

Os benefícios da modelagem de dados no armazenamento de dados

DEZEMBRO DE 2008

Índice

Resumo Executivo	1
SEÇÃO 1	2
Introdução	2
SEÇÃO 2	2
Projetando o Armazenamento de Dados	2
Tabela de Dados	3
Dimensões	3
SEÇÃO 3	7
Extrair, Transformar e Carregar	7
SEÇÃO 4	8
A Importância da Modelagem de Dados como uma Melhor Prática de Armazenamento de Dados	8
Como Reunir os Requisitos de Negócios	8
Otimizando a Performance da Base de Dados	8
Proporcionando Documentação de Sistemas de Fonte e Meta	9
SEÇÃO 5	9
Conclusão	9

Copyright © 2008 CA. Todos os direitos reservados. Todas as marcas registradas, nomes comerciais, marcas de serviços e logotipos aqui mencionados pertencem às suas respectivas empresas. Este documento é somente para fins informativos. Até onde for permitido pela lei aplicável, a CA proporciona este documento “nas condições em que se encontra”, sem garantia de qualquer tipo, incluindo sem limitações, qualquer garantia implícita de comerciabilidade, adequação para qualquer propósito em particular ou não-infratora. Em nenhum caso a CA será responsável por quaisquer perdas ou danos, diretos ou indiretos, pelo uso deste documento, incluindo sem limitações, perda de lucros, interrupção dos negócios, prestígio, ou perda de dados, mesmo se a CA for expressamente advertida de tais danos.

Resumo Executivo

DESAFIOS

Atualmente, as organizações possuem grandes quantidades de dados. Embora estes dados contenham informações que são úteis para o negócio, pode ser extremamente difícil reuni-los e fazer relatórios sobre estas informações. Existem vários desafios fundamentais que precisam ser abordados:

- Descobrir, coletar, e transformar dados numa única fonte de registros.
- Garantir que os dados sejam relevantes e precisos para a elaboração de relatórios de negócios.
- Armazenar dados históricos num formato que permita realizar pesquisas rápidas através de grandes quantidades de dados.

OPORTUNIDADES

Caso os dados que o negócio mantém possam ser desbloqueados e proporcionar informações significativas aos usuários da empresa, existem várias possibilidades:

- Os usuários da empresa podem ter acesso a informações relevantes que lhes permitam tomar decisões bem informadas.
- Consultas rápidas tornam os dados mais acessíveis e dados históricos permitem que as tendências sejam identificadas.
- Os dados podem ser avaliados em todos os níveis, desde uma compra individual até as vendas totais de uma corporação multinacional.

BENEFÍCIOS

Projetar corretamente armazenamentos de dados utilizando um modelo de dados ajudará a vencer muitos dos desafios da atualidade. Os benefícios principais incluem:

- Projetar estruturas que permitam especificamente realizar consultas rápidas para elaborar relatórios centralizados na empresa.
- Garantir que os requisitos da empresa sejam satisfeitos, e os relatórios sejam precisos e tenham sentido.
- Documentar sistemas fonte e meta corretamente para auxiliar no desenvolvimento, assegurar o controle efetivo das versões, e melhorar a compreensão dos sistemas.

SEÇÃO 1

Introdução

A maioria das organizações possui grandes quantidades de dados. Os dados são coletados constantemente, conforme cada transação é feita, cada revisão do empregado é completada, e cada venda em potencial é perseguida. Estes dados estão no centro dos sistemas que administram uma organização, e as bases de dados que fortalecem estes sistemas foram projetadas de forma a tornar os processos comerciais de uma organização tão eficientes quanto for possível. O problema surge quando os usuários da empresa precisam reportar estas informações, para determinar quantas transações foram feitas este ano, por exemplo.

Existem vários desafios ao usar sistemas transacionais para a elaboração de relatórios comerciais:

- O projeto de base de dados que é requerido para elaborar relatórios é muito diferente do projeto que é requerido para otimizar a performance dos sistemas transacionais.
- Executar relatórios com os sistemas transacionais essenciais reduz a sua performance, afetando negativamente os sistemas que administram a organização.
- Os dados que estão armazenados num sistema transacional de base de dados não estão centralizados; não existe fonte única de informações com as quais os relatórios possam ser gerados.

Para resolver estes desafios, devemos criar um armazenamento de dados projetado especificamente pensando na elaboração de relatórios comerciais, e que contenha todas as informações relevantes para elaborar relatórios.

O armazenamento de dados é um importante fornecedor de informações para a empresa, por isso é essencial modelar tanto os projetos físicos como os lógicos. O projeto físico determina a performance e a funcionalidade do armazenamento de dados, e o projeto lógico é a visão que apresentamos aos desenvolvedores e usuários para compreendam os requisitos da empresa.

SEÇÃO 2

Projetando o Armazenamento de Dados

Todos os nossos dados são armazenados no armazenamento de dados de forma que a performance de consulta seja priorizada, em vez da performance transacional ou os volumes de armazenamento. Podemos acessar informações sobre qualquer área da nossa empresa, e como estas entidades se relacionam às métricas de performance ou a outras entidades. Esta visão da empresa pode proporcionar uma visão do funcionamento da organização, o qual ajuda no planejamento e proporciona uma vantagem competitiva.

Um verdadeiro armazenamento de dados contém todos os dados disponíveis, embora com frequência, a base de dados armazene um subconjunto enfocado em certos dados e então passe a ser chamada tecnicamente de data marts.¹

Os armazenamentos de dados normalmente armazenam dados históricos, os quais nos permitem consultar com precisão os eventos anteriores. Por exemplo, um sistema de processamento de transação on-line (OLTP) deve armazenar o importador atual de um produto e, se consultado, deveria retornar este o dado deste importador, independentemente de se o importador atual era o mesmo quando a transação ocorreu. Um armazenamento de dados, por outro lado, deve armazenar normalmente todos os importadores do produto em uma forma que nos permitisse entrar em contato com precisão com a compra com o seu importador.

É essencial modelar o projeto de armazenamento de dados para que as perguntas que a empresa faz possam ser respondidas efetivamente. Esta seção cobre algumas das escolhas

de projeto para armazenamento de dados, que nós criamos num tipo específico de modelo de dados chamado de modelo de dados dimensional.

Tabela de Dados

A tabela central de um projeto de armazenamento de dados é chamada de tabela de dados. Esta tabela tem uma linha para cada fato ou evento. Cada fato tem uma ou mais medidas numéricas quantificáveis, por exemplo, preço. A tabela de dados também contém vários valores de dimensão. Valores de dimensões são descrevem o fato e pode incluir valores tais como tempo, empregado, cliente, e localização.

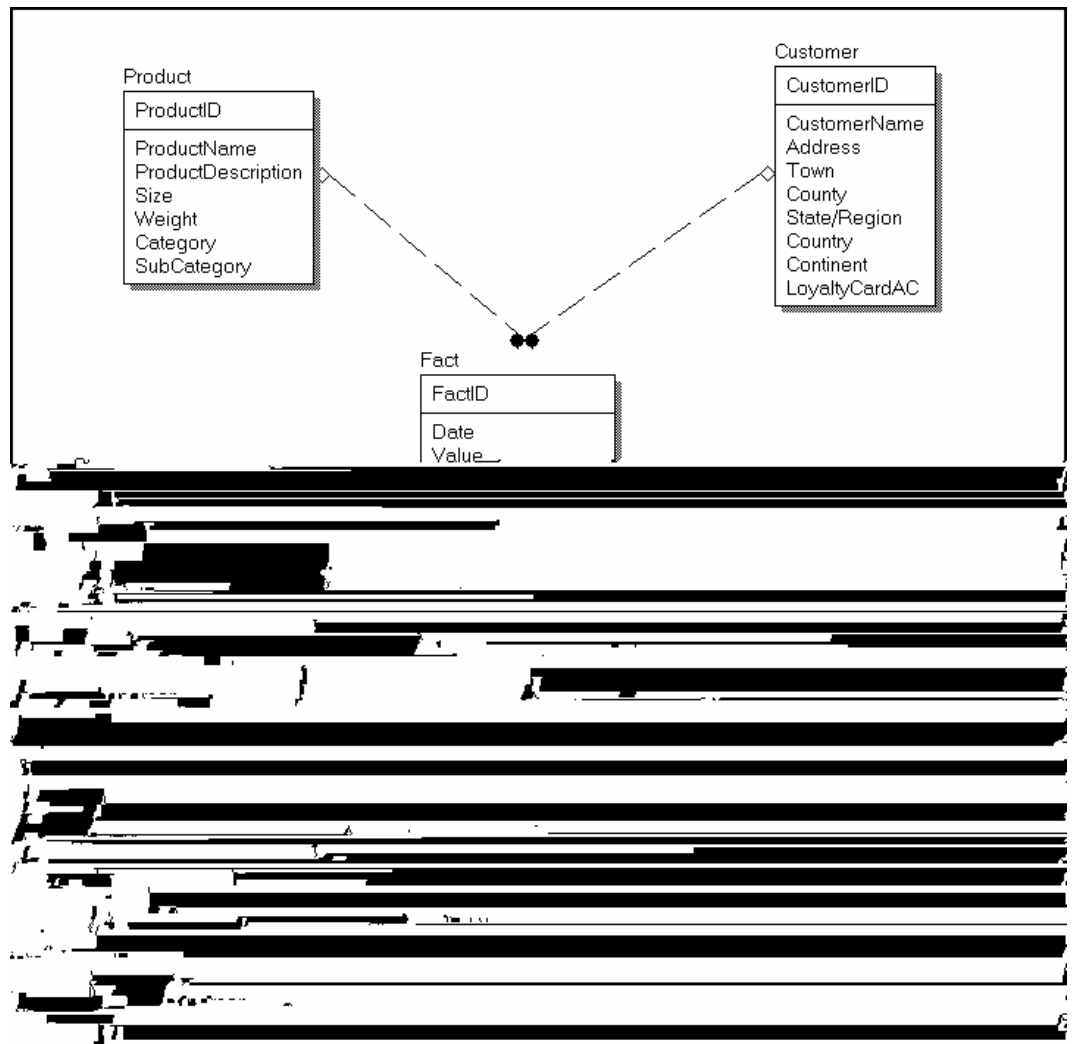
Dimensões

Valores de dimensões são normalmente armazenados como chaves externas na tabela de dados. Estas chaves estão relacionadas com as chaves primárias das tabelas de dimensões. As tabelas de dimensões descrevem cada membro da dimensão. Por exemplo, uma tabela de dados inclui o ID de empregado de um vendedor e a dimensão Empregado inclui o ID, nome, localização, e gerente do empregado. Nem todas as dimensões requerem uma tabela de dimensões. Por exemplo, a dimensão Tempo pode existir inteiramente na tabela de dados porque não existem outras propriedades além do próprio tempo. Caso existissem outras propriedades, tais como feriados oficiais, é requerida uma tabela de dimensões.

ESQUEMA STAR

Se uma tabela de dados tem um nível de tabelas de dimensões, o projeto de base de dados é conhecido como um esquema estrela (veja Figura 1).

FIGURA 1: ESQUEMA ESTRELA



MODELO DE DADOS MOSTRANDO UM ESQUEMA ESTRELA

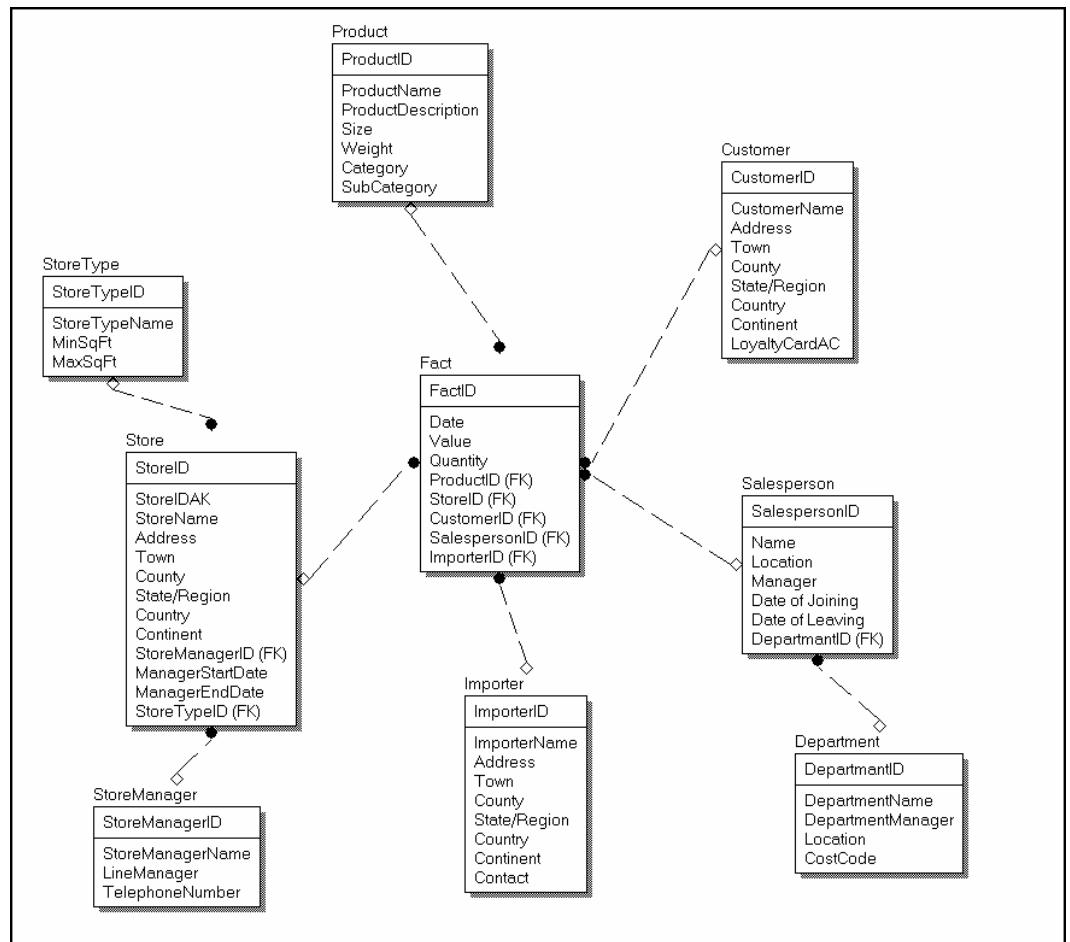
Os esquemas estrela são úteis porque cada propriedade de uma dimensão pode ser recuperada unindo a tabela de dados com a tabela de dimensão relevante. Isto melhora a performance da consulta, mas aumenta os volumes de dados

ESQUEMA FLOCO DE NEVE

Alguns detalhes são usados raramente em consultas, e devem ser modeladas de maneira diferente. Por exemplo, raramente consultamos qual é o departamento de um empregado. Todos os vendedores estão no departamento de Vendas, e, portanto, é de pouca utilidade analisar dados referentes ao departamento de Vendas numa consulta. Não obstante, queremos armazenar este dado. Podemos criar uma tabela adicional de dimensões que se relacione com a dimensão Empregado. Este é um esquema floco de neve (veja Figura 2) e é útil porque elimina a duplicação que poderia ocorrer caso duplicássemos as informações de departamento de cada empregado. Devemos sempre considerar com que frequência uma busca usará os dados do esquema floco de neve, entretanto, como ele requer uma união extra, é portanto, mais lento.

Devido aos problemas de performance, geralmente devemos evitar os esquemas floco de neve em nosso modelo.

FIGURA 2: ESQUEMA FLOCO DE NEVE



MODELO DE DADOS INCLUINDO ESQUEMAS FLOCO DE NEVE

Esquemas estrela e floco de neve são modelados no nível de dimensões e não se aplicam ao projeto de todo o armazenamento de dados. Na Figura 2, as dimensões Loja e Vendedores têm um esquema floco de neve, mas as dimensões Produto, Cliente, e Importador têm um esquema estrela.

No projeto das tabelas de dados e de dimensões podemos ver que o projeto do armazenamento de dados está muito mal distribuído. A normalização busca melhorar a eficiência dos sistemas OLTP a través da eliminação da duplicação, mas em um sistema projetado simplesmente para a normalização rápida da consulta, ela é prejudicial para a performance.

DIMENSÕES QUE MUDAM LENTAMENTE

Por enquanto, tomamos cada dimensão como uma imagem instantânea (snapshot) oportuna, mas, na verdade, os atributos das dimensões mudam. Por exemplo, um cliente pode estar registrado como residindo no Canadá, mas que recentemente mudou para a França. Não devemos designar ao Canadá qualquer fato que ocorreu na França. Igualmente, os empregados mudam de departamento e as lojas mudam de gerente.

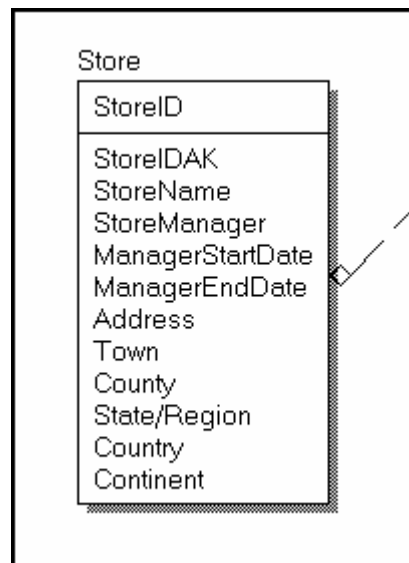
Para armazenar eficientemente as mudanças históricas para dimensionar membros, precisamos criar no nosso modelo dimensões que mudem lentamente (slowly changing dimensions - SCDs). Esta é uma área onde a modelagem é essencial porque ela requer planejamento para alcançar os resultados certos e uma abordagem criada para este fim é

pouco provável que tenha sucesso. Existem três tipos de SCD em uso que cobrem a maioria dos requisitos históricos.

SCDs Tipo 1 são efetivamente dimensões-padrão que permitem mudanças. Se um atributo precisa ser mudado, simplesmente o mudamos e não tomamos mais nenhuma ação. Isto mantém nossos registros atualizados, mas não resolve quaisquer dos problemas históricos que foram mencionados anteriormente.

SCDs Tipo 2 (veja Figura 3) resolvem o problema de mudanças históricas ao criar vários registros para cada membro da dimensão quando exista uma mudança. Por exemplo, se um gerente de loja se mudar, devemos criar outro registro que contenha os dados do novo gerente. Isto cria uma chave duplicada para o registro. Portanto, podemos armazenar a chave original da loja como uma chave alternativa e criar uma nova chave primária única. O outro problema com um SCD Tipo 2 é que precisamos conhecer qual registro se aplica a cada fato na tabela de dados. Isto é possível armazenando as datas de início e fim de cada registro.

FIGURA 3: SCD TIPO 2



SCD TIPO 2 COM CHAVE PRIMÁRIA ALTERNATIVA E DATAS DE INÍCIO E FIM

A Figura 3 mostra a chave alternativa, StoreIDAK, e as datas de início e fim do gerente. É simples encontrar o gerente atual porque este é o único registro com um campo NULL ManagerEndDate.

SCDs Tipo 3 são um compromisso entre os SCDs Tipo 1 e os SCDs Tipo 2. Precisamos conhecer somente os valores originais e atuais do membro da dimensão. Por exemplo, podemos querer um preço inicial e um preço atual. Não requeremos nenhum outro preço, por isso não requeremos a complexidade de um SCD Tipo 2. Podemos simplesmente armazenar os valores originais e atuais como atributos separados da dimensão e somente é necessário uma linha para cada membro. Embora isto simplifique o armazenamento, a funcionalidade limitada significa que a maioria dos modelos usam SCDs Tipo 1 ou Tipo 2.

Como pode ser observado, um armazenamento de dados tem um Projeto substancialmente diferente de um sistema OLTP, e precisamos implementar técnicas especiais de projeto para otimizar um armazenamento de dados para a elaboração de relatórios comerciais.

SEÇÃO 3

Extrair, Transformar e Carregar

Embora um armazenamento de dados seja projetado de maneira substancialmente diferente de um sistema OLTP, os sistemas OLTP são a principal fonte de dados para um armazenamento de dados. Precisamos planejar cuidadosamente como mudar os dados dos sistemas OLTP para o armazenamento de dados. Este processo é conhecido como extrair, transformar e carregar (extract, transform, and load - ETL), e esta seção procura algumas das decisões que precisamos tomar para modelar um sistema ETL.

A primeira tarefa é encontrar os dados disponíveis. Isto é fácil em uma organização enfocada em TI, pequena e recém fundada. Entretanto, em uma grande corporação multinacional que esteja no negócio desde antes que os computadores existissem, e teve múltiplas fusões, é muito mais complexo. É necessário investigar cada um dos sistemas em operação e verificar a documentação existente ou, se não existir, documentar os armazenamento de dados. Isto pode ser um processo muito demorado e laborioso, e destaca a necessidade da modelagem e da documentação. Podemos aplicar engenharia reversa em muitos sistemas ao usar software de modelagem de dados para reduzir o tempo necessário para documentar o armazenamento de dados.

Após identificar os dados a serem carregados no armazenamento de dados, devemos planejar a operação de extração. A maioria dos modernos sistemas utiliza normas reconhecidas para a conectividade, tais como Conectividade de Base de Dados Aberta (Open Database Connectivity - ODBC), que torna a extração de dados simples. Entretanto, sistemas legados podem requerer análise adicional, tal como a criação de perfis de dados antes que estes sistemas possam ser modelados e carregados efetivamente no armazenamento de dados.

É muito pouco provável que os dados estejam já no formato correto para o armazenamento de dados. Como discutimos anteriormente, o armazenamento de dados e sistemas OLTP são projetos fundamentalmente de maneira diferente, por isso os dados devem ser transformados.

A transformação de dados pode ser tão simples quanto uma consulta de Structured Query Language (SQL) que gere o formato correto nos resultados; entretanto, a maioria dos sistemas requerem uma fase de transformação muito mais complexa. Com frequência existem muitas inconsistências nos dados da fonte, como demonstrado nos seguintes exemplos:

- A cidade de Mumbai era conhecida anteriormente como Bombaim, e a cidade de São Petersburgo era conhecida como Leningrado.
- Muitas moedas diferentes podem ser usadas.
- Atributos requeridos podem ser omitidos em alguns armazenamentos de dados.
- Diferentes códigos podem ser usados para representar o mesmo valor.

Precisamos padronizar os atributos das entidades para permitir que os dados possam ser analisados adequadamente. Usar uma ferramenta de modelagem de dados para analisar os sistemas fonte e metas, especialmente um sistema baseado em repositório, ajuda na criação e implementação destes padrões.

Após documentar e projetar os dados de fonte e meta podemos usar uma ferramenta de ETL para executar a maioria das operações de transformação, mas as transformações mais complexas podem se beneficiar ou requerer programação.

Comparado com a extração e transformação, o carregamento é relativamente simples. Após extrair e transformar os dados eles devem estar no formato correto para serem carregados no armazenamento de dados. A mais importante consideração é o sincronismo. Este é um processo que utiliza muitos recursos e deve ser executado quando o sistema for

usado raramente, ou não for usado em absoluto. Isto é comum à noite ou durante o fim de semana.

As operações de ETL requerem que tenhamos um conhecimento completo da fonte e dos sistemas de base de dados de destino. A modelagem de dados proporciona a documentação destes projetos e ajuda a garantir um projeto de processo de ETL correto.

SEÇÃO 4

A Importância da Modelagem de Dados como uma Melhor Prática de Armazenamento de Dados

A maioria dos projetistas de armazenamento de dados utiliza uma ferramenta de modelagem de dados para criar o projeto lógico e físico do armazenamento de dados. O projeto lógico garante que todos os requisitos, definições, e regras do negócio sejam suportados. O projeto físico assegura a ótima performance no planejamento de índices, relacionamentos, tipos de dados, e propriedades. Para dar suporte aos desenvolvedores de sistemas OLAP, mineração de dados, e elaboração de relatórios, o modelo de dados também atua como documentação para o armazenamento final dos dados.

Como Reunir os Requisitos de Negócios

É particularmente útil criar modelos lógicos, bem como modelos físicos para o armazenamento de dados. Geralmente, devemos iniciar um projeto de armazenamento de dados com usuários da empresa e arquitetos de dados que decidam quais entidades são necessárias no armazenamento de dados e quais fatos devem ser registrados. Este projeto inicial apresenta a visão do armazenamento de dados e com frequência tem muitas iterações antes que todas as partes estejam satisfeitas com o projeto. Nesta etapa, devemos nos esforçar para evitar as ciladas comuns dos projetos de armazenamento de dados. Por exemplo, devido a que estamos carregando o armazenamento de dados dos sistemas existentes, é muito fácil esquecer elementos destes sistemas no projeto acabado, ou mesmo usar grandes partes do projeto do modelo existente para economizar tempo. Ao usar a modelagem de dados, é possível ver estes problemas em uma etapa inicial.

Não devemos pensar no modelo lógico somente como sendo um bloco de edifícios para o modelo físico. Embora os modelos lógicos sejam uma etapa essencial na criação de um modelo físico, eles também têm muitos usos após termos criado o modelo físico e o armazenamento de dados. O modelo lógico captura os requisitos de negócios. Ele utiliza convenções de nomeação que coincidem estreitamente com os termos comerciais que uma organização usa e esse é o projeto que é apresentado ao mundo exterior. Desenvolvedores de outros sistemas usam este projeto para criar interfaces no armazenamento de dados. Podemos criar vários modelos lógicos que coincidam com as necessidades dos consumidores de dados, enquanto que, no fundo, o modelo físico continua sendo o mesmo. Ao continuar com o desenvolvimento e manutenção do modelo lógico, evitamos o risco de ter um modelo físico que tenta executar tarefas de modelagem física e lógica, e sofrer com o resultado.

Otimizando a Performance da Base de Dados

A performance da consulta é crítica para o armazenamento de dados. Sacrificamos volumes de dados e a performance transacional para garantir o mais alto nível de performance para as nossas consultas, mas as consultas somente são executadas otimamente se for usado o projeto adequado da base de dados.

Isto envolve enormes volumes de dados, por isso é muito difícil usar a abordagem de tentativa e erro para o projeto de armazenamento de dados. Portanto, é bom usar produtos de modelagem de dados para automatizar este processo e fazê-lo fácil de gerenciar todos os metadados que estão associados com o armazenamento de dados e os dados que são consumidos pelos sistemas de inteligência empresarial (business intelligence - BI). Após termos projetado as entidades de dimensão e a tabela de dados, podemos decidir os

relacionamentos entre tabelas. Agora, a equipe mais extensa de BI pode revisar e avaliar este projeto, e nós podemos incorporar qualquer mudança. Fazer mudanças nesta etapa é muito simples, especialmente quando as comparamos com a complexidade de modificar um armazenamento de dados já finalizado. Uma vez aprovado o projeto lógico, podemos então iniciar o trabalho no projeto físico.

O projeto físico adiciona detalhes tais como tipos de dados e indexação, mas também pode mudar o projeto da entidade básica por razões de performance. O projeto físico é particularmente importante para a performance. Em um sistema OLTP, é muito comum ver tipos de dados um pouco grandes para os dados que eles armazenam. Por exemplo, freqüentemente é usado um número inteiro de oito dígitos onde um número inteiro de quatro dígitos seria suficiente. Em um sistema que contenha milhões de registros, estas pequenas ineficiências são amplificadas. A indexação do armazenamento de dados é ainda mais importante. O armazenamento de dados é projetado para a performance da consulta; os aspectos mais importantes disto são o esquema, tais como estrela ou floco de neve, e projeto de índices. Também podemos querer mudar o esquema do armazenamento de dados, talvez de um esquema estrela para um esquema floco de neve.

Proporcionando Documentação de Sistemas de Fonte e Meta

Quando um sistema ETL for modelado, é essencial verificar os modelos lógicos e físicos do sistema da fonte bem como os modelos de destino. Freqüentemente é necessário criar um modelo intermediário para uma área de armazenamento temporal porque muitas operações não podem se realizadas numa só etapa. Também, com freqüência, as operações de extração, transformação e carregamento não podem acontecer ao mesmo tempo devido aos requisitos dos sistemas da fonte e destino.

SEÇÃO 5

Conclusão

Se pudermos aproveitar a vasta quantidade de dados que está disponível nas nossas organizações, podemos obter enormes benefícios comerciais. Podemos analisar com precisão os resultados passados, fornecer estas informações aos sistemas de BI para encontrar correlações em nossos dados, e apresentar informações de uma forma amigável para os usuários da empresa.

Para alcançar estas metas, devemos modelar o armazenamento de dados muito cuidadosamente. Os sistemas de fonte são, com freqüência, variados e podem carecer de precisão. Os sistemas de destino com freqüência têm diferentes requisitos e, igualmente, cada usuário da empresa pode ter diferentes necessidades. Devemos dedicar uma quantidade substancial de tempo em modelar o sistema do armazenamento de dados. Projetar armazenamento de dados é caro e levam muito tempo, mas, quando projetados corretamente, eles podem proporcionar enormes benefícios ao negócio.

Para alcançar nossas metas, devemos modelar cuidadosamente o sistema de armazenamento de dados que projetamos para atender nossas metas; devemos criar modelos lógicos para os muitos consumidores dos nossos dados e modelos físicos para garantir a adequada performance da base de dados; e mais importante ainda, devemos nos esforçar em satisfazer as necessidades da empresa.