

ORACLE®

# Howdy, I'm **Blaine Carter**

Oracle Developer Advocate for Open Source  
Oracle Corporation

Email: [blaine.carter@oracle.com](mailto:blaine.carter@oracle.com)

Twitter: [@OraBlaineOS](https://twitter.com/OraBlaineOS)

Blog: [learncodeshare.net](http://learncodeshare.net)

Team: [community.oracle.com/docs/DOC-917690](http://community.oracle.com/docs/DOC-917690)



# Database Change Management – The Hard Way

MasterCreate directory:

RunAllVer5.sql

CreateLB\_Groups.sql

CreateLB\_People.sql

LoadLB\_GroupsData.sql

Update4to5 directory:

UpdateAllVer4to5.sql

AddColGroupsRules.sql

AddColPeopleFavorite\_color.sql

Problems:

Which script has been run?

Update or Rollback to a specific version?

Test data? → LoadTestData.sql?

# Leveraging Open Source for Oracle Database Development

## Liquibase

Cross platform database change management.

<http://www.liquibase.org/>

<https://github.com/liquibase/liquibase>

**Blaine Carter**

# Version Control / Change Management

## Basics:

- Track changes.
- Rollback / Switch to a specific revision.
- Branch / Merge
- Diffs

# What is Liquibase?

Liquibase is change management for your database.

- Database changes are made using changeSets.
- When you run an update, the changeSets are run in order.
- Liquibase tracks what has been run, when and who ran it.
- You can roll back to a specific version.
- Populate default and/or test data.
- Contexts - <http://www.liquibase.org/documentation/contexts.html>
- Diffs - <http://www.liquibase.org/documentation/diff.html>
- Documentation - <http://www.liquibase.org/documentation/dbdoc.html>
- SQL Output - [http://www.liquibase.org/documentation/sql\\_output.html](http://www.liquibase.org/documentation/sql_output.html)
- Offline - <http://www.liquibase.org/documentation/offline.html>

# How Does It Work?

# Changelog Master

```
{  
  "databaseChangeLog": [  
    {"include": {"file": "changelog/db.changelog-1.json"}}  
  ]  
}
```



# ChangeSet

```
{"databaseChangeLog": [  
  {"preConditions": [{  
    "runningAs": {  
      "username": "lb_demo"  
    }  
  }]  
},  
  {"changeSet": {  
    "id": "1",  
    "author": "BlaineCarter",  
    "changes": [  
      {"createTable": {  
        "tableName": "lb_person",  
        "columns": [  
          {"column": {  
            "name": "id",  
            "type": "int",  
            "autoIncrement": true,  
            "constraints": {  
              "primaryKey": true,  
              "nullable": false
```

.....

# Java Command Line

```
java $JAVA_OPTS -jar /opt/liquibase/liquibase.jar  
--driver=oracle.jdbc.OracleDriver  
--classpath="/usr/lib/oracle/12.1/client64/lib/ojdbc7.jar"  
--url=jdbc:oracle:thin:lb_demo/dd@dbaccess  
--changeLogFile=changelog/db.changelog-master.json updateSQL >> output.sql
```

options

```
--logLevel=DEBUG --logFile=liquibase.logFile
```

\$JAVA\_OPTS is used for my Oracle Exadata Express cloud connection.

# Properties File (liquibase.properties)

```
driver: oracle.jdbc.OracleDriver  
classpath: /usr/lib/oracle/12.1/client64/lib/ojdbc7.jar  
url: jdbc:oracle:thin:lb_demo/dd@dbaccess  
changeLogFile: changelog/db.changelog-master.json
```

# Liquibase ~~Best~~ Recommended Practices

<http://www.liquibase.org/bestpractices.html>

Ignore the directory structure I'm using in the demo's.

# Software We'll Be Using

- SQL Developer
- Oracle Exadata Express Cloud Database or Virtual Box
- Liquibase - <http://www.liquibase.org>

# Install

1. Download - <http://www.liquibase.org/download/index.html>
2. Install – It's just Java, extract it and go.

Enough blah blah, let's see it.

# Create Changelog and ChangeSet Files

Changelog Example:

<http://www.liquibase.org/quickstart.html> - step 1

Changeset Example:

<http://www.liquibase.org/quickstart.html> - step 2

Recommended to only perform one change per Change set, but you can do more if you need.



# Run It

<http://www.liquibase.org/quickstart.html> - step 3

Jar File: `java $JAVA_OPTS -jar /opt/liquibase/liquibase.jar updateSQL`

Shell Script: `liquibase updateSQL`

Batch File: `liquibase.bat updateSQL`

Put the Liquibase directory in your path and run it:  
`liquibase update`

The shell script / batch files use the same `JAVA_OPTS` environment variable for my Exadata Express Cloud connection.

# Rollback X Number of Change Sets

Warning: Rolling back anything in the database can be tricky.  
Be very careful if you ever do this in Production.

```
liquibase rollbackCount 1
```

# Tag a Changeset

**You can tag a changeset in the file:**

```
"changeSet": {  
  "id": "1",  
  "author": "BlaineCarter",  
  "tagDatabase": {"tag": "ver-1"},  
  "changes": [  

```

**You can also use the command line:**

```
liquibase tag ver-1
```

# Rollback to Tag

<http://www.liquibase.org/documentation/rollback.html>

liquibase rollback ver-1

## **Other Options:**

[http://www.liquibase.org/documentation/command\\_line.html](http://www.liquibase.org/documentation/command_line.html)

# Load Data From a File

```
"changes": [  
  {  
    "loadData": {  
      "file": "changelog/groups.csv",  
      "schemaName": "lb_demo",  
      "tableName": "lb_groups"  
    }  
  }  
]
```

## **groups.csv**

name,description

Trucks,People who like trucks

Rockets,People who like rockets

Horses,People who like horses

Snakes,People who like snakes

# Insert Data

```
"changes": [  
  {  
    "insert": {  
      "schemaName": "lb_demo",  
      "tableName": "lb_people",  
      "columns": [  
        {  
          "column": {  
            "name": "firstname",  
            "value": "Bob"  
          }  
        }, {  
          "column": {  
            "name": "group_id",  
            "valueComputed": "(SELECT id FROM lb_groups where name = 'Trucks')"  
          }  
        }  
      ]  
    }  
  }  
]
```

...

# Context Test

In the change set:

```
"changeSet": {  
  "id": "loadData-example",  
  "author": "liquibase-docs",  
  "context": "test",
```

```
Liquibase.properties  
contexts: !test
```

Command Line overrides the properties file:  
liquibase --contexts=test update

Context vs Labels

<http://www.liquibase.org/2014/11/contexts-vs-labels.html>

# Data Changes Are NOT Automatically Rolled Back.

## Define your own Rollback

```
{
  "rollback": {
    "delete": {
      "tableName": "lb_people"
    }
  }
}
```



# Diff

<http://www.liquibase.org/documentation/diff.html>

## Command Line:

```
liquibase.sh --url=jdbc:oracle:thin:lb_demo/dd@unknownDB \  
diff \  
--referenceUrl=jdbc:oracle:thin:lb_demo/dd@goodDB
```

## Liquibase.properties:

```
referenceUrl: jdbc:oracle:thin:lb_diff_demo/dd@dbaccess
```

# Generate Changelog (Reverse Engineer)

[http://www.liquibase.org/documentation/generating\\_changelogs.html](http://www.liquibase.org/documentation/generating_changelogs.html)

Note that this command currently has some limitations. It does not export the following types of objects:

- Stored procedures, functions, packages & Triggers

```
liquibase --changeLogFile=generated.json generateChangeLog
```

# Review The Output

The output won't be perfect.

Generated by update:

```
id INTEGER GENERATED BY DEFAULT AS IDENTITY NOT NULL
```

Generated by generateChangeLog:

```
ID NUMBER(*, 0) DEFAULT "LB_DEMO"."ISEQ$$_69966".nnextval NOT NULL
```

# DBDoc

<http://www.liquibase.org/documentation/dbdoc.html>

liquibase DBDoc docs

# Extensions / Plugins

<https://liquibase.jira.com/wiki/display/CONTRIB/LiquiBase+Extensions+Portal>

# We Can Make It Better

<http://www.liquibase.org/>



Github Repo - <https://github.com/liquibase/liquibase>

- Fork
- Change
- Pull Request

There are many ways to contribute.

- Enhance code
- Add tests
- Improve documentation
- Submit / review issues



# Howdy, I'm **Blaine Carter**

Oracle Developer Advocate for Open Source  
Oracle Corporation

Email: [blaine.carter@oracle.com](mailto:blaine.carter@oracle.com)

Twitter: [@OraBlaineOS](https://twitter.com/OraBlaineOS)

Blog: [learncodeshare.net](http://learncodeshare.net)

Team: [community.oracle.com/docs/DOC-917690](http://community.oracle.com/docs/DOC-917690)



# Commands Used For The Demo (by slide number)

#17

liquibase updateSQL  
liquibase update

#18

liquibase rollbackCount 1  
liquibase update

#19

liquibase update  
liquibase tag myCoolTag

#20

liquibase rollback ver-1  
liquibase update

#21

liquibase update

#23

liquibase --contexts=test update

#24

liquibase rollbackCount 2  
liquibase rollbackCount 1  
liquibase --contexts=test rollbackCount 2  
liquibase --contexts=test update

#25

liquibase diff  
liquibase --diffTypes=tables,columns diff  
liquibase diffChangeLog

#26

liquibase --changeLogFile=generated.json generateChangeLog

#28

liquibase dropAll  
liquibase DBDoc docs  
liquibase update  
liquibase DBDoc docs