

# Documentos

Dezembro, 2003 35

ISSN 1677-9274

## Projeto Físico de Data Warehouse no Oracle: um Estudo de Caso



República Federativa do Brasil

*Luiz Inácio Lula da Silva*

Presidente

Ministério da Agricultura, Pecuária e Abastecimento

*Roberto Rodrigues*

Ministro

Empresa Brasileira de Pesquisa Agropecuária – Embrapa

Conselho de Administração

*José Amauri Dimázio*

Presidente

*Clayton Campanhola*

Vice-Presidente

*Alexandre Kalil Pires*

*Dietrich Gerhard Quast*

*Sérgio Fausto*

*Urbano Campos Ribeiral*

Membros

Diretoria Executiva da Embrapa

*Clayton Campanhola*

Diretor-Presidente

*Gustavo Kauark Chianca*

*Herbert Cavalcante de Lima*

*Mariza Marilena T. Luz Barbosa*

Diretores-Executivos

Embrapa Informática Agropecuária

José Gilberto Jardine

Chefe-Geral

*Tércia Zavaglia Torres*

Chefe-Adjunto de Administração

*Sônia Ternes Frassetto*

Chefe-Adjunto de Pesquisa e Desenvolvimento

*Álvaro Seixas Neto*

Supervisor da Área de Comunicação e Negócios



*Empresa Brasileira de Pesquisa Agropecuária  
Embrapa Informática Agropecuária  
Ministério da Agricultura, Pecuária e Abastecimento*

*ISSN 1677-9274  
Dezembro, 2003*

# Documentos 35

## Projeto Físico de Data Warehouse no Oracle: um Estudo de Caso

Carlos Alberto Alves Meira

Campinas, SP  
2003

Embrapa Informática Agropecuária  
Área de Comunicação e Negócios (ACN)  
Av. André Tosello, 209  
Cidade Universitária "Zeferino Vaz" Barão Geraldo  
Caixa Postal 6041  
13083-970 Campinas, SP  
Telefone (19) 3789-5743 Fax (19) 3289-9594  
URL: <http://www.cnptia.embrapa.br>  
e-mail: [sac@cnptia.embrapa.br](mailto:sac@cnptia.embrapa.br)

Comitê de Publicações

*Carla Geovana Nascimento Macário*  
*José Ruy Porto de Carvalho*  
*Luciana Alvim Santos Romani (Presidente)*  
*Marcia Izabel Fugisawa Souza*  
*Marcos Lordello Chaim*  
*Suzilei Almeida Carneiro*

Suplentes

*Carlos Alberto Alves Meira*  
*Eduardo Delgado Assad*  
*Maria Angelica de Andrade Leite*  
*Maria Fernanda Moura*  
*Maria Goretti Gurgel Praxedis*

Supervisor editorial: *Ivanilde Dispatto*  
Normalização bibliográfica: *Maria Goretti Gurgel Praxedis*  
Editoração eletrônica: *Área de Comunicação e Negócios (ACN)*

1ª edição on-line - 2003  
Todos os direitos reservados.

---

Meira, Carlos Alberto Alves.

Projeto físico de Data Warehouse no Oracle: um estudo de caso / Carlos Alberto Alves Meira. — Campinas: Embrapa Informática Agropecuária, 2003.  
18 p.: il. — (Documentos / Embrapa Informática Agropecuária; 35)

ISSN 1677-9274

1. Data Warehouse. 2. Banco de dados. 3. Armazém de dados. I. Título. II. Série.

CDD 005.74 (21<sup>st</sup> ed.)

# Autor

Carlos Alberto Alves Meira  
M.Sc. em Ciência da Computação e  
Matemática Computacional,  
Pesquisador da Embrapa Informática  
Agropecuária, Caixa Postal 6041,  
Barão Geraldo  
13083-970 - Campinas, SP  
Telefone (19) 3789-5806  
e-mail: carlos@cnptia.embrapa.br



# Apresentação

Data warehouse (DW) ou, traduzindo, armazém de dados, é um banco de dados com características que o diferenciam dos bancos de dados convencionais projetados para gerenciamento de transações. Um DW é construído geralmente para suporte a decisões, bastante útil quando as decisões devem estar baseadas em análises de grandes quantidades de dados. É requerido para os níveis tático e estratégico, enquanto suas fontes de dados executam, na maioria das vezes, no nível operacional.

Tanto o projeto lógico quanto o projeto físico de banco de dados devem ser elaborados de acordo com as características próprias de um DW. Os aspectos relacionados com o projeto lógico são facilmente encontrados na literatura sobre o assunto. Os aspectos de projeto físico, por outro lado, são melhor exemplificados por relatos de experiência e, muitas vezes, são dependentes do sistema gerenciador de banco de dados (SGBD).

Este trabalho compartilha a experiência no projeto físico do Armazém de Dados da Fruticultura (ADF), um DW desenvolvido pela Embrapa Informática Agropecuária utilizando o SGBD Oracle. São apresentadas características do Oracle que favorecem a manutenção e o desempenho de consultas a um DW, utilizando como exemplo o ADF.

O ADF é um produto desenvolvido para o Ministério da Agricultura, Pecuária e Abastecimento (MAPA) como parte de uma infra-estrutura em tecnologia da informação capaz de prover informações confiáveis e de qualidade para suporte a processos de tomada de decisão no âmbito do Programa de Desenvolvimento da Fruticultura (PROFRUTA).

*José Gilberto Jardine*  
Chefe-Geral





# Sumário

Introdução .....	9
Favorecendo a Manutenção do DW.....	11
Favorecendo o Desempenho do DW.....	14
Considerações Finais.....	16
Referências Bibliográficas.....	18



# Projeto Físico de Data Warehouse no Oracle: um Estudo de Caso

---

*Carlos Alberto Alves Meira*

## Introdução

Data warehouse (DW) é um banco de dados projetado para consultas e análises ao invés de processamento de transações. Geralmente contém dados históricos provenientes de registros de transações, mas pode incluir dados de outras fontes. DW separa o processamento analítico das transações e permite a uma organização consolidar dados de diversas fontes (Lane & Lumpkin, 1999).

Um dos aspectos importantes relacionados com o desenvolvimento de DW é a modelagem dos dados. O "esquema estrela" é o mais conhecido e talvez o mais simples modelo lógico de dados. É chamado assim porque o diagrama entidade-relacionamento lembra o formato de uma estrela, com pontas irradiando de uma entidade central. O centro da estrela é chamado de tabela de fatos e as pontas são conhecidas como tabelas de dimensão (Garcia-Molina et al., 1999).

Um esquema estrela (Fig. 1) é caracterizado por uma ou mais tabelas de fatos grandes, que contêm os dados primários do DW, e tabelas de dimensão menores, cada uma contendo dados relacionados com valores de um determinado atributo das tabelas de fatos.

O projeto físico de um DW é tão importante quanto o seu projeto lógico. Além de aplicar corretamente os conceitos de integração de dados e elaborar um bom modelo lógico, é preciso planejar o uso dos mecanismos de estrutura física oferecidos pelo sistema gerenciador de banco de dados (SGBD) que se vai utilizar, que, quando bem aproveitados, facilitam a manutenção do DW e melhoram o desempenho das consultas aos dados integrados.

Este trabalho visa compartilhar a experiência no projeto físico de um DW desenvolvido pela Embrapa Informática Agropecuária utilizando o SGBD Oracle. São apresentadas algumas características do Oracle que favorecem a manutenção e o desempenho de consultas a um DW em geral, utilizando como exemplo o DW específico.

É importante ressaltar que este trabalho não apresenta todo o conjunto de características que podem ser usadas no projeto físico de um DW, nem aborda todos os detalhes das características apresentadas. Os leitores interessados em aprofundar os conhecimentos são direcionados para os manuais de referência correspondentes da documentação do Oracle.

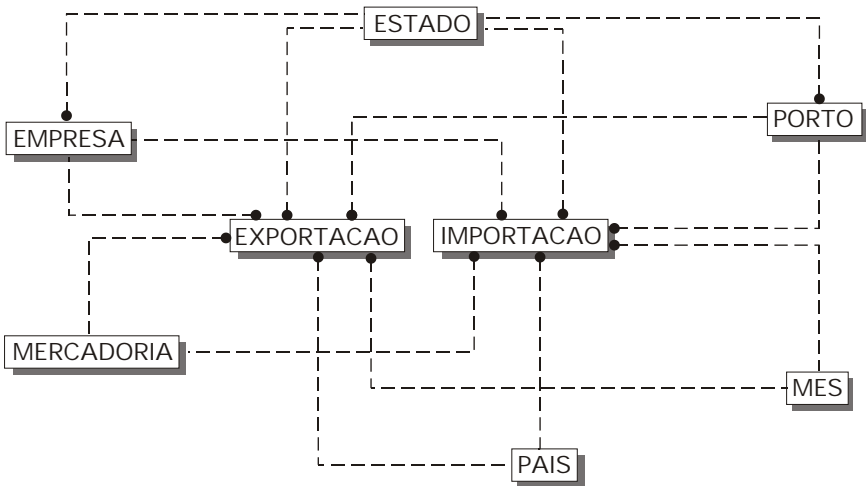


Fig. 1. Esquema estrela<sup>1</sup> de dados de comércio exterior.

O DW utilizado como estudo de caso é o Armazém de Dados da Fruticultura (ADF), onde se encontram reunidas informações de bancos de dados distribuídos e heterogêneos mantidos por diferentes instituições (Meira et al., 2003). O seu objetivo é tornar disponíveis os dados integrados, permitindo análises para a tomada de decisões (Meira et al., 2001, Meira et al., 2002b) como, por exemplo, indicação, acompanhamento e direcionamento de programas de incentivo à produção e à exportação de frutas.

O ADF atende uma demanda do Ministério da Agricultura, Pecuária e Abastecimento (MAPA) para melhorar o processo de análise existente àquela época, ano 2000, quando as consultas eram feitas de forma independente em cada banco de dados das diferentes instituições e os resultados eram transportados e analisados em planilhas eletrônicas. Atualmente, esse processo está sistematizado e automatizado, resultando em rapidez de acesso, flexibilidade na criação e na manutenção das análises, e confiabilidade nos resultados obtidos.

O ADF integra dados primários de produção e de comércio exterior de frutas (Meira et al., 2002a), que são extraídos de dois levantamentos de produção agrícola do Instituto Brasileiro de Geografia e Estatística (IBGE) e de um sistema de informação da Secretaria de Comércio Exterior (SECEX). Pretende-se também incorporar outras fontes de informação referentes à cadeia das frutas, como dados

<sup>1</sup> As tabelas de exportação e de importação, de acordo com a nomenclatura de DW, são as tabelas de fatos. As demais tabelas são denominadas tabelas de dimensão.

de comercialização interna, dados de produção e comercialização mundial, dados socioeconômicos e dados de tecnologia agropecuária.

A discussão e os exemplos apresentados neste trabalho referentes ao projeto físico do ADF estão restritos à sua porção de comércio exterior por razões didáticas. Não há comprometimento quanto ao conteúdo decorrente dessa restrição, pois os mesmos critérios de projeto foram utilizados na parte referente à produção de frutas.

A fonte de dados da SECEX é o Sistema Integrado de Comércio Exterior (Siscomex), que registra todas as exportações e as importações de mercadorias realizadas pelas empresas nacionais. Seus dados, representados na Fig. 1, indicam o ano e o mês de exportação/importação, o estado que exportou/importou a mercadoria, qual o porto utilizado, o país importador/exportador, a quantidade e o valor da negociação.

## Favorecendo a Manutenção do DW

A preocupação com a manutenção do DW é um dos cuidados que se deve ter no seu projeto físico. A manutenção envolve o carregamento periódico de dados atualizados e tarefas de cópia de segurança (*back up*) e restauração de dados.

O cuidado com a manutenção começa na criação da estrutura física primária de armazenamento do Oracle, o *tablespace*. Um *tablespace* é um recipiente de estruturas físicas de armazenamento no banco de dados.

É recomendável que um DW, como qualquer banco de dados a ser criado no Oracle, seja dividido em diferentes *tablespaces*. Tabelas devem estar separadas de seus índices e tabelas pequenas podem estar separadas de grandes tabelas. *Tablespaces* podem também representar unidades de negócio distintas, como departamentos de uma organização.

Um *tablespace* independente para as tabelas facilita e otimiza as operações de *back up* e de restauração de dados. Não guardar cópia de segurança de índices e restrições diminui o espaço de armazenamento necessário e, sem problemas, esses índices e restrições podem ser recriados após uma operação de restauração que se faça necessária.

No caso do ADF, foram criados um *tablespace* para as tabelas (D\_PROFRUTA) e outro para os índices e as restrições de chave primária e de chave estrangeira (I\_PROFRUTA). O nome desses *tablespaces* está baseado no padrão utilizado pela Coordenadoria de Informática (COINF) do MAPA nos seus bancos de dados, onde 'D' corresponde a dados e 'I' corresponde a índices.

Outra decisão de projeto físico que favorece a manutenção de um DW é a divisão de tabelas grandes em partições (*partitioned tables*), permitindo que os dados sejam organizados em pedaços menores e mais fáceis de lidar (Baylis & Fee, 1999).

Dentre os métodos existentes para definir as partições de uma tabela, o mais comum é a divisão baseada em intervalos de valores de um ou mais atributos (*range partitioning*). Esse tipo de divisão é útil quando se tem sentido trabalhar com os dados distribuídos em intervalos lógicos como, por exemplo, meses do ano.

Além da divisão lógica, a partição das tabelas de fatos, normalmente as maiores tabelas, deve se basear no processo de atualização do DW. O carregamento de dados geralmente é feito em períodos de tempo regulares: toda noite, toda semana ou todo mês. Nesse cenário, sugere-se que as tabelas estejam divididas por data ou por período de tempo.

O recebimento de dados de comércio exterior no ADF é mensal. A SECEX envia todos os meses um arquivo com dados sobre exportações e outro arquivo com dados sobre importações. Cada arquivo contém os dados do período de janeiro do ano corrente até o mês anterior ao de recebimento. Por exemplo, em outubro de 2002 foi recebido o arquivo com os registros de exportação de janeiro até setembro de 2002.

Sendo assim, como são recebidos a cada mês dados desde o início do ano, decidiu-se a divisão por anos para a tabela de exportações e para a tabela de importações. O comando em SQL (Structured Query Language) de criação da tabela de exportações é apresentado a seguir (Lorentz, 1999). A criação dessa tabela foi feita num momento onde se tinha um histórico de dados de 1997 até 2002, por isso o comando de criação inicia a divisão em partições no ano 1997 (`VALUES LESS THAN ('1998')`) e prossegue até o ano 2002.

```
CREATE TABLE EXPORTACAO (
    ano                CHAR(4) NOT NULL,
    mes                CHAR(2) NOT NULL,
    cod_ncm             CHAR(8) NOT NULL,
    cod_pais            CHAR(3) NOT NULL,
    pessoa_jur_fis     CHAR(1) NOT NULL,
    cnpj_cpf           VARCHAR2(14) NOT NULL,
    cod_porto          CHAR(4) NOT NULL,
    cod_uf              CHAR(2) NOT NULL,
    cod_unid_merc      CHAR(2) NOT NULL,
    quantidade         NUMBER(13) NOT NULL,
    peso               NUMBER(13) NOT NULL,
    us_fob             NUMBER(15) NOT NULL
) TABLESPACE D_PROFRUTA
PARTITION BY RANGE (ano) (
    PARTITION EXP1997 VALUES LESS THAN ('1998'),
    PARTITION EXP1998 VALUES LESS THAN ('1999'),
    PARTITION EXP1999 VALUES LESS THAN ('2000'),
    PARTITION EXP2000 VALUES LESS THAN ('2001'),
    PARTITION EXP2001 VALUES LESS THAN ('2002'),
    PARTITION EXP2002 VALUES LESS THAN ('2003')
);
```

As vantagens esperadas dessa decisão de projeto físico se confirmaram no "dia-a-dia" de uso do ADF. A principal vantagem é no carregamento mensal de novos dados. Para percebê-la, é preciso primeiro imaginar como seria o processo de carregamento de dados para uma tabela de exportações sem divisão em partições.

Como exemplificado anteriormente, em outubro de 2002 foi preciso carregar no ADF o arquivo com os registros de exportações de janeiro até setembro de 2002. No caso da tabela sem as partições, seria necessário o seguinte procedimento:

- remover os registros referentes ao período de janeiro a agosto de 2002<sup>2</sup> da tabela EXPORTACAO;
- inserir os novos registros na tabela EXPORTACAO;
- recriar os índices relacionados com a tabela EXPORTACAO.

O procedimento em uso no ADF de carregamento desses dados, que utiliza a divisão em partições, é o seguinte:

- carregar os novos dados para uma tabela temporária separada, por exemplo EXP\_TEMP;
- criar os índices e as restrições na tabela EXP\_TEMP. Os índices e as restrições criados nessa tabela devem ser idênticos aos criados na tabela EXPORTACAO;
- trocar a tabela EXP\_TEMP pela partição EXP2002 com um simples comando:

```
ALTER TABLE EXPORTACAO EXCHANGE PARTITION EXP2002 WITH TABLE EXP_TEMP;
```

Apesar dos dois procedimentos apresentados possuírem três passos, o último apresenta vantagens significativas, pois os novos dados são carregados com utilização mínima de recursos, já que são carregados numa tabela totalmente independente e o processamento de índices e de restrições é efetuado apenas na nova partição. O tempo de carregamento é proporcional à quantidade de dados sendo carregada e não ao tamanho total da tabela.

Além disso, os novos dados são carregados com impacto mínimo em consultas concorrentes. Até a troca da tabela temporária com a partição, os dados e índices da tabela EXPORTACAO permanecem inalterados. E a execução do comando EXCHANGE é instantânea, permitindo aos usuários acesso imediato aos novos dados assim que a troca é feita.

Quando a partição correspondente não existir, no caso do carregamento de dados de um novo ano, é preciso apenas criá-la vazia antes da execução do comando EXCHANGE. Por exemplo, o comando que cria a partição para o ano 2003 é o seguinte:

---

<sup>2</sup> Esses registros foram carregados no mês anterior, mas podem ter sido atualizados no novo arquivo recebido.

```
ALTER TABLE EXPORTACAO ADD PARTITION EXP2003 VALUES LESS THAN ( '2004' );
```

Uma outra vantagem da divisão em partições para o ADF é a facilidade de remoção de dados passados que não se tem mais interesse. Suponha que a análise de comércio exterior deva considerar até cinco anos antes do ano corrente. Então, criada a partição EXP2003, pode-se remover a partição EXP1997 com um simples comando:

```
ALTER TABLE EXPORTACAO DROP PARTITION EXP1997;
```

## Favorecendo o Desempenho do DW

A primeira providência para favorecer o desempenho de consultas a um DW é criá-lo numa instância de banco de dados Oracle apropriada para aplicações OLAP (Online Analytical Processing). Se a instância existente está ajustada para aplicações mais comuns de processamento de transações, chamadas OLTP (Online Transactional Processing), é recomendado que se crie uma instância própria para OLAP, que é uma das opções de criação de instância de banco de dados no SGBD Oracle.

A divisão das tabelas de fatos em partições também favorece o desempenho das consultas a um DW. O otimizador de consultas do Oracle analisa as cláusulas FROM e WHERE dos comandos SQL e elimina partições desnecessárias na construção de uma lista de partições de acesso, o que é chamado de poda de partições (*partition pruning*). Isso permite que o Oracle execute operações apenas nas partições relevantes para o comando SQL. Veja a seguinte consulta:

```
SELECT MES, SUM(QUANTIDADE) FROM EXPORTACAO  
WHERE ANO = '2002'  
GROUP BY MES;
```

Esta consulta calcula a soma das quantidades exportadas para todos os meses de 2002. Nesse caso, o otimizador vai indicar ao Oracle que é necessário apenas acesso à partição EXP2002, diminuindo o tempo de resposta já que não é necessário acesso à tabela completa de exportações.

Uma estrutura física muito comum em banco de dados para melhorar o desempenho de consultas é o índice. A função do índice é proporcionar acesso rápido aos dados por meio de ponteiros para linhas de uma tabela que possuem um determinado valor de chave para um atributo. Num índice convencional, isso é conseguido com a criação de uma lista de identificadores de linhas para cada valor de chave, correspondendo às linhas da tabela com aquele valor de atributo. O Oracle armazena repetidamente cada valor de chave com cada identificador de linha.



No índice bitmap, um tipo diferente de índice, é usado um mapa de bits para cada valor de chave ao invés de uma lista de identificadores de linhas (Leverenz et al., 1999). Cada bit no mapa de bits corresponde a uma linha da tabela. Se o bit estiver marcado (valor 1), significa que a linha identificada contém aquele valor de chave para o atributo. Um índice bitmap é criado usando-se a palavra-chave BITMAP no comando de criação do índice (veja exemplo mais adiante). Indexar totalmente uma tabela grande com índices tradicionais de árvore-B pode se tornar proibitivo dado o espaço necessário, já que os índices podem somar um tamanho maior que o tamanho da tabela. Índices bitmap, ao contrário, geralmente são uma fração do tamanho dos dados indexados da tabela.

As vantagens de usar índices bitmap são maiores para atributos com baixa cardinalidade, isto é, atributos para os quais o número de valores distintos é pequeno em comparação com o número de linhas da tabela. Se os valores numa coluna são repetidos mais do que uma centena de vezes, então a coluna é candidata para índice bitmap. Por exemplo, os atributos MES e COD\_PAIS da tabela EXPORTACAO são indicados para índices bitmap. Índices de árvore-B devem ser usados em DW apenas para atributos únicos ou outras colunas com alta cardinalidade, como nomes próprios.

Índices bitmap melhoraram sensivelmente o tempo de resposta em consultas. Condições AND e OR em cláusulas WHERE podem ser resolvidas rapidamente resolvendo-se as operações booleanas correspondentes diretamente nos bitmaps, e depois convertendo-se o bitmap resultante em identificadores de linhas.

Um índice bitmap de uma tabela dividida em partições deve ser local, também dividido em partições. Num índice local, todas as chaves de uma determinada partição do índice referem-se a linhas armazenadas numa única partição da tabela. Um índice local é criado usando-se a palavra-chave LOCAL no comando de criação do índice (veja exemplo a seguir).

Como exemplo de comando para criação de um índice bitmap para uma tabela dividida em partições, segue o comando SQL de criação do índice para o atributo MES da tabela EXPORTACAO:

```
CREATE BITMAP INDEX BMX_MES_EXP ON EXPORTACAO (
    mes ASC
) TABLESPACE I_PROFRUTA
LOCAL (
    PARTITION EXP1997,
    PARTITION EXP1998,
    PARTITION EXP1999,
    PARTITION EXP2000,
    PARTITION EXP2001,
    PARTITION EXP2002
);
```

O comando que adiciona uma partição a uma tabela (ADD PARTITION), apresentado anteriormente, adiciona automaticamente partições nos índices bitmap da tabela. E o comando que troca uma partição por uma tabela (EXCHANGE PARTITION), também exemplificado anteriormente, troca as partições correspondentes dos índices bitmap pelos índices da tabela de troca.

Outra técnica empregada em DW para melhorar o desempenho de consultas é a criação de resumos (*summaries*). Resumo é um tipo especial de visão que diminui o tempo de execução de consultas ao pré-calcular junções e agregações, e armazenar os resultados numa tabela do banco de dados. Por exemplo, pode-se criar um resumo com o agregado das exportações de cada ano, por fruta e por região. Resumos são criados no Oracle por meio do objeto de esquema chamado visão materializada.

O otimizador do Oracle usa visões materializadas para melhorar o desempenho de consultas reconhecendo quando uma dessas visões pode e deve ser usada para satisfazer uma requisição de dados. Ele automaticamente, de forma transparente, reescreve a consulta e a direciona para a visão materializada no lugar da tabela subjacente.

O uso de visões materializadas no desenvolvimento do ADF foi auxiliado pela versão administrativa da ferramenta de inteligência de negócios (*business intelligence*) Oracle Discoverer™ (Oracle Corporation, 2002), um software de análise multidimensional que permite ao usuário analisar os dados em diferentes dimensões. Neste caso, não será feita a apresentação de um exemplo, pois iria requerer uma apresentação mais detalhada do Discoverer, portanto fora do escopo deste trabalho.

## Considerações Finais

Numa análise informal das características do Oracle utilizadas na construção do ADF e dos benefícios obtidos com a adoção dessas características, destacam-se os seguintes pontos:

- facilidade de manutenção do ADF obtida com a partição das tabelas de fatos;
- impacto praticamente inexistente do processo de carregamento de novos dados em consultas concorrentes de usuários acessando os relatórios de análise a partir do ADF;
- espaço ocupado pelos índices bitmap como uma fração do espaço ocupado pelos dados das tabelas;
- melhora de desempenho das consultas quando visões materializadas estão disponíveis.

Mesmo sem resultados de uma análise mais criteriosa, comparando-se tempos de execução com e sem a adoção de determinada característica dentre as apresentadas neste trabalho, conclui-se que foi de suma importância o estudo do ambiente de *data warehousing* do Oracle e o uso planejado dos recursos disponíveis no projeto físico do ADF.

Pretende-se que este documento sirva como uma visão inicial e sucinta dos recursos existentes para quem tem interesse na construção de DW no SGBD Oracle. Após a leitura deste, encoraja-se consultas às referências bibliográficas indicadas a fim de se obter mais detalhes das características apresentadas e de outras disponíveis que podem melhor servir o projeto específico de interesse.

## Referências bibliográficas

- BAYLIS, R.; FEE, J. Oracle8i: administrator's guide, release 2 (8.1.6). Redwood City: Oracle Corporation, 1999. Paginação irregular.
- GARCIA-MOLINA, H.; ULLMAN, J. D.; WIDOW, J. Information integration. In: GARCIA-MOLINA, H.; ULLMAN, J. D.; WIDOW, J. Database system implementation. New York: Prentice Hall, 1999. Ch.11, p. 595-641.
- LANE, P.; LUMPKIN, G. Oracle8i: data warehousing guide, release 2 (8.1.6). Redwood City: Oracle Corporation, 1999. Paginação irregular.
- LEVERENZ, L.; REHFELD, D.; BAIRD, C. Oracle8i: concepts, release 2 (8.1.6). Redwood City: Oracle Corporation, 1999. Paginação irregular.
- LORENTZ, D. Oracle8i: SQL reference, release 2 (8.1.6). Redwood City: Oracle Corporation, 1999. Paginação irregular.
- MEIRA, C. A. A.; SEIXAS NETO, A.; NAKA, J.; CUNHA, L. M. S. Análise de comércio exterior de frutas a partir do armazém de dados da fruticultura brasileira. Campinas: Embrapa Informática Agropecuária, 2001. 6 p. (Embrapa Informática Agropecuária. Comunicado Técnico, 17).
- MEIRA, C. A. A.; CUNHA, L. M. S.; SEIXAS NETO, A.; NAKA, J. Integração de informação da fruticultura. In: CONGRESSO BRASILEIRO DE FRUTICULTURA, 17., 2002a, Belém. Os novos desafios da fruticultura brasileira: anais. Belém: Sociedade Brasileira de Fruticultura, 2002. 1 CD-ROM.
- MEIRA, C. A. A.; SEIXAS NETO, A.; NAKA, J.; CUNHA, L. M. S. Análise da produção brasileira de frutas a partir do Armazém de Dados da Fruticultura. Campinas: Embrapa Informática Agropecuária, 2002b. 6 p. (Embrapa Informática Agropecuária. Comunicado Técnico, 18).
- MEIRA, C. A. A.; SEIXAS NETO, Á.; NAKA, J.; CUNHA, L. M. S. Armazém de Dados da Fruticultura. In: CONGRESSO BRASILEIRO DA SOCIEDADE BRASILEIRA DE INFORMÁTICA APLICADA À AGROPECUÁRIA E À AGROINDÚSTRIA, 4., 2003, Porto Seguro. Anais... Lavras: SBIAGRO, 2003. 1 CD-ROM.
- ORACLE CORPORATION. Oracle9i Discoverer Administrator: administration guide, version 9.0.2. Redwood City, 2002. Paginação irregular.



---

*Informática Agropecuária*

Ministério da Agricultura, Governo  
Pecuária e Abastecimento Federal