



Data Science Academy

www.datascienceacademy.com.br

Design e Implementação de Data Warehouses

Integridade Referencial

Na era da informação sob demanda não podemos nos contentar que os dados são apenas registros armazenados em um banco de dados. Sem a devida garantia de sua integridade também não é possível transformá-los em informação.

De nada adianta termos imensos bancos de dados armazenando incontáveis bytes de dados, que são constantemente inseridos, modificados, eliminados, se não podemos transformá-los em informação. E, de preferência, informação útil. E sem garantir que os dados armazenados estão íntegros, dificilmente conseguimos transformar esses bytes em informação real. Não importa quão avançada sejam suas ferramentas ou técnicas de aplicação em seu ambiente, nada irá adiantar se não houver um investimento na qualidade de seus dados.

A Integridade Referencial é utilizada para garantir a Integridade dos dados entre as tabelas relacionadas. Por exemplo, considere um relacionamento do tipo um-para-vários entre a tabela Clientes e a tabela Pedidos (um cliente pode fazer vários pedidos). Com a Integridade Referencial, o banco de dados não permite que seja cadastrado um pedido para um cliente que ainda não foi cadastrado. Em outras palavras, ao cadastrar um pedido, o banco de dados verifica se o código do cliente que foi digitado já existe na tabela Clientes. Se não existir, o cadastro do pedido não será aceito. Com o uso da Integridade Referencial é possível ter as seguintes garantias (ainda usando o exemplo entre as tabelas Clientes e Pedidos):

- Quando o Código de um cliente for alterado na tabela Clientes, podemos configurar para o banco de dados atualizar, automaticamente, todos os Códigos do Cliente na tabela Pedidos, de tal maneira que não fiquem Registros Órfãos, isto é, registros de Pedidos com um Código de Cliente para o qual não existe mais um correspondente na tabela Clientes. Essa ação é conhecida como "Propagar atualização dos campos relacionados".
- Quando um Cliente for excluído da Tabela Clientes, podemos configurar para que o banco de dados exclua, automaticamente, na tabela Pedidos, todos os Pedidos para o Cliente que está sendo excluído. Essa opção é conhecida como "Propagar exclusão dos registros relacionados".

Essas opções podem ser configuradas quando da definição dos relacionamentos. Estas opções não são obrigatórias, isto é, podemos optar por não atualizar ou não excluir em cascata. A Opção de "Propagar atualização dos campos relacionados" é utilizada na maioria das situações, já a opção de "Propagar exclusão dos registros relacionados" deve ser estudada caso a caso. Por exemplo, se nos quiséssemos manter um histórico com os Pedidos de cada Cliente, não utilizaríamos a opção "Propagar exclusão dos registros relacionados"; caso não nos interessasse manter um histórico dos pedidos, poderíamos utilizar esta opção. As regras de integridade referencial impedem que os utilizadores de uma base de dados quebrem acidentalmente o mapeamento entre duas colunas.

Essencialmente, a integridade referencial estabelece que um banco de dados não pode conter quaisquer valores de chaves estrangeiras incomparáveis. Uma chave estrangeira é uma coluna em uma tabela de banco de dados que contém valores também encontrados na coluna de chave primária - um identificador exclusivo que rotula uma linha na tabela - em outra tabela. Por exemplo, considere uma tabela de banco de dados chamado "departamento", na qual uma coluna chamada "dept - não" é a chave primária. Relaciona-se com uma outra tabela chamada "empregado", no qual "Dept_No" é uma chave estrangeira. Um funcionário não pode pertencer a um departamento em particular se o correspondente "Dept_No" já não existe na tabela "departamento". Se o programa para adicionar funcionários impõe a integridade referencial, qualquer tentativa de inserir um empregado em um departamento desconhecido não ocorrerá.

Além de garantir que as referências entre dados estão intactas e válidas, definir a integridade referencial em um banco de dados tem inúmeras vantagens. A integridade referencial usa código existente em um motor de banco de dados em vez de exigir aos programadores escrevam código de programa personalizado a partir do zero. Como resultado, o desenvolvimento do programa é mais rápido, menos propenso a erros e consistente em vários programas aplicativos que acessam um banco de dados.