**Mongo DB**

Nos últimos anos, vimos um interesse crescente em sistemas de gerenciamento de bancos dados que diferem do modelo relacional tradicional. No centro disso está o conceito de NoSQL, um termo usado coletivamente para denotar software de banco de dados que não usa SQL (Structured Query Language) para interagir com o banco de dados. Um dos projetos NoSQL mais notáveis até o momento é o MongoDB, uma banco de dados orientado a documentos e de software livre que armazena dados em coleções de documentos semelhantes a JSON. O que distingue o MongoDB de outros bancos de dados NoSQL é a sua poderosa linguagem de consulta baseada em documento, que torna a transição de um banco de dados relacional para o MongoDB fácil, porque as consultas são convertidas com bastante facilidade.O MongoDB é escrito em C++. Ele armazena dados dentro de documentos semelhantes a JSON (usando BSON — uma versão binária de JSON), que retém os dados usando pares de chave/valor. Um recurso que diferencia o MongoDB de outro bancos de dados de documentos é que é muito simples converter instruções SQL em chamadas de função de consulta do MongoDB. Isso facilita a migração de organizações que atualmente usam bancos de dados relacionais fazerem a migração. Ele também é muito simples de instalar e usar, com binários e drivers disponíveis para os principais sistemas operacionais e linguagens de programação.O MongoDB é um projeto de software livre, com o banco de dados em si licenciado sob a GNU AGPL (Affero General Public License) versão 3.0. Essa licença é uma versão modificada da GNU GPL que fecha uma brecha na qual as restrições de copyleft não se aplicam ao uso do software, mas apenas à sua distribuição. Isso, naturalmente, é importante em software armazenada na nuvem e geralmente não instalado em dispositivos cliente. Usar a GPL regular poderia resultar em alguém perceber que nenhuma distribuição de fato acontece e, portanto, potencialmente contornar os termos da licença.A AGPL aplica-se apenas ao próprio aplicativo de banco de dados, e não a outros elementos do MongoDB. Os drivers oficiais que permitem que os desenvolvedores conectem o MongoDB a várias linguagens de programação são distribuídos sob a Licença Apache, Versão 2.0. A documentação do MongoDB está disponível sob a licença Creative Commons.

**Bancos de dados orientados a documentos**Os bancos de dados orientados a documentos são bastante diferentes dos tradicionais bancos de dados relacionais. Em vez de armazenar dados em estruturas rígidas, como tabelas, eles os armazenam em documentos vagamente definidos. No caso de tabelas de sistemas de gerenciamento de bancos de dados relacionais (RDBMS), se for preciso acrescentar uma nova coluna, será necessário mudar a definição da própria tabela, que acrescentará aquela coluna a todos os registros existentes (embora potencialmente com valor nulo). Isso se deve ao estrito design baseado em esquema do RDBMS. No entanto, em documentos, é possível acrescentar novos atributos aos documentos individuais sem que outros sejam alterados. Isso porque o design de bancos de dados orientados a documentos costuma ser sem esquema.

Outra diferença fundamental é que bancos de dados orientados a documentos não fornecem relacionamentos estritos entre documentos, o que ajuda a manter seu design sem esquema. Isso difere muito de bancos de dados relacionais, que dependem fortemente de relacionamentos para normalizar o armazenamento de dados. Em vez de armazenar dados "relacionados" em uma área de armazenamento separada, em bancos de dados de documentos eles são integrados ao próprio documento. Isso é muit;o mais rápido do que armazenar uma referência a outro documento onde os dados relacionados são armazenados, visto que cada referência exigiria uma consulta adicional.

Contudo, se for preciso armazenar dados relacionados em separado, é possível fazer isso facilmente no MongoDB usando uma coleção separada. Outro bom exemplo é armazenar informações de pedidos de clientes em documentos do MongoDB. Geralmente isso inclui informações sobre o cliente, o próprio pedido, itens de linha do pedido e informações sobre o produto. Usando o MongoDB, provavelmente seriam armazenados os clientes, produtos e pedidos em coleções individuais, mas seriam integrados dados de itens de linha dentro do documento de pedido relevante. Seria então feito referência às coleções products e customers usando IDs em estilo de chave estrangeira, assim como faria em um banco de dados relacional. A simplicidade dessa abordagem híbrida faz do MongoDB uma escolha excelente para quem está acostumado a trabalhar com SQL. Isso posto, tire tempo e tome cuidado para decidir a abordagem que precisará ser usada em cada caso de uso individual, visto que os ganhos de desempenho podem ser significativos ao incorporar os dados dentro do documento, em vez de fazer referência a eles em outras coleções.

**Recursos:**

MongoDB é muito mais do que apenas um armazenamento básico de chaves/valores. Vamos dar uma breve olhada em algum de seus recursos:

* Binários oficiais disponíveis para Windows®, Mac OS X, Linux® e Solaris, distribuição de código de origem disponível para autoconstrução
* Drivers oficiais disponíveis para C, C#, C++, Haskell, Java™, JavaScript, Perl, PHP, Python, Ruby e Scala, com uma ampla variedade de drivers suportados pela comunidade disponíveis para outras linguagens
* Consultas JavaScript ad-hoc que permitem localizar dados usando qualquer critério de qualquer atributo de documento. Essas consultas são um espelho da funcionalidade de consultas SQL, o que torna muito fácil para desenvolvedores SQL escreverem consultas em MongoDB.
* Suporte a expressões regulares em consultas
* Os resultados da consulta em MongoDB são armazenados em cursores que fornecem muitas funções para filtragem, agregação e classificação, incluindo limit(), skip(), sort(), count(), distinct() e group().
* map/reduce -- implementação para agregação avançada
* Armazenamento de arquivos grandes usando GridFS
* Suporte a indexação de atributo semelhante a RDBMS, onde é possível criar índices diretamente em atributos selecionados de um documento
* Recursos de otimização de consulta usando dicas, planos de explicação e perfis
* Replicação mestre/escravo similar ao MySQL
* Armazenamento de objeto baseado em coleta, permitindo consulta referencial onde forem necessários dados normalizados

Fonte:

www.ibm.com/developerworks/br/library/os-**mongodb**4/