

NO control or KNOW control. Time to rule your database changes people!



Rafal Grzegorczyk

Oracle APEX & PL/SQL Developer

Pretius Low-Code

rgrzegorczyk@pretius.com

NO control?



You can't / don't track database changes because?

- NO time
- NO money
- NO resources
- you don't need it
- it's your own ordered mess
- other reason

So, how are you handling your DB scripts?



- 1
- 2
- 3
- 4
- 5

Homegrown solution?



Emails? Slack ? Other communicators?



Attached to tasks or included in comments (JIRA etc.)?



GIT (checking what has changed) ?



Other solutions?

One of my past projects....

- 5 developers
- NO tracking tool
- NO git
- many tasks, changes and features

Create a script for UAT including changes from tasks 1 and 2, without 3,4,5.
Easy and fast, right?



How many hours are you spending finding out **who** changed **what, where** and **why** in your database?



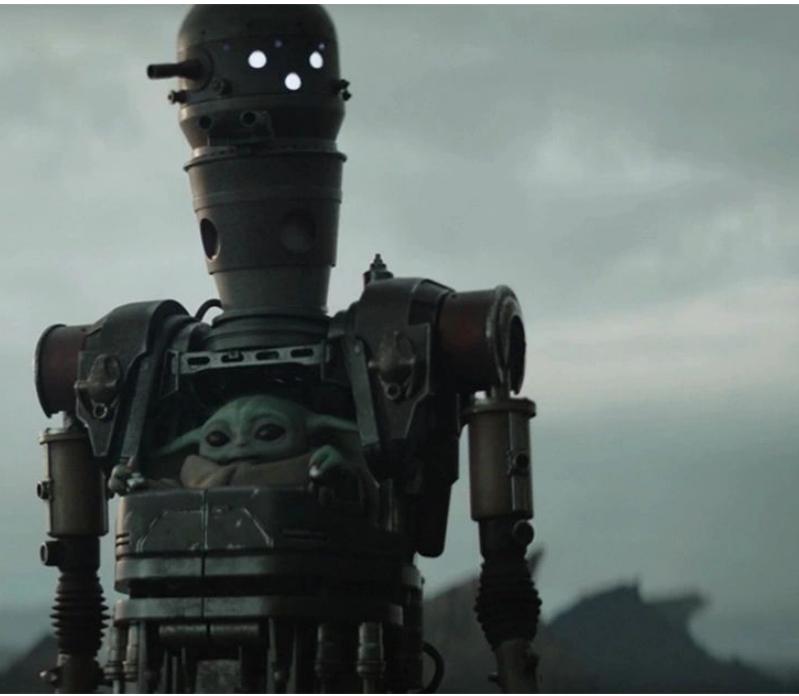
How about production hotfixes?



KNOW control



Does it require implementing full-blown DevOps?



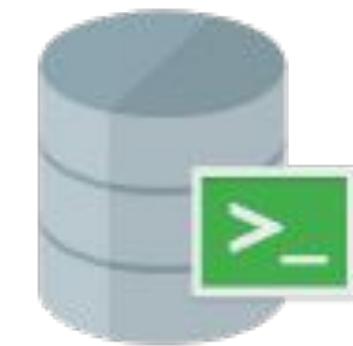
No.



Just one tool (standalone or built-in SQLcl)



ORACLE®

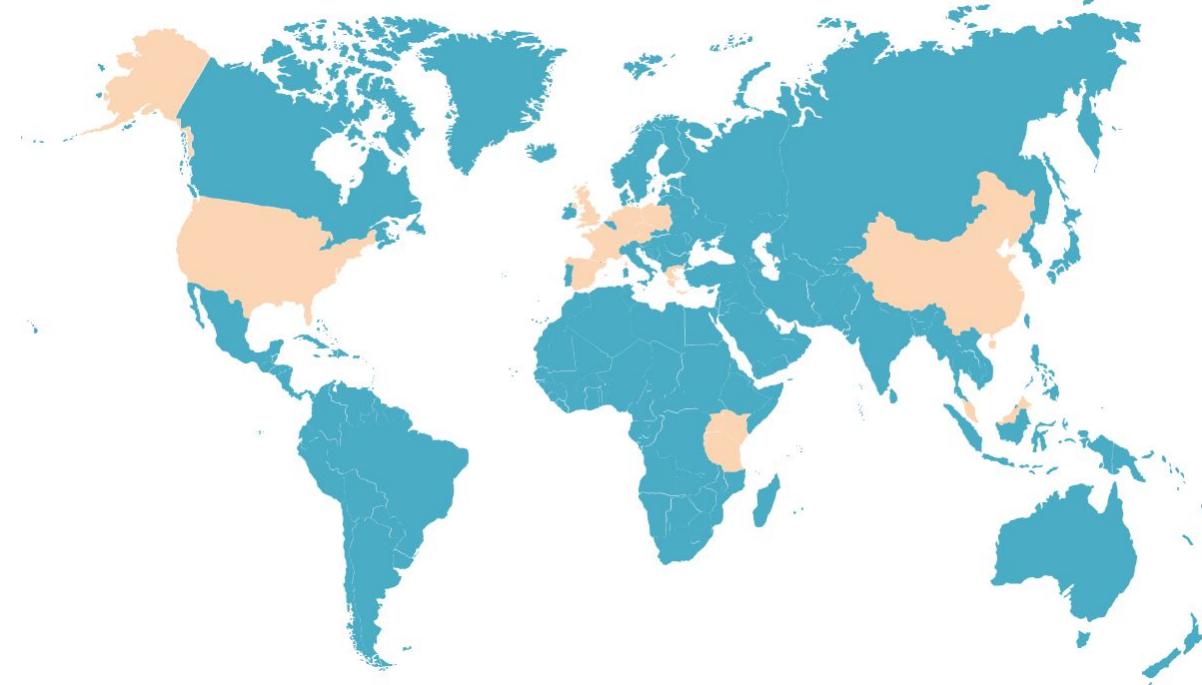




Rafal Grzegorczyk
Oracle APEX & PL/SQL Developer
rgrzegorczyk@pretius.com
[@r_grzegorczyk](https://twitter.com/r_grzegorczyk) 
[rgrzegorczyk](https://www.linkedin.com/in/rgrzegorczyk/) 
rafał.hasnode.dev

About me

- Oracle APEX & PL/SQL Developer at Pretius Low-Code
- Over 10 years in IT
- Financial systems for government administration
- Energy, banking and logistics industry
- Database automation fan
- Liquibase certified user
- Speaker since 2021 (Kscope, Apex World, DOAG, Switzerland, Croatia, Scotland, Poland)
- Blogger at rafał.hashnode.dev and pretius.com/blog/



LIVE DEMO 1

SQLcl + built-in Liquibase (part 1 / 2)

Move all objects from DEV to new UAT with...only two commands.

What is Liquibase?

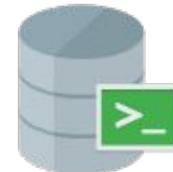
Open-Source Java based tool



Developed since 2006



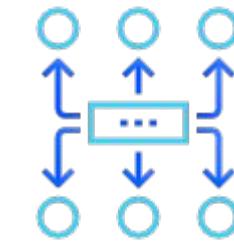
Enhanced & implemented into SQLCI in 2019



Command-line based



How liquibase works?



Changelog

Ledger of changes. It holds information about what and when to execute.



Changeset

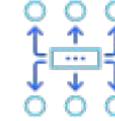
Single DB change identified by ID, author and file path.
SQL, XML, YAML or JSON



Tracking table

History of changes. Who, when and what. Comments, labels, tags.

Changelog



```
</> controller.xml X

sqlcl_liquibase > liquibase > </> controller.xml > databaseChangeLog
1   <?xml version="1.0" encoding="UTF-8"?>
2   <databaseChangeLog
3     xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
4     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5     xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog
6     http://www.liquibase.org/xml/ns/dbchangelog-dbchangelog-latest.xsd">
7
8     <include file="database/table/regions.sql"/>...
9     <include file="database/table/countries.sql"/>...
10    <include file="database/table/application_params.sql"/>...
11
12    <includeAll path="database/procedure"/>
13    <includeAll path="database/trigger"/>...
14    <includeAll path="database/view"/>...
15
16  </databaseChangeLog>
```

Changesets



liquibase

- database
 - PROCEDURE
 - SECURE_DML.sql
 - TABLE
 - APPLICATION_PARAMS.sql
 - COUNTRIES.sql
 - REGIONS.sql
 - TRIGGER
 - SECURE_COUNTRIES.sql
 - SECURE_REGIONS.sql
 - VIEW
 - EMP_DETAILS_VIEW.sql

APPLICATION_PARAMS.sql

```

sqlcl_liquibase > liquibase > database > TABLE > APPLICATION_PARAMS.sql > ...
1 --liquibase-formatted.sql
2 --changeset=rgrzegorczyk:APPLICATION_PARAMS--runOnChange:false
3 --comment=APPLICATION_PARAMS-table
4 --CREATE TABLE "APPLICATION_PARAMS"
5   --> "CODE" VARCHAR2(50),
6   --> "VALUE" VARCHAR2(500)
7 ;

```

SECURE_COUNTRIES.sql

```

sqlcl_liquibase > liquibase > database > TRIGGER > SECURE_COUNTRIES.sql
1 --liquibase-formatted.sql
2 --changeset=rgrzegorczyk:SECURE_COUNTRIES--runOnChange:true
3 --comment=SECURE_COUNTRIES-first-version
4 CREATE OR REPLACE EDITIONABLE TRIGGER "SECURE_COUNTRIES"
5 before insert or update or delete on countries
6 begin
7   ...secure_dml;
8 end secure_countries;
9
10 /
11 ALTER TRIGGER "SECURE_COUNTRIES" DISABLE;

```

EMP_DETAILS_VIEW.sql

```

sqlcl_liquibase > liquibase > database > VIEW > EMP_DETAILS_VIEW.sql > ...
1 --liquibase-formatted.sql
2 --changeset=rgrzegorczyk:EMP_DETAILS_VIEW--runOnChange:true
3 --comment=EMP_DETAILS_VIEW-first-version
4 --CREATE OR REPLACE VIEW "EMP_DETAILS_VIEW" ("COUNTRY_NAME", "REGION_NAME")
5 AS
6 select c.country_name,
7       r.region_name
8   from countries c
9      join regions r
10     on c.region_id = r.region_id;
11

```

Tracking table



DATABASECHANGELOG

Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL

Sort.. Filter:

ID	AUTHOR	FILENAME	DATEEXECUTED	MD5SUM	COMMENTS
1	rgrzegorczyk	database/table/regions.sql	2023-03-07 07:52:02,300175000	8:f43434582d009980021e85851fb9626d	REGIONS table
2	rgrzegorczyk	database/table/countries.sql	2023-03-07 07:52:02,968867000	8:9fd1a2073de3191651054949aaab1580	COUNTRIES table
3	rgrzegorczyk	database/table/application_params.sql	2023-03-07 07:52:03,594816000	8:f8e8f3d9bf37034fdd3ecac69c34ed47	APPLICATION_PARAMS table
4	rgrzegorczyk	database/table/application_params.sql	2023-03-07 07:52:04,414706000	8:e8e0571ab2e0a41299500ab4cbdf6b6a	(null)
5	rgrzegorczyk	database/procedure/SECURE_DML.sql	2023-03-07 07:52:05,074303000	8:1a2d8b94fd04ed0840b1f35ecd8358f7	Secure dml proc first version
6	rgrzegorczyk	database/trigger/SECURE_COUNTRIES.sql	2023-03-07 07:52:05,985844000	8:f27caee7682cb79842ebce1109a44c07	SECURE_COUNTRIES first version
7	rgrzegorczyk	database/trigger/SECURE_REGIONS.sql	2023-03-07 07:52:06,892042000	8:a858d6ded934520369300f53b4352173	SECURE_REGIONS first version
8	rgrzegorczyk	database/view/EMP_DETAILS_VIEW.sql	2023-03-07 07:52:07,753123000	8:67c206a624d7f791199fafcbc903a668	EMP_DETAILS_VIEW first version

APPLICATION_PARAMS.sql

```

liquibase > database > TABLE > APPLICATION_PARAMS.sql > ...
1  --liquibase.formatted.sql
2  --changeset=rgrzegorczyk:APPLICATION_PARAMS--runOnChange:false
3  --comment APPLICATION_PARAMS
4  CREATE TABLE "APPLICATION_PARAMS"
5  ... ("CODE" VARCHAR2(50),
6  ... "VALUE" VARCHAR2(500)
7  ... );
8
9  --changeset=rgrzegorczyk:add_col_description--runOnChange:false
10 --comment Added column name
11 ALTER TABLE APPLICATION_PARAMS ADD DESCRIPTION VARCHAR2(50);
12

```

Update command



- 1 Deployes any changes in the changelog file that have not been deployed to database yet.
- 2 If the unique identifiers do not exist, Liquibase will apply the changeset to the database.
- 3 If the unique identifiers exist, the MD5Sum of the changeset is compared to the one in the database.
- 4 If they are different, Liquibase will produce an error unless runOnChange parameter is set to true.

ID	AUTHOR	FILENAME	MD5SUM
1 add_col_description	rgrzegorczyk	database/table/application_params.sql	8:b1fee6c60845ab29eae6c4c3f8b5a741
2 APPLICATION_PARAMS	rgrzegorczyk	database/table/application_params.sql	8:4e131c1437edc63f631de6e1074c3d04

LIVE DEMO 1

SQLcl + built-in Liquibase (part 2 / 2)

Move only changed/new objects from DEV to UAT

Questions & Answers

LIVE DEMO 2

Liquibase (standalone)

- use with existing database
- deploy selected changes
- compare environments and auto-generate SQL script

Questions & Answers

LIVE DEMO 3

Generate and deploy APEX scripts with SQLcl Liquibase

Presentation and examples: https://github.com/rgrzegorczyk/no_control_or_know_control

New, detailed blogpost will be ready in Q3 2023

Want more? Pretius resources:

- Liquibase tutorial: <https://pretius.com/blog/liquibase-tutorial/>
- Liquibase rollbacks: <https://pretius.com/blog/liquibase-rollback-jenkins-oracle-apex/>
- Flyway, Gradle & APEX <https://pretius.com/gradle-flyway-oracle-apex-a-few-words-about-migration-part-1/>

Other resources:

- Liquibase standalone documentation: <https://docs.liquibase.com/home.html>
- SQLcl Liquibase documentation : <https://docs.oracle.com/en/database/oracle/sql-developer-command-line/22.4/index.html>



Rafał Grzegorczyk

Oracle APEX & PL/SQL Developer

rgrzegorczyk@pretius.com

r_grzegorczyk

rgrzegorczyk

rafal.hasnode.dev



Thank you