Administração de Bases de Dados

Engenharia Informática – Universidade do Minho

Trabalho Prático – 2022/2023

Os grupos de trabalho devem ser constituídos por 5 (cinco) elementos, todos inscritos na Unidade Curricular. O resultado do trabalho é um relatório escrito. O relatório deve omitir considerações genéricas sobre as ferramentas utilizadas, focando a apresentação e justificação dos objetivos atingidos. A entrega do relatório é feita na área da Unidade Curricular no *e-Learning*. O relatório deve identificar também o nome e número de todos os elementos do grupo. A data limite é 26 de maio de 2023.

1 Contexto

O trabalho prático consiste na configuração, otimização, e avaliação do *benchmark* que usa dados IMDb[1] e contém operações transacionais e analíticas. O benchmark está disponível na plataforma *e-Learning*.

A componente transacional simula uma pequena parte de um serviço de *streaming* de video, baseado no dataset IMDb juntamente com informação extra relativa aos utilizadores. Este *benchmark* fornece as seguintes operações:

- `addNewTitleToList` – adiciona um título (filme, episódio, vídeo, etc.) à lista de títulos "para ver" de um utilizador;
- `getWatchListInformation` – obtém a informação dos títulos da lista "para ver" de um utilizador;
- `viewTitle` – adiciona informação de visualização de um título ao "histórico" de um utilizador, como a data e a duração. Caso um utilizador veja um título na sua totalidade, este é removido da lista "para ver';
- `rateTitle` – adiciona ao histórico de um utilizador uma classificação relativa a um título visto;
- `searchTitles` – pequisa títulos por nome.

As interrogações analíticas simulam operações estatísticas e de *data warehousing* sobre o conjunto de dados IMDb + informação de utilizadores.

Apesar de apenas as tabelas com os dados dos utilizadores sofrerem modificações com a carga transacional, devem considerar que na prática seriam constantemente adicionados novos títulos, pessoas, episódios, etc. Contudo, podem assumir que um título e as suas respetivas informações, depois de adicionados, não são mais alterados.

[1] https://www.imdb.com/interfaces/

2 Objetivos

A configuração de referência é uma máquina virtual na *Google Cloud* do tipo E2, 8 vCPUs, 16 GB RAM, disco SSD 500 GB, e sistema operativo Linux Ubuntu 22.04 LTS x86/64. Usando esta configuração devem:

1. Otimizar o desempenho das interrogações analíticas no PostgreSQL.
2. Exportar os dados de PostgreSQL para ficheiros e otimizar o desempenho das interrogações analíticas no Spark.
3. Otimizar o desempenho da carga transacional.

A realização destes objetivos deve considerar **redundância** (e.g., índices e vistas materializadas), **paralelismo**, **parâmetros de configuração**, e até **código SQL/Java**. Recomenda-se primeiro a otimização através de redundância, paralelismo, e/ou código, e só depois a otimização através dos parâmetros de configuração de modo a obter diferenças mais significativas neste último ponto.

3 Notas

- Como medidas de desempenho da carga transacional deve considerar-se principalmente o **debito máximo** atingível. Nas operações analíticas, deve considerar-se o **tempo de resposta**.
- Poderão modificar as interrogações SQL e o código Java. Devem explicar no relatório em que medida essas alterações preservam o funcionamento da aplicação original.
- Em cada um dos objetivos, não poderão considerar todas as otimizações possíveis nas suas várias combinações... Devem focar-se nas que consideram mais prometedoras e que mais vos interessam, justificando no relatório essas opções.
- No caso de não obterem melhorias de desempenho, devem explicar porque é que a configuração de referência já era ótima. Por outro lado, a simples apresentação de uma melhoria de desempenho, não justificada, não é muito interessante.
- A utilização de ferramentas de monitorização e diagnóstico do PostgreSQL, do Spark, e do sistema operativo (e.g., pgbadger, iostat, psutil) valoriza o trabalho.
- Devem considerar uma instalação nativa para o PostgreSQL e uma instalação virtualizada (Docker) para o cluster Spark.
- A automatização da instalação e execução do *benchmark* permitirá obter resultados em maior quantidade e uma análise mais profunda, valorizando o trabalho.
- Devem também procurar estratégias para poupar recursos na *Google Cloud*, por exemplo, armazenando os dados em *Cloud Storage* e reutilizando-os, em vez de re-executar o `load.py`. Considerem as opções mais baratas na *Google Cloud* (i.e., regiões nos EUA, *spot*, ...) e de **não deixar máquinas virtuais ativas a consumir recursos desnecessariamente!**