

liquibase quick start

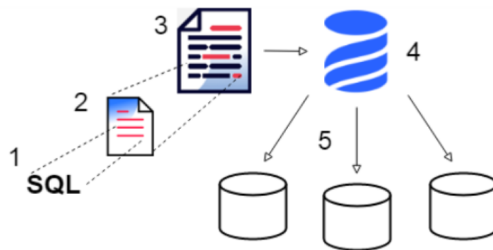
核心概念



Getting Started with Liquibase and SQL on Windows

After you install Liquibase, get started with our tutorial and learn how Liquibase works.

With Liquibase, SQL statements [1] are declared as Liquibase *changesets* [2] in a *changelog* file [3].



Liquibase [4] then uses the changelog to apply the changesets to target databases [5].

In this tutorial, you will use an example changelog to apply two rounds of updates to an H2 database that is included with the Liquibase installation files.

docs.liquibase.com/concepts/introduction-to-liquibase.html


Changesets or database change management system that enables you to track and manage database changes across and safer from development to production.


Tip: Check out [Why Liquibase?](#) for more information.


To keep it simple, you can write migrations in SQL – a database-specific syntax supported for PL/SQL. To take advantage of abstraction abilities, use a database-agnostic way by specifying your changes in XML, JSON, or YAML.


Liquibase uses SQL, XML, JSON, and YAML *changelog* files to list database changes in sequential order. Database changes have the format of *changesets*. Changesets contain *Change Type*, which are types of operations to apply to the database, such as adding a column or primary key. *Context*, *label*, and *precondition* changelog tags help precisely control when a database change is made and to which database environment it is deployed.


To set the connection between Liquibase with your database, you need the *database connection information* and parameters. Liquibase comes with the configuration file called *liquibase.properties* to store your database connection information and parameters that do not frequently change. Setting the parameters as *environment variables* to handle sensitive database information or running them at the command prompt is an alternative option.

**Changesets: Define database changes**

**Changelog: Add database changes to a file**

**Update: Run the command to deploy database changes**

**DATABASECHANGELOG and DATABASECHANGELOGLOCK**
Track and version database changes

**Manage your database changes with other Liquibase commands**

Liquibase runs 6 basic types of *commands*: update, rollback, snapshot, diff, status, and utility commands. When you use the update command to deploy your first changes, Liquibase checks the database connection information, including credentials, database URL, and JDBC driver.

When you deploy your changes, Liquibase creates two tables in your database: [DATABASECHANGELOG](#) and [DATABASECHANGELOGLOCK](#).

The DATABASECHANGELOG table tracks deployed changes so that you have a record. Liquibase compares the changesets in the changelog file with the DATABASECHANGELOG tracking table and deploys only new changesets.

DATABASECHANGELOGLOCK prevents multiple instances of Liquibase from updating the database at the same time. The table manages access to the DATABASECHANGELOG table during deployment and ensures only one instance of Liquibase is updating the database.

Liquibase offers many ways to manage your database changes:

- Run the [command-line client](#) (CLI).
- Use the [Liquibase Java API](#) and integrate Liquibase into your application to deploy database changes on the application startup.
- Integrate Liquibase into your build processes using [Maven](#), [Spring Boot](#), [Ant](#), [Jenkins](#), [GitHub Actions](#), or other CI/CD tools.
- With ephemeral environments in [Docker](#).

Related links

- [Concepts](#)
- [Best Practices](#)

refer to:

<https://docs.liquibase.com/concepts/introduction-to-liquibase.html>

<https://docs.liquibase.com/concepts/home.html>

<https://docs.liquibase.com/start/getting-started-liquibase-sql-windows.html>

前置条件

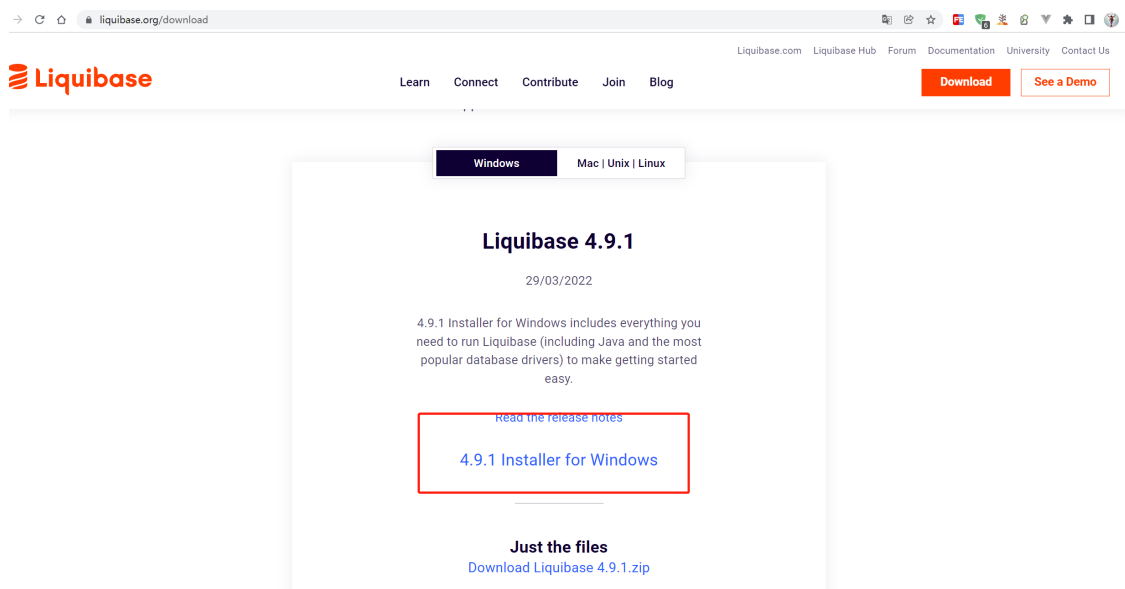
1. MySQL database
2. JDK8
3. apache-maven-3.8.4 or above
4. idea

command使用(win10 with community install version)

see branch liquibase-with-maven-plugin for [hello-liquibase](#)

安装

1. 下载 <https://liquibase.org/download>



2. 安装，记得安装过程中，选择加入环境变量
3. 验证，cmd窗口中输入 `liquibase -v`，控制台打印logo安装成功

```
C:\Windows\System32\cmd.exe

E:\a\liquibase-mysql-test-20220331>liquibase -v
#####
##
## Liquibase
##
##
## Get documentation at docs.liquibase.com
## Get certified courses at learn.liquibase.com
## Free schema change activity reports at
## https://hub.liquibase.com
##
#####
Starting Liquibase at 16:43:08 (version 4.9.0 #1885 built at 2022-03-16 16:58+0000)
Running Java under C:\Program Files\Java\jdk1.8.0_311\jre (Version 1.8.0_311)

Liquibase Version: 4.9.0
Liquibase Community 4.9.0 by Liquibase

E:\a\liquibase-mysql-test-20220331>
```

初试

1. 新建 liquibase-mysql-test-20220331 文件夹
2. 在文件夹 liquibase-mysql-test-20220331 下新增 mysql-connector-java-8.0.20.jar 依赖
3. 在文件夹 liquibase-mysql-test-20220331 下新增 liquibase.properties 文件,具体内容参照 demo,该配置文件很重要!!!

该文件包含

数据库配置

日志配置

- 日志级别选择 DEBUG ERROR INFO ALL...
- 输出到控制台还是文件
- 日志名称,位置配置

输入文件位置配置

- changeLogFile=master-changelog.xml

输出文件位置配置

refer to: <https://docs.liquibase.com/concepts/connections/creating-config-properties.html>

4. 在文件夹 liquibase-mysql-test-20220331 下新增 master-changelog.xml 文件,具体内容参照 demo

1. 该文件主要用于 include 分散的 changelog 文件


2. 注意坑

The reason why you would use the `<include>` tag rather than using XML's built-in include functionality is that the parser sees just one large XML document. Liquibase needs to uniquely identify each *changeset* with an id, author, and file name.

How to use the `<include>` tag


1. Create a master *changelog* file, if you do not have one already.

The master *changelog* file works as a configuration file that will hold all the references to all your other *changelogs*. Your master *changelog* file must be in an XML, YAML, or JSON format.

 **Note:** At this time the `<include>` tag cannot be used in an SQL formatted master *changelog*.

2. Add the `<include>` tag and file references to your master *changelog*.

In this example, we are using an XML master *changelog* with the `<include>` tag referencing two SQL *changelogs*:

 **Note:** Your reference *changelog* can be SQL, XML, YAML, or JSON file types.

```
<?xml version="1.0" encoding="UTF-8"?>
<databaseChangeLog
  xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:ext="http://www.liquibase.org/xml/ns/dbchangelog-ext"
  xmlns:pro="http://www.liquibase.org/xml/ns/pro"
  xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog
    http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-4.9.0.xsd
    http://www.liquibase.org/xml/ns/dbchangelog-ext http://www.liquibase.org/xml/ns/dbchangelog/dbchang
    http://www.liquibase.org/xml/ns/pro http://www.liquibase.org/xml/ns/pro/liquibase-pro-4.9.0.xsd">
  <include file="com/example/news/news.changelog.sql"/>
  <include file="com/example/directory/directory.changelog.sql"/>
</databaseChangeLog>
```

refer to: <https://docs.liquibase.com/concepts/changelogs/attributes/include.html>

5. 在文件夹 liquibase-mysql-test-20220331 下新增 changelogs 文件夹,并在文件夹中新增 changelog 文件

1. 新增 1-changeLog.sql
2. 新增 2-changeLog.sql
3. 新增 3-changeLog.sql
4. 新增 4-changeLog.sql
5. ...

这些文件将会在文件 master-changeLog.xml 中通过include标签被包含,如下

```
<include file="changeLogs/2-changeLog.sql"/>
```

以上xxx-changelog.sql文件介绍

changelog文件可以是 sql,xml,yaml,json格式

具体内容是对数据库的 ddl dml等操作

sql格式比较容易理解

包含关系:

changelog file > changesets > changetype

refer to:

<https://docs.liquibase.com/concepts/changelogs/sql-format.html>

<https://docs.liquibase.com/concepts/changelogs/working-with-changelogs.html>

generateChangeLog

refer to:

<https://docs.liquibase.com/concepts/connections/creating-config-properties.html>

<https://docsstage.liquibase.com/commands/generatechangelog.html>

1. 以mysql为例, 对已有数据库, 导出对应表结构changelog(包含)

```
liquibase --changeLogFile=20220331-exist-data-structure.mysql.sql  
generateChangeLog
```

2. 以mysql为例, 对已有数据库, 导出对应表数据changelog,注意配置

```
liquibase --changeLogFile=20220331-exist-data.mysql.sql --diffTypes=data  
generateChangeLog
```

update

refer to:

<https://docs.liquibase.com/commands/home.html>

<https://docs.liquibase.com/commands/update.html>

1. 将changelog更新到数据库, 在文件夹 liquibase-mysql-test-20220331 下打开cmd窗口, 执行一下命令

```
liquibase update
```

```
C:\Windows\System32\cmd.exe

E:\a\liquibase-mysql-test-20220331>liquibase update

#####
##                               ##
##  Liquibase                    ##
##                               ##
##  Get documentation at docs.liquibase.com  ##
##  Get certified courses at learn.liquibase.com  ##
##  Free schema change activity reports at  ##
##  https://hub.liquibase.com                ##
##                               ##
#####
Starting Liquibase at 17:58:17 (version 4.9.0 #1885 built at 2022-03-16 16:58+0000)
Liquibase Version: 4.9.0
Liquibase Community 4.9.0 by Liquibase
Running Changeset: changelogs/1-changelog.sql::1::your.name
Running Changeset: changelogs/2-changelog.sql::2::your.name
Liquibase command 'update' was executed successfully.

E:\a\liquibase-mysql-test-20220331>
```

2. 查询数据库中生成的表,发现changelog中create的表已经同步到数据库

18 show tables;

信息 结果 1 剖析 状态

Tables_in_liquibase									
company									
databasechangelog									
databasechangeloglock									
person									

○ 其中

■ databasechangelog

The DATABASECHANGELOG table tracks deployed changes so that you have a record. Liquibase compares the changesets in the changelog file with the DATABASECHANGELOG tracking table and deploys only new changesets.

信息	结果 1	剖析	状态
ID	AUTHOR	FILENAME	DATEEXECUTED
1	your.name	changelogs/1-changelog.sql	2022-03-31 17:5
2	your.name	changelogs/2-changelog.sql	2022-03-31 17:5

■ databasechangeloglock

DATABASECHANGELOGLOCK prevents multiple instances of Liquibase from updating the database at the same time. The table manages access to the DATABASECHANGELOG table during deployment and ensures only one instance of Liquibase is updating the database.

信息	结果 1	剖析	状态
ID	LOCKED	LOCKGRANTED	LOCKEDBY
1	0	(Null)	(Null)

rollback

rollback tag 回滚到某个版本

1. 执行完某次 update 操作以后需要给该次操作打标签

```
liquibase tag v1
```

2. 回滚到一个tag

```
liquibase rollback v2
```

refer to:

<https://docs.liquibase.com/commands/rollbackbytag.html>

rollbackToDate 回滚到指定时间

Attribute	Definition	Requirement
-- date/time	The date and time your database rolls back to. The date format is YYYY-MM-DD HH:MM:SS or YYYY-MM-DD'T'HH:MM:SS, however, it is possible to indicate the date or time only.	Required

```
liquibase rollbackToDate 2020-03-31 12:01:23
```

refer to:

<https://docs.liquibase.com/commands/rollbacktodate.html>

rollbackCount 表示rollback的changeset的个数

```
liquibase rollbackCount 1
```

或者

```
liquibase rollback-count 1
```

refer to:

<https://docs.liquibase.com/commands/rollbackcount.html>

rollback conclusion

Command	Description
---------	-------------

Command	Description
rollback	Rolls back the database to the state it was in when the tag was applied.
rollbackSQL	A helper command that produces the raw SQL Liquibase would run when using the <code>rollbackByTag</code> command.
rollbackToDate	Rolls back the database to the state it was in at the given date/time.
rollbackToDateSQL	A helper command that allows you to inspect the SQL Liquibase will run while using the rollbackToDate command.
rollbackCount	Rolls back the last <i>changeset**s</i> .
rollbackCountSQL	Writes SQL to roll back the last <i>changeset**s</i> to <code>STDOUT</code> .
rollbackOneChangeSet PRO	Rolls back a single <i>changeset</i> without needing to roll back other already applied <i>changeset</i> .
rollbackOneChangeSetSQL PRO	Writes SQL to roll back a single <i>changeset</i> to <code>STDOUT</code> .
rollbackOneUpdate PRO	Rolls back all <i>changesets</i> related by a specific <code>deploymentId</code> that was made during a previous change to your database.
rollbackOneUpdateSQL PRO	A helper command that allows you to inspect the SQL Liquibase will run to revert all <i>changeset</i> associated with the <code>deploymentID</code> specified in the rollbackOneUpdate command.
futureRollbackSQL	Writes SQL to roll back the database to the current state after the changes in the <i>changelog</i> have been applied.
futureRollbackCountSQL	Generates the SQL that Liquibase would use to sequentially revert the number of changes associated with undeplied <i>changeset**s</i> , which are added to a <i>changelog</i> file.
future-rollback-from-tag-sql	Produces the raw SQL Liquibase would need to roll back all undeplied changes made up to the specified tag.

refer to:

<https://docs.liquibase.com/commands/home.html>

maven-plugin 使用

see branch liquibase-with-maven-plugin for [hello-liquibase](#)

refer to:

<https://docs.liquibase.com/tools-integrations/maven/maven-pom-file.html>

<https://docs.liquibase.com/tools-integrations/maven/commands/home.html>

集成 springboot 使用

see branch liquibase-with-spring-boot for [hello-liquibase](#)

Jenkins 使用 todo
