UTILIZAÇÃO DE BANCOS DE DADOS RELACIONAIS PARA O ARMAZENAMENTO DE DADOS RDF

Tiago Heinrich[1] Rebeca Schroeder Freitas [2]

[1] Acadêmico do Curso de Bacharelado em Ciências da computação -CCT - bolsista PIPES
[2] Orientadora - Departamento de Ciências da Computação – CCT – rebeca.schroeder@udesc.br

Palavras-chave: RDF. NoSQL. Relacional

O crescimento do volume de informações encontradas na web é cada dia maior, mas os sistemas responsáveis pelo seu armazenamento e consulta, em geral, não acompanham este mesmo crescimento. Um dos formatos de dados que oferece um grande crescimento atualmente é o RDF *(Resource Description Framework)*. O modelo RDF é considerado como padrão para representar dados na Web 3.0. Através de seu formato de triplas, dados RDF são definidos como uma sentença constituída por *sujeito-predicado-objeto*.

Em geral, sistemas NoSQL têm sido adotados para o armazenamento de fontes RDF em virtude do elevado volume de dados que algumas fontes apresentam. Entretanto, o uso de sistemas deste tipo acrescentam uma maior complexidade ao desenvolvimento de aplicações pois, em geral, estes sistemas não apresentam algumas das características de um Sistema Gerenciador de Banco de Dados (SGBD) relacional[1]. Desta forma, para repositórios com volumes de dados de dimensões convencionais o uso de SGBDs relacionais se torna aplicável. No contexto de SGBDs Relacionais que suportam RDF, existem dois tipos de modelo aplicados. O primeiro, é conhecido como *triple-store*. Um *triple-store* define um banco RDF através de uma única relação composta pelos campos sujeito, predicado e objeto. Nesta relação, as tuplas correspondem a triplas RDF. O segundo modelo corresponde à utilização de conhecimentos sobre a estrutura RDF para definição de um esquema relacional baseado em tipos. Neste caso, cada tipo representa uma relação da base de dados.

Este trabalho tem por objetivo apresentar um estudo experimental que compara o desempenho em consultas utilizando os dois tipos de modelos aplicados por sistemas relacionais para RDF. Este estudo compara o *triple-store* Jena TDB Fuseki [3] como representante do primeiro modelo, e o ntSQL [1] como representante do segundo modelo. Os experimentos foram baseados no Berlin SPARQL Benchmark (BSBM)[2] através de consultas SPARQL e seu gerador de bases de dados.

A máquina utilizada para o experimento possui 4G de memória RAM, um processador AMD phenom II x4 e o sistema operacional linux-ubuntu 14.04. Para a realização do experimento, foram utilizadas 11 consultas SPARQL do BSBM. Para verificar a escalabilidade dos sistemas comparados utilizaram-se bases com tamanhos variados. No BSBM o fator de escala da base corresponde à quantidade de produtos de um sistema de *e-commerce*. No caso do experimento foram utilizadas bases de 100, 200, 400, 800, 1600 e 2000 produtos. Cada consulta foi executada dez vezes para cada tamanho de base, dos quais foram retirados para cada base a média e a

mediana. A Fig. 1 e Fig.2 apresentam, respectivamente, a mediana e a média referentes aos tempos de resposta dos 2 sistemas para a consulta 3 do BSBM.

Observa-se em todas as consultas que o Fuseki apresenta um desempenho muito inferior em comparação com o ntSQL. O ntSQL provou ter um melhor tempo para efetuar as consultas, demonstrando um desempenho com um crescimento quase constante, não possuindo nenhuma variação drástica com o crescimento do tamanho do banco. Em relação ao Fuseki, seu crescimento apresenta picos de acordo com o crescimento do tamanho dos bancos, com aumento do tempo de resposta muito superior ao ntSQL. Acredita-se que esta diferença possa ser explicada pelo modelo de armazenamento utilizado por *triple-stores* ao processar consultas com diversos padrões de triplas. Ao colocar todas as triplas em uma mesma tabela, além de gerar grandes arquivos para as relações, surge a necessidade da execução de diversas auto-junções para a recuperação dos diversos padrões de triplas das consultas. O modelo empregado pelo ntSQL mostra-se mais adequado neste sentido por distribuir os dados em diferentes relações, agrupando-os de acordo com seus tipos.

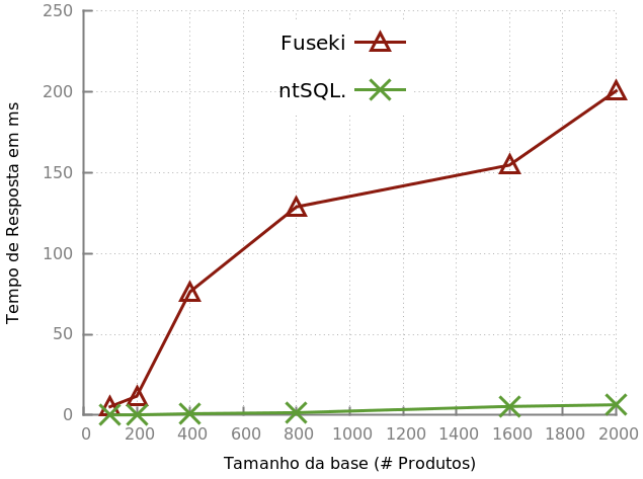

Fig. 1 *Mediana (Consulta 3)*

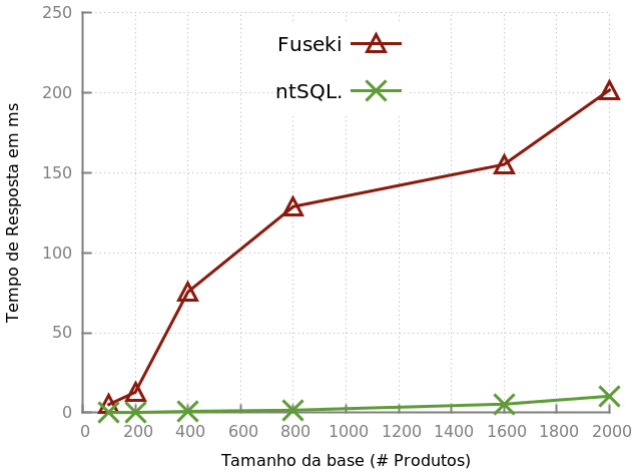

Fig. 2 *Média (Consulta 3)*

Este trabalho desenvolveu um estudo experimental que compara dois modelos de armazenamento de dados RDF em sistemas relacionais através dos sistemas Jena-TDB Fuseki e ntSQL. Como esperado, o Jena-TDB Fuseki apresentou um desempenho inferior ao ntSQL, atestando as desvantagens do uso de *triple-stores* já apontadas por outros trabalhos na literatura. O melhor desempenho do ntSQL pode ser atribuído ao uso de um esquema relacional mais robusto do que os utilizados por *triple-stores*. Como trabalho futuro, pretende-se envolver outros sistemas na avaliação e outros *benchmarks*. Além disto, pretende-se desenvolver uma análise mais detalhada que possa justificar o desempenho dos sistemas através de características de consultas e do esquema de banco de dados.

[1] Bayer et al. 2014 Bayer, F. R., Nesi, L. L., and Schroeder, R. **ntSQL: Um Conversor de Documentos RDF para SQL**. In Anais da Escola Regional de Banco de Dados. SBC, 2014.
[2] Christian Bizer, Andreas Schultz: **The Berlin SPARQL Benchmark**. In: International Journal on Semantic Web & Information Systems, Vol. 5, Issue 2, Pages 1-24, 2009.
[3] JENA, Apache. **Apache Jena Fuseki**. Disponível em: https://jena.apache.org/documentation/fuseki2. Acesso em: 25 jul. 2016.