PÓS-GRADUAÇÃO EM ENGENHARIA DE DADOS

IEC PUC MINAS

Disciplina:

BANCOS DE DADOS NoSQL

Professor: Augusto Zadra

2.2 Propriedades NoSQL

INTRODUÇÃO

- Conforme verificamos, as diferenças entre as tecnologia são relacionadas à Arquitetura, modelo e distribuição de dados, além de modelos de desenvolvimento e a necessidade de altíssima performance como abordado anteriormente.
- NoSQL utilizam uma abordagem diferente para resolver estes problemas que são complexos, criando aplicativos mais simples que distribuem os recursos necessários pela rede.
- A manutenção de seus componentes arquitetônicos simples viabiliza a reutilização entre aplicativos e torna mais fácil portar seu aplicativo para novas arquiteturas.

INTRODUÇÃO

- Apesar de serem implementadas distintamente nas soluções de SGBD NoSQL, a escalabilidade e a disponibilidade são propriedades comuns, e se tornam fatores preponderantes para garantia de níveis satisfatórios de desempenho dessas soluções na manipulação de grandes volumes de dados.
- Afinal de contas esse é o maior mote da utilização de bancos de dados NoSQL nas organizações e/ou produtos assim por dizer.
- Vamos apresentar rapidamente o conceito destas características.

ESCALABILIDADE

- A escalabilidade está relacionada à capacidade de um sistema computacional qualquer satisfazer um requisito de aumento da carga de trabalho pela adição de uma quantidade proporcional de recursos, enquanto continua a satisfazer toda demanda que lhe foi atribuída.
- E quando o conjunto de dados a ser manipulado é maior do que a capacidade de recursos disponíveis (processamento, memória e armazenamento), é necessidade de criar uma condição para garantir o dimensionamento necessário para atendimento da demanda. Existem dia formas que podem ser adotadas:

Bancos de dados NoSQL

Escalabilidade

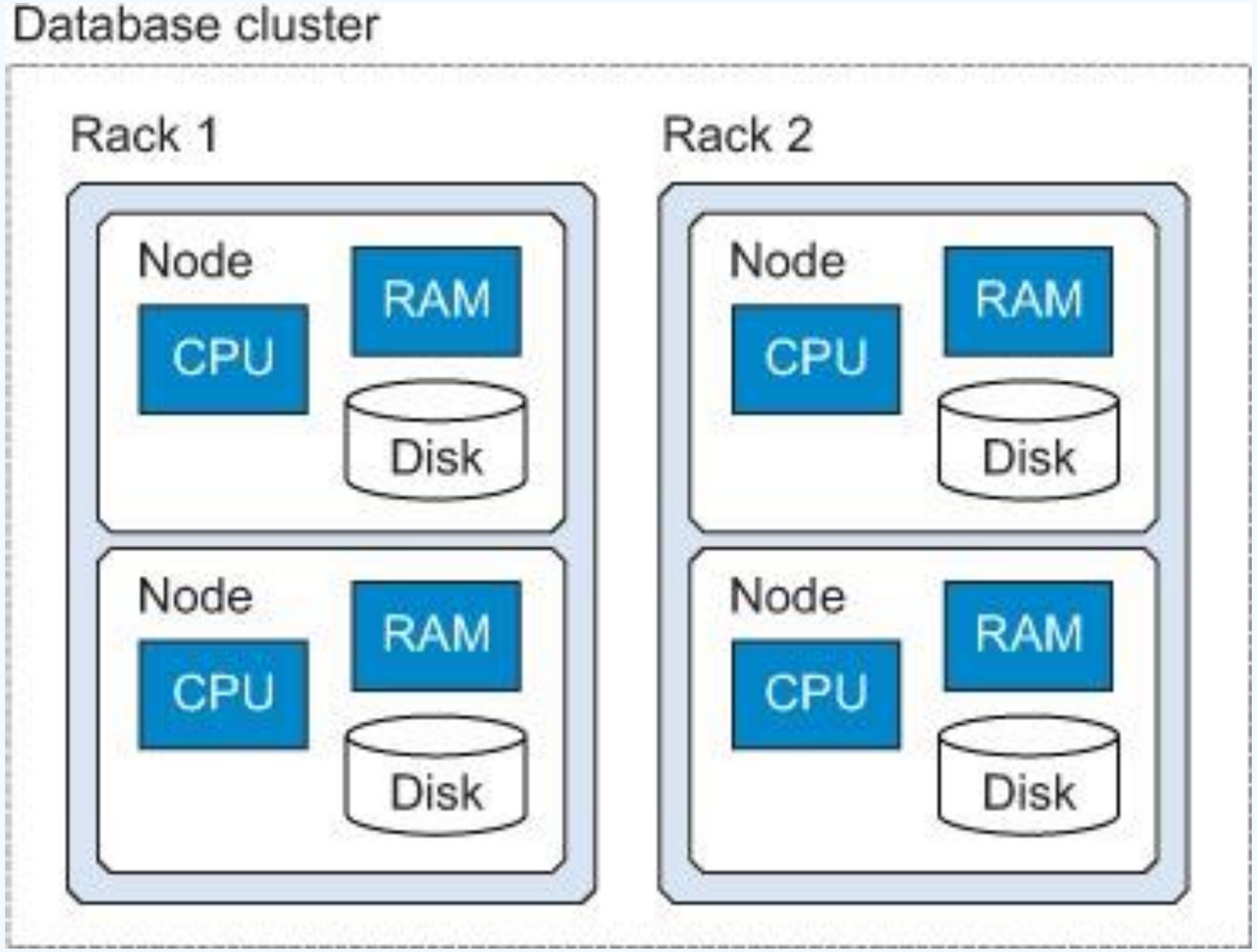
- **Vertical**: consiste no acréscimo de mais recursos de hardware ao servidor que mantem o SGBD (processamento, memória e armazenamento);

- **Horizontal**: consiste na divisão e alocação dos dados de um SGBD entre vários servidores denominados “nós”, formando um agrupamento (*cluster*) de nós, que são alocados sob demanda.

PUC Minas

Bancos de dados NoSQL

Escalabilidade



PROBLEMATIZAÇÃO

- Depois de decidir distribuir os dados, como eles devem ser distribuídos?
- A distribuição dos dados se dá através da utilização de uma chave de particionamento.
- Uma das maneiras de se fazer isto seria com uma chave e aplicar uma função *hash* conforme já verificamos.
- Outra maneira de fazer atribuições seria pegar todo o *keyspace* (espaço para armazenamento das chaves) e dividi-lo em um conjunto de intervalos.

SHARDING

- A redistribuição dos dados é feita e cada intervalo corresponde a um *shard* (fragmento) é atribuído a um determinado nó no cluster.
- A escalabilidade dos bancos NoSQL está diretamente relacionada a este conceito que traz custos operacionais mais baixos quando implantamos mecanismos de *autosharding*.
- Com este recurso os bancos de dados NoSQL minimizam o tempo de inatividade.

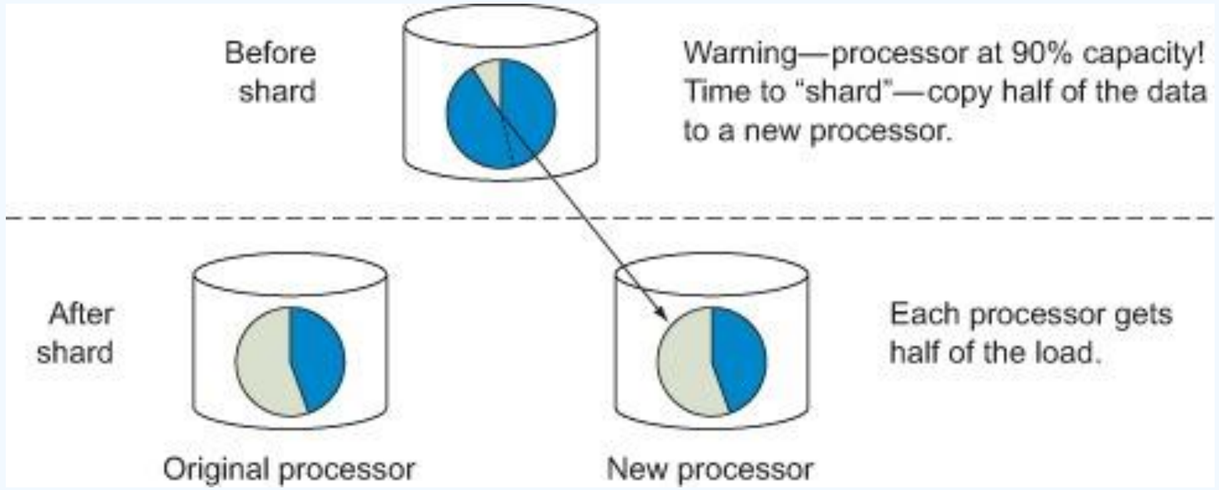
SHARDING

- A técnica de *sharding* resolve um problema antigo onde, como já dissemos a distribuição dos dados em vários nós causa indisponibilidade.
- Sistemas NoSQL fazem isso automaticamente e a partir daí um banco de dados cresce e sua tolerância para particionamento automático de dados é importante para sistemas.
- O sharding tornou-se um processo altamente automatizado em sistemas de tolerantes a falhas.

PUC Minas

Bancos de dados NoSQL

Sharding



- Quando um único processador não pode lidar com os requisitos de rendimento de um sistema é desejável mover os dados para dois sistemas, cada um com metade do trabalho.
- Muitos sistemas NoSQL têm fragmentação automática integrada, de forma que você só precisa adicionar um novo servidor.

VARIÇÕES DE ARQUITETURA NoSQL

- Os padrões de armazenamento de valor-chave, grafos, armazenamento *Bigtable* (orientado a colunas) e armazenamento de documento podem ser modificados com foco em um aspecto diferente da implementação do sistema.
- Alguns produtos NoSQL são projetados para funcionar especificamente com um tipo de memória.

PUC Minas

Bancos de dados NoSQL

Variações de arquitetura NoSQL

Memcache

- É um armazenamento de valor-chave que foi projetado especificamente para ver se os itens estão na RAM em vários servidores.
- Um armazenamento de valor-chave que usa apenas RAM é chamado de cache de RAM.
- É flexível e tem ferramentas gerais que os desenvolvedores de aplicativos podem usar para armazenar variáveis globais, arquivos de configuração ou resultados intermediários de transformações de documentos.

PUC Minas

Bancos de dados NoSQL

Variações de arquitetura NoSQL

Memcache

- Um cache de RAM é rápido e confiável e pode ser considerado como outra construção de programação, como um *array*, um mapa ou um sistema de pesquisa.
- Os armazenamentos de valores-chave residentes na RAM simples geralmente ficam vazios quando o servidor é inicializado e só podem ser preenchidos com valores sob demanda.
- As informações residentes na RAM podem ser salvas em outro sistema de armazenamento se você quiser persisti-las.

PUC Minas

Bancos de dados NoSQL

Variações de arquitetura NoSQL

Memcache

- Os sistemas SSD fornecem armazenamento permanente e são quase tão rápidos quanto a RAM para operações de leitura.
- As operações de gravação em SSDs geralmente podem ser armazenadas em grandes caches de RAM, resultando em tempos de gravação rápidos até que a RAM fique cheia.

PUC Minas

Bancos de dados NoSQL

Variações de arquitetura NoSQL

Armazenamento Distribuído

- Idealmente, o processo de distribuição de dados é transparente para o usuário, o que significa que a API não exige que você saiba como ou onde seus dados são armazenados.
- Mas saber que seu software NoSQL pode escalar e como ele faz isso é fundamental no processo de seleção de software.
- Se seu aplicativo usa muitos servidores da web, cada um armazenando em cache o resultado de uma consulta de longa duração, é mais eficiente ter um método que permita que os servidores trabalhem juntos para evitar a duplicação.

PUC Minas

Bancos de dados NoSQL

Variações de arquitetura NoSQL

- Esteja você usando NoSQL ou sistemas SQL tradicionais, a RAM é o recurso mais caro e precioso na configuração de um servidor de aplicativos.
- Se você não tiver RAM suficiente, seu aplicativo não será escalado.
- A solução usada em um armazenamento de valor-chave distribuído é criar um protocolo simples e leve que verifica se algum outro servidor tem um item em seu cache.

PUC Minas

Bancos de dados NoSQL

Variações de arquitetura NoSQL

- Em caso afirmativo, este item é rapidamente devolvido ao solicitante e nenhuma pesquisa adicional é necessária.
- O protocolo é simples: cada servidor *memcache* tem uma lista dos outros servidores *memcache* com os quais está trabalhando.
- Sempre que um servidor *memcache* recebe uma solicitação que não está em seu próprio cache, ele verifica com os outros servidores de mesmo nível, enviando-lhes a chave.

PUC Minas

Bancos de dados NoSQL

Variações de arquitetura NoSQL

- Na prática, quase todos os sistemas NoSQL distribuídos podem ser configurados para armazenar itens em cache em dois ou três servidores diferentes.
- A decisão de qual servidor armazena qual chave pode ser determinada pela implementação de um sistema round-robin simples ou de distribuição aleatória.

PUC Minas

Bancos de dados NoSQL

Variações de arquitetura NoSQL

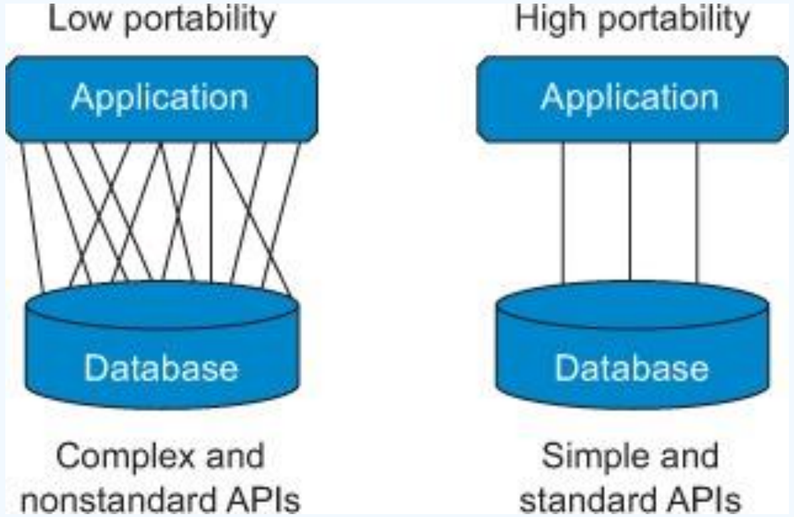
- Para tornar esta implementação de valor-chave mais fácil e eficiente, seu armazenamento pode ser modificado para incluir informações adicionais na estrutura da chave para indicar que o par de valor-chave está associado a outro par de valor-chave, criando uma coleção ou estruturas de propósito usadas para agrupar recursos.
- Embora cada sistema de armazenamento de valor-chave possa chamá-lo de algo diferente (como pastas, diretórios ou depósitos), o conceito é o mesmo.

PUC Minas

Bancos de dados NoSQL

Variações de arquitetura NoSQL

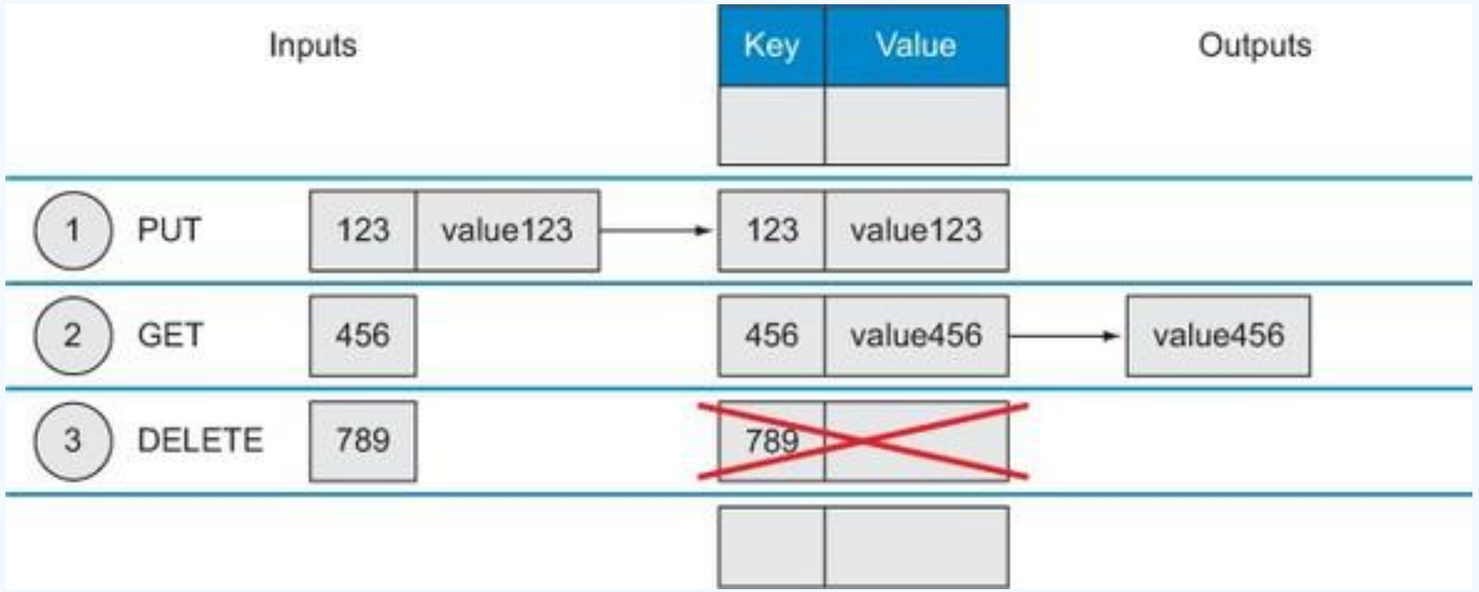
	Key	Value
Image name →	image-12345.jpg	Binary image file
Web page URL →	http://www.example.com/my-web-page.html	HTML of a web page
File path name →	N:/folder/subfolder/myfile.pdf	PDF document
MD5 hash →	9e107d9d372bb6826bd81d3542a419d6	The quick brown fox jumps over the lazy dog
REST web service call →	view-person?person-id=12345&format=xml	<Person><id>12345</id .</Person>
SQL query →	SELECT PERSON FROM PEOPLE WHERE PID="12345"	<Person><id>12345</id .</Person>

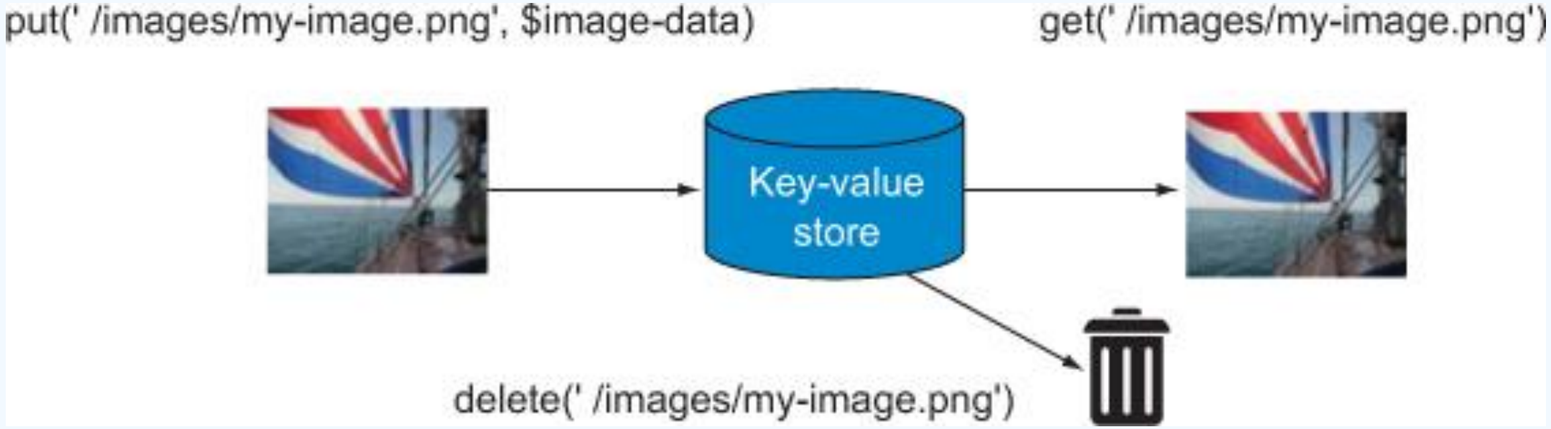


PUC Minas

Bancos de dados NoSQL

Variações de arquitetura NoSQL





PUC Minas

REFERÊNCIAS BIBLIOGRÁFICAS

DIANA, M. de;; GEROSA, M. A. NOSQL na Web 2.0: Um estudo comparativo de bancos não-relacionais para armazenamento de dados web 2.0. In: Workshop de Teses e Dissertações em BD - WTDB, 9., 2010, Belo Horizonte. Anais... Belo Horizonte: SBC, 2010.

LI, Y.;; MANOHARAN, S. A performance comparison of SQL and NoSQL databases. In: IEEE Pacific Rim Conference on Communications, Computers and Signal Processing - PACRIM, 14., 2013, Victoria, B.C., Canadá. Proceedings... IEEE, 2013.

LÓSCIO, B. F.;; OLIVEIRA, H. R. de;; PONTES, J. C. de S. NoSQL no desenvolvimento de aplicações web colaborativas. In: Simpósio Brasileiro de Sistemas Colaborativos – SBSC, 8., 2011, Paraty (RJ). Anais... Paraty: SBC, 2011.

VIEIRA, M. R. et al. Bancos de Dados NoSQL: conceitos, ferramentas, linguagens e estudos de casos no contexto de Big Data. In: Simpósio Brasileiro de Bancos de Dados, 27., 2012, São Paulo. Anais... São Paulo: SBC, 2012.