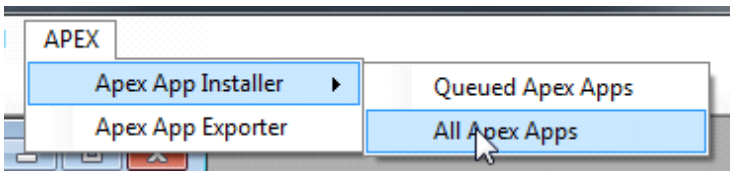
GitPatcher Tutorial 2

Tuesday, 2 April 2019 2:18 PM

GitPatcher

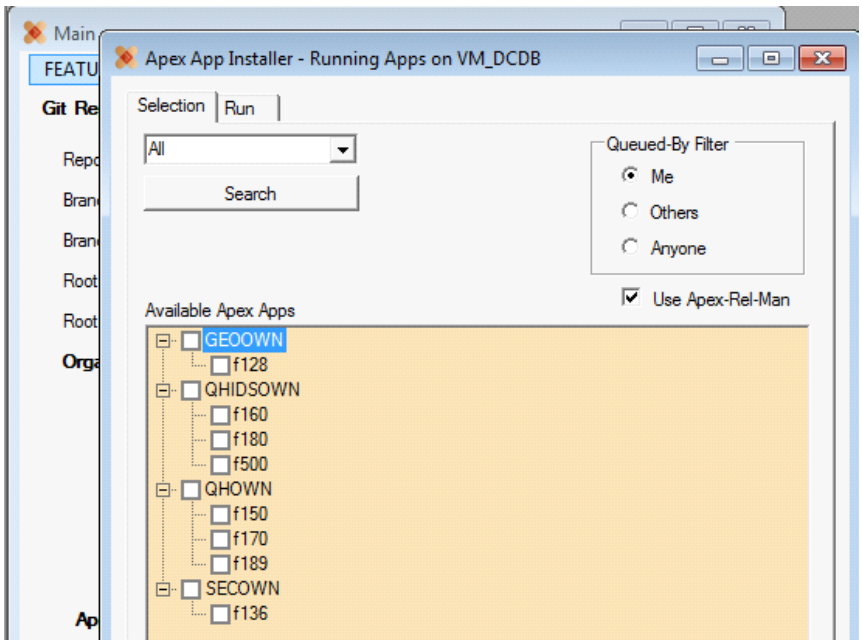
Apex App Installer

Apex App Installer can be launched from the menu, and is also included in workflows.



- Choose All or Queued Apex Apps to start the wizard.

Selection tab

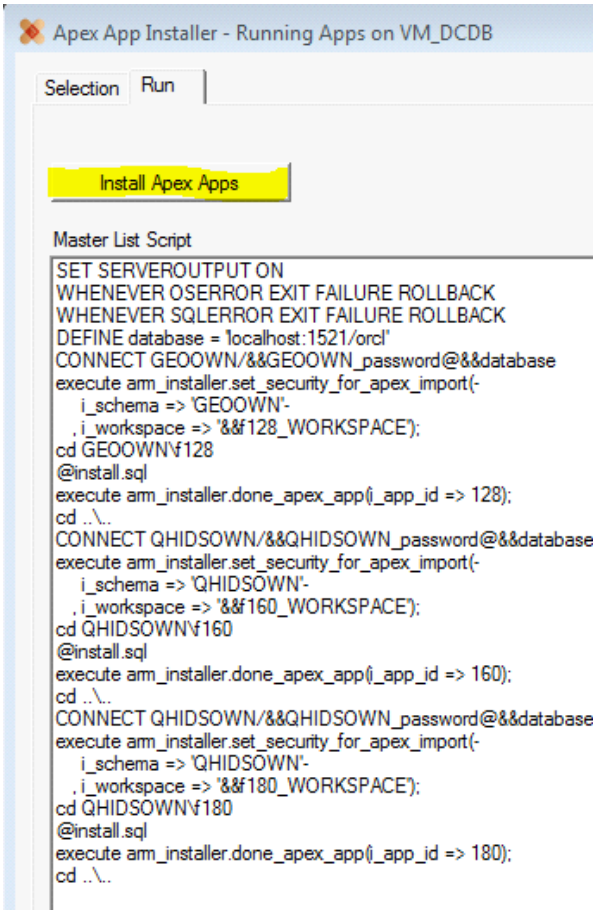


Lists the Apps available in the current repo.

- Choose the Apps to install

Run tab

Displays the master file to run the Apex Apps.



- Install Apex Apps to start write and run the master install script.

Answers

```
Enter value for GEOOWN_password: oracle
Connected.
Enter value for f128_WORKSPACE: QHIDS

PL/SQL procedure successfully completed.

API Last Extended:20180524
Your Current Version:20180524
This import is compatible with version: 20180524
COMPATIBLE (You should be able to run this import without issues.)
```

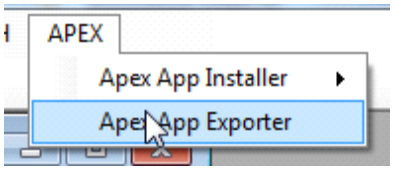
If you a non-existent workspace - will get NO DATA FOUND.

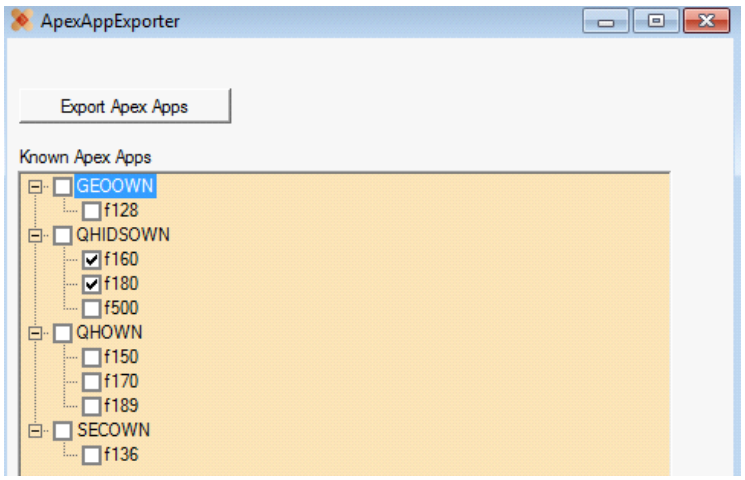
Results

```
...done
DONE APEX APP: 128


PL/SQL procedure successfully completed.

Enter value for QHIDSOWN_password:
```

Apex App Exporter

Apex App Exporter can be launched from the menu, and is also included in workflows.

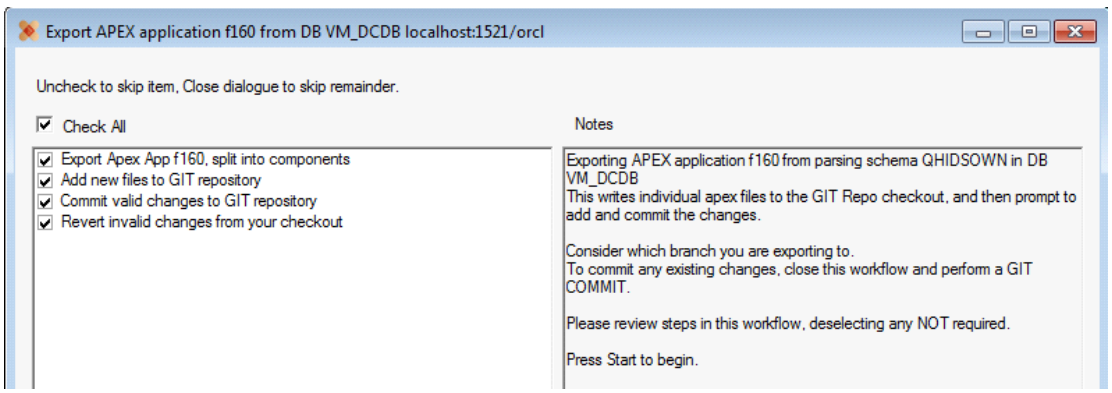




Lists the Apps available in the current repo checkout.
Choose 1 or more apps to Export

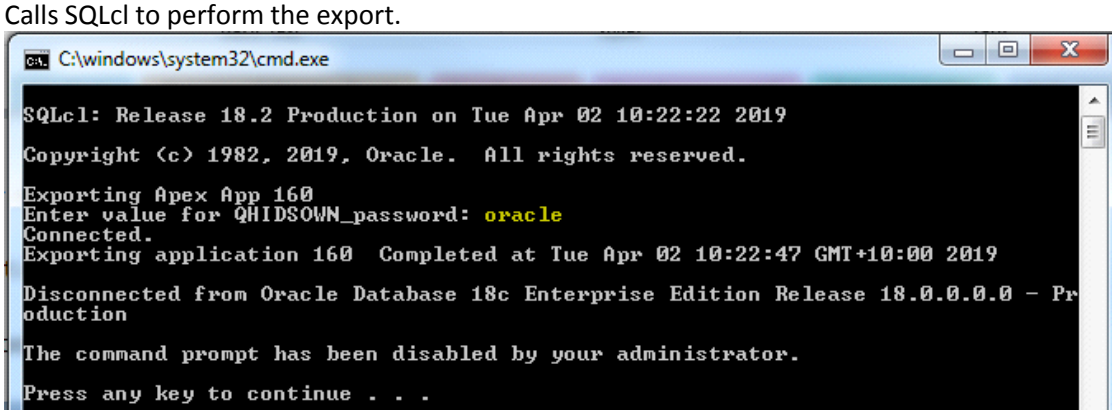
- Export Apex Apps

Calls the Export Apex App workflow, once for each selected App.



Export via SQLcl

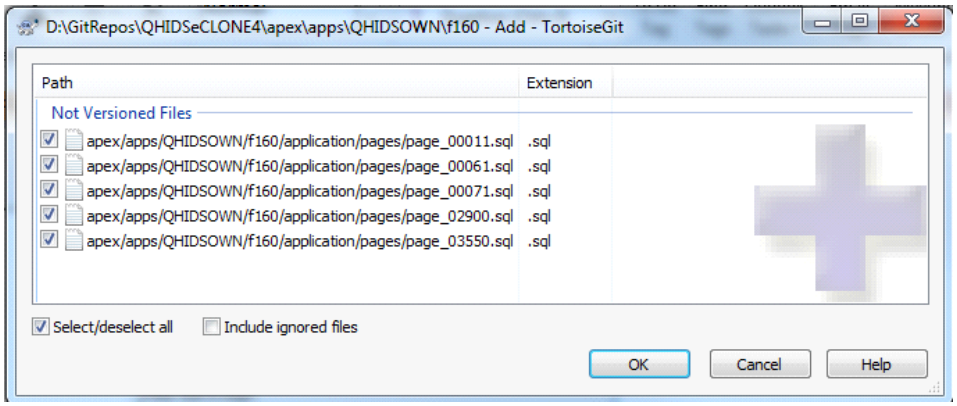
Calls SQLcl to perform the export.



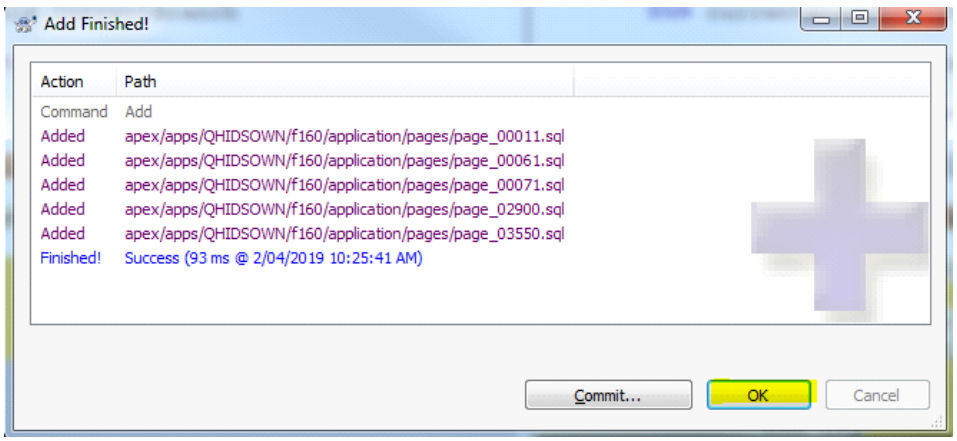
Enter the user password.

- After export - Close the window

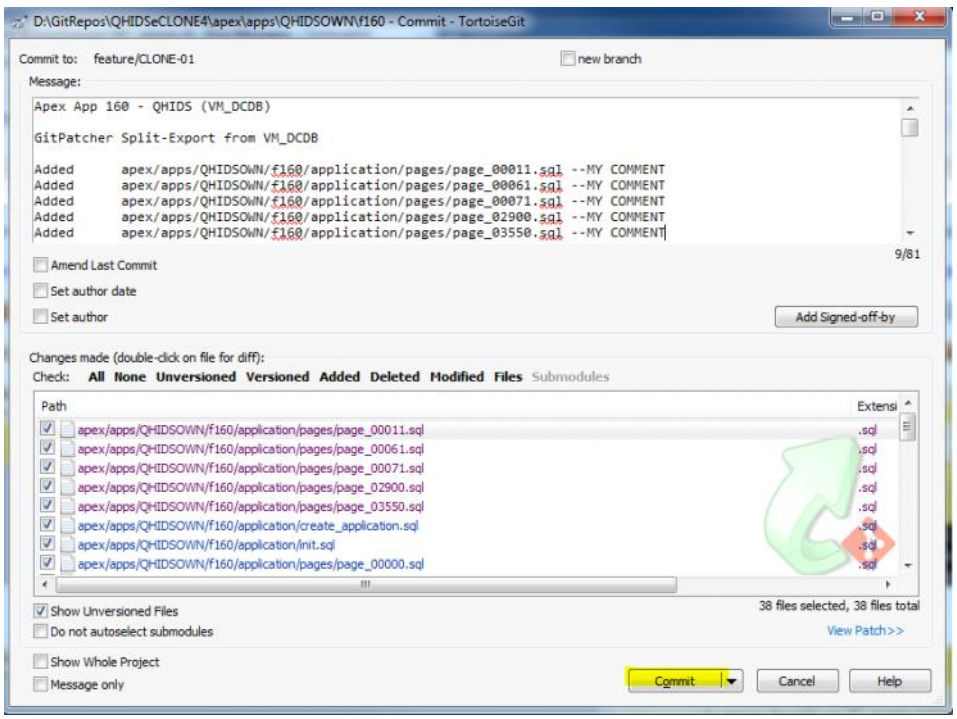
Add via TortioseGit



- Add new files (Cancel if no new files)

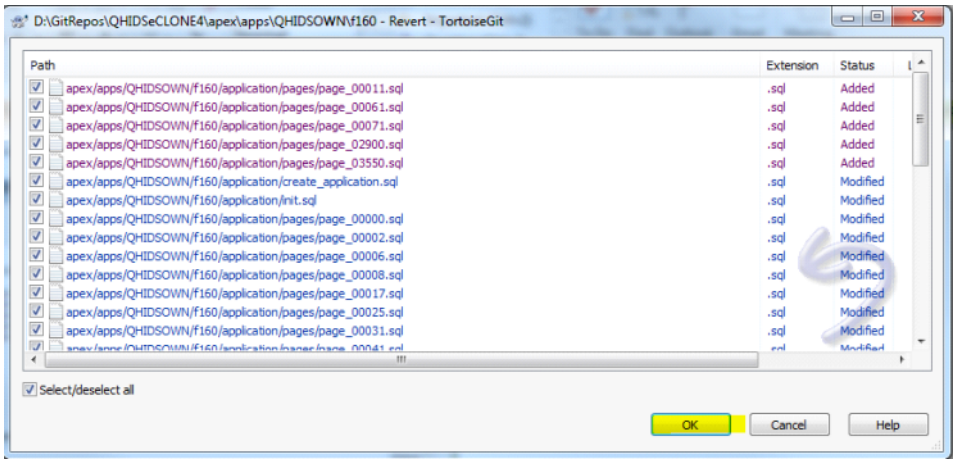


Commit via TortioseGit

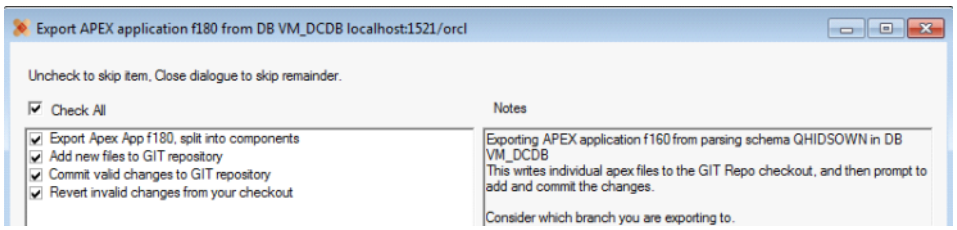


- Commit your changes with your comments. (Untick changes not to be committed)

Revert via TortioseGit



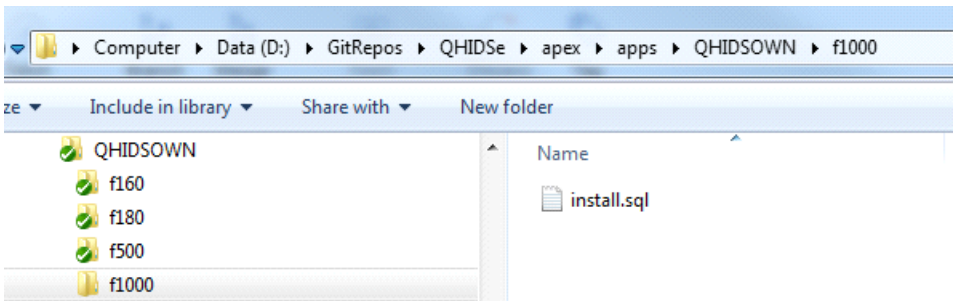
- Revert anything you did not Commit.



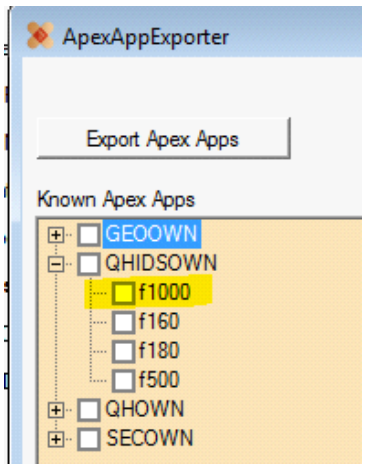
- Repeats for each selected App.

What if I have a brand new app?

- To export a new app (that you have just created) eg 1000
  a. create an empty folder in the local checkout
  b. Copy any install.sql file into the dir.
  c. DO NOT COMMIT.



Apex App Exporter searches for install.sql files, to determine entries for the Known Apex Apps tree.



Close and reopen the wizard to get the new app in the list.

- Export the app - to replace the dummy dir with the real export.

SmartGen app

Getting Started

Configure db-spooler

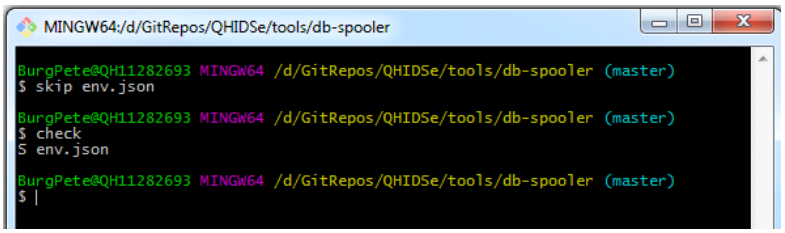
Set username and passwords in the file
D:\GitRepos\QHIDSe\tools\db-spooler\env.json

```
{ "debugging" : ""
 ,"sqlcl_path" : "C:/oracle/sqlcl/bin/sql.exe"
 ,"queued_by" : "BURGPETE"
 ,"default_promo" : 0
 ,"promos" : [
   {
```

You can also set passwords in this file.

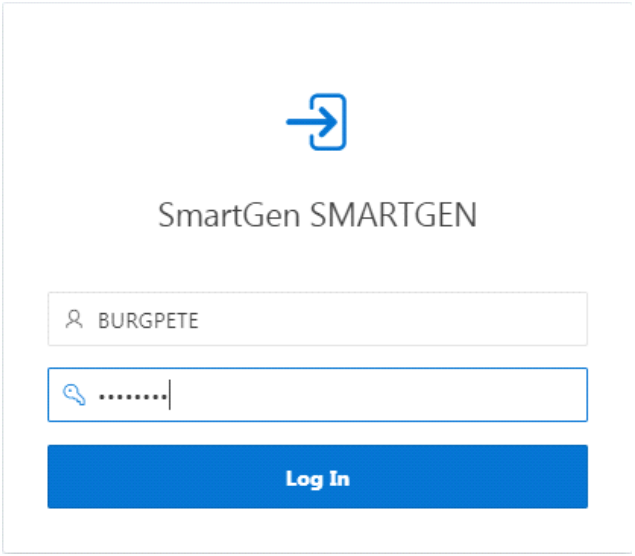
```
,"envs_vm" : [ {
   "env_name"         : "QHIDSOWN"
  ,"choice_label"     : "&1 QHIDSOWN"
  ,"choice_help"      : "VM QHIDSOWN"
  ,"connection"       : "localhost:1521/orcl"
  ,"sid"              : "ORCL"
  ,"dbname"           : "ORCL"
  ,"user"             : "QHIDSOWN"
  ,"pword"            : "oracle"
  ,"db_dir"           : ""
  ,"db_dir_rel"       : "../../database"
  ,"owner_dir"        : "QHIDSOWN"
  ,"apps_dir"         : ""
  ,"apps_dir_rel"     : "../../apex/apps"
  ,"workspace"        : "QHIDS"
 }
```

Also use GitBash and the skip command alias to set skipworktree on the env.json
This depends on the creation of .bashrc that you did in the initial setup for the QHIDSe git repo.

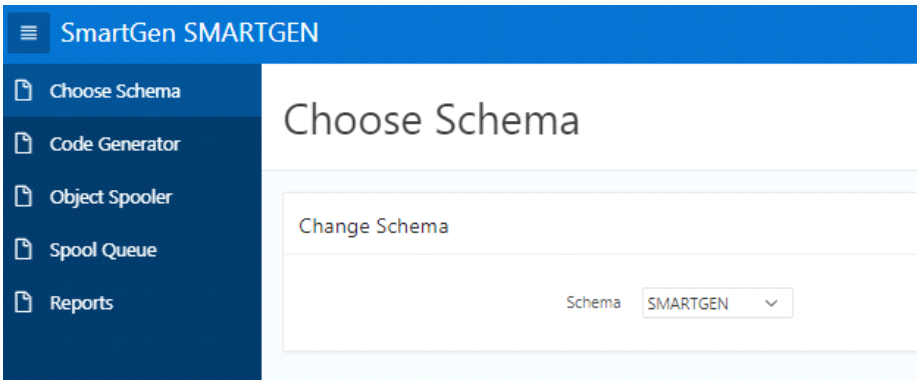


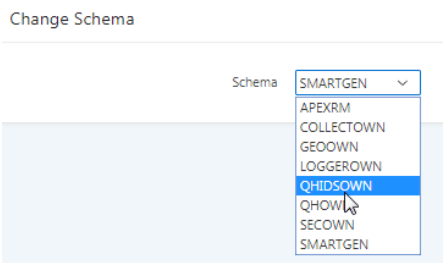
Logon to SmartGen

SmartGen is apex app 146 - in the QHIDS workspace.



Use your QHIDS developer logon

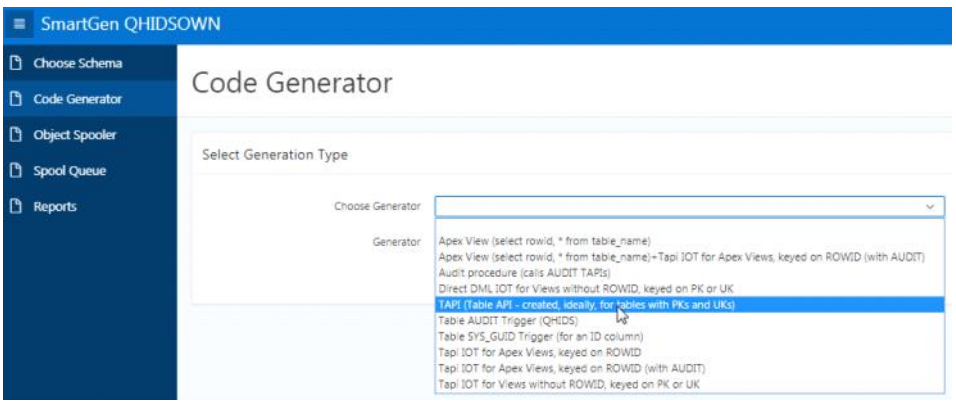




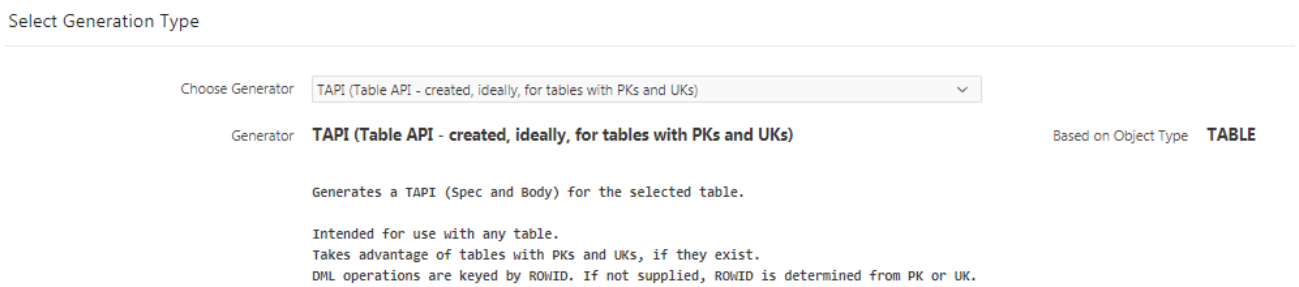
- Choose a Schema

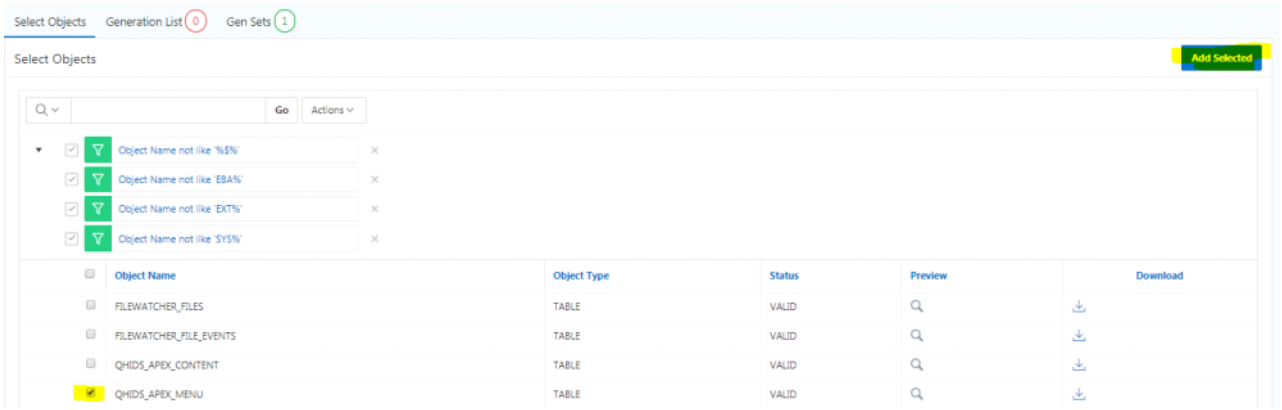
Changes the parsing schema of the app.

Code Generation from Templates

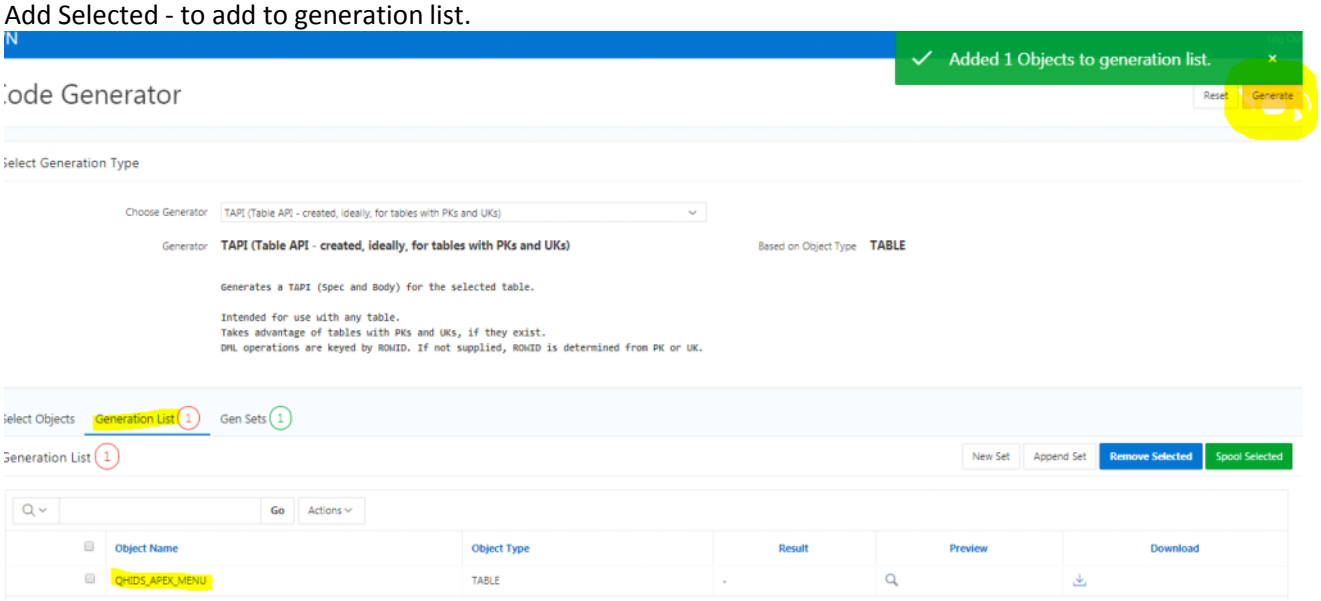


Select the generator type - eg Tapi

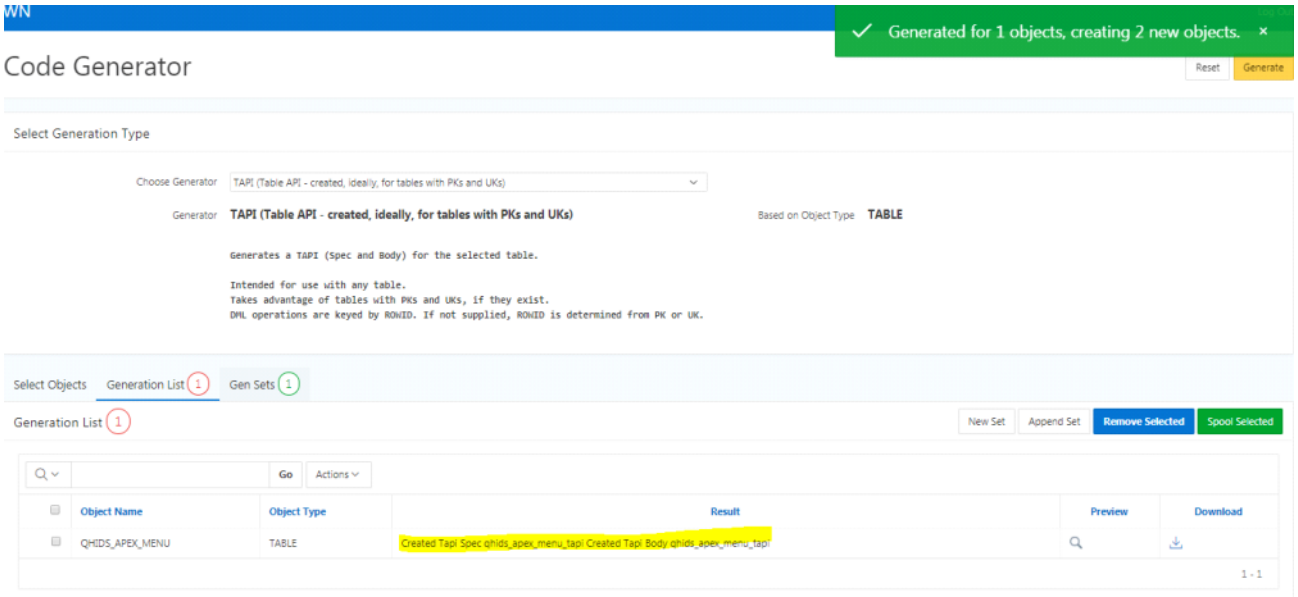




- Choose base objects - in this case tapis are based on tables.
- Add Selected - to add to generation list.

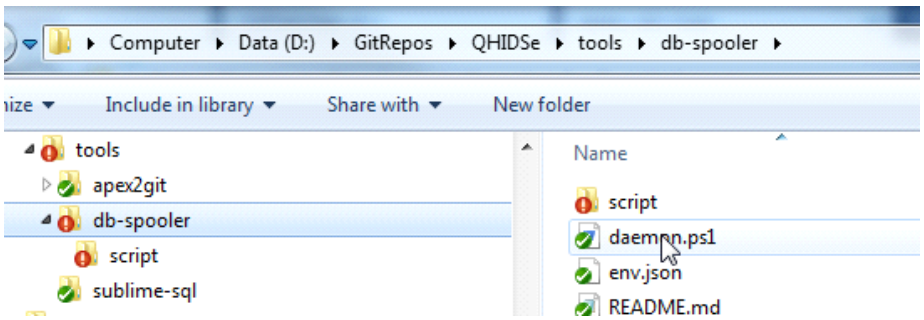


- Generate
- wait for code to generate… time depends on the complexity of the template and number of objects selected.



The valid objects are queued for spooling.

Start the spooler daemon



Right-click on
D:\GitRepos\QHIDSe\tools\db-spooler\daemon.ps1

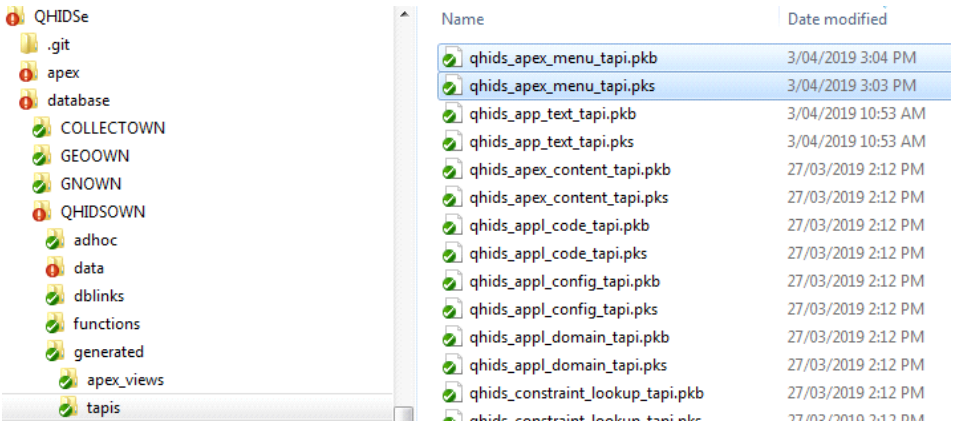
Open with Powershell.
Or Associate ps1 with powershell and then just double-click.

- Choose V to spool from VM

- Choose 1 for spool from QHIDSOWN

There is a collection of sql and javascript files used to spool the output the filesystem.

```
Now using the VM QHIDSOWN connection.

SQLcl: Release 18.2 Production on Wed Apr 03 15:03:38 2019

Copyright (c) 1982, 2019, Oracle.  All rights reserved.

Connected to:
Oracle Database 18c Enterprise Edition Release 18.0.0.0.0 - Production

Spool From Queue


0:spool_from_queue.js
1:D:\GitRepos\QHIDSe\tools\db-spooler/../../database
2:BURGPETE
3:QHIDSOWN
Schema: QHIDSOWN




**********************************************************************
** Matching Objects of Owner QHIDSOWN requestor BURGPETE
**********************************************************************
PACKAGE QHIDS_APEX_MENU_TAPI
PACKAGE_BODY    QHIDS_APEX_MENU_TAPI
**********************************************************************
```



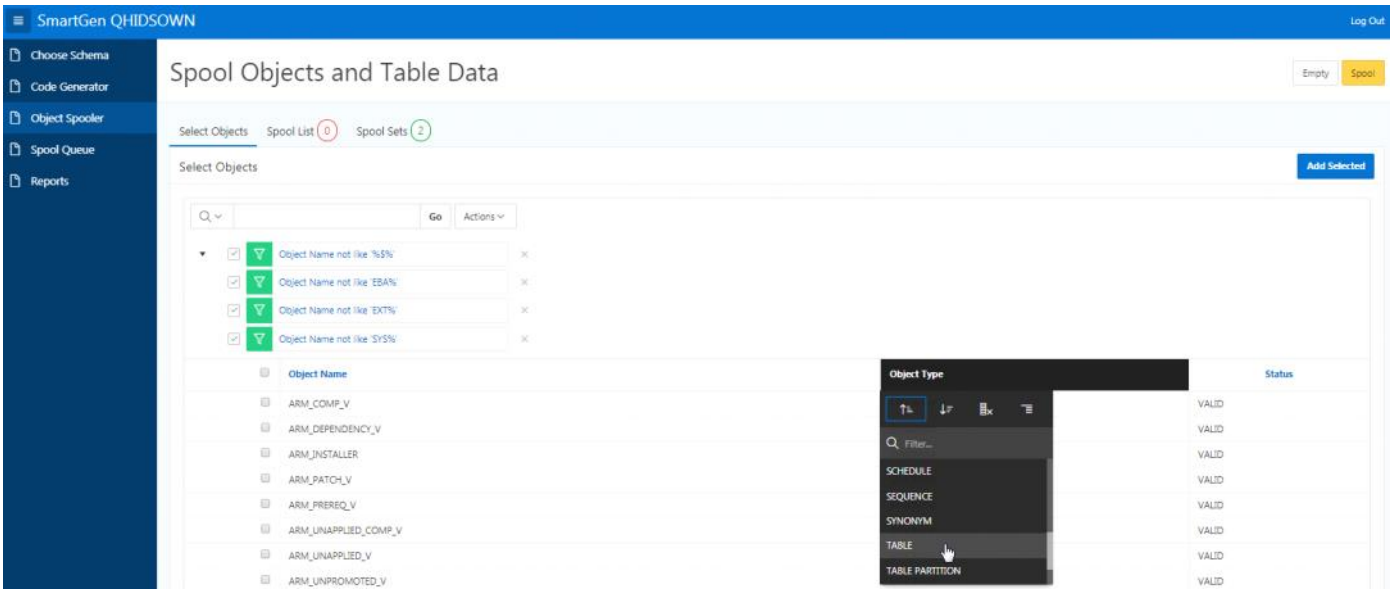
Check the filesystem for the newly spool tapis.

Spooling Objects scripts

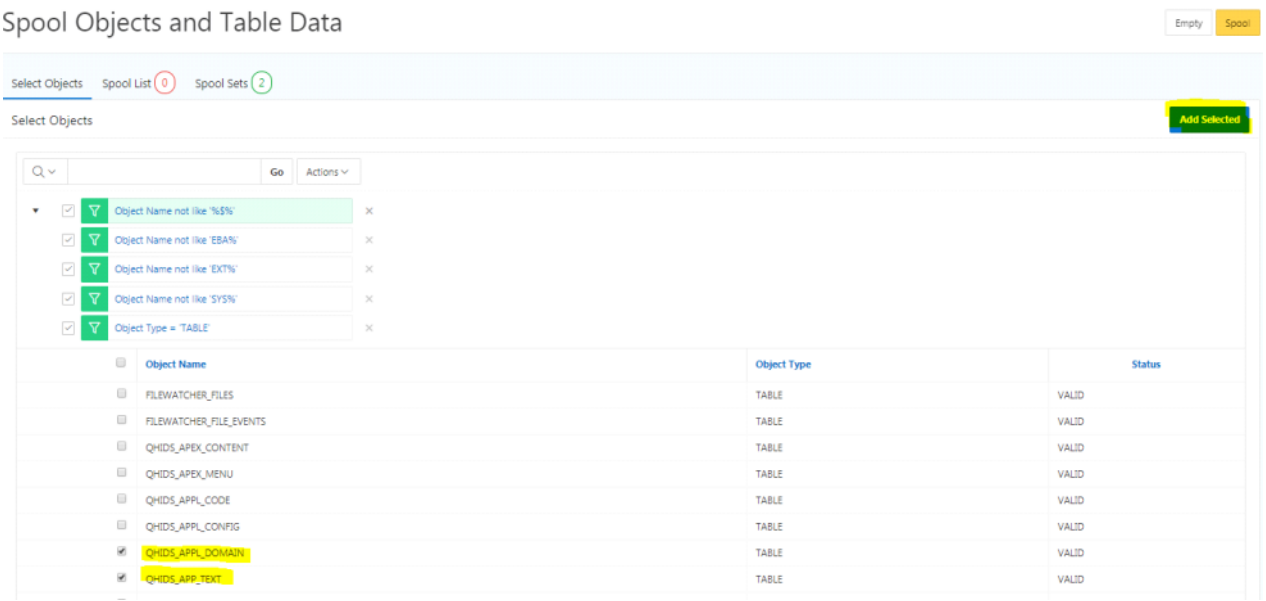
Tables

Table scripts are the most complex object script, because they are an aggregation of a table's

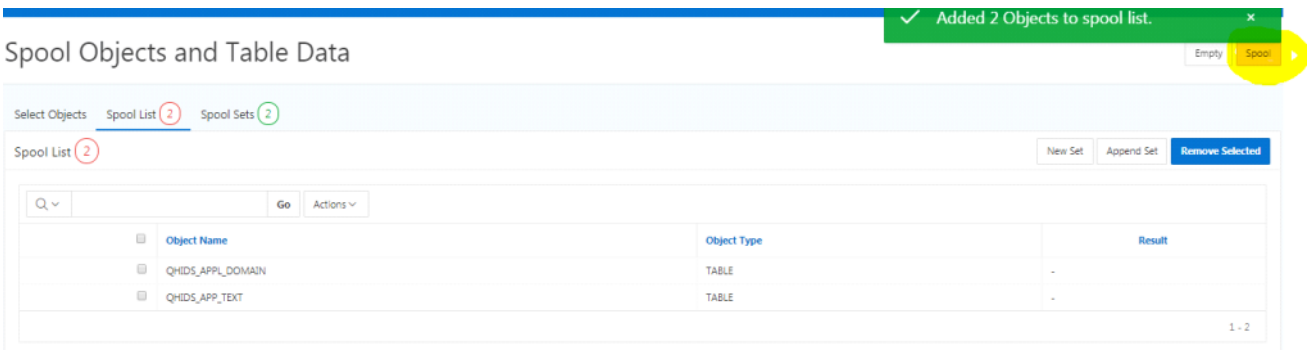
- CREATE TABLE
- INDEXES
- CONSTRAINTS
- REFERENTIAL CONSTRAINTS
- COMMENTS
- MVIEW LOGS
- TRIGGERS
- GRANTS
- SYNONYMS
- SEQUENCES - if mentioned in a column comment.



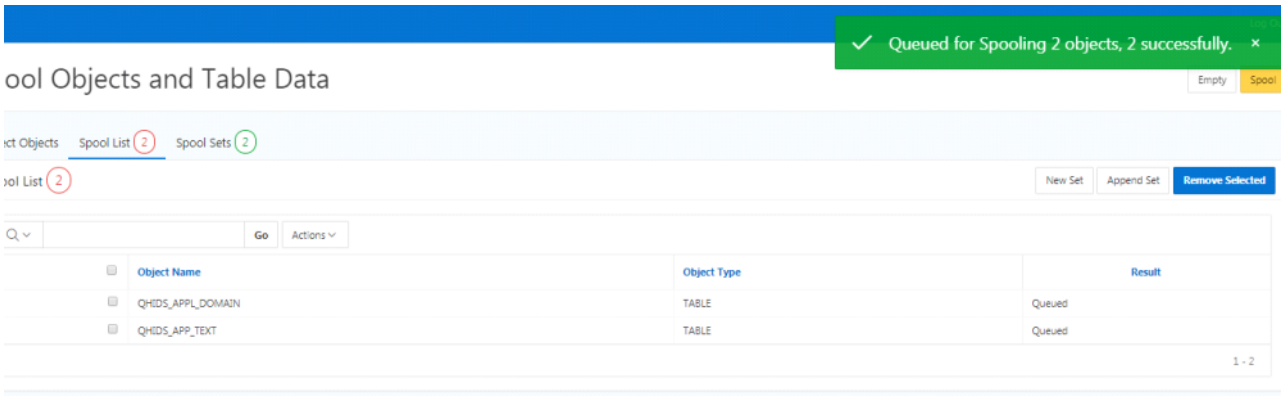
- Filter by Object Type - TABLE



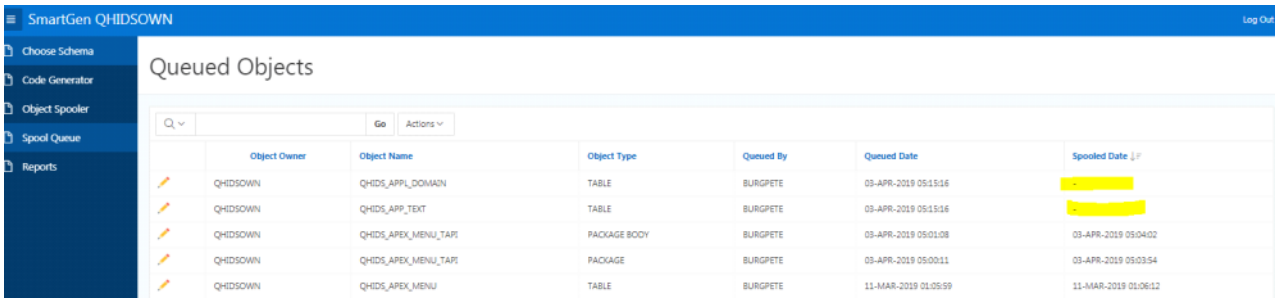
- Choose the tables
- Add Selected



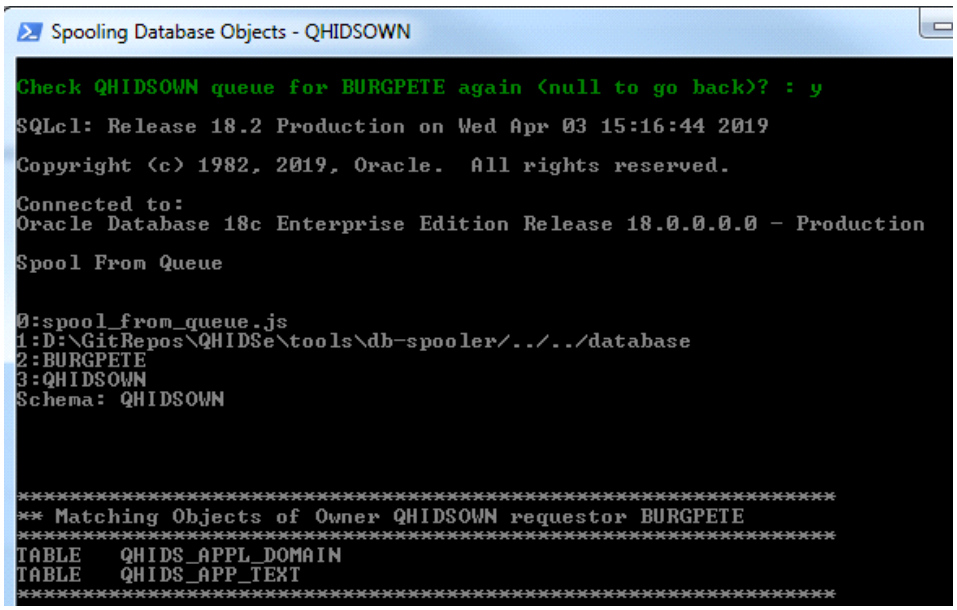
- Spool the table

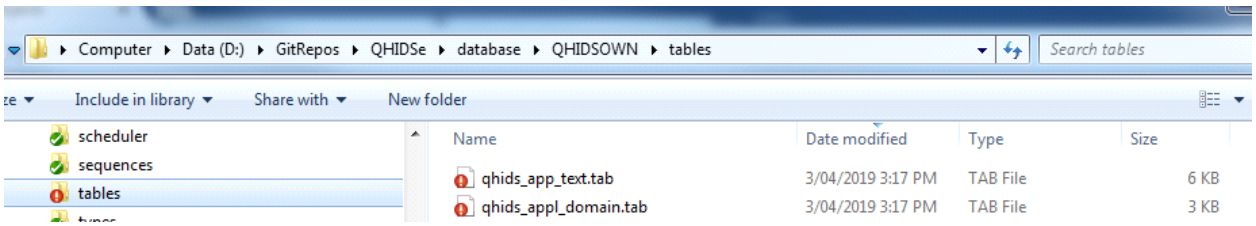


Now added to the spool queue

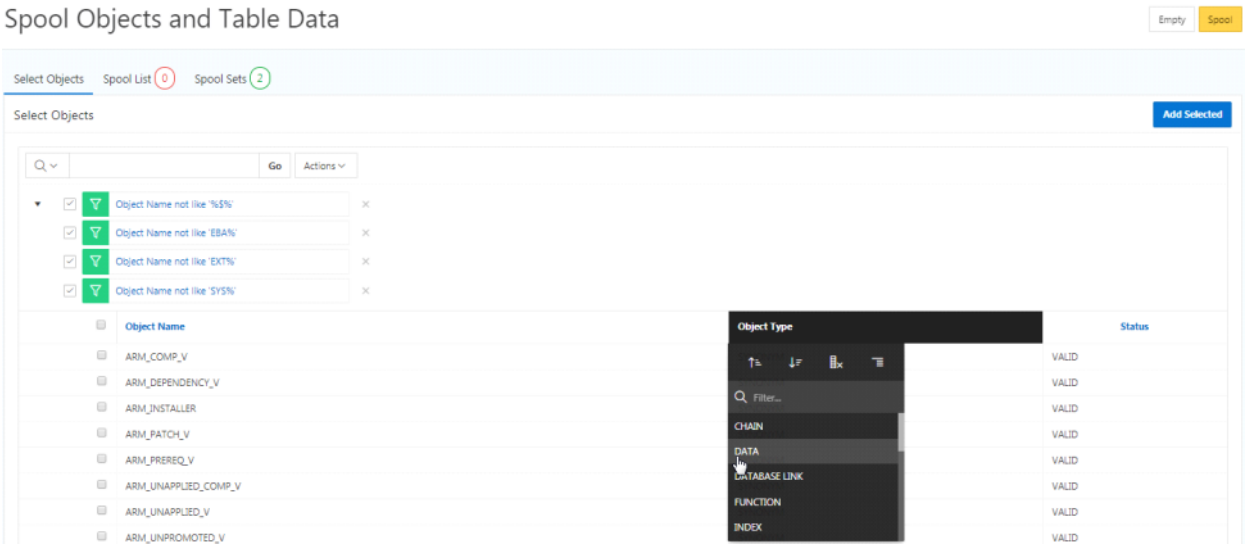


- Use Spool daemon to get the output.

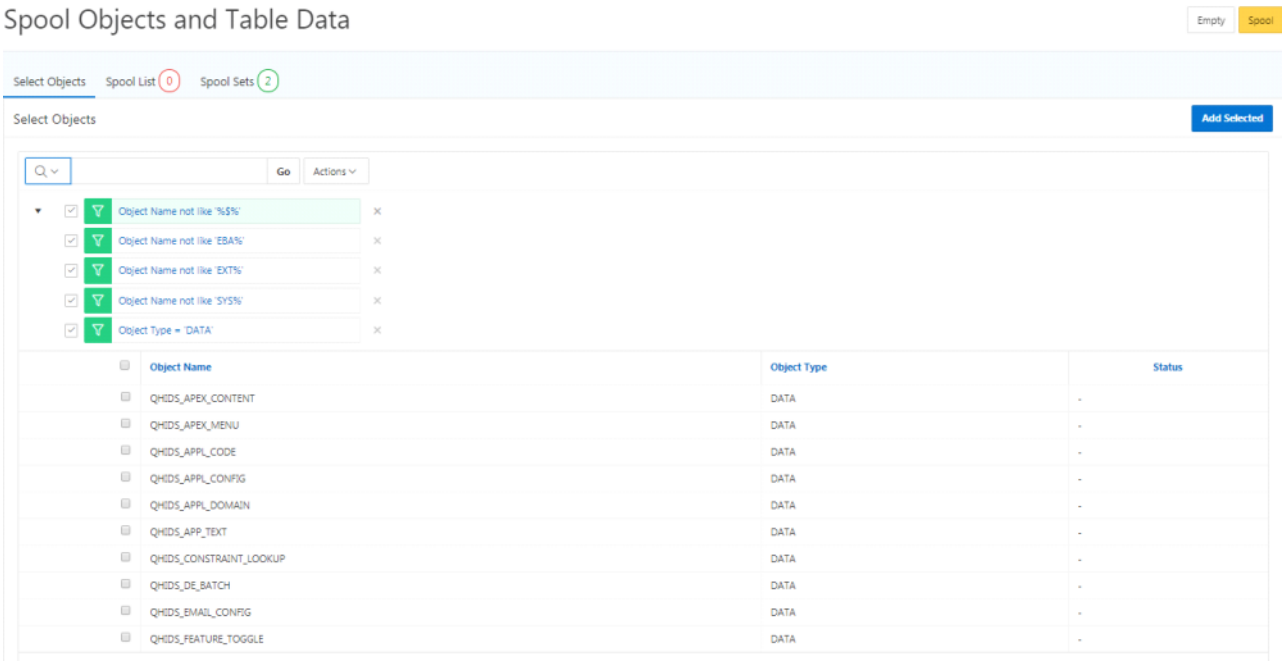




Spooling Data extracts

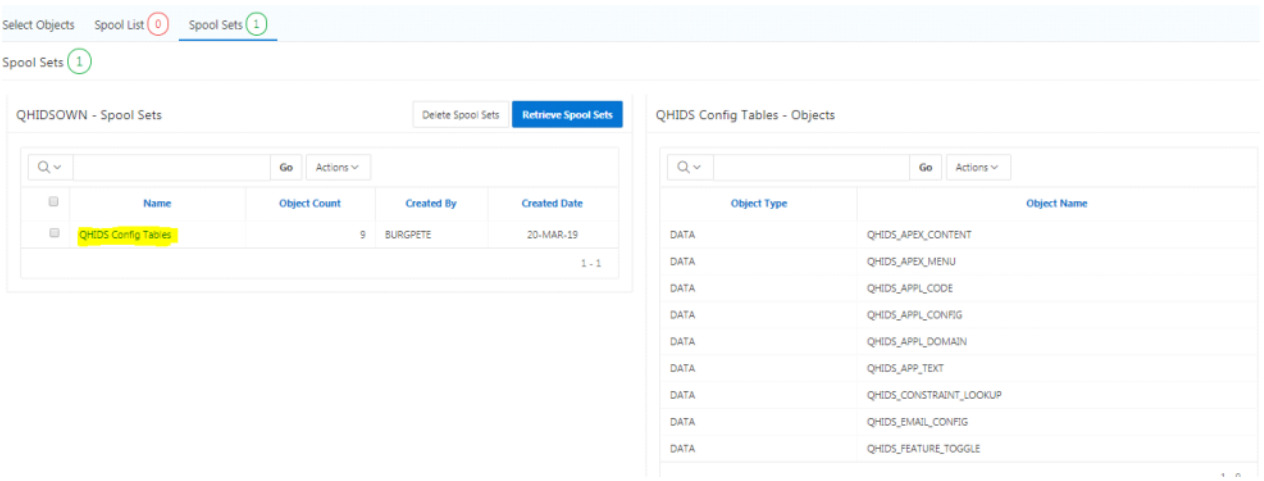


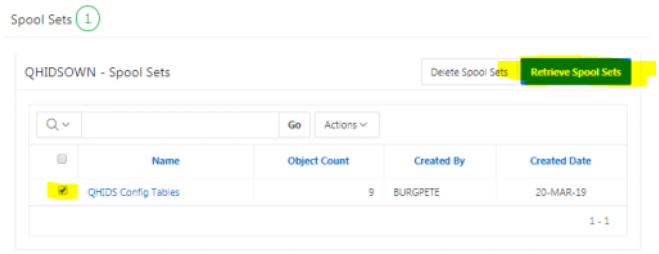
Filter by Object Type - DATA

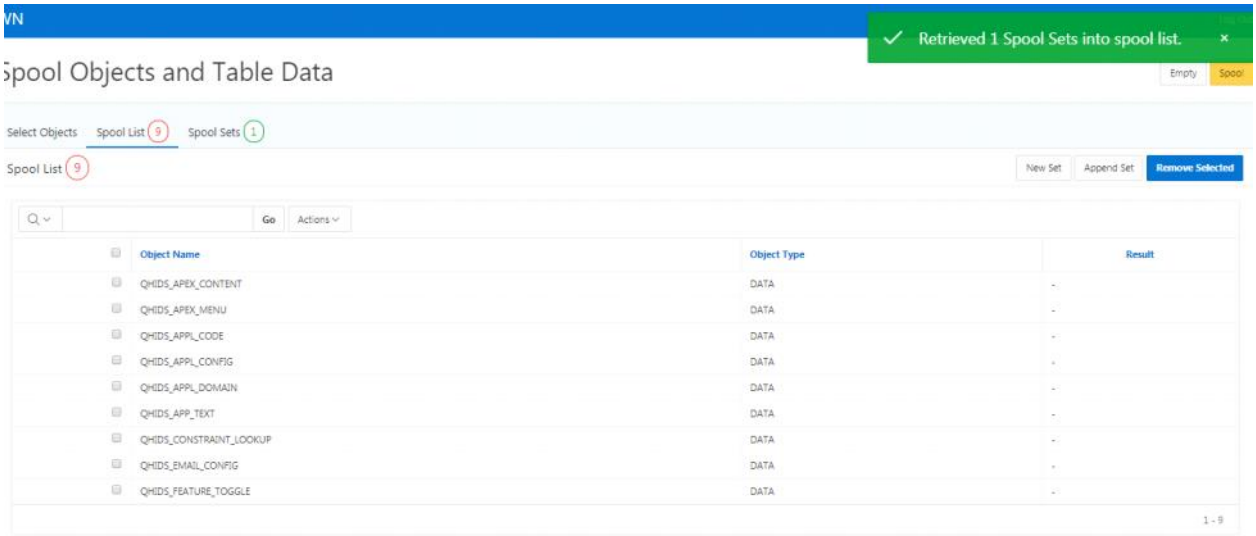


Will display tables that already have a tapi.

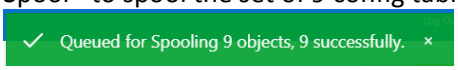
Alternatively - Retrieve a Spool Set from a saved set (Spool Sets tab)

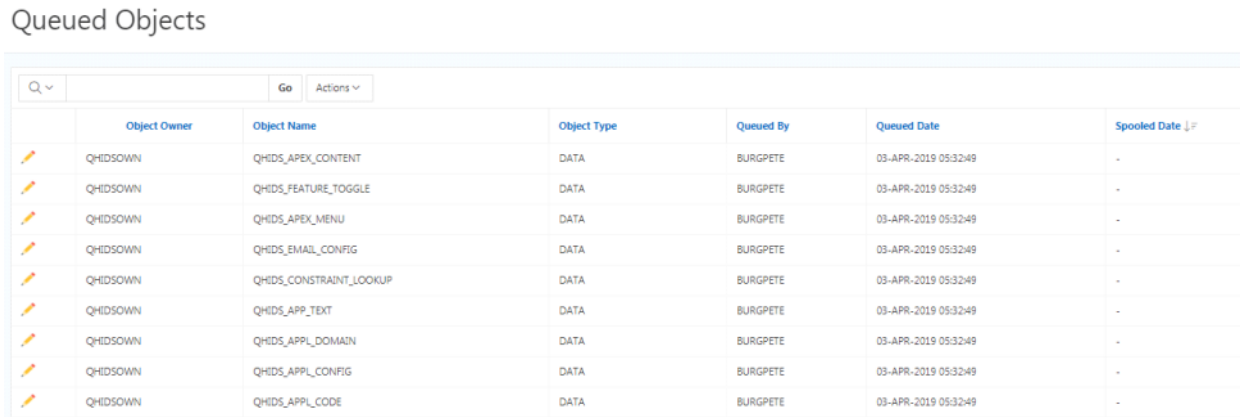




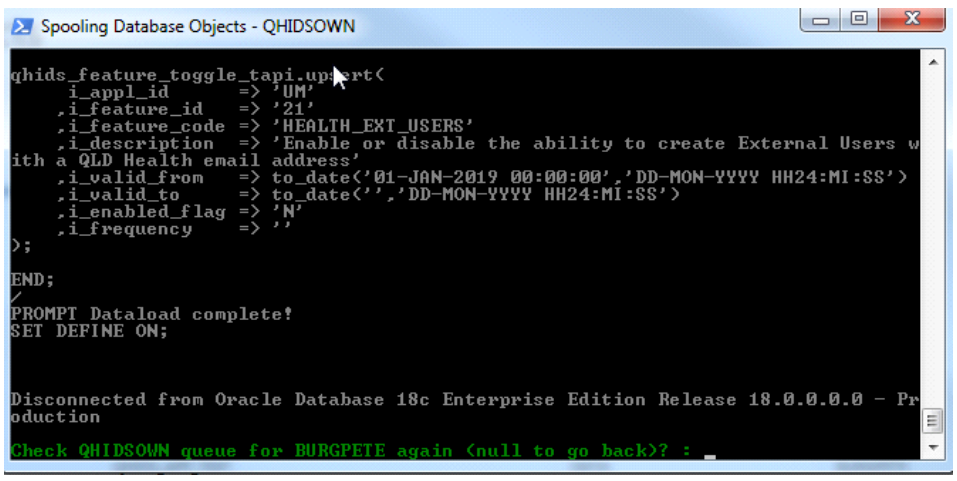


- Spool - to spool the set of 9 config tables.





Use the Daemon again..



Done.

Check files..

qhids_app_text.pop	3/04/2019 3:34 PM	POP File	5 KB
qhids_constraint_lookup.pop	3/04/2019 3:34 PM	POP File	16 KB
qhids_email_config.pop	3/04/2019 3:34 PM	POP File	13 KB
qhids_feature_toggle.pop	3/04/2019 3:34 PM	POP File	1 KB
qhids_appl_code.pop	3/04/2019 3:34 PM	POP File	31 KB
qhids_appl_config.pop	3/04/2019 3:34 PM	POP File	29 KB
qhids_appl_domain.pop	3/04/2019 3:34 PM	POP File	3 KB
qhids_apex_menu.pop	3/04/2019 3:34 PM	POP File	544 KB
qhids_apex_content.pop	3/04/2019 3:34 PM	POP File	22 KB
qhids_apex_menu_DEV.pop	27/03/2019 4:17 PM	POP File	544 KB
qhids_appl_config_DEV.pop	27/03/2019 4:17 PM	POP File	29 KB
qhids_apex_menu_TEST.pop	27/03/2019 2:12 PM	POP File	309 KB
qhids_appl_config_TEST.pop	27/03/2019 2:12 PM	POP File	28 KB

NB - in the list above 2 tables have alternative scripts for DEV and TEST.

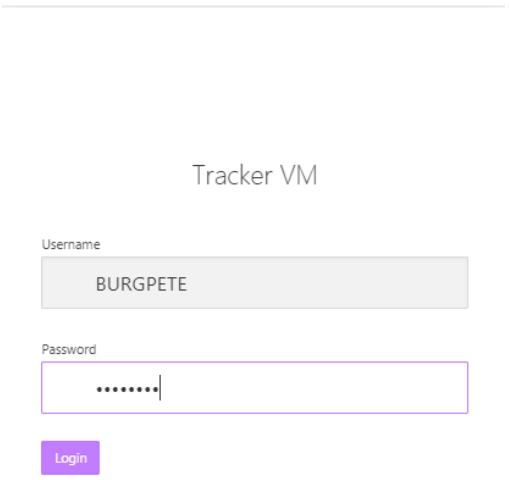
Since I have exported from the VM, which is using the DEV config, I need to merge my changes into the DEV script.
New records can also be copied directly into the TEST script, but updates will have to be manually merged.
(There is still a reconciliation activity to be done for these 2 tables, between DEV, TEST, PAT, PROD.)

Tracker app

Tracker is apex app 2000 - in the APEXRM workspace.

The app is installed in VM, DEV, TEST, and will also be installed in PAT and PROD.
You will need to create yourself a user in APEXRM workspace to login.

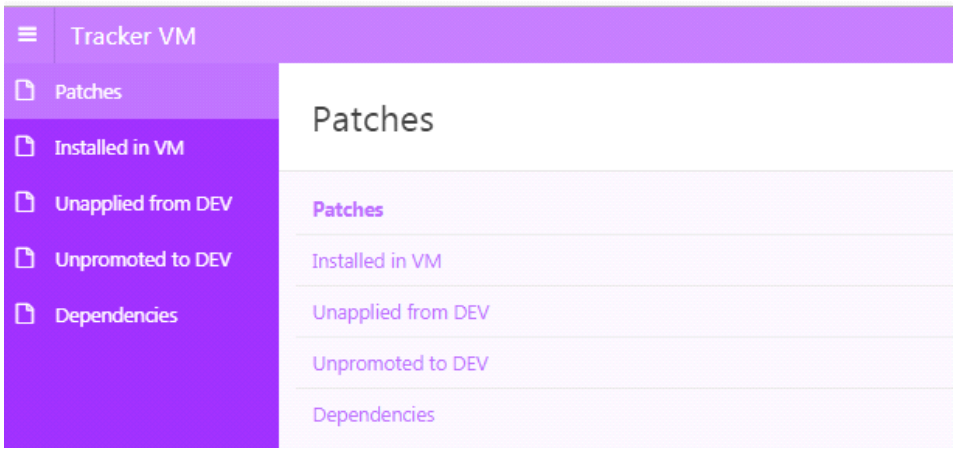
Tracker VM



This app interrogates the APEXRM release management tables, which are populated by the GitPatcher install.sql files.
It's main purpose is to Track and display the state of the databases, but can also be used to change

the settings on a patch after it has been run. Eg to Retire a patch.

Landing Page



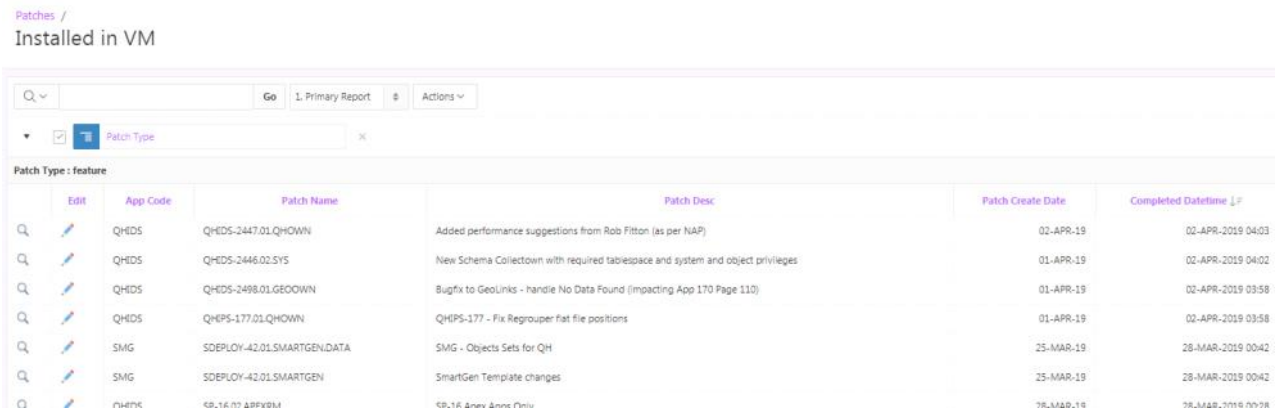
Above I have logged into my VM.
The page names change to reflect the appropriate local and linked environment names.

Eg

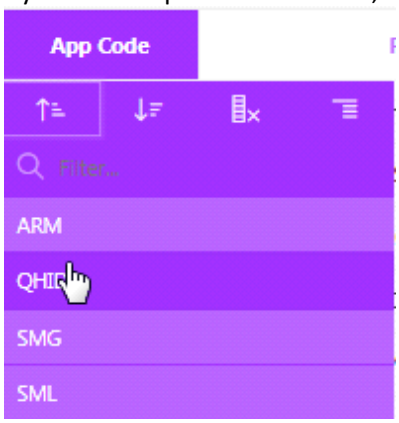
- VM links downstream to DEV and also upstream to DEV.
- DEV links upstream to TEST
- TEST will link downstream to DEV and upstream to PAT.

Installed in VM

This shows all the patches that have been installed in this DB. Listed in DESC order of completed datetime.



By default all patches are listed, but you may find it useful to filter by App Code.



- Select QHIDS to hide the other tools.

Patches /

Installed in VM

Go | 1. Primary Report | Actions

App Code = 'QHIDS'

Patch Type

Patch Type : feature

Edit	App Code	Patch Name	Patch Desc	Patch Create Date	Completed Datetime
	QHIDS	QHIDS-2447.01.QHOWN	Added performance suggestions from Rob Fitton (as per NAP)	02-APR-19	02-APR-2019 04:03
	QHIDS	QHIDS-2446.02.SYS	New Schema Collectown with required tablespace and system and object privileges	01-APR-19	02-APR-2019 04:02
	QHIDS	QHIDS-2498.01.GEOOWN	Bugfix to GeoLinks - handle No Data Found (impacting App 170 Page 110)	01-APR-19	02-APR-2019 03:58
	QHIDS	QHIPS-177.01.QHOWN	QHIPS-177 - Fix Regrouper flat file positions	01-APR-19	02-APR-2019 03:58
	QHIDS	SP-16.02.APEXRM	SP-16 Apex Apps Only	28-MAR-19	28-MAR-2019 00:28
	QHIDS	SP-16.01.GEOOWN	SP-16 GEOOWN	27-MAR-19	28-MAR-2019 00:28
	QHIDS	SP-16.01.QHOWN	SP-16 QHOWN	27-MAR-19	28-MAR-2019 00:28

Patches /

Installed in VM

Go | 2. Detailed | Actions

App Code = 'QHIDS'

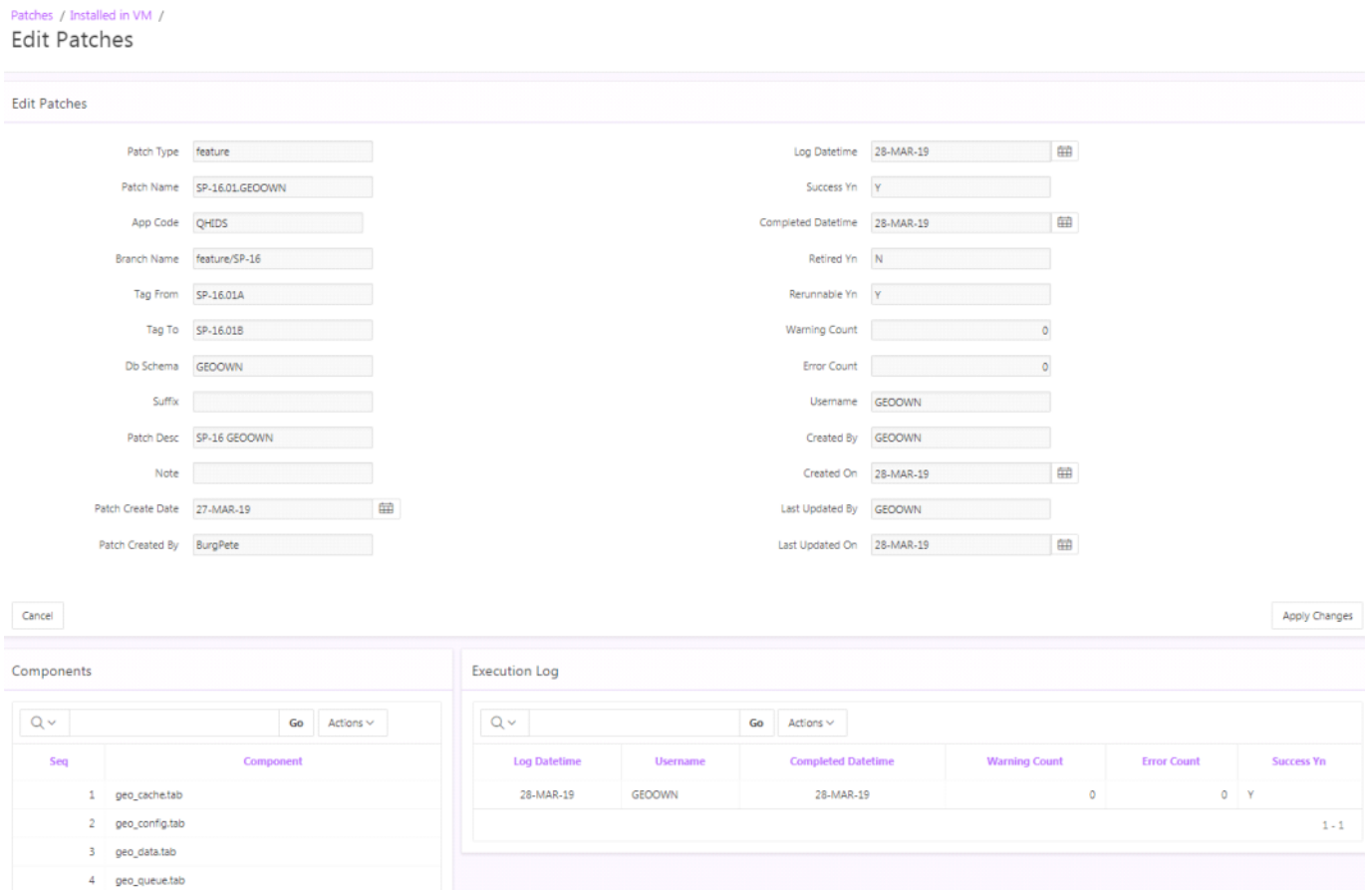
Patch Type

Patch Type : feature

Edit	App Code	Patch Name	Patch Desc	Allow Alt Schema	Patch Create Date	Completed Datetime	Has Pre Reqs	Is Pre Req For	Components	Execution Log
	QHIDS	QHIDS-2447.01.QHOWN	Added performance suggestions from Rob Fitton (as per NAP)	N	02-APR-19	02-APR-2019 04:03	2	0	qh_process_load.pkb	QHOWN 02-APR-2019 04:02
	QHIDS	QHIDS-2446.02.SYS	New Schema Collectown with required tablespace and system and object privileges	N	01-APR-19	02-APR-2019 04:02	1	0	collectown.user	SYS 02-APR-2019 03:58
	QHIDS	QHIDS-2498.01.GEOOWN	Bugfix to GeoLinks - handle No Data Found (impacting App 170 Page 110)	N	01-APR-19	02-APR-2019 03:58	2	0	geo_apex_util.pkb	GEOOWN 02-APR-2019 03:58
	QHIDS	QHIPS-177.01.QHOWN	QHIPS-177 - Fix Regrouper flat file positions	N	01-APR-19	02-APR-2019 03:58	2	0	qh_regrouper.pkb	QHOWN 02-APR-2019 03:58
	QHIDS	SP-16.02.APEXRM	SP-16 Apex Apps Only	N	28-MAR-19	28-MAR-2019 00:28	1	0	-	APEXRM 28-MAR-2019 00:11 APEXRM 28-MAR-2019 00:28
	QHIDS	SP-16.01.GEOOWN	SP-16 GEOOWN	N	27-MAR-19	28-MAR-2019 00:28	3	1	geo_cache.tab geo_config.tab geo_data.tab geo_queue.tab geo_street_type.tab geo_apex_util.pks geo_coding.pks geo_coding_collection.pks geo_collection_data_vw.vw geo_apex_util.pkb geo_coding.pkb geo_coding_ws_secure.pkb	GEOOWN 28-MAR-2019 00:28

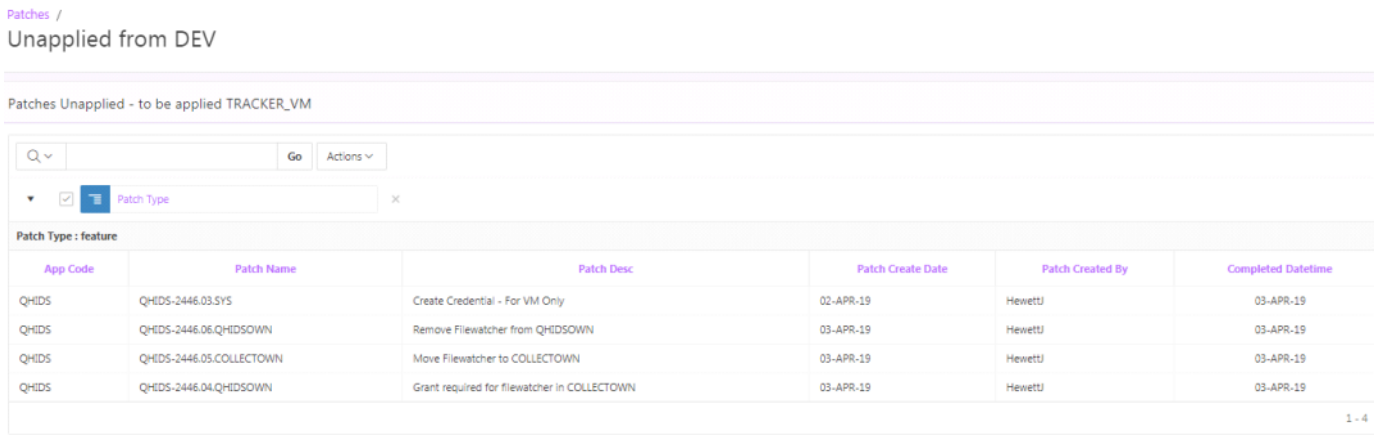
The Detailed report, shows the components lists.

The edit link allows for updates of the patch data, and also lists components and the execution log.



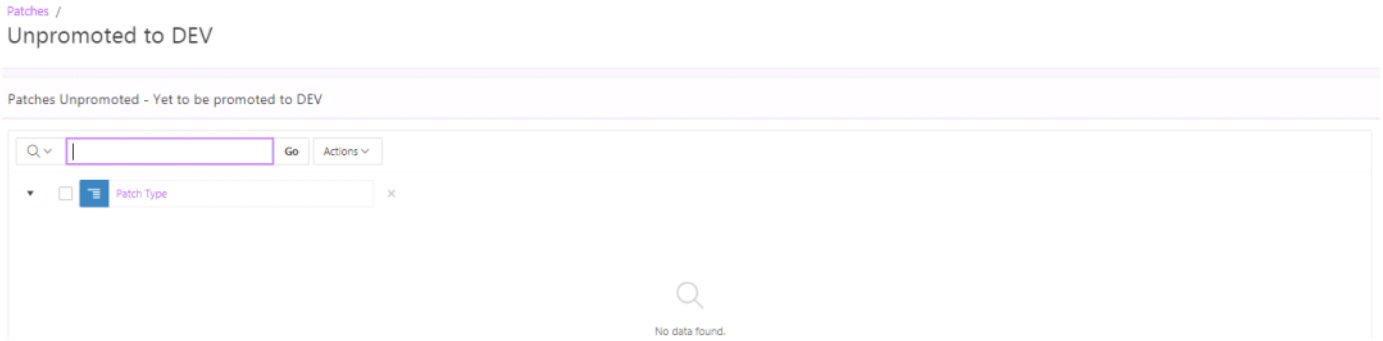
Unapplied from DEV

This shows the patches installed downstream at DEV, and not yet in my VM.



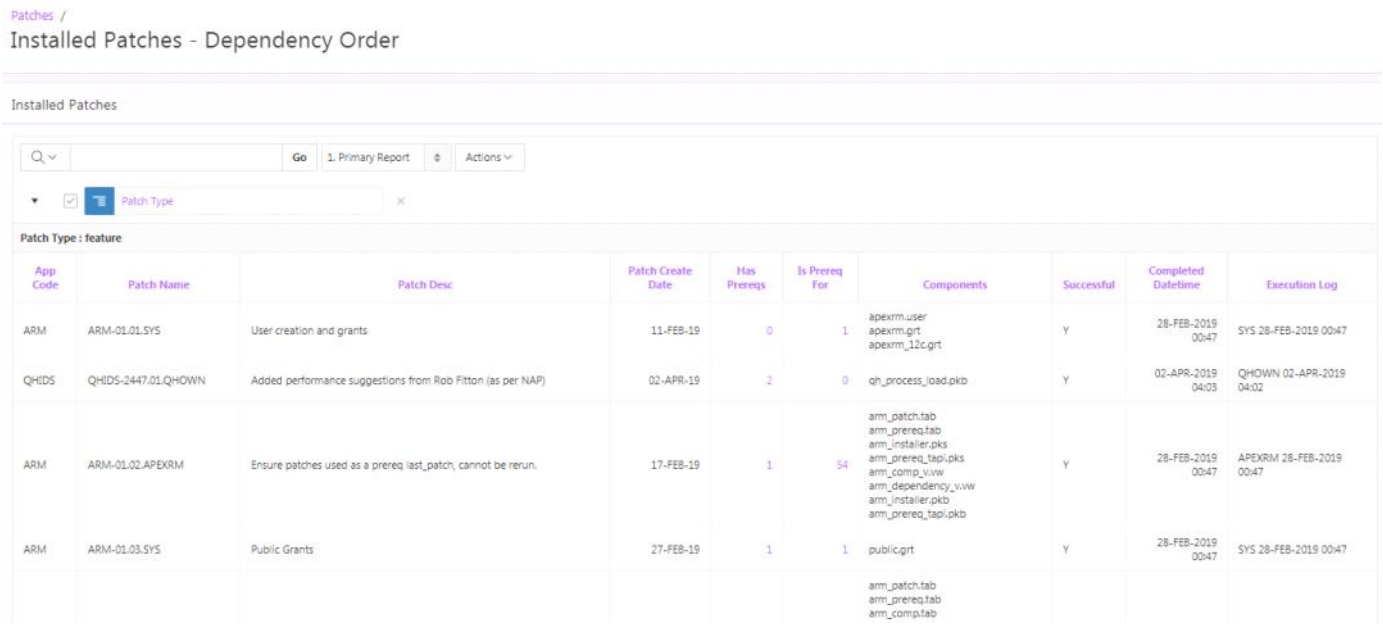
Unapplied to DEV

This shows the patches that are in my VM, but that have not yet been installed upstream at DEV.
In most cases I will have created these patches, and will be promoting them soon.



Dependencies

This shows all patches installed on my VM in order of dependency
Earliest to latest.
If patches appear out of order here, it is possible that some patches were created without any dependency records. In the absence of any dependency records, the completed datetime is meant to be used to determine best order.



Tracker DEV

Similarly you can logon Tracker on DEV with this link

https://oasdcasdev.test.health.qld.gov.au:8888/pls/apex/f?p=2000:LOGIN_DESKTOP::::::