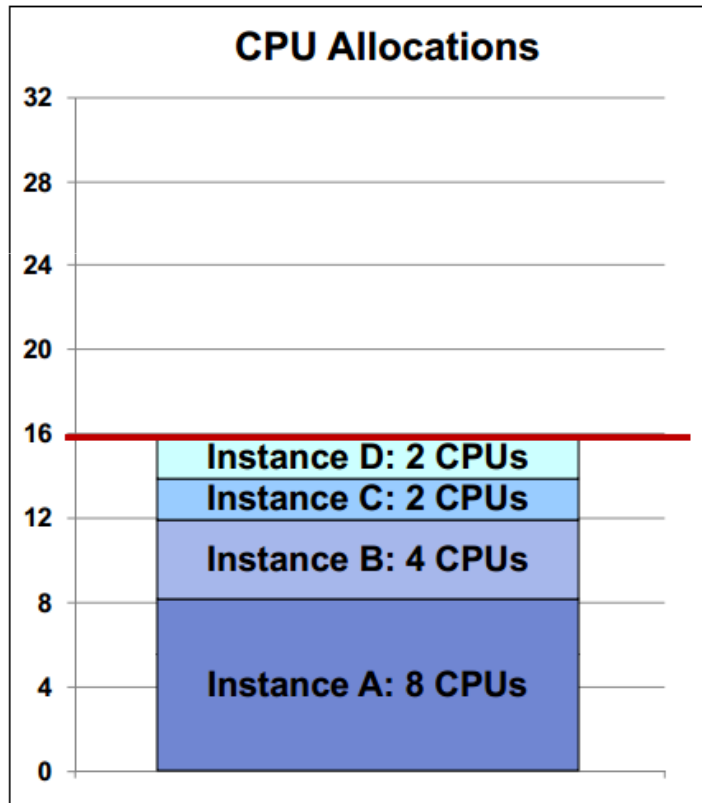
- Instance Caging:
 - Permite definir quantas CPU's podem ser utilizadas pela base de dados;
 - Em um ambiente com consolidação definir a quantidade de CPU disponível é interessante;
 - No Exadata temos 16(V2), 24 (X2) ou 32 (X3) processadores por DBNODE:
 - Em Cluster isso dá muito poder de processamento para uma única base de dados.
 - Contam-se as “threads” e não somente os “cores”;
 - Um detalhe, no Exadata temos economia de CPU, pois o storage fica responsável por parte da carga que o banco iria fazer:
 - CELLOFFLOAD;

Gerenciamento de Recursos

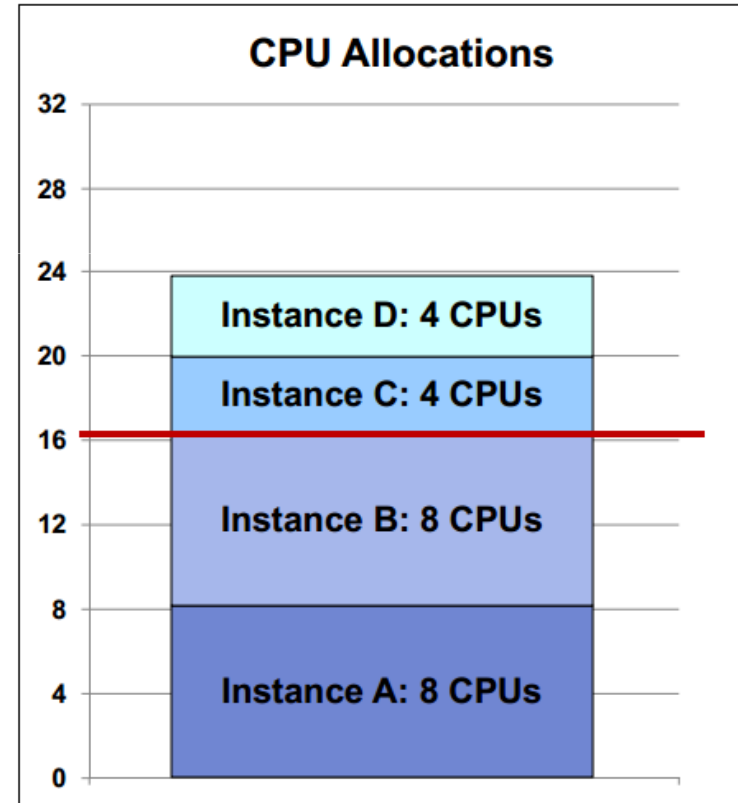
- Instance Caging:
 - Definido através do parâmetro CPU_COUNT.
 - Dois modos:
 - **Partitioning**: a quantidade definida em todas as bases não extrapola o número de CPU's do servidor;
 - **Over-Subscribe**: a quantidade definida é maior que o de CPU's por servidor.
 - Boa prática: A soma de todas as bases não deve ser maior que 3X o número de CPU's disponíveis por servidor;

Gerenciamento de Recursos

Partitioning



Over-Provisioning



Number of CPUs on Server

Gerenciamento de Recursos

- Instance Caging:
 - Deve ser avaliado antes de aplicado, o parâmetro CPU_COUNT é utilizado em diversos locais;
 - Paralelismo:
 - CPU_COUNT está na fórmula do DOP;
 - Qualquer mudança de CPU deve refletir no DOP:
 - Manual Oracle Database VLDB and Partitioning Guide (Capítulo 8);
 - Configuring and Monitoring Instance Caging (Doc ID 1362445.1);

Gerenciamento de Recursos

- Resource Manager:
 - Permite definir a porcentagem de CPU (%) que estará disponível;
 - Ativo através de um plano de recursos:
 - Os planos podem ser alterados e intercambiáveis a qualquer momento;
 - Planos baseados com grupos de consumidores (**CONSUMER GROUPS**);
 - Existem planos “simples” e “complexos”;
 - Podem ser utilizados em qualquer banco de dados de single instance até RAC e Exadata;
 - Para o Exadata são conhecidos como “**intradatabase plan**”;

Gerenciamento de Recursos

- Resource Manager:
 - Trabalha com planos divididos em 8 níveis onde são especificadas as porcentagens de CPU para cada consumer group.
 - Na divisão de recursos o nível 1 tem 100% garantido, o nível 2 recebe a sobra de recursos do nível 1 (o que não foi utilizado), o nível 3 a sobra do nível 2 e assim consequentemente.
 - *“Consumer groups at level 2 get resources that were not allocated at level 1 or were not consumed by a consumer group at level 1. Similarly, consumer groups at level 3 are allocated resources only when some allocation remains from levels 1 and 2... Any unused allocation goes to the next level, not to other consumer groups at the same level...”*
 - Os planos podem definir limites além de CPU, como UNDO, Paralelismo , quantidade de conexões. Mas estes não tem níveis;
 - Planos simples contêm somente consumer groups, planos complexos podem definir outros planos em seus níveis;
 - Os recursos são alocados por níveis, **o que não é utilizado é repassado ao próximo nível;**
 - A ligação do resource manager do banco com o controle de storage do Exadata é através de categorias;

Gerenciamento de Recursos

- Resource manager:
 - Após aplicar o plano o monitoramento pode ser feito através do Grid Control/Enterprise Manager;
 - GV\$RSRCMGRMETRIC/GV\$RSRCMGRMETRIC_HISTORY:
 - **CPU_CONSUMED_TIME**: Consumo de CPU em milissegundos;
 - **CPU_WAIT_TIME**: Tempo de Espera em por CPU em milissegundos;
 - **AVG_RUNNING_SESSIONS**: Número médio de sessões ativas no último minuto. Basicamente $CPU_CONSUMED_TIME/60000$.
 - **AVG_WAITING_SESSIONS**: Número médio de sessões que esperaram por CPU no último minuto. $CPU_WAIT_TIME/60000$.
 - **AVG_CPU_UTILIZATION**: Média do consumo de CPU. **Não é confiável**, usa a fórmula $CPU_CONSUMED_TIME/600/\#_CPU_Servidor$.
 - Basicamente não converte nada (pois divide milissegundos por 600) e não leva em consideração Instance Caging.
 - Mais garantido calcular na mão e ajustando os valores corretos, ou somar os gastos de CPU do nível e dividir pelo consumo do grupo no momento.
 - Usando os bugs **#17017939** e **#5118953** mais nota **1338988.1** podemos chegar na seguinte fórmula:
 - » $(CPU_CONSUMED_TIME/60000) / ((\%Garantida\ para\ CG/100)*CPU\ Disponível) * 100$
 - Não existe uma unidade para CPU disponível, a nota 1338988.1 fala em número de sessões concorrentes que podem executar utilizando 100% do processador;

Gerenciamento de Recursos

- Exemplo Plano Resource Manager;
- Enterprise Manager;

Gerenciamento de Recursos

- Resource Manager:
 - Cuidados:
 - Nunca deixar um grupo com 100% em um nível (desde que não seja o último).
 - Sempre terminar o plano com o consumer group OTHERS e criar um específico para os usuários SYS e SYSTEM (definido no plano como primeiro nível e com 100%);
 - Dar grant GRANT_SWITCH_CONSUMER_GROUP para permitir que a conexão seja mapeada para o CG;
 - Referências:
 - Master Note: Overview of Oracle Resource Manager and DBMS_RESOURCE_MANAGER (Doc ID 1484302.1);
 - Database Resource Manager samples (Doc ID 106948.1);
 - Bug 5118953 : RESOURCE MANAGER NOT WORKING AS DOCUMENTED;
 - CPU time calculation 'mismatch' in Oracle9 (Doc ID 215848.1);
 - Scripts and Tips for Monitoring CPU Resource Manager (Doc ID 1338988.1);
 - Oracle® Database Administrator's Guide 11g Release 2 (11.2) E25494-03 - Monitoring Oracle Database Resource Manager;
 - Oracle® Database Reference 11g Release 2 (11.2) E25513-04 - V\$RSRC_CONSUMER_GROUP e V\$RSRCMGRMETRIC;
 - Se disponível, Oracle® Exadata Storage Server Software - E13861;

Gerenciamento de Recursos

- IORM:
 - Resource manager do Exadata, definido diretamente em cada storage node. **I/O Resource Manager (IORM)** e chamado também de “**interdatabase plan**”;
 - Permite definir, da mesma forma que o Database Resource Manager, a porcentagem de I/O que pode ser utilizada
 - É especificado através de um plano de recursos, chamado de **IORMPLAN** dividido em até 8 níveis;
 - Por ser no storage, não leva em consideração somente um único banco, pode utilizar categorias também;
 - Quando ativo, marca cada I/O recebido e distribui aquele que será respondido através das prioridades definidas no plano.
 - Basicamente cria filas para o I/O e responde de forma mais rápida aqueles com mais prioridade;

Gerenciamento de Recursos

- IORM:
 - **Catplan:**
 - O plano não está preso ao banco de origem do I/O, analisa cada um através de sua categoria;
 - Isso permite que duas conexões de um mesmo usuário de um banco tenham prioridades diferentes;
 - Indo mais além, permite que I/O's oriundos de bancos diferentes sejam tratadas iguais. Por exemplo dois bancos com processos de BATCH seriam tratados com a mesma prioridade;
 - Tem prioridade sobre o DBplan;
 - **DBplan:**
 - O plano leva em consideração somente o banco de origem do I/O;
 - Quando utilizado com Catplan, ele define qual o banco que terá mais prioridade caso tenham a mesma categoria;
 - Permite definir se um banco usará ou não a FLASH;
 - Integrado com o DataGuard, o banco especificado somente será utilizado pelo plano quando for standby ou primary;

Gerenciamento de Recursos

- IORM:
 - Expande em muito o controle que o DBA tem;
 - Por exemplo, em um ambiente tradicional o que você faria se estivesse com problema de I/O, onde dois bancos concorrem pelo mesmo disco/raid group no Storage?
 - Mudaria de disco?
 - E se fosse o mesmo banco?
 - Mudaria a tabela de tablespace?
 - E se fossem duas aplicações operando sobre a mesma tabela?
 - Outro exemplo, se você tivessem em um mesmo Exadata o ambiente de Produção, Homologação e Desenvolvimento?
 - O loop do estagiário iria matar a sua base de produção?
 - Com IORM é possível gerenciar tudo sem indisponibilidade do ambiente;

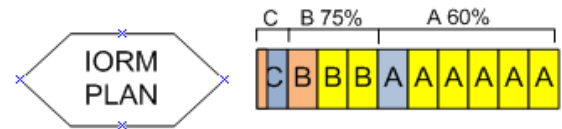
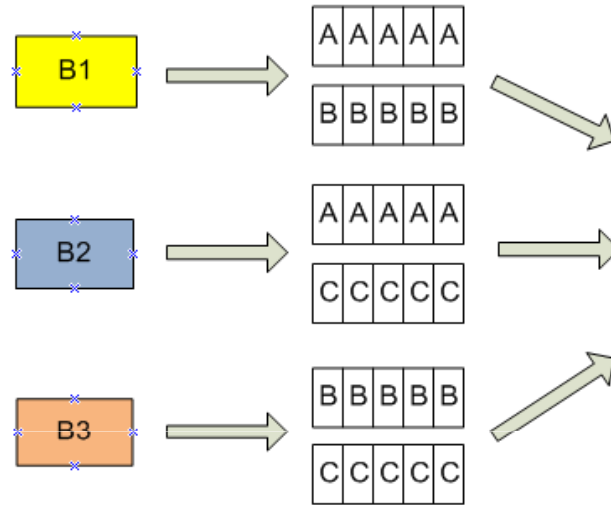
Gerenciamento de Recursos

- IORM:
 - Ao ativado, cada I/O passa a ganhar uma marca (“selo”) e o plano é aplicado. Este “selo” pode vir do banco de dados através da definição de categorias;
 - Categorias somente podem ser criadas através de PL/SQL;
 - Bancos distintos podem ter as mesmas categorias
 - No TJSC temos um aplicativo que utiliza 4 bancos distintos, e em cada um destes existem categorias comuns;
 - As categorias são associadas aos consumer groups através da `DBMS_RESOURCE_MANAGER.CREATE_CATEGORY`;
 - Na tabela **`dba_rsrc_consumer_groups`** existe uma coluna chamada `category` que mostra a categoria de cada consumer group;

ALTER IORMPLAN

```
catPlan=(
  (name=OLTP_ALTA_PRIORIDADE      , level=1, allocation=40),
  (name=DSS_ALTA_PRIORIDADE       , level=1, allocation=30),
  (name=OLTP_MEDIAALTA_PRIORIDADE , level=2, allocation=40),
  (name=DSS_MEDIA_PRIORIDADE      , level=2, allocation=20),
  (name=OLTP_MEDIA_PRIORIDADE     , level=3, allocation=70),
  (name=OLTP_BAIXA_PRIORIDADE     , level=4, allocation=40),
  (name=DSS_BAIXA_PRIORIDADE      , level=4, allocation=20),
  (name=OTHER                      , level=5, allocation=100)
),
dbPlan=(
  (name=DB1           , level=1, allocation=60),
  (name=DB2           , level=2, allocation=35),
  (name=DB3           , level=2, allocation=30),
  (name=DB4           , level=3, allocation=40),
  (name=DB5           , level=4, allocation=40),
  (name=OTHER         , level=5, allocation=100, flashCache=OFF)
),
objective=low_latency
```

- Neste exemplo, todo o I/O que oriundo da categoria OLTP_ALTA_PRIORIDADE receberia 40% do I/O do storage, e estes 40% seria divididos em:
 - 60% para o DB1 caso ele tivesse a categoria, o que ele não utilizaria (como está no nível 1) seria passado para o nível 2
 - Para o nível 2 no DBPLAN teríamos 40% que não foi alocado no primeiro nível mais a porcentagem não usada pelo nível 1(neste caso por exemplo 70% não foi usado). Isso daria $40\% + 42\% (60\% * 70\% / 100) = 82\%$ de I/O para o segundo nível.
 - Assim 35% destes 82% seriam alocados para o DB2 e 30% para o DB3, o que eles não utilizassem seria passado para o nível 3 e assim sucessivamente.
 - Por fim, tudo o que não for destas categorias seria classificado como OTHER, e qualquer banco que não os especificados não estaria na flash. Assim aquele banco de teste nunca estaria roubando espaço dos bancos de produção na FLASHCACHE.
 - E tudo isso para uma única categoria. É complexo, mas dá uma segurança e horas a mais de sono ☺



```

IORMPLAN =
  catPlan=( (name=A, level=1, allocation=60),
            (name=B, level=2, allocation=75),
            (name=C, level=2, allocation=25)
            ),
  dbPlan=( (name=B1, level=1, allocation=83),
            (name=B2, level=2, allocation=66),
            (name=B3, level=3, allocation=100)
            )
  )
    
```

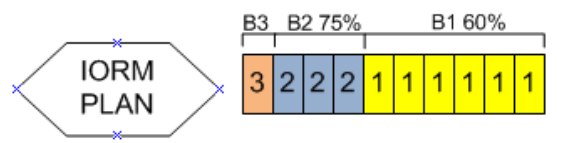
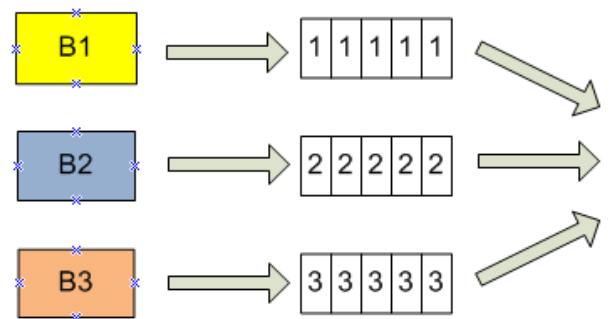
Exemplo Categorias:
 Slots = 10
 Primeiro Nível: 10 Slots
 A = 60% = 6 Slots

Segundo Nível: 4 Slots
 B = 75% = 3 Slots
 C = 25% = 1 Slot

Exemplo Bancos:
 Categoria A: 6 Slots
 B1 = 83% = 5Slots
 B2 = 66% = - de 1 Slot
 B3 = 100% = Não Usou

Categoria B: 3 Slots
 B1 = 83% = +-2,5 Slots
 B2 = 66% = Não Usou
 B3 = 100% = 1 Slot

Categoria C: 1 Slot
 B1 = 83% = Não Usou
 B2 = 66% = +-1
 B3 = 100% = Em espera



```

IORMPLAN =
  dbPlan=( (name=B1, level=1, allocation=60),
            (name=B2, level=2, allocation=75),
            (name=B3, level=3, allocation=100)
            )
  )
    
```

Exemplo Bancos:
 Slots = 10
 Primeiro Nível: 10 Slots
 B1 = 60% = 6 Slots

Segundo Nível: 4 Slots
 B = 75% = 3 Slots

Terceiro Nível: 1 Slot
 B3 = 100% = 1Slot

Gerenciamento de Recursos

- Controlando o ambiente:
 - O primeiro passo é fazer um levantamento das demandas do ambiente/aplicativo/banco de dados;
 - Com base nas respostas termos as seguintes ações:
 - Criação de services para cada fim identificados no levantamento/questionário.
 - Por exemplo, um para aplicações desktop e outro para WEB;
 - Definir as demandas de CPU e definir o instace caging;
 - Criar um plano recursos através do Resource Manager do banco de Dados;
 - Criar as categorias e associar aos consumer groups;
 - Criar o plano do IORM:
 - Pode ser somente para bancos, ou categorias;
 - E por fim...

Análise do Ambiente

- Monitoramento:
 - Não existe gerenciamento de recursos sem constante acompanhamento do ambiente.
 - *Com grandes poderes vêm grandes responsabilidades;*
 - Para monitoramento do Exadata temos:
 - Enterprise Manager;
 - SQL/Linha de comando;
 - Para o Enterprise Manager a recomendação é o uso da versão 12C;

Análise do Ambiente

The screenshot displays the Oracle Enterprise Manager Cloud Control 12c interface. The main view is titled "DB Machine" and shows an overview of the database machine's components. The overview includes:

- Racks: 1
- Incidents: 6 (5 critical, 1 warning), 24 errors, 1 warning, 0 flags
- Compute Nodes: 4 (all up)
- Ethernet Switches: 1 (all up)
- KVM: 1 (all up)
- PDU: 1 (all down)
- Exadata Cells: 7 (all up)
- IB Switches: 2 (all up)

The "Database Machine Schematic" section shows a rack labeled "Rack 1" with a list of components and their status:

Component	Status
exasw-ib3	Up
exasw-kvm	Down
exasw-ip	Up
exasw-ib2	Up
exadb04	Up
exadb03	Up
exadb02	Up
exadb01	Up
exacel07	Up
exacel06	Up
exacel05	Up
exacel04	Up
exacel03	Up
exacel02	Up
exacel01	Up
exasw-ib1	Down

A legend defines the status indicators:

- Green circle: Up
- Red square: Down
- Black square: Blackout
- Dark blue square: Exadata Cell
- Light blue square: Compute Node
- Grey square: Infiniband Switch
- White square: Ethernet Switch
- Light grey square: Keyboard-Video-Mouse
- Dark grey square: Unallocated

Vertical text on the right side of the schematic reads "EXTREME PERFORMANCE".

Análise do Ambiente

Oracle Enterprise Manager Cloud Control 12c

Enterprise | Targets | Favorites | History | Search Target Name

Exadata Grid

Exadata Storage Server Grid

Page Refreshed Jul 29, 2013 5:23:53 PM BRT

Performance

Average IO Load

Metric	Value
Flash Disk IO Load	~1.2
Hard Disk IO Load	~3.8

Average CPU Utilization

Metric	Value (%)
CPU Utilization	~10

Total Network Utilization

Metric	Value (KB/sec)
Total Received	~0.8
Total Sent	~0.4

Average Response Time

Metric	Value (ms)
Hard Disk Read	~3.8
Hard Disk Write	~0.5
Flash Disk Read	~0.5
Flash Disk Write	~0.5

ASM Diskgroup Summary

ASM	Disk Group	Size (GB)	No. of Griddisks	Database Names
+ASM_exa-cluster	DATA	25200	84	ODJE,OIND,ONET,OPG5,OSTS,ADVSC,OTJSC,OUNIC,OINDTJ,OJURIS,ONETTJ,OSG5TJ,OSGETJ,OTJSCDW
+ASM_exa-cluster	RECO	19209	84	DBFS,ODJE,OIND,ONET,OPG5,OSTS,ADVSC,OTJSC,OUNIC,OINDTJ,OJURIS,ONETTJ,OSG5TJ,OSGETJ,OTJSCDW
+ASM_exa-cluster	SYSTEMDG	2038	70	

Incidents

View: Target | Local target and related targets | Category: All | 0 | 0 | 0 | 0

Summary	Target	Severity	Status	Escalation level	Type	Time since last update
No matching incidents or problems found.						

Columns Hidden: 13 | Updated in last 31 days

Análise do Ambiente

ORACLE Enterprise Manager Cloud Control 12c

Enterprise ▾ Targets ▾ Favorites ▾ History ▾ Search Target Name ▾

Target Navigation

- DB Machine
 - Compute Nodes
 - Exadata Grid**
 - exacel01
 - exacel02
 - exacel03
 - exacel04
 - exacel05
 - exacel06
 - exacel07

Exadata Grid Exadata Storage Server Grid

Page Refreshed Jul 29, 2013 5:28:40 PM BRT

I/O Resource Manager (IORM) Settings:

I/O Resource Manager controls how databases utilize the disks and flash cache, based on the settings specified here. [Update All](#)

Status: Active Inactive

Disk I/O Objective: Common objective is not set on all cells. Select

Target	Status
exacel01	Active
exacel02	Active
exacel03	Active
exacel04	Active
exacel05	Active
exacel06	Active
exacel07	Active

Inter-Database Plan: Inter-database plans are different among the Exadata Storage Servers

Tip: Specify the plan below to configure it for all Exadata Storage Servers.

Display: [+ Add](#) [- Remove](#)

Database Name	Disk I/O Utilization Limit(%)	Disk I/O Allocation(%)								Use Flash Cache	Use Flash Log
		Level 1	Level 2	Level 3	Level 4	Level 5	Level 6	Level 7	Level 8		

Tip: Oracle recommends the same IORM settings for the group of cells used by the same set of databases. Click on Update All button to change the IORM settings for all cells in this Group.

03:30 PM - 05:30 PM All Year Month Week Day **2 hours** 15 minutes [Slider](#)

Select Database:

Average Throttle Time for Disk I/Os: All Databases

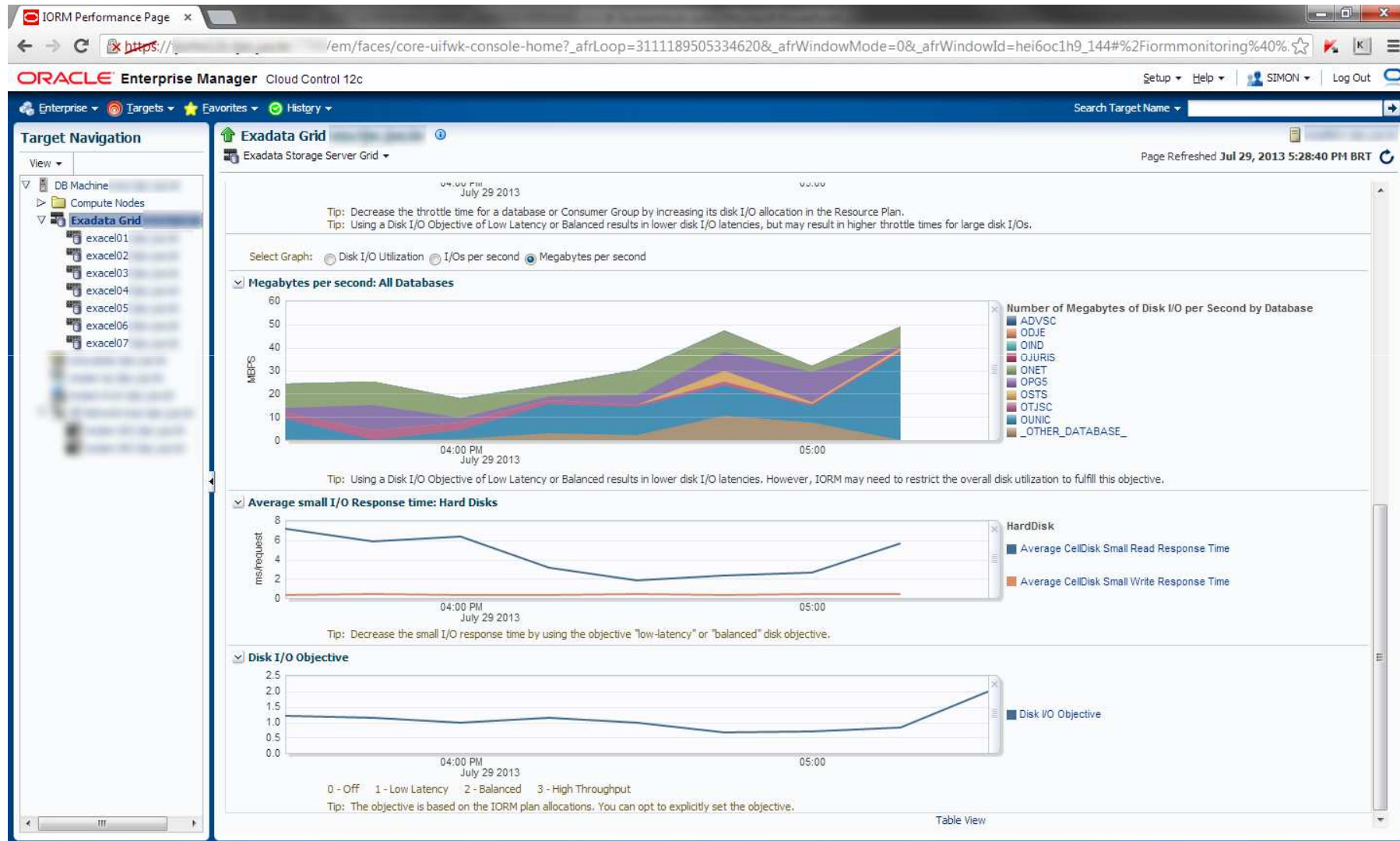
Average Throttle Time per Disk I/O by Database

- ADVSC
- ODJE
- ONID
- OJURIS
- ONET
- OPGS
- OSTS
- OTJSC
- OUNIC
- _OTHER_DATABASE_

Y-axis: Milliseconds (0 to 30)

X-axis: 04:00 PM to 05:00

Análise do Ambiente



Análise do Ambiente

- Banco de dados:
 - AWR;
 - Grid Control/Enterprise Manager;
 - DBA_HIST*
 - DBA_HIST_SYSMETRIC_SUMMARY;
 - DBA_HIST_SERVICE_STAT;
 - V\$SESSTAT;
 - V\$SYSSTAT;
 - V\$RSRCMGRMETRIC;
 - V\$RSRCMGRMETRIC_HISTORY;

Análise do Ambiente

- CELLCLI:
 - Não tenha medo da ferramenta, ela é muito poderosa e permite um acompanhamento fino do ambiente;
 - Para IORM é fundamental o seu uso:
 - Planos que utilizem categorias não são visíveis no Enterprise Manager
 - Para análise de recursos o comando é LIST METRICCURRENT;

Análise do Ambiente

- CELLCLI - Métricas:
 - São divididas em dois grupos:
 - SMALL e LARGE
 - Todo e qualquer I/O maior que 128KB é considerado large;
 - Tipos de métrica:
 - Cumulative;
 - Instantaneous;
 - Rate;

Análise do Ambiente

- CELLCLI – Métricas:
 - Métricas importantes:
 - **CT_FC_IO_BY_SEC/CG_FC_IO_BY_SEC/DB_FC_IO_BY_SEC**: MB/s processados exclusivamente pelo flash;
 - **CT_FC_IO_RQ_SEC/CG_FC_IO_RQ_SEC/DB_FC_IO_RQ_SEC**: requisições por segundo processados pela flash;
 - **CT_IO_BY_SEC/CG_IO_BY_SEC/DB_IO_BY_SEC**: MB/s processados pelos discos;
 - **CT_IO_WT_*/CG_IO_WT_*/DB_IO_WT_***: Quantidade de espera para processar o I/O. Apesar de ser cumulativo, valores altos indicam que o IORM está atuando e talvez a alocação esteja baixa;

Análise do Ambiente

- Exemplo métricas IORM;

Análise do Ambiente

- Economia do Exadata;

- Utilizando o conjunto de services no RAC, instance caging, resource manager e IORM temos o controle total do Exadata;
- Podemos controlar desde a entrada da conexão (services), o cpu utilizado (instance caging e resource manager) além do I/O (IORM);
- Sendo todo o fluxo passível de controle, análise e manipulação:
 - Você tem idéia do que isso significa?
 - Você tem idéia do poder que o DBA tem em mãos?

Fernando Simon

fernando.simon.br@gmail.com

<http://www.fernandosimon.com/blog>

- Dúvidas?