

Investigando o Uso de Bancos de Dados não Convencionais para Gerenciar Informações da Administração Pública

Diogo A. P. Wanzeller

Universidade de Brasília

20 de Dezembro de 2013

Overview

- 1 Revisão Teórica
- 2 O Projeto
- 3 Resultados
- 4 Considerações Finais e Trabalhos Futuros
- 5 Cronograma
- 6 Referências

BIG DATA ?

- A sociedade está lidando com uma quantidade de dados cada vez maior.
- Os dados precisam de tratamentos;
- Os dados geram informações importantes.
- A informação é o diferencial
- Massa de dados de grande volume, velocidade e variedade.

Relacional x NoSQL

- O modelo relacional representa o banco de dados como uma coleção de relações;
- Validação, verificações e garantias de integridade, controle de concorrências...
- Problemas causados pelo layout rígido.
- PostgreSQL, MySQL, Oracle, MS SQL, DB2

Relacional x NoSQL

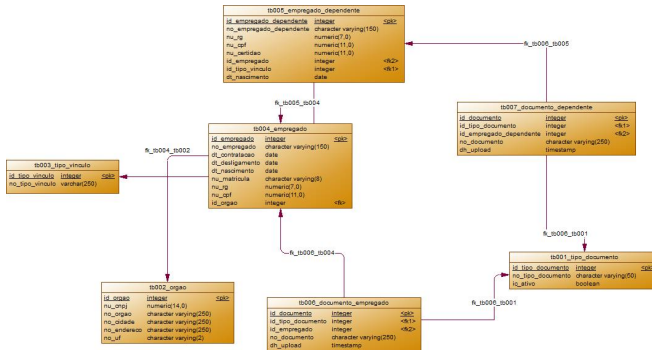


Figure: Modelo Relacional

Relacional x NoSQL

- Armazenamento de dados de forma não relacional;
- Schema free;
- Chave-Valor / Orientados a Documentos / Orientados a Colunas / Baseados em Grafos;
- Google, Amazon, Facebook...
- MongoDB, Cassandra, NEO4j, Redis

Relacional x NoSQL

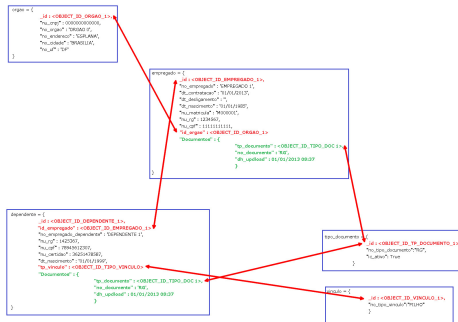


Figure: Modelagem orientada a documentos implementada no protótipo.

O Projeto

- Grande número de pastas funcionais físicas geram custos para manter a qualidade dos arquivos permanentes;
- Digitalização das pastas funcionais;

O Projeto

- AFD - Criação de um dossiê, em mídia digital;
- Fonte Primária de informações cadastrais do Servidor Público Civil Federal. *citar site do SIGEPE*

Problema

- Determinar se um banco de dados NoSQL é indicado para um caso como o citado e se temos um tipo de banco mais adequado.

Motivação

- Tema relativamente novo;
- Caso Real;
- Mercado -> Academia;
- Novas Tecnologias;

Objetivo Geral

- Comparar os modelos relacional e não relacional (Orientada a Documento) de armazenamento para o contexto do Assentamento Digital Funcional (AFD).

Objetivos Específicos

- Compreender, abstrair e modelar os conceitos e operações do AFD;
- Modelar os conceitos do AFD utilizando a estratégia relacional;
- Modelar os conceitos do AFD utilizando a estratégia orientada a documentos;
- Implementar os modelos em SGBDs relacionais e orientados a documentos;

Objetivos Específicos

- Implementar uma arquitetura SOA para realizar as operações do AFD, utilizando os dois bancos para a persistência;
- Projetar, implementar e realizar testes de desempenho que nos permitam tirar conclusões sobre quais dos modelos são mais propícios para o armazenamento dos dados do AFD.

Resultados Esperados

- Obter informações o bastante para escolher o banco de dados mais indicado para esse caso e semelhantes.

Teste de Software

- Qualidade de software;
- Teste: atividades nas quais um sistema ou um componente é executado sob determinadas condições e os resultados são observados ou gravados, e uma avaliação é feita observando determinado comportamento do sistema ou do componente;
- Vários tipos: Testes de sistema, usabilidade, back-to-back, carga, performance.
- Com a complexidade e tamanho das aplicações os testes manuais se tornaram inviáveis.
- Solução: Testes automatizados.

Protótipo AFD

- Arquitetura SOA;
- Principais capacidades para manter os dados do AFD;
- Inserção e consulta de órgãos, empregados, dependentes, documentos;
- Empregados ativos;
- Desligamento de empregados e exclusão de dependentes.

Arquitetura

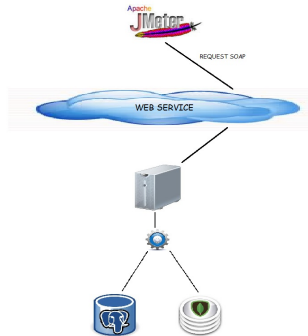


Figure: Arquitetura de Testes

Plano de testes

- Criação de um plano de testes para as principais capacidades do protótipo;
- Plano de testes parametrizáveis : Usuários virtuais, carga de dados via arquivo CSV;
- Monitoramento dos resultados;
- Geração de gráficos;

Execução dos testes

- Criação de um plano de testes para as principais capacidades do protótipo;
- Plano de testes parametrizáveis : Usuários virtuais, carga de dados via arquivo CSV;
- Monitoramento dos resultados;
- Geração de gráficos;
- As capacidades foram testadas com 10, 100 e, quando possível, com 500 usuários simultâneos;

Execução dos testes

Os testes foram realizados em uma máquina física com as seguintes configurações:

- Sistema Operacional: Debian GNU/Linux 6.0
- Processador: Intel Pentium Quad Core
- Quantidade de Memória RAM: 4 GB
- MongoDB: Versão 2.4.0 padrão
- PostgreSQL: Versão 8.4.16 padrão
- Driver Python MongoDB: pymongo
- Driver Python PostgreSQL: psycopg

Massa de dados

- Quantidade de unidades pagadoras (órgãos): 10
- Quantidade de empregados por órgão: 100
- Quantidade de dependentes por empregado: 2
- Quantidade de documentos por empregado: 5
- Quantidade de documentos por dependente: 2
- Quantidade de dependentes excluídos: 250
- Quantidade de empregados desligados: 250

Métrica

- Tempo de Resposta [molineux][raj jain]

Resultados

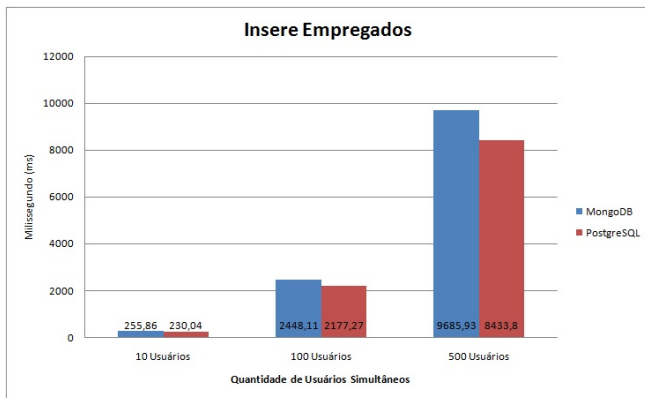


Figure: Resultados - Inserir Empregados

Resultados

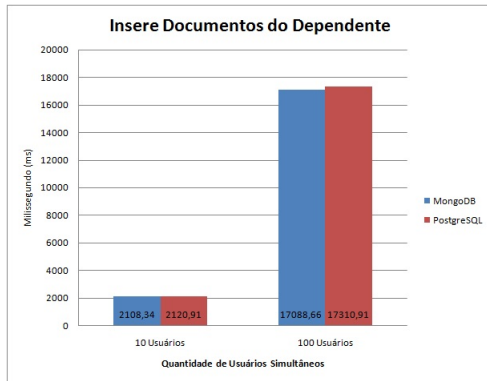


Figure: Resultados - Inserir Documento do Dependente

Resultados

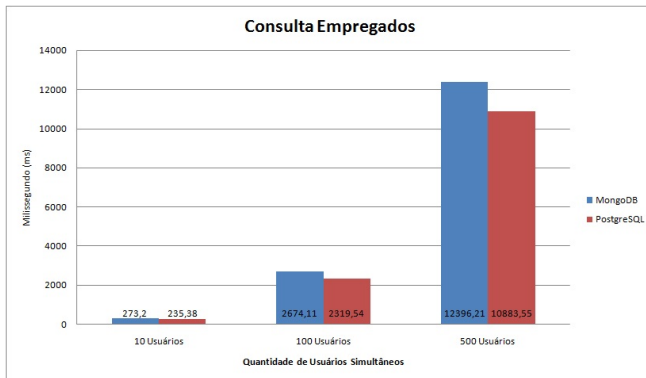


Figure: Resultados - Lista Empregados

Resultados

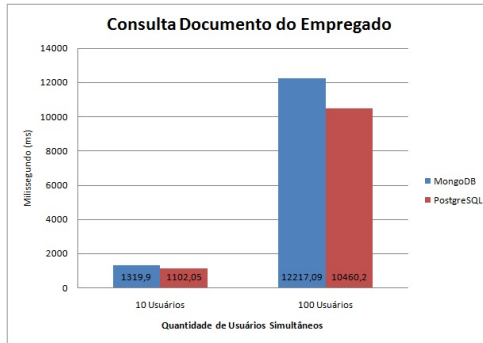


Figure: Resultados - Lista Documentos do Empregado

Resultados

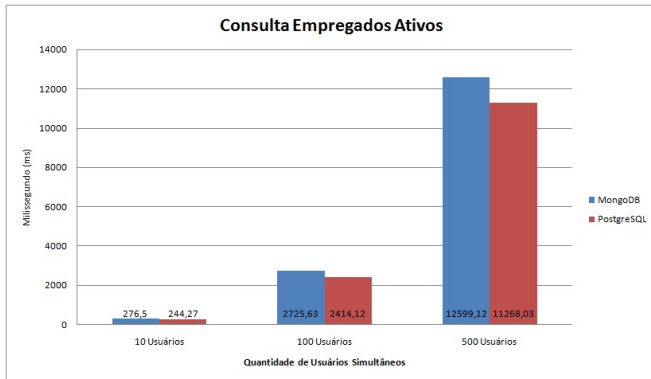


Figure: Resultados - Consulta Empregados Ativos

Resultados

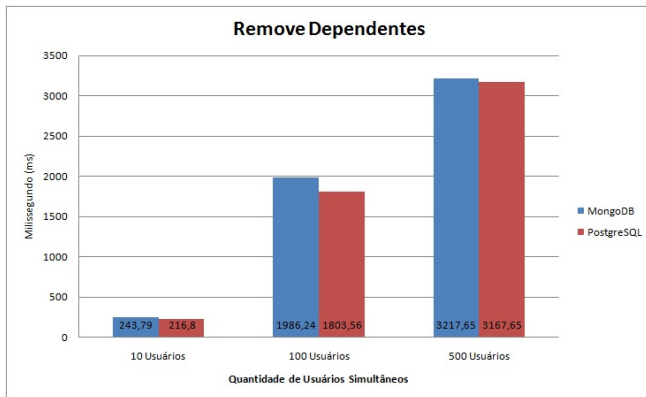


Figure: Resultados - Remove Dependentes

Resultados

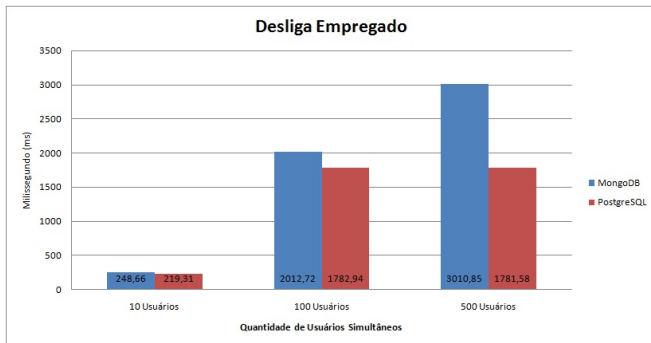


Figure: Resultados - Desliga Empregado

Resultados

Ao final dos testes pode-se verificar que nenhum banco de dados foi consideravelmente mais veloz que o outro e que, para praticamente todos os testes realizados, o PostgreSQL se mostrou mais rápido. Dessa maneira, para o cenário de manutenção dos dados do AFD, com a arquitetura, massa de dados e modelagem utilizadas, não é vantajoso utilizar o MongoDB para a persistência dos dados, visto que, mesmo com todos os recursos de segurança e controle de transações oferecidos pelo PostgreSQL, ele ainda continua sendo mais performático.

Considerações Finais

- Cenários definido;
- Tanto o PostgreSQL quanto o MongoDB possuem seu lugar no mercado; item Fontes no GitHub;

Trabalhos Futuros

- Outra modelagem orientada a documentos com uma maior utilização de subdocumentos;
- Uso de uma arquitetura mais robusta item Expandir os testes para outros bancos de dados como o MySQL, Cassandra, HBase...

Tarefas	JUL	AGO	SET	OUT	NOV	DEZ
Escrita da Monografia	x	x	x	x	x	x
Compreender AFD	x	x				
Modelagem Relacional			x			
Modelagem Orientada a Documentos			x	x		
Implem. Mod. de Dados			x	x		
Arquitetura SOA		x	x	x	x	
Projeto e Implem. Testes			x	x	x	
Execução dos Testes					x	x

Bibliografia I

- E. Hewitt. Cassandra: The Definitive Guide. Definitive Guide Series. O'Reilly Media, 2010.
- S. Tiwari. Professional NoSQL. Wrox Programmer to Programmer. Wiley, 2011.
- Mongodb oficial site. <http://www.mongodb.org/>

Bibliografia II

- R. Hecht and S. Jablonski. Nosql evaluation: A use case oriented survey. In Cloud and Service Computing (CSC), 2011 International Conference on, pages 336 - 341, dec. 2011.

Bibliografia III

- Vinayak R. Borkar, Michael J. Carey, and Chen Li. Big data platforms: What's next? XRDS, September 2012.
- Ricardo W. Brito. Banco de dados nosql x sgbd's relacionais: Análise comparativa. 2010.

Bibliografia

Obrigado pela atenção!

Obrigado!