

Tecnologias de Banco de Dados

Msc. Eliezio Soares
eliezio.soares@ifrn.edu.br



INSTITUTO FEDERAL
Rio Grande do Norte

Campus
Carrais Novos

Teste de Carga em Banco de Dados

O **Apache JMeter™** é um software *open source*, 100% projetado em Java para testes funcionais e medir performance. Foi projetado originalmente para testes de aplicações web, mas tem sido expandido para outras funções de teste.

É possível elaborar um plano de testes para um servidor de banco de dados, instanciando:

- Um número X de usuários;
- N requisições para o servidor de banco de dados;
- Cada usuário poderá executar seus testes Y vezes.

Dessa forma, o número total de requisições equivale a **JDBCrequests = X * N * Y**.

Exemplo:

- X = 50
- N = 2
- Y = 100
- JDBCrequests = 10.000

Instalação

1. Baixe o JMeter em:
 - a. http://mirror.nbtelecom.com.br/apache//jmeter/source/apache-jmeter-5.0_src.tgz
2. Descompacte-o e copie para /opt
3. Baixe o driver JDBC para o PostgreSQL:
 - a. <https://jdbc.postgresql.org/download/postgresql-42.2.5.jar>
4. Copie o jar do driver JDBC para /opt/apache-jmeter-5.0/lib/
5. Execute o jmeter indo em /opt/apache-jmeter-5.0/bin:
 - a. `java -jar ApacheJMeter.jar`

Criando usuários - Thread Group

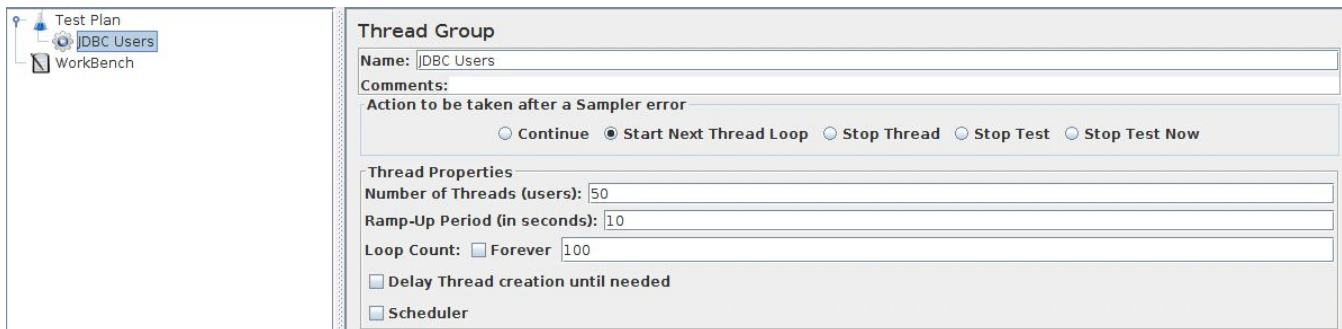
Para adicionar usuários ao plano de testes, adiciona-se o elemento Thread Group.

Name: Um nome descritivo para o grupo de usuários;

Number of Threads (users): Quantidade de usuários simulado no grupo;

Rum-Up period (in seconds): Intervalo de tempo que o JMeter levará para iniciar todos os usuários (Tempo / Usuários);

Loop Count: Quantidade de repetições do teste;

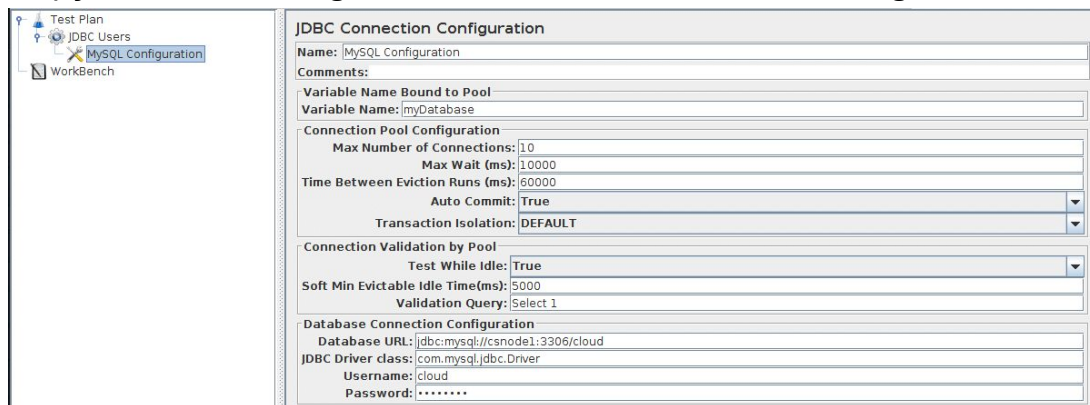


Definindo as tarefas - JDBC Connection

Para definir as tarefas que cada usuário deve executar, especifica-se uma requisição JDBC (Java DataBase Connectivity). Em um plano de testes pode-se adicionar N requisições.

Para a execução da tarefa, nesse caso uma requisição JDBC ao servidor de banco de dados, é necessário especificar os parâmetros de conexão com tal servidor.

1. Selecionando o elemento equivalente ao grupo de usuários, clique com o botão direito do mouse e em seguida na opção **ADD -> Config Element -> JDBC Connection Configuration**.



Definindo as tarefas - JDBC Connection

- **Variable name:** Será a referência para o pool de conexões e precisa ser único.
- **Database URL:** **`jdbc:postgresql://localhost:5432/academico`**
- **JDBC Driver class:** **`org.postgresql.Driver`**
- **Username:** Nome de usuário para autenticação no servidor de banco de dados.
- **Password:** Senha de acesso ao servidor de banco de dados.

JDBC Connection Configuration

Name: MySQL Configuration
Comments:

Variable Name Bound to Pool
Variable Name: myDatabase

Connection Pool Configuration

Max Number of Connections: 10
Max Wait (ms): 10000
Time Between Eviction Runs (ms): 60000
Auto Commit: True
Transaction Isolation: DEFAULT

Connection Validation by Pool

Test While Idle: True
Soft Min Evictable Idle Time(ms): 5000
Validation Query: Select 1

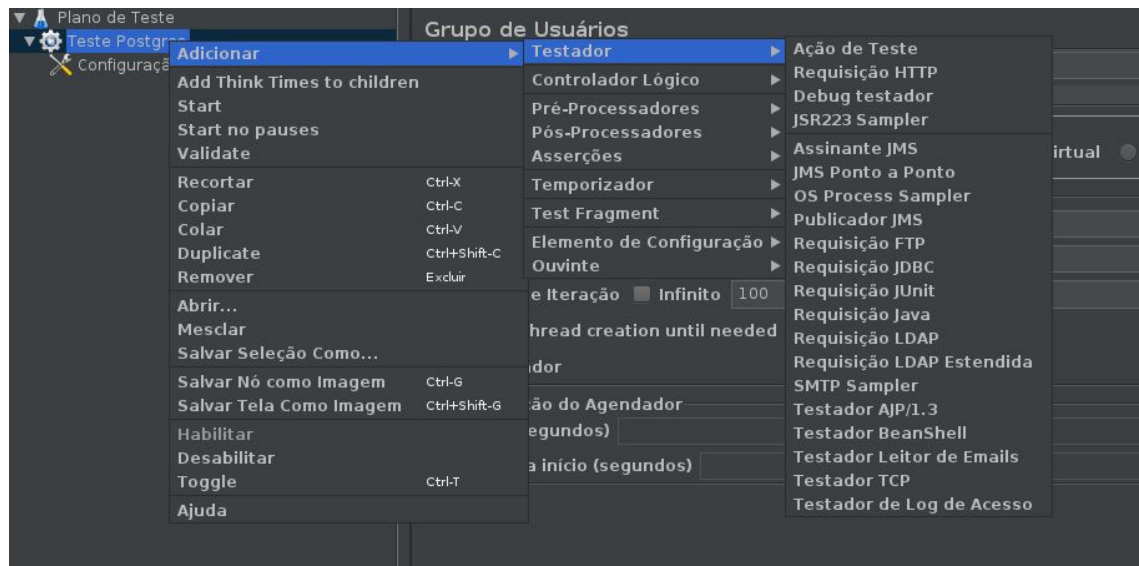
Database Connection Configuration

Database URL: jdbc:mysql://cnode1:3306/cloud
JDBC Driver class: com.mysql.jdbc.Driver
Username: cloud
Password:

Definindo as tarefas - JDBC Request

Para a execução da tarefa, cria-se uma nova requisição JDBC ao servidor de banco de dados:

1. Selecionando o elemento equivalente ao grupo de usuários, clique com o botão direito do mouse e em seguida na opção **ADD -> Sampler (testador) -> JDBC Request**.



Definindo as tarefas - JDBC Request

- **Name:** Nome da tarefa (requisição);
- **Variable Name:** Nome da variável da conexão JDBC a ser utilizada;
- **SQL Query String:** Comando SQL a ser executado;

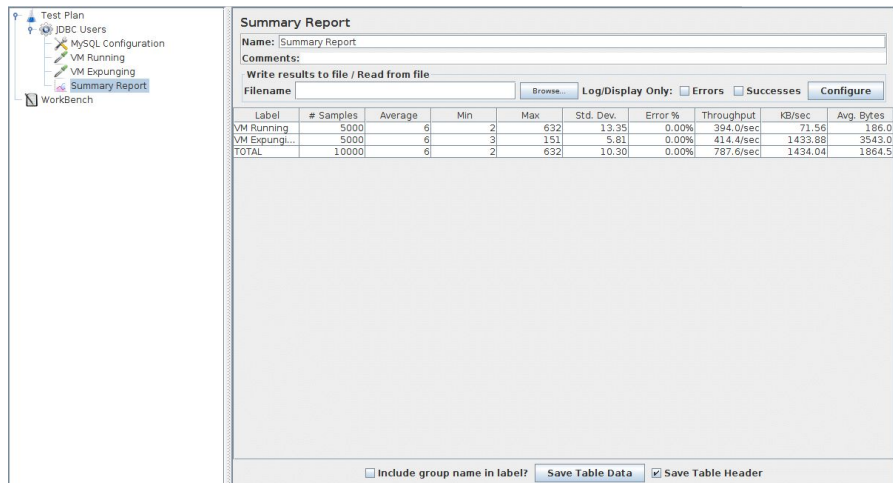
The screenshot shows the configuration interface for a JDBC Request task. On the left, a tree view shows the hierarchy: Test Plan > JDBC Users > MySQL Configuration > JDBC Request. The main panel is titled 'JDBC Request' and contains the following fields:

- Name:** JDBC Request
- Comments:** (empty text area)
- Variable Name Bound to Pool:** Variable Name: (empty text field)
- SQL Query:** Query Type: Select Statement (dropdown menu)
- Query:** A large text area with a yellow highlight at the top, containing the number '1' in the first column.
- Parameter values:** (empty text field)
- Parameter types:** (empty text field)
- Variable names:** (empty text field)
- Result variable name:** (empty text field)
- Query timeout (s):** (empty text field)
- Handle ResultSet:** Store as String (dropdown menu)

Summary Report

O último elemento a ser adicionado é um ouvinte (*listener*). Esse elemento é responsável por armazenar os resultados das requisições e apresentá-los.

1. Selecionando o elemento equivalente ao grupo de usuários, clique com o botão direito do mouse e em seguida na opção **ADD -> Listener (Ouvinte) -> Summary Report**.



The screenshot shows the 'Summary Report' configuration window. The left sidebar shows a tree view with 'Test Plan' expanded, containing 'JDBC Users', 'MySQL Configuration', 'VM Running', 'VM Expunging', and 'Summary Report'. The main window has the following configuration:

Summary Report
Name: Summary Report
Comments:
Write results to file / Read from file
Filename: Log/Display Only: Errors Successes

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	KB/sec	Avg. Bytes
VM Running	5000	6	2	632	13.35	0.00%	394.0/sec	71.56	186.0
VM Expunging...	5000	6	3	151	5.81	0.00%	414.4/sec	1453.68	3543.0
TOTAL	10000	6	2	632	10.30	0.00%	787.6/sec	1434.04	1864.5

At the bottom, there are checkboxes for 'Include group name in label?' (unchecked), 'Save Table Data' (checked), and 'Save Table Header' (checked).

Atividade

1. **Elabore dois planos de testes com o JMeter:**

- Cada plano de testes deve ter uma configuração específica, uma com maior TPS (observar os parâmetros do plano de testes) do que a outra;

2. **Monitore os testes com as ferramentas apropriadas:**

- Top
- Vmstat
- iostat
- Pg_activity
- pgAdmin

3. **Elabore um relatório comparando os resultados observados nos dois testes.**

Bibliografia Utilizada

CAIUT, Fábio. Administração de banco de dados. 1ª Edição. Rio de Janeiro. RNP/ESR, 2015.

PostgreSQL 10.5

Documentation. Disponível em:
<<https://www.postgresql.org/docs/10/static/index.html>>. Acesso em 09 Ago. 2018.

