Building and Running Agile BIRD

## Introduction

We describe the steps below to run Agile BIRD, they can easily be run in Microsofts Free VSCode editor, or in Github CodeSpaces (VSCode in the cloud) which requires no local install or local execution of code.

## Install and run steps

### Step 1

Get the SQLDeveloper files that you want to use to create the LDM (Later we can create the Input layer also ) information in Agile BIRD. You can edit the SQLDeveloper model first if you want to play with a locally edited model (Agile BIRD enable local edits of anything in BIRD) or get from branch that you like. Follow the wiki <https://github.com/eclipse/efbt/wiki/Export-Files-from-SQLDeveloper>

Put these files in the resources director (birds\_nest/resources)

### Step 2

Get the detailed subtyping information from SQLDeveloper using the script get\_arcs.txt under /birds\_nest/sqldev\_scripts by following the wiki <https://github.com/eclipse/efbt/wiki/Running-the-Get_arcs-script>

Put the result in resources director (birds\_nest/resources)

### Step 3

Get the csv files needed from the BIRD websites technical export , mostly related to output information (e.g. report templates) by following the wiki <https://github.com/eclipse/efbt/wiki/Downloading-Files-from-the-ECB-website>

In addition to files requested in the wiki you will also need cube\_mapping.csv, member\_mapping.csv, variable\_mapping.csv, variable\_mapping\_item.csv,

Put these in the resources director (birds\_nest/resources)

### Step 4

Get or create some configuration information.

Currently there are 4 configuration files, and ready made versions are available at <https://github.com/regcommunity/BIRD_Lite>

Theses files are [in\_scope\_reports\_FINREP\_REF.csv](https://github.com/regcommunity/BIRD_Lite/blob/main/in_scope_reports_FINREP_REF.csv) , [table\_part\_ldm\_definitions\_FINREP\_REF.csv](https://github.com/regcommunity/BIRD_Lite/blob/main/table_part_ldm_definitions_FINREP_REF.csv) ,

And [table\_part\_to\_reference\_category\_FINREP\_REF.c](https://github.com/regcommunity/BIRD_Lite/blob/main/table_part_to_reference_category_FINREP_REF.csv)sv

Documentation on creating these is to follow, basically they list the reports in scope, and show which tables in BIRD input layer or LDM are used for each products used in reports (like debt securities used in Finrep) .

Put these in the resources director (birds\_nest/resources)

### Step 5

Setup environment:

Install Python if not installed

Install an editor (tested with VSCode on windows and Github codespaces which is VSCode on Linux in the cloud in a web-browser)……current paths in the code should work well in Github codespaces Github codespaces but can be a bit slow compared to a good local development machine.

Install prerequisites:

Run these commande in the terminal:

python -m pip install Django

pip install pyecore

pip install unidecode

change to the birds\_nest directory:

cd birds\_nest

setup environment variables:

unix/linux:

PYTHONPATH='.'

export PYTHONPATH

DJANGO\_SETTINGS\_MODULE='birds\_nest.settings'

export DJANGO\_SETTINGS\_MODULE

windows:

$env:DJANGO\_SETTINGS\_MODULE='birds\_nest.settings'

$env:PYTHONPATH='.'

### Step 6

configure for local paths:

The python files that will run are all in the birds\_nest/agilebird/entry\_points/ directory.

These have some hard coded paths which currently work well If running the code in github codespaces. You will need to change these to local paths if running locally, also if running on windows you will need to use ‘\\’ as the path sperator.

For example when I use windows I search the code for /workspaces/efbt/birds\_nest/

And replace with C:\\Users\\neil\\freebirdtools-develop-July11\\git\\efbt\\birds\_nest\\

### Step 7

Create the Django file form SQLdevelopor export files:

Delete existing birds\_nest\results\models.py and birds\_nest\results\admin.py if they exist.

Run ‘python agilebird/entry\_points/create\_django\_models.py’ (note path separator will be different in windows/linux)

You will see a few warning messages like ‘association with cardinality of N does not have an opposite relationship’ which is fine.

Look in birds\_nest\results directoryfor the results which is a models.py file and an admin.py file.

Copy the content of models.py into birds\_nest\agilebird\ldm\_models.py

Delete these 2 lines (which are accidental included twice ) from ldm\_models.py

*from* .ldm\_models *import* NT\_MMBR\_EU

admin.site.register(NT\_MMBR\_EU)

Append the contents of the birds\_nest\results\admin.py file (except for the first line ‘from django.contrib import admin’ to birds\_nest\agilebird\admin.py

### Step 8:

create the empty sqllite database:

First delete the old db (birds\_nest/db.sqlite3) and any migrations file birds\_nest\agilebird\migrations\0001\_initial.py if they exist.

Run the Django scripts to create the db

python manage.py makemigrations agilebird

python manage.py migrate

you will see some warnings like ‘ SQLite does not support comments on columns (db\_comment) ‘ which is fine

### Step 9:

make the admin user:

python manage.py createsuperuser

complete the prompts for user name , email and password

check that you can login to the UI:

python manage.py runserver

go to the website link shown, add /admin to the link

<http://127.0.0.1:8000/admin> (don’t forget the /admin at the end or you will see a 404 page not found error)

type CTRL-C to kill the webserver, you can start it again later.

### Step 10

create the basic info in the sqllite database from the website export:

python .\agilebird\entry\_points\import\_basic\_info\_from\_website.py

### Step 11

Create the Initial reference content from SQLDEveloper:

python .\agilebird\entry\_points\import\_reference\_info\_from\_sqldev.py

### Step 12

Optionally supplement the Initial reference content from the website export:

python .\agilebird\entry\_points\import\_reference\_info\_from\_website.py

### Step 13

create the Non-refernce content in the sqllite database from the website export:

python .\agilebird\entry\_points\import\_non\_reference\_info\_from\_website.py

### Step 14

Add the mappings to the database from the export:

python .\agilebird\entry\_points\import\_input\_output\_analysis\_from\_website.py

### Step 15

Create the output layers and combinations from the mapping and non-ref info:

python .\agilebird\entry\_points\create\_output\_layer\_concepts.py

### Step 16

Run the gap analysis to create the transformation metadata from input layer/LDM tp output layer:

python .\agilebird\entry\_points\create\_transformations\_metadata.py

### Step 17

Create executable Python transformations from the transformation meta data:

Currently step 17 is under development, it works in the previous Java version of Eclipse Free BIRD Tools so we ae porting that to Python. It is largely an automated process but needs some manual editing because the BIRD transformation are not fully executable (e.g. derivation rules are in Pseudocode, but the lineage is sored in a forma way)