



Oracle Performance Diagnostics & Tuning
Ricardo Portilho Proni

Mitificação



Métodos antigos

- Experiência
- Intuição
- Imprecisão
- Tempo
- Sorte
- Recursos

Top Tuning

- Verificar maior consumidor de CPU
- Verificar o SQL agressor
- Alterar o SQL e esperar que o desempenho melhore
- Adicionar índices e esperar que o desempenho melhore
- Se não melhorar, matar a sessão
- Se o desempenho não melhorar, voltar ao início

Checklist Tuning

- Verificar Sistema Operacional
- Verificar SGA e PGA
- Verificar coleta de estatísticas e fragmentação
- Verificar parâmetros do Oracle
- Verificar LOCKs
- Verificar SQLs que consomem mais recursos
- ...
- Construir uma teoria baseada nos dados observados
- Alterar algo e esperar que o desempenho melhore
- Se o cliente não gostar da teoria, apenas cite e altere alguns parâmetros relacionados
- Se o desempenho não melhorar, voltar ao início

Ratios Tuning

- Verificar Buffer Cache Hit Ratio
- Verificar Data Dictionary Hit Ratio
- Verificar SQL Cache Hit Ratio
- Verificar Library Cache Hit Ratio
- ...
- Construir uma teoria baseada nos dados observados
- Alterar algo (geralmente aumentar) e esperar que o desempenho melhore
- Se o desempenho não melhorar, voltar ao início

KIWI Tuning

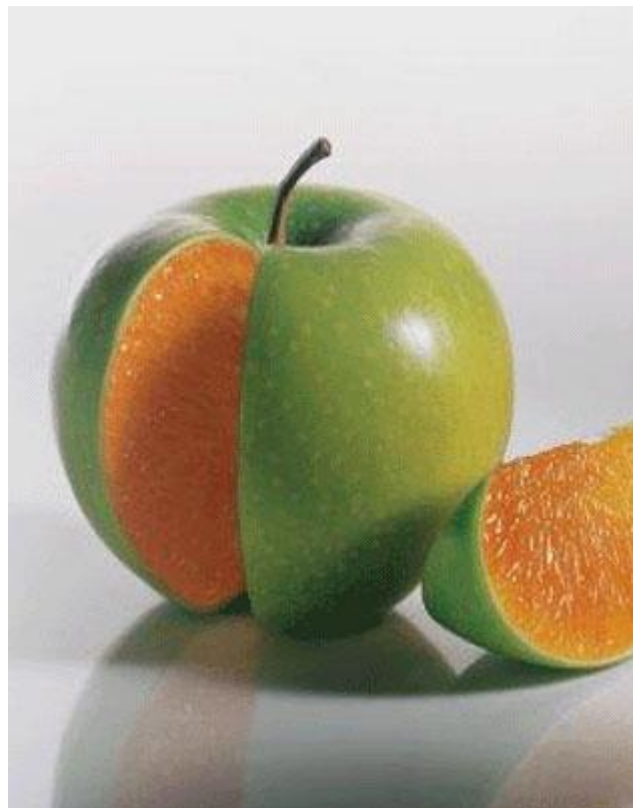
- KIWI = Kill It With Iron
- Adicionar Memória RAM
- Adicionar CPUs
- Melhorar o I/O
- Migrar para um Servidor maior
- Migrar para RAC
- Adicionar Nós no RAC
- ...
- Pagar a conta, e esperar que o desempenho melhore
- Se o desempenho não melhorar, voltar ao início

Manager Tuning

- Migrar Banco para outro servidor
- Executar Upgrade de Banco de Dados
- Executar Upgrade da Aplicação
- Executar Upgrade do Application Server
- Juntar Aplicação e Banco de Dados
- Separar Aplicação e Banco de Dados
- Mudar o Application Server
- Voltar Backups
- ...
- Se o desempenho não melhorar, tentar outra coisa, até melhorar

O que está errado?

Paradigma



O carro e o leite

- Seu filho leva 2 horas para comprar leite na padaria, de carro.

Como melhorar este tempo?

- É necessário um carro mais rápido?
- São necessários dois carros?
- É necessário tornar a estrada mais larga?
- É melhor só comprar 1 litro de leite de cada vez?
- Deve-se utilizar uma padaria que só tenha 1 tipo de leite?
- A porta da garagem deve estar sempre aberta?

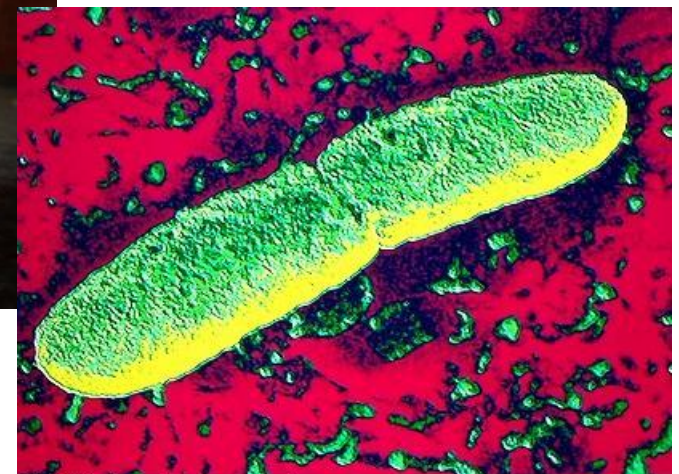
O chefe e o atraso



Resultados

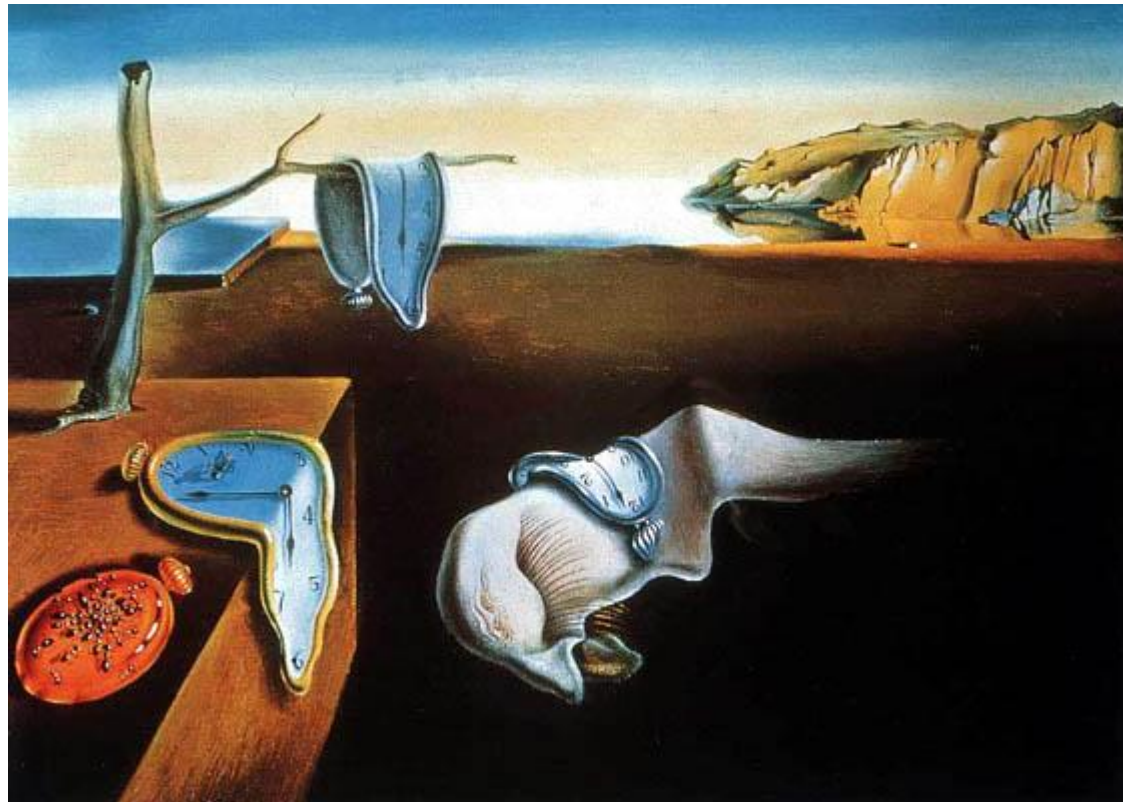


O método correto



© 1997 The Learning Company, Inc.

O Tempo



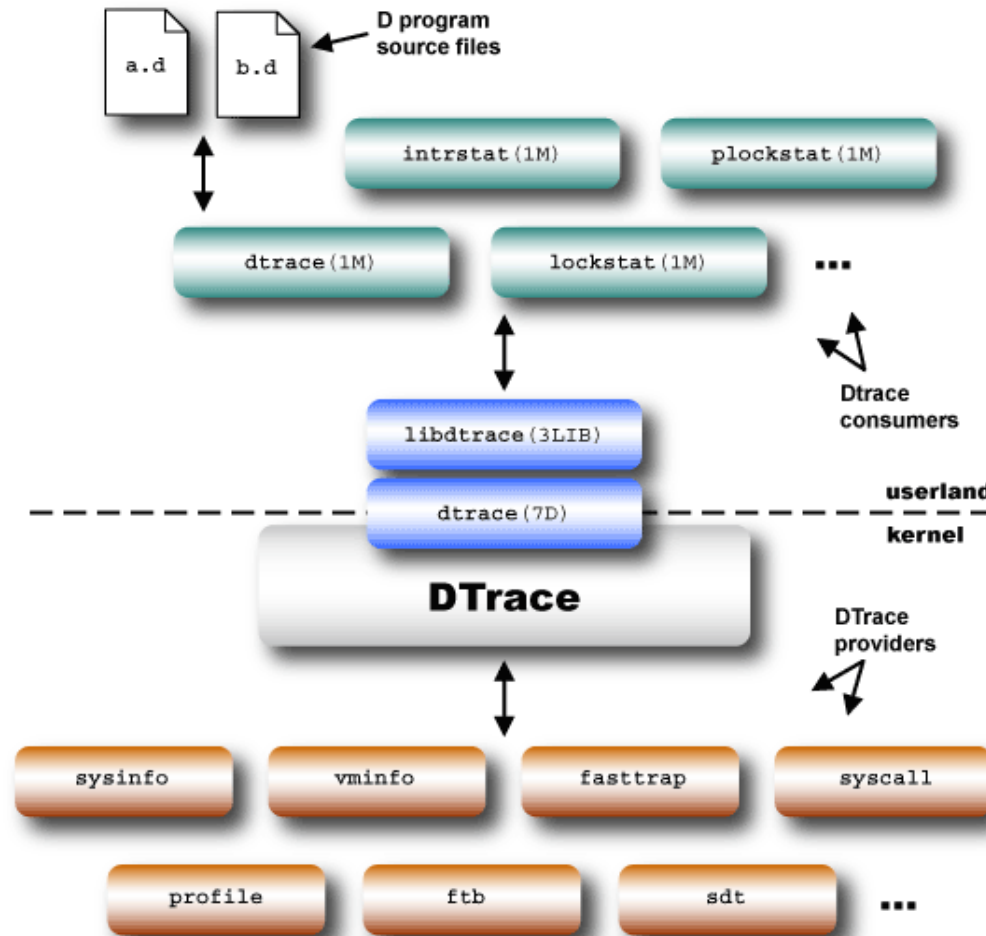
$$R = S + W$$

Response Time = Service Time + Wait Time

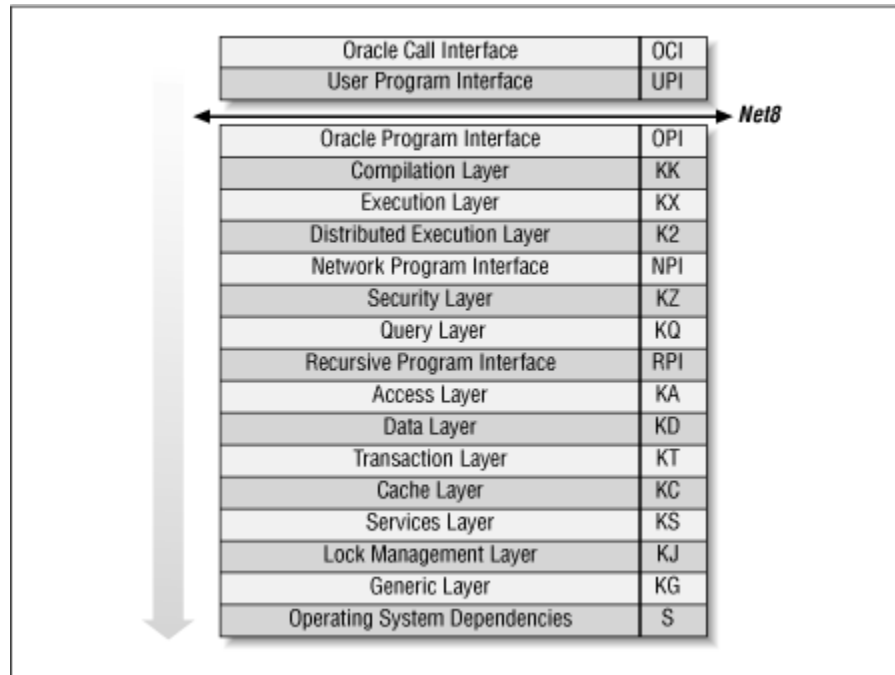
Instrumentação



Instrumentação



Oracle Wait Interface



Nascimento da OWI

- Benchmark 7.0.12: Juan Loaiza
- YAPP Paper: Anjo Kolk



Evolução da OWI

- Versão 7.0.12: 104 Wait Events
- Versão 8: 140 Wait Events
- Versão 8i: 220 Wait Events
- Versão 9i: 400 Waits Events
- Versão 10gR1: >800 Wait Events
- Versão 11gR2: >1100 Wait Events

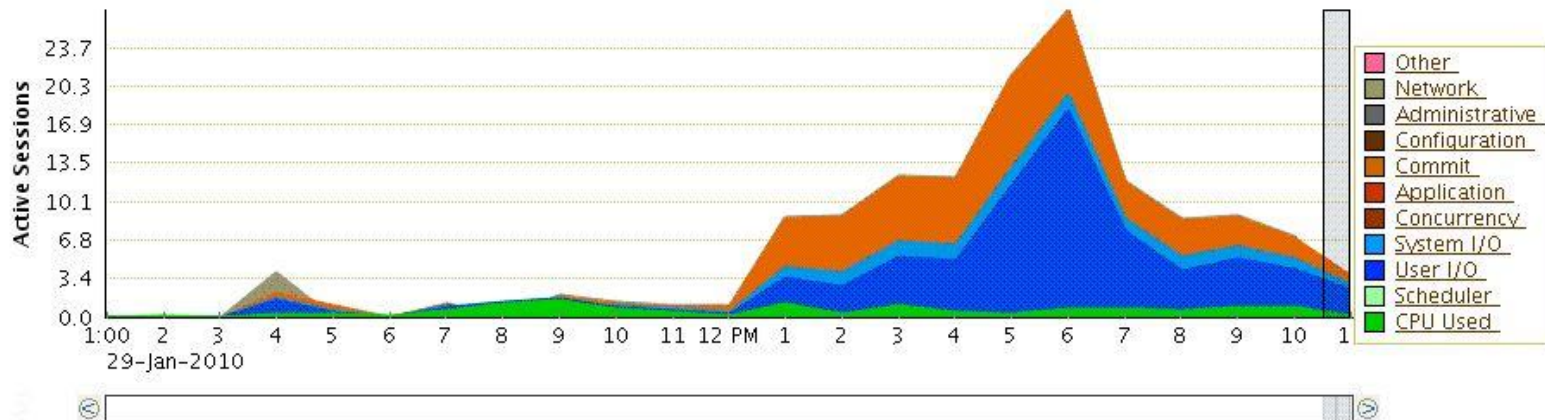
Wait Events mais comuns

- buffer busy
- control file parallel write
- db file parallel read
- db file parallel write
- db file scatterread read
- db file sequential read
- db file single write
- direct path read
- direct path write
- enqueue
- free buffer
- latch free
- library cache pin
- library cache lock
- log buffer space
- log file sequential read
- log file switch (archiving needed)
- log file switch (checkpoint incomplete)
- log file switch completion
- log file sync
- SQL*Net message from client
- SQL*Net message to client

Enterprise Manager

Top Activity

Click on the band below the chart to change the time period for the detail section below.



Detail for Selected 30 Minute Interval

Start Time 29-Jan-2010 22:30:37 o'clock GMT

Exemplo de Wait Event

db file scattered read

P1: File number

P2: Starting block

P3: Number of blocks

Explicação: durante *Full Table Scan*, a leitura do *Datafile* espera pelo mecanismo de I/O.

Causas: DB_CACHE_SIZE insuficiente, *Full Table Scan* incorreto ou desnecessário, I/O ineficiente (RAID 5?).

Correção:

- Aumente o DB_CACHE_SIZE;
- Elimine o *Full Table Scan*;
- Otimize o DB_FILE_MULTIBLOCK_READ_COUNT para seu I/O;
- Melhore o mecanismo de I/O.

Exemplo de Wait Event

log file switch completion

P1: Not used

P2: Not used

P3: Not used

Explicação: todos os grupos de *Redo Logs* foram utilizados e ainda são necessários para um eventual *Recover*, pois o DBWR ainda não gravou os blocos alterados nos *Datafiles*.

Causas: *Redo Logs* sub-dimensionados para a gravação corrente, DB_CACHE_SIZE insuficiente, I/O ineficiente.

Correção:

- Aumentar os *Redo Logs* em quantidade e/ou tamanho;
- Aumente o DB_CACHE_SIZE;
- Melhore o mecanismo de I/O.

Granularidades de Análise

- SQL Statement
- Session
- Instance

Ferramentas de Análise

- Dynamic Performance Views
- Extended SQL Trace (Event 10046)
- Statspack / AWR

OWI Dynamic Performance Views

- V\$SYSTEM_EVENT
- V\$SESSION_EVENT
- V\$SESSION_WAIT

Cenários de Análise

- O Banco de Dados está lento agora
- O Banco de Dados estava lento ontem
- Este SQL está lento

O Banco de Dados está lento

- Encontrar indícios do gargalo na V\$SYSTEM_EVENT
- Verificar os maiores gargalos na V\$SESSION_EVENT
- Verificar os maiores gargalos na V\$SESSION_WAIT, repetidamente
- Encontrar o SID ofensor na V\$SESSION_WAIT
- Encontrar o maior Wait Event deste SID na V\$SESSION_EVENT
- Corrigir o maior Wait Event possível
- Se o tempo esta satisfatório, finalizar a análise

O Banco de Dados está lento

```
SQL> CREATE TABLE T2 AS SELECT * FROM T;
```

Table created.

```
SQL> SELECT EVENT, TIME_WAITED FROM V$SESSION_EVENT WHERE SID = 184 ORDER
      BY TIME_WAITED DESC;
```

EVENT	TIME_WAITED
SQL*Net message from client	96602
direct path read	8403
log buffer space	7337
direct path write	2698
log file switch (checkpoint incomplete)	1037
log file switch completion	396
buffer busy waits	346
log file switch (private strand flush incomplete)	218
Data file init write	61
...	

O Banco de Dados estava lento

- Encontrar indícios do gargalo na V\$SYSTEM_EVENT
- Encontrar o maior Wait Event via Statspack / AWR
- Corrigir o maior Wait Event possível
- Se o tempo esta satisfatório, finalizar a análise

Este SQL está lento

- Executar o comando SQL com Extended SQL Trace
- Encontrar o maior Wait Event via tkprof
- Corrigir o maior Wait Event possível
- Se o tempo esta satisfatório, finalizar a análise

Este SQL está lento

*** 2010-03-22 11:43:12.276

```

WAIT #9: nam='db file scattered read' ela= 183330 file#=4 block#=9124 blocks=26 obj#=74574
WAIT #9: nam='db file scattered read' ela= 2528 file#=4 block#=9150 blocks=26 obj#=74574
WAIT #9: nam='db file scattered read' ela= 170358 file#=4 block#=9176 blocks=26 obj#=74574
WAIT #9: nam='db file scattered read' ela= 96261 file#=4 block#=9202 blocks=26 obj#=74574
WAIT #9: nam='db file scattered read' ela= 1669 file#=4 block#=9228 blocks=26 obj#=74574
WAIT #9: nam='db file scattered read' ela= 26055 file#=4 block#=9254 blocks=26 obj#=74574
WAIT #9: nam='db file scattered read' ela= 4760 file#=4 block#=9280 blocks=26 obj#=74574
WAIT #9: nam='db file scattered read' ela= 108783 file#=4 block#=9306 blocks=26 obj#=74574
tim=1269268992840594

```

=====

Limitações OWI: Views

- Não é um monitoramento End-to-End
- Sem dados de consumo de CPU
- Sem dados de consumo de Memória
- Sem histórico
- Bugs
- Imprecisões

Limitações OWI: Extended Trace

- Muitos dados
- Altíssima granularidade
- Desempenho
- Correlação de informações
- Bugs
- Imprecisões
- Sessões PARALLEL
- Sessões SHARED SERVER
- Waits só disponíveis em $\geq 9iR1$
- Suporte oficial só em $>10gR1$

Limitações OWI: Statspack / AWR

- Baixa granularidade

Minha abordagem em Tuning

- Performance de Sistemas Computacionais só pode ser medida em TEMPO.
- Performance Tuning deve ser reativa.
- Performance Tuning deve ter ROI.
- Apenas os maiores gargalos devem ser solucionados.
- O processo deve ser Diagnostics, e depois Tuning.
- Alto consumo de CPU não é um problema.
- O usuário não executa um SQL por prazer.
- O desenvolvedor não deveria saber como fazer um bom SQL (COBOL?).
- Ferramentas Gráficas / Enterprise Manager / Wizards / Automação são bons auxiliares.

Minha abordagem em Tuning

- Bancos com bom desempenho devem ser observados.
- Não acredite em nada (separar tabelas e índices?). Teste.
- Se houvesse um parâmetro que sempre deixasse o Oracle mais rápido, sem nenhum efeito colateral, ele já viria habilitado.
- Desenvolva um método de convencimento gerencial.
- Por algo chamar-se Storage, não quer dizer que ele não tenha problemas.
- KISS (Keep It Simple, Stupid): a probabilidade de falha cresce linearmente com o aumento de complexidade.
- Conheça outros RDBMSs: TI não é lugar para paixões.



Perguntas?

Blog: <http://nervinformatica.com.br>

Email: ricardo@nervinformatica.com.br

Twitter: <http://twitter.com/rportilhoproni>