# Introduction

This document contains instructions for data migration from the original Oracle SFA database to the new SQL database.

# Prerequisites

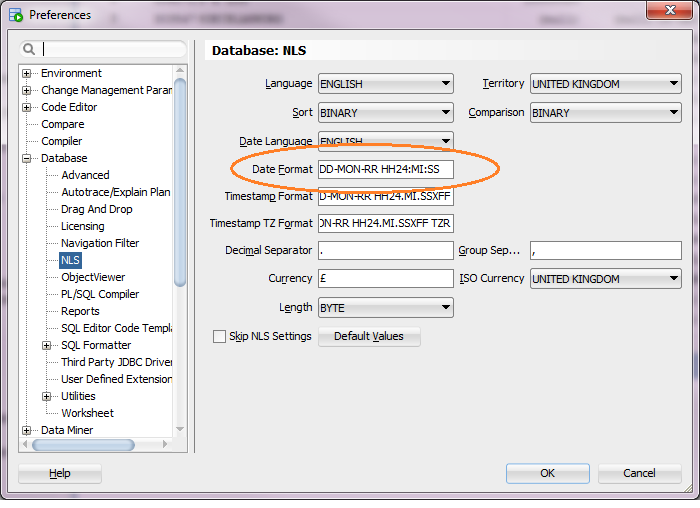
SQLDeveloper is required set up and connected to the original Oracle database(s) in order to export data as CSVs.

* A published but empty SQL2014 database. This should be published from the solution where it will be set up with the necessary lookup data, but will be devoid of any existing providers and course data.
* MS SQL Management Studio installed connected to the new empty database.
* Four script files that will prepare and migrate the data over. These are in the solution under the Data Migration folder numbered 1 to 4.
* These instructions.
* While the import only takes around 5 minutes, the preparation may take up to an hour, but can be done in advance provided the original data is not changing.

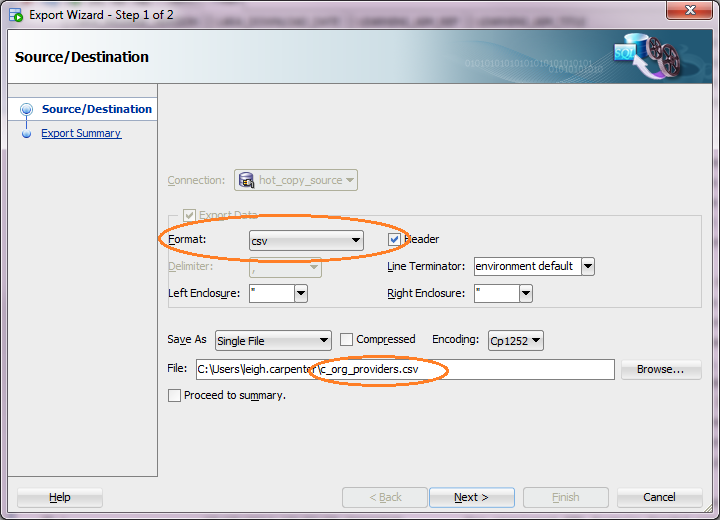
# Preparation

Preparation consists of exporting the data tables from the original Oracle database, and importing them as temporary extra tables in to the new SQL Database. Finally 2 scripts are run to validation the data and add various indexes and functions to support the data migration.

## Export data from the original Oracle database

Ensure that the SQLDeveloper date formatting is set to format date with the time on any column with a datetime object, as otherwise the time portion is lost. This can be done from Tools -> Options then select NLS under the Database option.

Export all the tables listed below from the Oracle database as CSVs by using SQLDeveloper. To export, view the data for the table in SQLDeveloper, right click the data and choose Export…, make sure the format is CSV and the file name is named after the table (see example), click Next then Finish. Do this for all required tables in turn.



Tables to export from **hot\_ndlpp**

c\_org\_providers

c\_provider\_attribute\_values

c\_provider\_tracking\_codes

c\_provider\_users

c\_providers

c\_venue\_attribute\_values

c\_venues

o\_course\_attribute\_values

o\_courses

o\_learning\_aims

o\_opp\_attribute\_values

o\_opp\_locations

o\_opp\_start\_dates

o\_opportunities

s\_lad\_qt\_map

s\_postcode\_locations

s\_regions

u\_users

w\_address

w\_provider\_address

Tables to export from **hot\_copy\_source**

ld\_awarding\_organisations

ld\_learning\_aim

ld\_lars\_lrnaimreftype

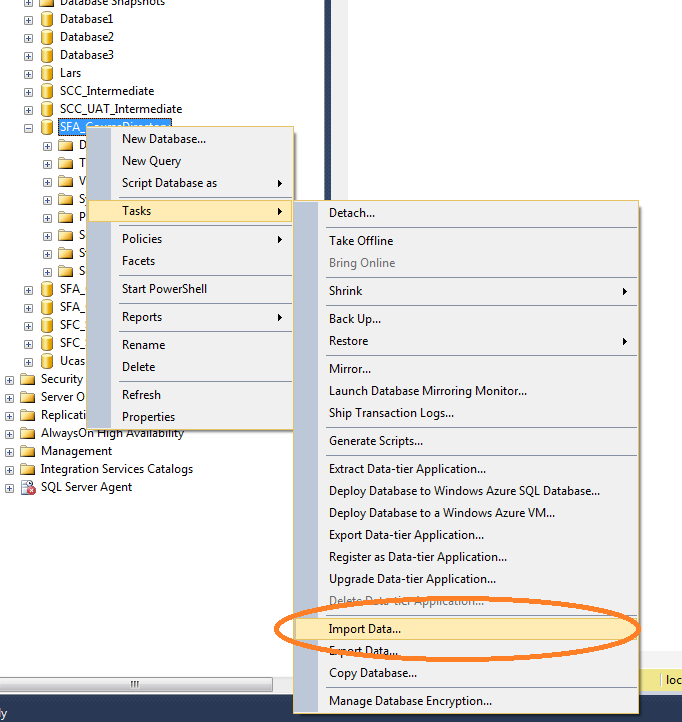
s\_learning\_aims \*

\* s\_learning\_aims is troublesome to import, when importing following the instructions below set the following fields to Strings, Section\_96\_Apprvl\_Status\_Code, Skills\_Ending\_Apprv\_Status\_Code, SSA\_Tier1\_Code, SSA\_Tier2\_Code, SSA\_Tier3\_Code

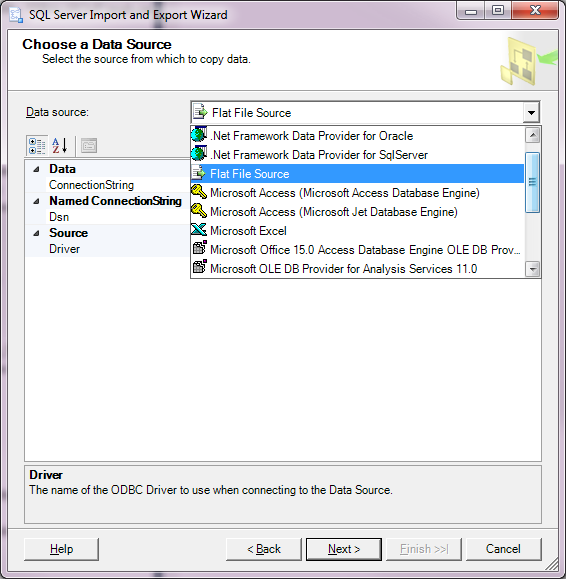
## Import the data to the new database

Import the CSVs to the new SFA database. The database should have been created from the Database project which will have pre-populated common lookup tables, but is otherwise empty.

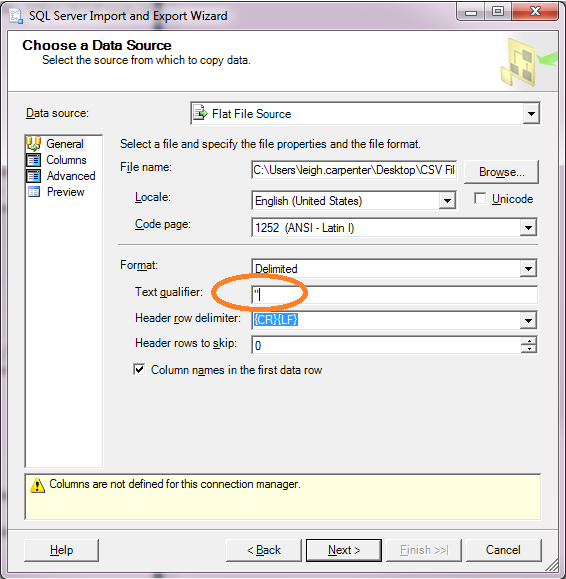
Import each CSV in turn from SQL Management Studio. Right click the database to import to, select Tasks then Import Data…



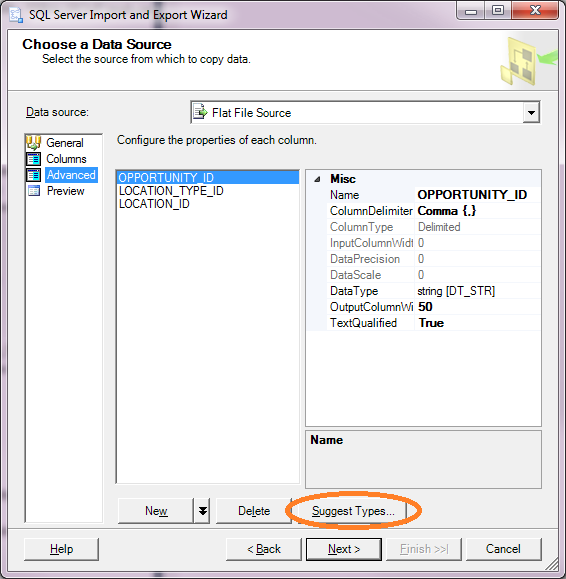
From the wizard select flat file source.



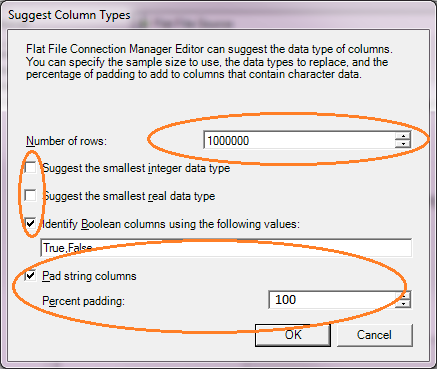
Set text qualifier to a double quote (“).



In the left hand side click Advanced, then click Suggest Types…



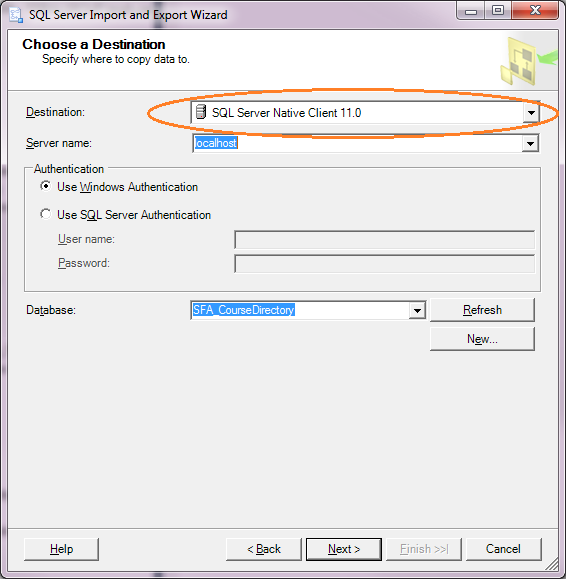
On suggested types dialogue make these settings for each import.



Click OK, this may take a minute or so with large tables to work out the suggested field types for the import.

When complete click Next.

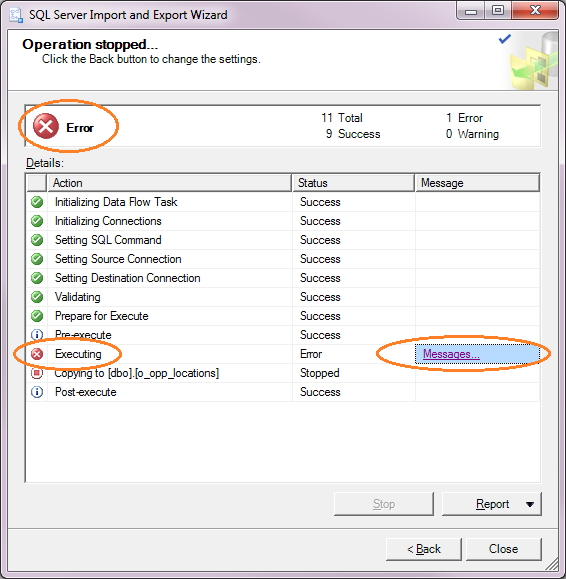
Select the destination type as SQL Server Native Client 11.0 (or 10 if that is highest version). Select the server and database, these should pre-set to the database you are importing in to.

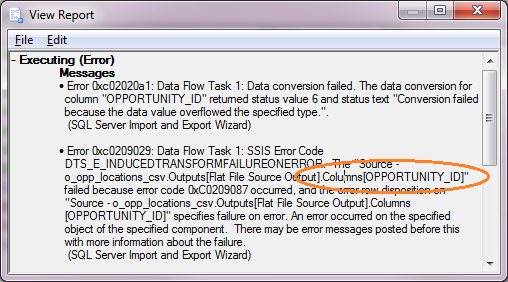


Click Next, the Select Source and Tables’ selection is already made by defaulting to the text file name. Click next and Finish to run immediately and the CSV file will be imported, creating a table that matches the CSV file name.

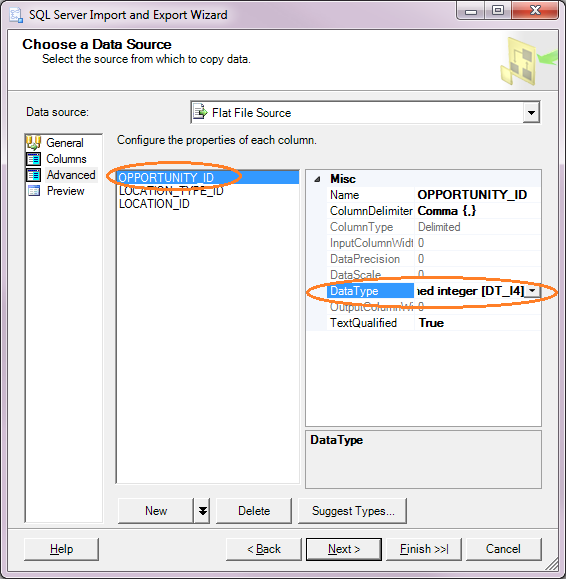
Issues

Sometimes the ‘suggest data type’ makes a mistake and the import will fail due to a field becoming truncated, to correct this, find the field having problems and view the error messages.





Click back in the wizard and return to the Choose a Data Source page, find the field and make the data type larger

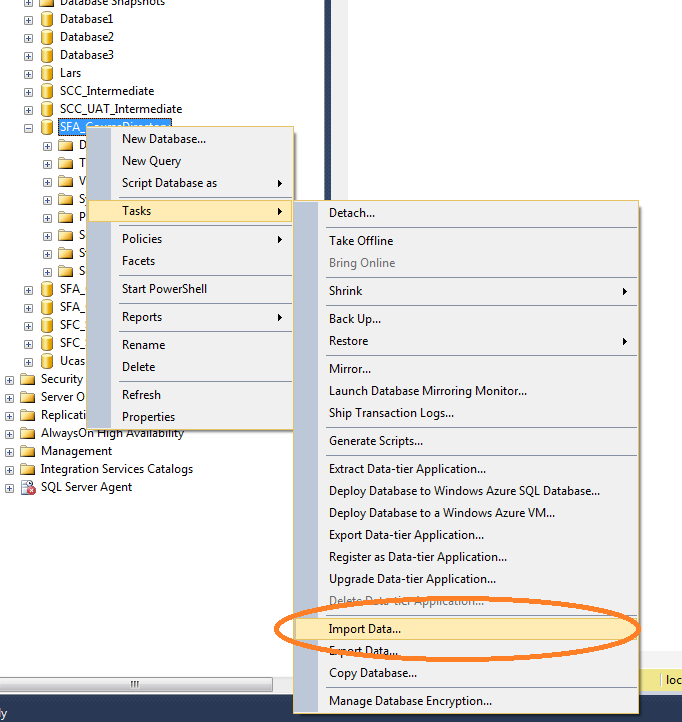


Delete the table from the database that will have been created during the first attempt, then continue through the wizard and import again.

Tip, if these CSVs are imported to a separate SQL database, they can then be imported quicker using the Import wizard in one go to the required database. This means if something goes wrong and the import needs doing again, a new empty database can be published, and these tables imported from the separate SQL database in one pass rather than loading each CSV individually.

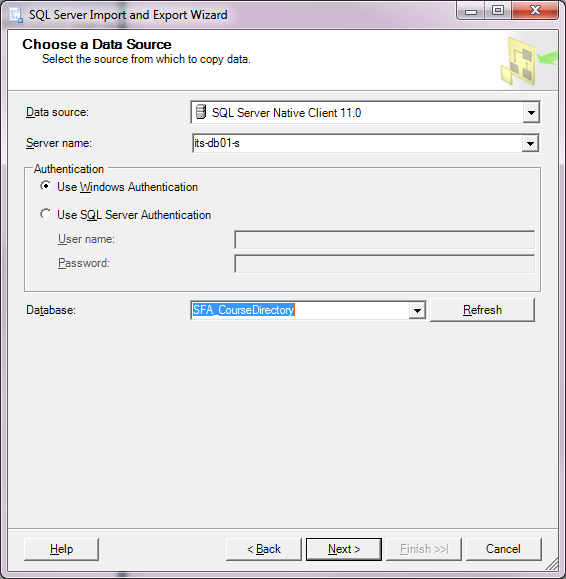
## Other required data

Import the AddressBase and Geolocation data, the fastest method is to import from an existing version of the MS SQL SFA database using the import wizard. Select to import…

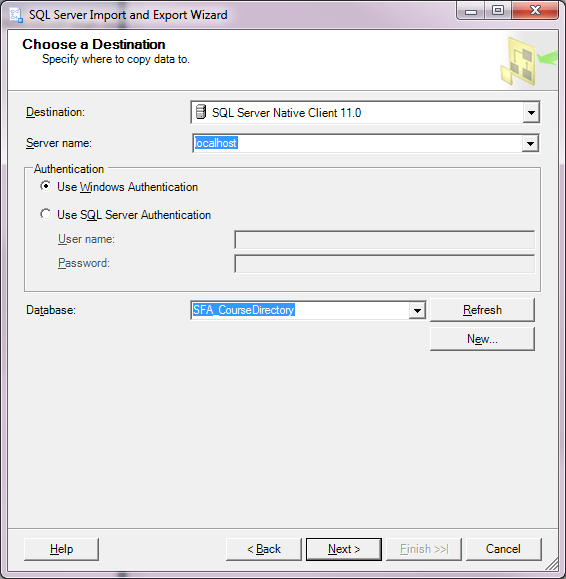


In the import source dialogue select the Data Source as SQL Server Native Client 11.0 (or 10 if that is the highest version).

Enter the server where an existing SFA database resides with populated postcode data. Then click Next…



For destination select SQL Server Native Client 11.0 (or 10 if that is the highest one) and the destination should default to this database.



On the Specify Table Copy or Query dialogue, select ‘Copy date from one or more tables or views’, click Next, then select the table AddressBase and GeoLocation, click Next and Finish to run immediately. This import takes several minutes.

# Prepare for data migration

To pre-pare for the migration, from MS SQL Management Studio open the first SQL script called ‘1 – Check data tables.sql’. Run this, where it will check all the necessary tables from the Oracle database are imported okay. If tables are reported missing then import them before continuing.

Next run the script ‘2 – Add functions and indexes to prepare for import.sql’, this adds indexes and some mapping functions to speed up the import, these are removed later. Note this may take several seconds to complete while indexes are built.

# Data migration

Run the import SQL called ‘3- Data migration.sql’, by default this is set for test mode, so it will run in a transaction, print the export log with any warnings or errors, then rollback all changes. See the top of the script file and change @TestImport to 0 when happy to commit the changes, then execute again. The import will take around 5 minutes to run.

# Post migration tasks

After committing the data migration, clean up the temporary indexes and imported tables by running ‘4 – Tidy up.sql’, this will drop all the imported Oracle data tables and remove indexes added to speed up the migration. This task maybe postponed in order to troubleshoot or check the import against the original data tables.

## Default users

Default users are added at the point of initially creating the SQL database from the project. These default users have full developer permissions, so may require deleting or restricting after the import. First make sure access is available by a genuine user account, then review this list and restrict or remove the users as required.

To initially log in, use the account ‘admin@tribalgroup.com’ using password: Adm!n\*log7in and set up an admin user. **This admin account, if it remains enabled, should have the password changed once access is obtained.**

**Default accounts to be reviewed or removed as appropriate.**

[jon.ripley@tribaldefault.com](mailto:jon.ripley@tribaldefault.com)

[adam.davidson@tribaldefault.com](mailto:adam.davidson@tribaldefault.com)

[leigh.carpenter@tribaldefault.com](mailto:leigh.carpenter@tribaldefault.com)

steven.fulleylove@tribaldefault.com

graeme.tizard@tribaldefault.com