

# REFERÊNCIA

## Compatibilidade

Liquibase 3.5.5

### Conectores

- PostgreSQL - 42.2.5
- Oracle - Ojdbc7
- MySQL - 5.1.45
- SQLServer - 4.2

### Notas

- O SGBD referenciado na string de conexão será utilizado para carregar o conector a ser utilizado

## Origem do código fonte

Variável	Descrição	Exemplo
SOURCE_METHOD	Origem do código fonte	<b>GIT</b> ou <b>VOLUME</b>

### Git

O clone do repositório, na imagem será feito em /opt/source

Variável	Descrição	Valor Padrão
GIT_EMAIL	E-mail do usuário responsável pelo commit ( será o email do usuário configurado no gitconfig )	N/A
GIT_URL	URL do repositório	<a href="https://github.com/integr8/integr8/alpine-liquibase-helper/">https://github.com/integr8/integr8/alpine-liquibase-helper/</a>
GIT_USER	Usuário utilizado para autenticação no repositório	N/A
GIT_PASS	Senha utilização para autenticação no repositório	N/A

### Volume

Caso os procedimentos serão executados em um código já clonado, o root do repositório deverá ser referenciado como um volume em /opt/source

*Exemplo: Utilização da imagem com volume*

```
docker run --rm \
-e 'SOURCE_METHOD=VOLUME' \
-e 'LIQUIBASE_DB_URL=jdbc:postgresql://172.17.0.3:5432/db' \
-e 'LIQUIBASE_DB_SCHEMA=public' \
-e 'LIQUIBASE_DB_USER=user' \
-e 'LIQUIBASE_DB_PASS=password' \
-v $(pwd)/source:/opt/source \
integr8/alpine-liquibase-helper status
```

□ Referencia ao volume, sendo que a localização atual no host é na raiz do repositório do projeto.

# Instruções do Liquibase

## Generate

O propósito desta funcionalidade é extrair os ChangeSets

Table 1. Parâmetros Opcionais

Variável	Descrição	Valor Padrão
CHANGELOG_FILE	Arquivo que lista todos os ChangeSets pertencentes ao sistema	changelog.xml
CHANGESET_PATH	Diretório que guarda os changesets utilizados pelo ChangeLog	\${dirname \${CHANGELOG_FILE}}/changeset
CHANGESET_FILE	Nome padrão dos ChangeSets	changeset_YYYY-MM-DD_HH-MM.xml
LIQUIBASE_AUTHOR	Autor que será referenciado nos ChangeSets	liquibase
LIQUIBASE_WITH_DATA	Flag para habilitar a inclusão dos dados no changeset	0
LIQUIBASE_DEBUG	Flag para habilitar o debug	0
LIQUIBASE_DB_SCHEMA	Schema a ser utilizado pelo liquibase	N/A
LIQUIBASE_CONTEXT	Contexto utilizado pelo liquibase	N/A

Table 2. Parâmetros Obrigatórios

Variável	Descrição	Exemplo
LIQUIBASE_DB_URL	String de conexão com o banco de dados no padrão utilizado pelo JDBC	jdbc:postgresql://172.17.0.2:5432/sistema
LIQUIBASE_DB_USER	Usuário com permissão de leitura no \${LIQUIBASE_DB_SCHEMA} requerido	N/A
LIQUIBASE_DB_PASS	Senha do usuário	N/A

Caso o arquivo referenciado em \${CHANGELOG\_FILE}, não seja encontrado, será criado um changelog com o conteúdo abaixo:

Example 1. Template changelog.xml

```
<databaseChangeLog
  xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
  xmlns:ext="http://www.liquibase.org/xml/ns/dbchangelog-ext"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog-ext
http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-ext.xsd
http://www.liquibase.org/xml/ns/dbchangelog
http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-3.5.xsd">

</databaseChangeLog>
```

Exemplo: Utilização do comando **generate** com o PostgreSQL

```
docker run --rm \
-e 'SOURCE_METHOD=VOLUME' \
-e 'LIQUIBASE_DB_URL=jdbc:postgresql://172.17.0.2:5432/db' \
-e 'LIQUIBASE_DB_USER=username' \
-e 'LIQUIBASE_DB_PASS=password' \
-v $(pwd)/source:/opt/source \
integr8/alpine-liquibase-helper generate
```

## Status

O propósito desta funcionalidade retornar o estado atual do banco de dados relacionando os ChangeSets existentes com os aplicados e rastreados na tabela DATABASECHANGELOG.

Table 3. Parâmetros Opcionais

Variável	Descrição	Valor Padrão
CHANGELOG_FILE	Arquivo que lista todos os ChangeSets pertencentes ao sistema	changelog.xml
LIQUIBASE_DEBUG	Flag para habilitar o debug	0
LIQUIBASE_DB_SCHEMA	Schema a ser utilizado pelo liquibase	N/A
LIQUIBASE_CONTEXT	Contexto utilizado pelo liquibase	N/A

Table 4. Parâmetros Obrigatórios

Variável	Descrição	Exemplo
LIQUIBASE_DB_URL	String de conexão com o banco de dados no padrão utilizado pelo JDBC	jdbc:postgresql://172.17.0.2:5432/sistema
LIQUIBASE_DB_USER	Usuário com permissão de leitura no \${LIQUIBASE_DB_SCHEMA} requerido	N/A
LIQUIBASE_DB_PASS	Senha do usuário	N/A

Exemplo: Utilização do comando **status** com o PostgreSQL

```
docker run --rm \
-e 'SOURCE_METHOD=VOLUME' \
-e 'LIQUIBASE_DB_URL=jdbc:postgresql://172.17.0.2:5432/db' \
-e 'LIQUIBASE_DB_USER=username' \
-e 'LIQUIBASE_DB_PASS=password' \
-v $(pwd)/source:/opt/source \
integr8/alpine-liquibase-helper status
```

## Update

Atualiza o banco de dados para a versão mais recente, consultando a tabela DATABASECHANGELOG, quais os ChangeSets não foram executados.

Table 5. Parâmetros Opcionais

Variável	Descrição	Valor Padrão
CHANGELOG_FILE	Arquivo que lista todos os ChangeSets pertencentes ao sistema	changelog.xml
LIQUIBASE_DEBUG	Flag para habilitar o debug	0
LIQUIBASE_DB_SCHEMA	Schema a ser utilizado pelo liquibase	N/A
LIQUIBASE_CONTEXT	Contexto utilizado pelo liquibase	N/A

Table 6. Parâmetros Obrigatórios

Variável	Descrição	Exemplo
LIQUIBASE_DB_URL	String de conexão com o banco de dados no padrão utilizado pelo JDBC	jdbc:postgresql://172.17.0.2:5432/sistema
LIQUIBASE_DB_USER	Usuário com permissão de leitura no \${LIQUIBASE_DB_SCHEMA} requerido	N/A
LIQUIBASE_DB_PASS	Senha do usuário	N/A

Exemplo: Utilização do comando **update** com o PostgreSQL

```
docker run --rm \
-e 'SOURCE_METHOD=VOLUME' \
-e 'LIQUIBASE_DB_URL=jdbc:postgresql://172.17.0.2:5432/db' \
-e 'LIQUIBASE_DB_USER=username' \
-e 'LIQUIBASE_DB_PASS=password' \
-v $(pwd)/source:/opt/source \
integr8/alpine-liquibase-helper update
```

## Diff

imprime a diferença entre dois bancos de dados para simples conferência

Table 7. Parâmetros Opcionais

Variável	Descrição	Valor Padrão
LIQUIBASE_DEBUG	Flag para habilitar o debug	0
LIQUIBASE_DB_SCHEMA	Schema a ser utilizado pelo liquibase	N/A
LIQUIBASE_DB_REFERENCE_SCHEMA	Schema a ser utilizado pelo liquibase no banco de referência	N/A
LIQUIBASE_CONTEXT	Contexto utilizado pelo liquibase	N/A

Table 8. Parâmetros Obrigatórios

Variável	Descrição	Exemplo
LIQUIBASE_DB_URL	String de conexão com o banco de dados no padrão utilizado pelo JDBC	jdbc:postgresql://172.17.0.2:5432/sistema
LIQUIBASE_DB_USER	Usuário com permissão de leitura no \${LIQUIBASE_DB_SCHEMA} requerido	N/A
LIQUIBASE_DB_PASS	Senha do usuário	N/A
LIQUIBASE_DB_REFERENCE_URL	String de conexão com o banco de dados de referência no padrão utilizado pelo JDBC	jdbc:postgresql://172.17.0.2:5432/sistema

Variável	Descrição	Exemplo
LIQUIBASE_DB_REFERENCE_USER	Usuário com permissão de leitura no \${LIQUIBASE_DB_SCHEMA} requerido no banco de dados de referência	N/A
LIQUIBASE_DB_REFERENCE_PASS	Senha do usuário no banco de dados de referência	N/A

Exemplo: Utilização do comando **diff** com o PostgreSQL

```
docker run --rm \
-e 'SOURCE_METHOD=VOLUME' \
-e 'LIQUIBASE_DB_URL=jdbc:postgresql://172.17.0.3:5432/db' \
-e 'LIQUIBASE_DB_USER=username' \
-e 'LIQUIBASE_DB_PASS=password' \
-e 'LIQUIBASE_DB_REFERENCE_URL=jdbc:postgresql://172.17.0.2:5432/db' \
-e 'LIQUIBASE_DB_REFERENCE_USER=username' \
-e 'LIQUIBASE_DB_REFERENCE_PASS=password' \
-v $(pwd)/source:/opt/source \
integr8/alpine-liquibase-helper diff
```

## GenerateDiff

O propósito desta funcionalidade comparar dois bancos de dados, e gerar o ChangeSet para que um fique idêntico ao outro. O script atualiza o arquivo de ChangeLog referenciado em \${CHANGELOG\_FILE}, adicionando o ChangeSet criado.

Table 9. Parâmetros Opcionais

Variável	Descrição	Valor Padrão
CHANGELOG_FILE	Arquivo que lista todos os ChangeSets pertencentes ao sistema	changelog.xml
CHANGESET_PATH	Diretório que guarda os changesets utilizados pelo ChangeLog	\$(dirname \${CHANGELOG_FILE})/changeset
CHANGESET_FILE	Nome padrão dos ChangeSets	changeset_YYYY-MM-DD_HH-MM.xml
LIQUIBASE_AUTHOR	Autor que será referenciado nos ChangeSets	liquibase
LIQUIBASE_DEBUG	Flag para habilitar o debug	0
LIQUIBASE_DB_SCHEMA	Schema a ser utilizado pelo liquibase	N/A
LIQUIBASE_DB_REFERENCE_SCHEMA	Schema a ser utilizado pelo liquibase no banco de referência	N/A
LIQUIBASE_CONTEXT	Contexto utilizado pelo liquibase	N/A

Table 10. Parâmetros Obrigatórios

Variável	Descrição	Exemplo
LIQUIBASE_DB_URL	String de conexão com o banco de dados no padrão utilizado pelo JDBC	jdbc:postgresql://172.17.0.2:5432/sistema
LIQUIBASE_DB_USER	Usuário com permissão de leitura no \${LIQUIBASE_DB_SCHEMA} requerido	N/A
LIQUIBASE_DB_PASS	Senha do usuário	N/A

Variável	Descrição	Exemplo
LIQUIBASE_DB_REFERENCE_URL	String de conexão com o banco de dados de referência no padrão utilizado pelo JDBC	jdbc:postgresql://172.17.0.2:5432/sistema
LIQUIBASE_DB_REFERENCE_USER	Usuário com permissão de leitura no \${LIQUIBASE_DB_SCHEMA} requerido no banco de dados de referência	N/A
LIQUIBASE_DB_REFERENCE_PASS	Senha do usuário no banco de dados de referência	N/A

Exemplo: Utilização do comando **generateDiff** com o PostgreSQL

```
docker run --rm \
-e 'SOURCE_METHOD=VOLUME' \
-e 'LIQUIBASE_DB_URL=jdbc:postgresql://172.17.0.2:5432/application' \
-e 'LIQUIBASE_DB_USER=application' \
-e 'LIQUIBASE_DB_PASS=application' \
-e 'LIQUIBASE_DB_REFERENCE_URL=jdbc:postgresql://172.17.0.3:5432/application' \
-e 'LIQUIBASE_DB_REFERENCE_USER=application' \
-e 'LIQUIBASE_DB_REFERENCE_PASS=application' \
-v $(pwd)/source:/opt/source \
integr8/alpine-liquibase-helper generateDiff
```

## Report

Gera um relatório contrastando o \${CHANGELOG\_FILE} com o estado do banco de dados atual, utilizado pelo Administrador de dados para conferência do que será mudado com a aplicação do ChangeLog.

Table 11. Parâmetros Opcionais

Variável	Descrição	Valor Padrão
CHANGELOG_FILE	Arquivo que lista todos os ChangeSets pertencentes ao sistema	changelog.xml
LIQUIBASE_DEBUG	Flag para habilitar o debug	0
LIQUIBASE_DB_SCHEMA	Schema a ser utilizado pelo liquibase	N/A
LIQUIBASE_CONTEXT	Contexto utilizado pelo liquibase	N/A

Table 12. Parâmetros Obrigatórios

Variável	Descrição	Exemplo
LIQUIBASE_DB_URL	String de conexão com o banco de dados no padrão utilizado pelo JDBC	jdbc:postgresql://172.17.0.2:5432/sistema
LIQUIBASE_DB_USER	Usuário com permissão de leitura no \${LIQUIBASE_DB_SCHEMA} requerido	N/A
LIQUIBASE_DB_PASS	Senha do usuário	N/A

Exemplo: Utilização do comando **report** com o PostgreSQL

```
docker run --rm \
-e 'SOURCE_METHOD=VOLUME' \
-e 'LIQUIBASE_DB_URL=jdbc:postgresql://172.17.0.2:5432/application' \
-e 'LIQUIBASE_DB_USER=application' \
-e 'LIQUIBASE_DB_PASS=application' \
-v $(pwd)/source:/opt/source \
-v $(pwd)/report:/opt/liquibase/report \
integr8/alpine-liquibase-helper report
```

□ Note a referência ao volume do report, para que este esteja disponível no host que estiver executado este container

## Raw Command

Executa um comando sem a utilização dos Wrappers criados por esta imagem

Table 13. Parâmetros Obrigatórios

Variável	Descrição	Exemplo
LIQUIBASE_DB_URL	String de conexão com o banco de dados no padrão utilizado pelo JDBC	jdbc:postgresql://172.17.0.2:5432/sistema
LIQUIBASE_DB_USER	Usuário no banco de dados	N/A
LIQUIBASE_DB_PASS	Senha do usuário no banco de dados	N/A

Exemplo: Utilização do comando **raw** com o PostgreSQL

```
docker run --rm \
-e 'SOURCE_METHOD=VOLUME' \
-e 'LIQUIBASE_DB_URL=jdbc:postgresql://172.17.0.2:5432/db' \
-e 'LIQUIBASE_DB_USER=user' \
-e 'LIQUIBASE_DB_PASS=password' \
-v $(pwd)/source:/opt/source \
integr8/alpine-liquibase-helper raw --logLevel=debug \
diff \
--referenceUrl=jdbc:postgresql://172.17.0.3:5432/db \
--referenceUsername=user \
--referencePassword=password
```

## DropAll

A execução desta funcionalidade apagará todos os objetos do banco de dados, não importando se foram executados pelo liquibase ou não.

Table 14. Parâmetros Opcionais

Variável	Descrição	Valor Padrão
LIQUIBASE_DEBUG	Flag para habilitar o debug	0
LIQUIBASE_DB_SCHEMA	Schema a ser utilizado pelo liquibase	N/A

Table 15. Parâmetros Obrigatórios

Variável	Descrição	Exemplo
LIQUIBASE_DB_URL	String de conexão com o banco de dados no padrão utilizado pelo JDBC	jdbc:postgresql://172.17.0.2:5432/sistema
LIQUIBASE_DB_USER	Usuário com permissão de leitura no \${LIQUIBASE_DB_SCHEMA} requerido	N/A
LIQUIBASE_DB_PASS	Senha do usuário	N/A

Exemplo: Utilização do comando **dropAll** com o PostgreSQL

```
docker run --rm \
-e 'SOURCE_METHOD=VOLUME' \
-e 'LIQUIBASE_DB_URL=jdbc:postgresql://172.17.0.2:5432/db' \
-e 'LIQUIBASE_DB_USER=username' \
-e 'LIQUIBASE_DB_PASS=password' \
-v $(pwd)/source:/opt/source \
integr8/alpine-liquibase-helper dropall
```