Project: BDC Release Track History

Version: 1.0

Date: 31/08/2018

Author: Israel Palomino Garcia

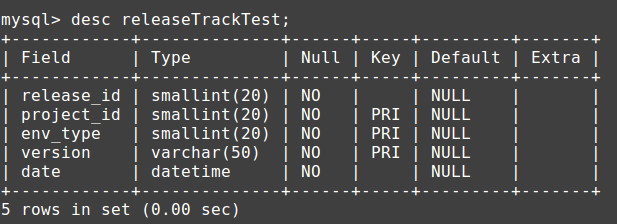
Rationale: Described below

When updating existing table “version” with a new version in BDC, a trigger “after\_update\_version” will execute an INSERT statement in a releases historic table “ releaseTrack” for environments:

14 INT, 8 REF, 11 PROD.

Another trigger “after\_insert\_version” will execute same INSERT in the historic table when a new application is created in “version” table.

The table releaseTrack has the following structure:



The attribute release\_id indicates in which release the corresponding “version” has been deployed in each environment,

e.g.: 1808 for August 2018.

This attribute will be used when querying for specific releases and associated projects/versions, generating statistics.

The trigger follows this logic:

The next release\_id will be generated after adding +1 to latest PROD release\_id.

A Stored Procedure “sp\_check\_previous\_versions” is called in order to check for previous versions in INT or REF which had not been previously deployed in the latest PROD release. These versions will be moved to the next release (updating release\_id).

The trigger will INSERT a record for each version in INT or REF, as long as the version value is not string ‘EMPTY’.

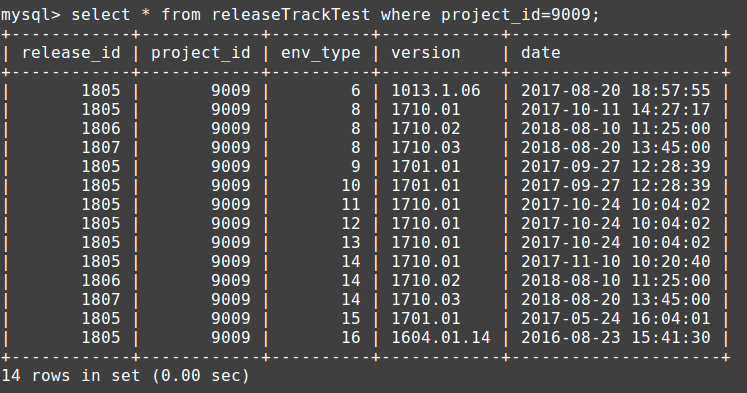
Initial releaseTrack table example for one project:

Scenario: Latest release in PROD was 1808, new versions will be linked with 1809.

Project ID: 9009

Latest release in PROD (11) for this project was version 1710.01 in release 1805

After release 1805, there were a couple of versions deployed in INT(14)/REF(8), one version 1710.02 initially planned for 1806, another version 1710.03 planned for 1807. These versions never reached PROD.



A new version 1710.04 is developed for this project, this version is linked with release 1809, as well as the previous versions which never reached PROD.

The Stored Procedure takes charge of moving the release\_id to the future release.

mysql> select max(release\_id) from releaseTrackTest;

+-----------------+

| max(release\_id) |

+-----------------+

| 1808 |

+-----------------+

mysql> update versionTest set version = '1710.04' where project\_id = 9009 and type IN (14,8);

Query OK, 2 rows affected (0.00 sec)

Rows matched: 2 Changed: 2 Warnings: 0

Below version and releaseTrack tables updated with versions to be deployed with release 1809 in INT/REF:

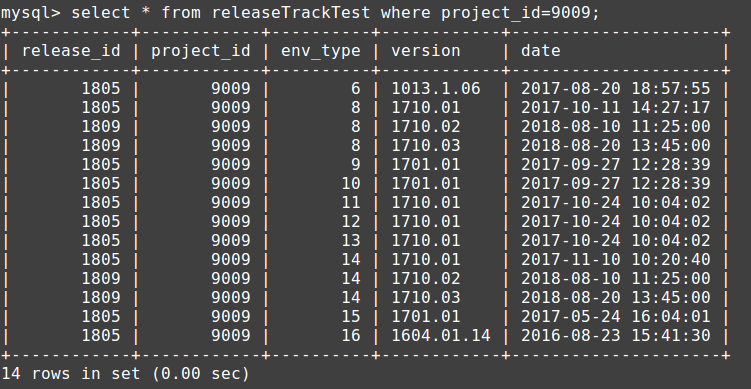
New version: 1710.04

+

Previous versions INT/REF not deployed in PROD:

1710.02 | 1710.03 have been moved to Next Release 1809.

While version linked with 1805 in PROD: 1710.01 is not updated.



When creating the queries for generating statistics for deployed project/versions in a specific release (e.g.: How many iterations for project 9009 have been deployed in INT/REF before reaching PROD?), it will only need to specify that the project has the environment type for PROD (11) deployed for this project, otherwise it will be excluded.