Moose Monitoring Database Application

Table of contents

Overview

NPS Moose Monitoring Database Application

Overview

Moose population structure and abundance are monitored the National Park Service in Alaska. Methods follow the protocols developed by the Arctic and Central Alaska Inventory and Monitoring Networks and rely on the GeoSpatial Estimator toolset described in Kellie and Delong, 2006.

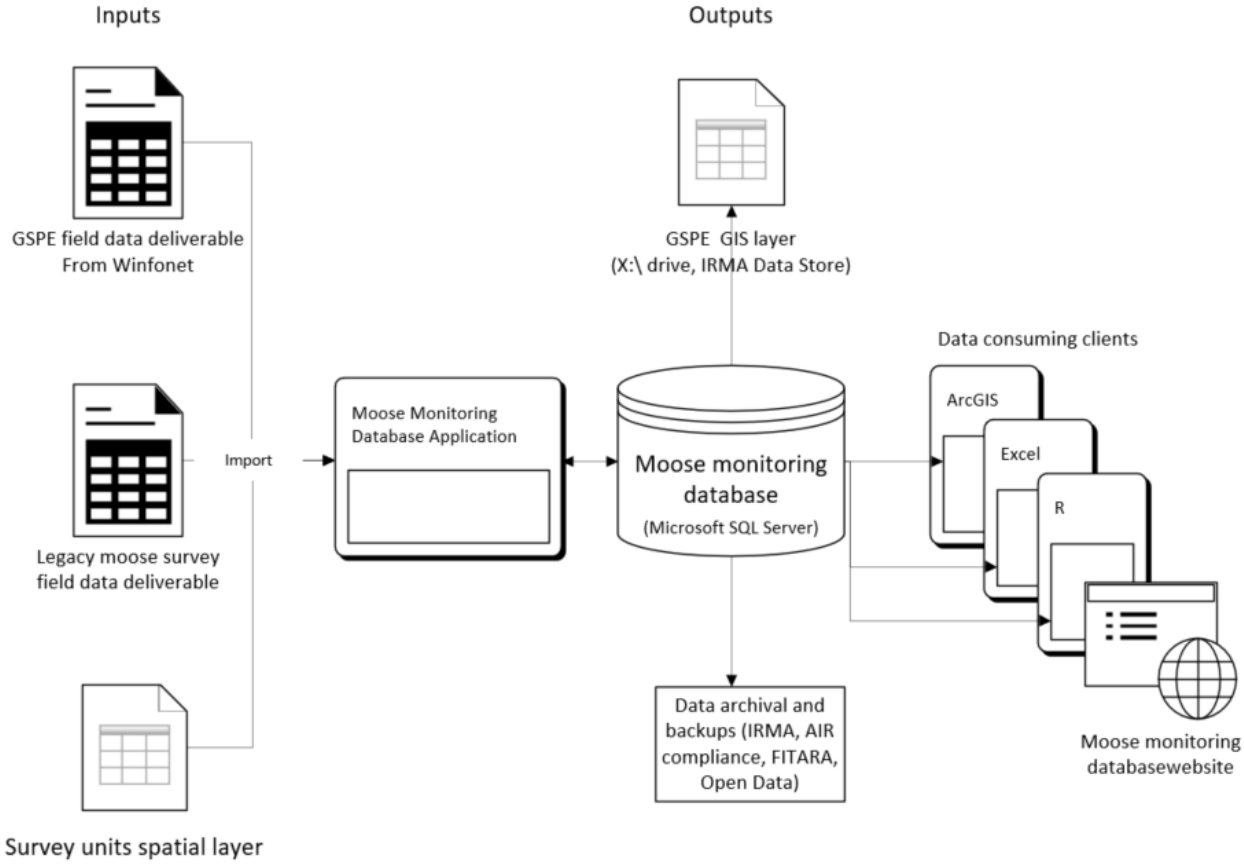
From 2006 until 2016 moose monitoring data was stored in ADF&G's Winfonet system. NPS policy now requires that the data reside in an authorized Department of the Interior data store. Development of a database began in 2015, data migration started in 2016 and was substantially finished in 2022, at which time a front-end application was developed. This document describes the moose monitoring data management system, database and front-end application.

Sensitivity statement

The Moose Monitoring Database Application contains information about a species of commercial interest, some or all of which may be subject to data sharing agreements with partner agencies. Please read and follow the directives of each participating Park superintendent as outlined in memorandums available in the IRMA Data Store regarding protected data (search on *'Handling of Protected Natural Resource Data'* for each Park in question. Obtain permission of the Park moose biologist and Park superintendent before sharing data beyond the NPS.

Conceptual diagram

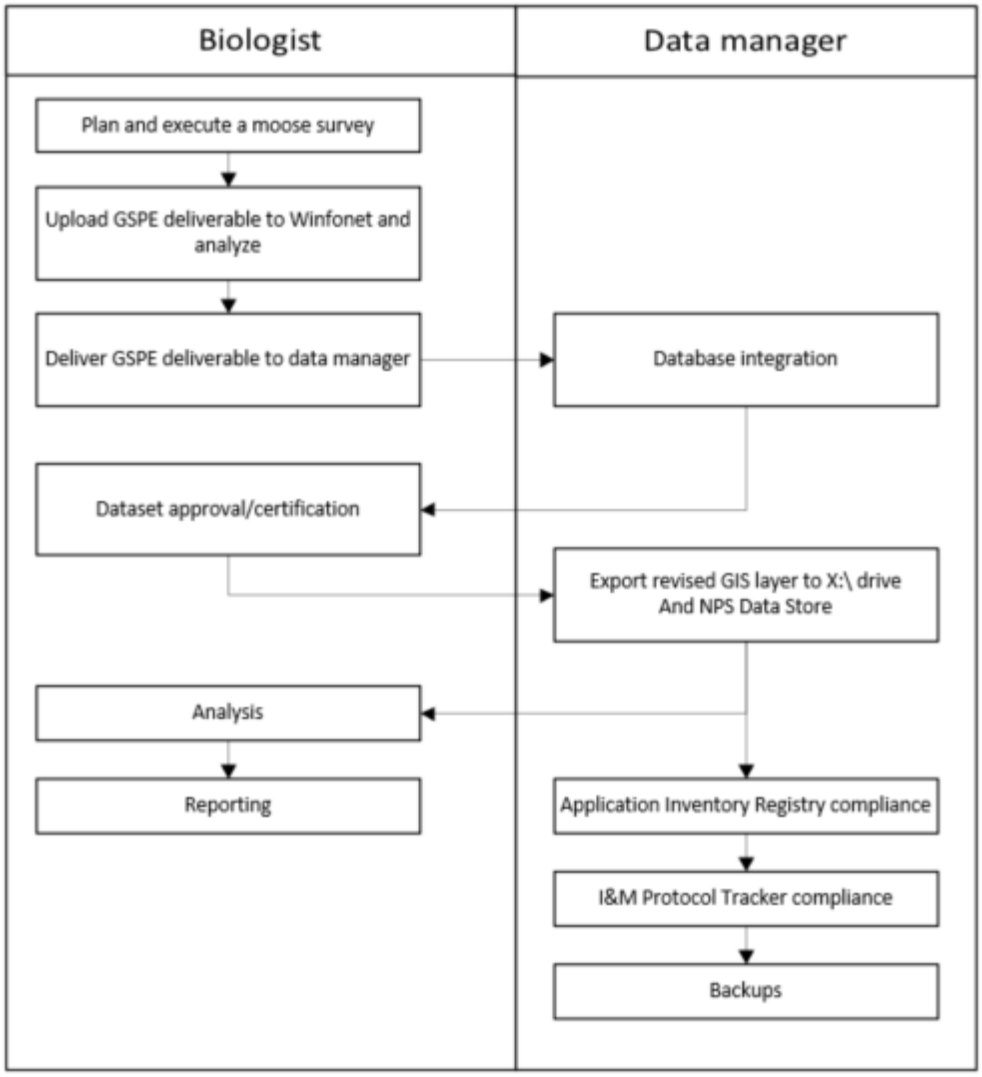
The heart of the moose monitoring system is the Moose database hosted on an SQL Server at the NPS Fairbanks Administrative Center.



Work flow

The moose monitoring work flow can be summarized simply in the steps below

1. Collect data according to ARCN/CAKN protocol
2. Process GSPE deliverable in Winfonet
3. Integrate GSPE deliverable to ARCN/CAKN moose monitoring database
4. Analyze
5. Report



Resources

Resources available to moose monitoring personnel include:\

Resource	Location
Data deliverable files	ARCN O:\Monitoring\Vital Signs\Moose\Data CAKN J:\Monitoring\Moose\Data
Moose monitoring master Project reference in the IRMA Data Store	ARCN: DataStore - Project - (Code: 2222140) (nps.gov) CAKN: DataStore - Project - (Code: 2220369) (nps.gov)
Moose monitoring database	See Database documentation
Moose monitoring database application	NPS-ARCN-CAKN/Moose3 (github.com)
Alaska Department of Fish and Game's Winfonet database	Home (alaska.gov)

System administration

ARCN/CAKN FAC IT

Obtaining the Moose Monitoring Database Application

Obtaining the Moose Monitoring Database Application

Copy the latest build from J:\Monitoring\Moose\Database application builds to your local computer and click Moose3.exe. This may not be the permanent home for builds; probably will move to a web deployment at some point.

Adding a new moose survey and GSPE data deliverable to the moose monitoring database

Adding a new moose survey and GSPE data deliverable to the moose monitoring database

Adding a new survey to the Moose Monitoring Database Application involves two (or possibly three) steps that must be performed in order:

1. Add a Survey record to the GSPE_Surveys database table
2. Add related GSPE records to the GSPE database table
3. If you are trying to add legacy moose survey data rather than standard GSPE data then it's possible the survey units do not exist in the database. In this case add the data as above but leave the SurveyUnitSet fields blank or NULL. You will have to get your data manager to add the survey unit polygons to the database and relate them to the survey data. There is currently no within-application functionality to accomplish this task.

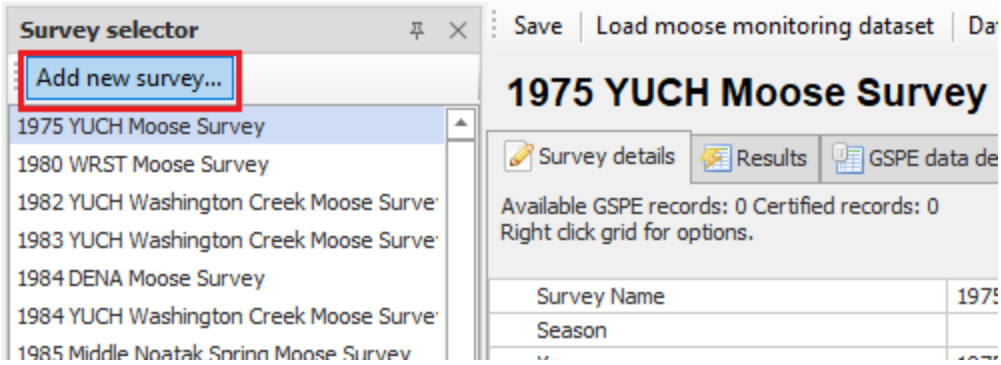
Prerequisites

- At minimum, datawriter privileges on the database. Contact your data manager.
- SQL Server Management Studio software.

Procedure

Adding a new Survey record

1. If the survey does not exist then add a new survey record to the database by clicking the Add new survey... button.

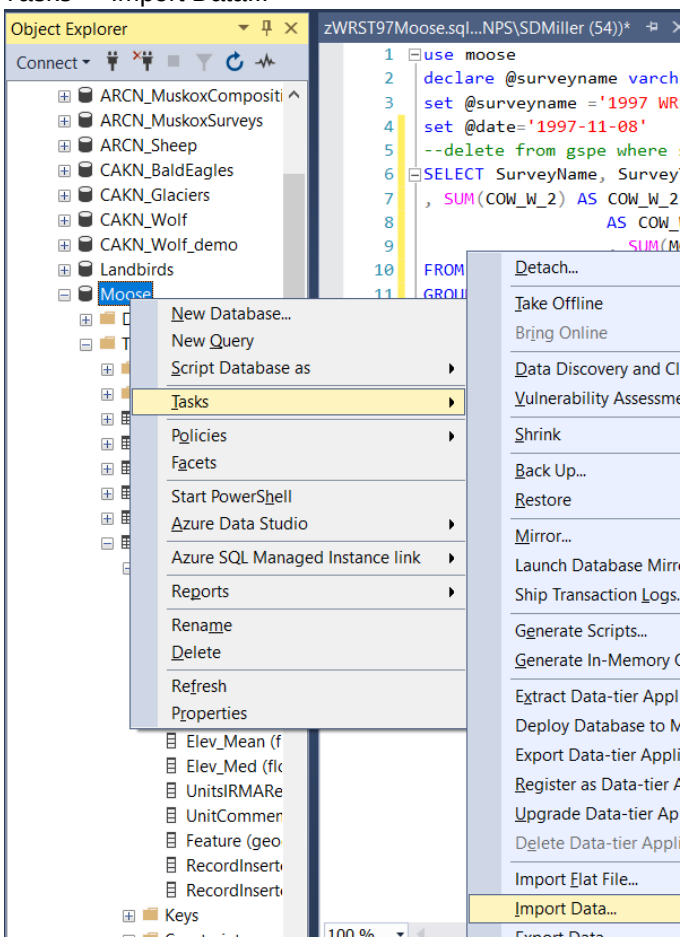
2. 

3. Answer the wizard questions and save your work after the new survey appears in the surveys inventory.

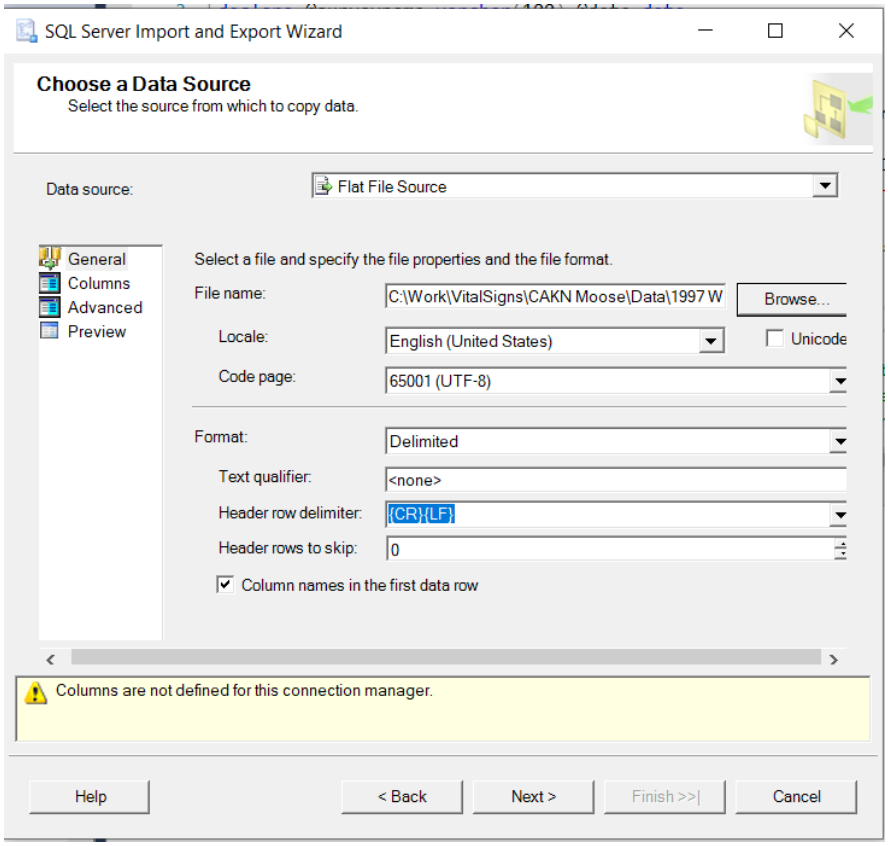
Importing GSPE records for the new Survey

As of 2022 there is no tool in the Moose Monitoring Database Application to directly import GSPE data for a new Survey. You can use Microsoft SQL Server Management Studio to import the records, or other data manipulation software, but really, the easiest way is to give the deliverable to the ARCN data manager for import. The method using SSMS is shown below.

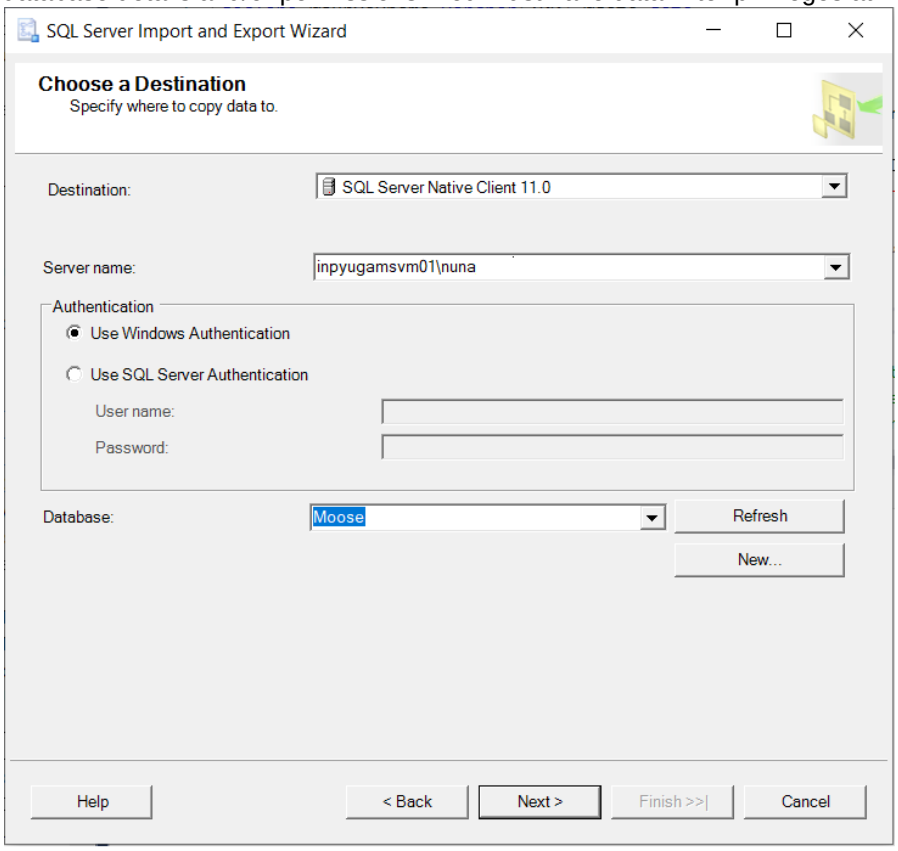
Procedure

1. Start SSMS
2. Navigate to the Moose database and then locate the Sql Server Import/Export Tool: Moose -> Tasks -> Import Data...
3. *TIP: Befor you import the data in the GSPE spreadsheet, make sure the source dataset does not have commas in any of the fields, or, if commas exist, that the text fields are surrounded by quotes.*
4. Start the Import Data... in Sql Server Management Studio by right clicking the database and selecting Tasks -> Import Data...
5. 

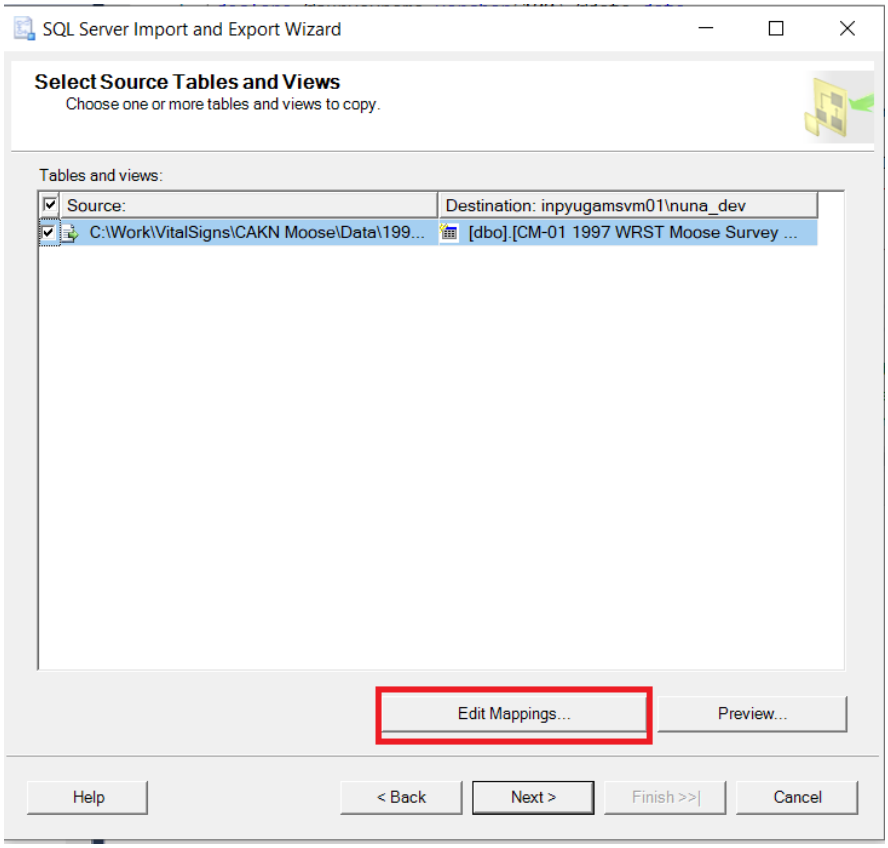
6. Choose the GSPE deliverable as a data source



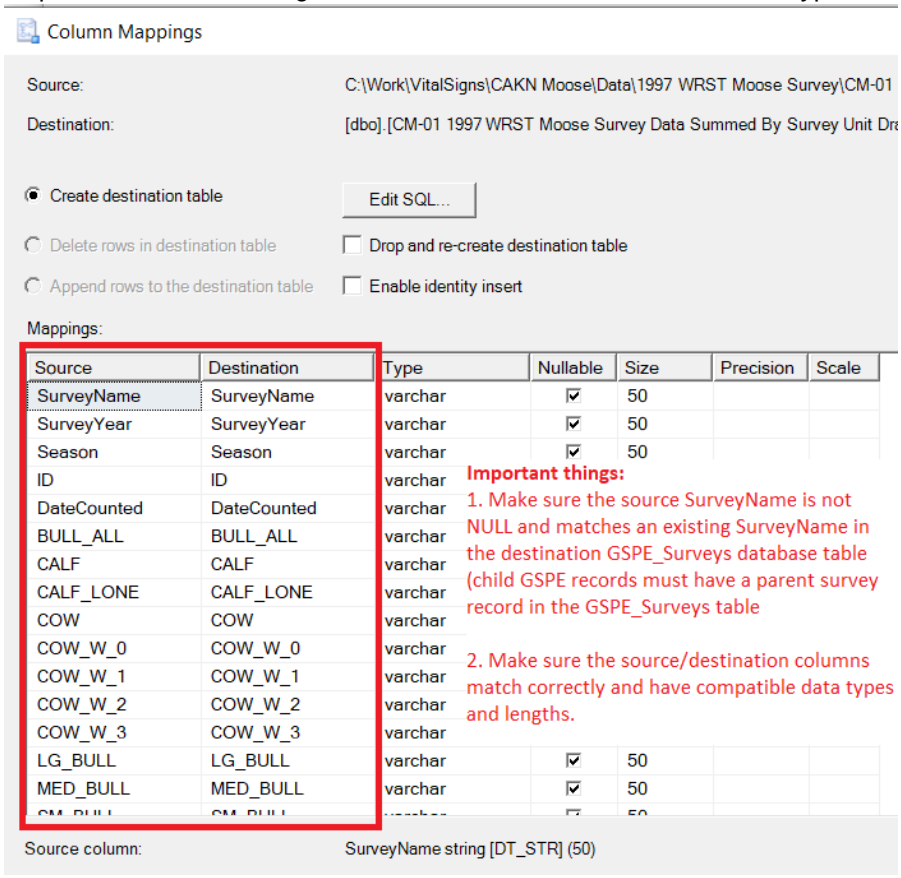
7.
8. Choose the Moose database on the nuna SQL Server (the. Contact the ARCN/CAKN data manager for database details and/or permissions. You must have datawriter privileges at minimum).



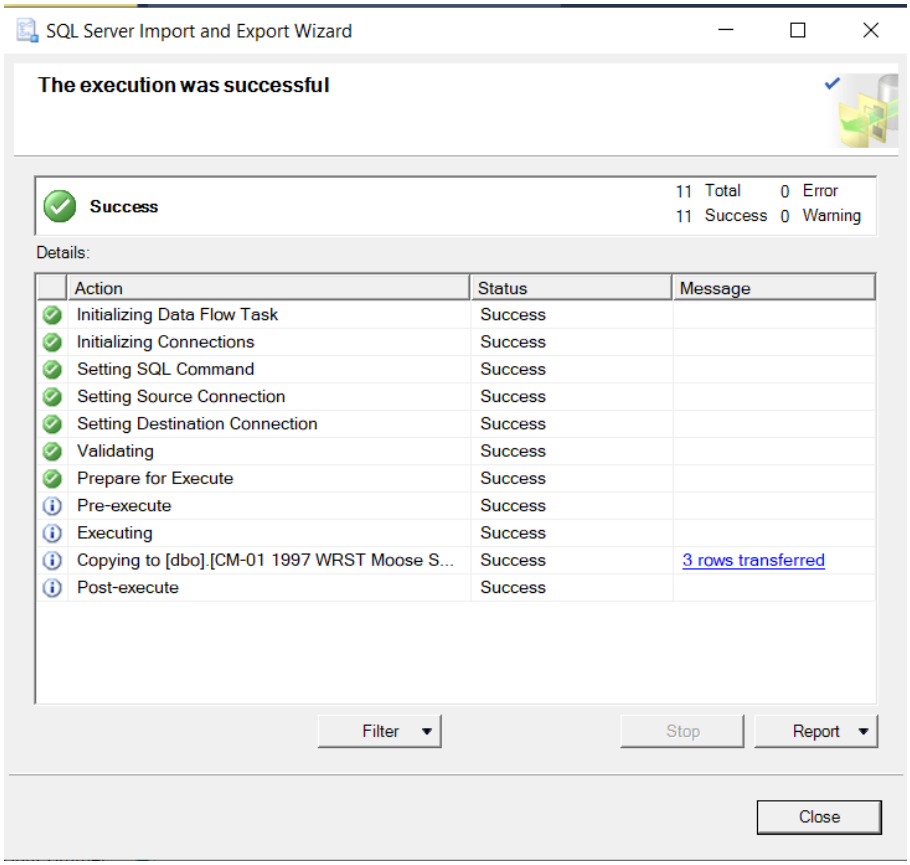
9.
10. Click Next
11. Map the source columns to the database columns. Choose Edit Mappings...

12. 

13. Map the columns ensuring the source/destination columns and data types are compatible



14.
15. Double check all settings are correct and execute. If all goes well you should see something like this:



16.
17. Check your work to be sure the data imported correctly.
18. Validate that the destination data exactly matches the source data. This is most easily done by cross-summing the source numeric values in a pivot table in Excel and comparing them to a similar SQL query or by using the Moose Monitoring Database Application's pivot tool. Scan the comments and text fields visually.
19. Certify the dataset, if warranted. Compare the record counts, summed values, etc., against the GSPE Winfonet report. If a formal report for the dataset exists, then attempt to regenerate any summary tables. If all looks good and the biologist approves, then certify the survey dataset.

Adding a new set of survey units

Adding a new set of survey units

There is no functionality withing the Moose Monitoring Database Application to add a new set of moose survey unit polygons. This is because new surveys invariable use the ADFG units which exist in the database. New unit sets are only added for legacy surveys so it is rare to have to add these to the databas. However, should a legacy survey be found you can modify the following arcpy script to produce an SQL script that will insert the units.

[arcpy script goes here]

Quality control

Quality control

Quality control (QC) procedures ensure that data in the database meet expectations. Defects should be documented in the Comments column and in the Survey record's attributes.

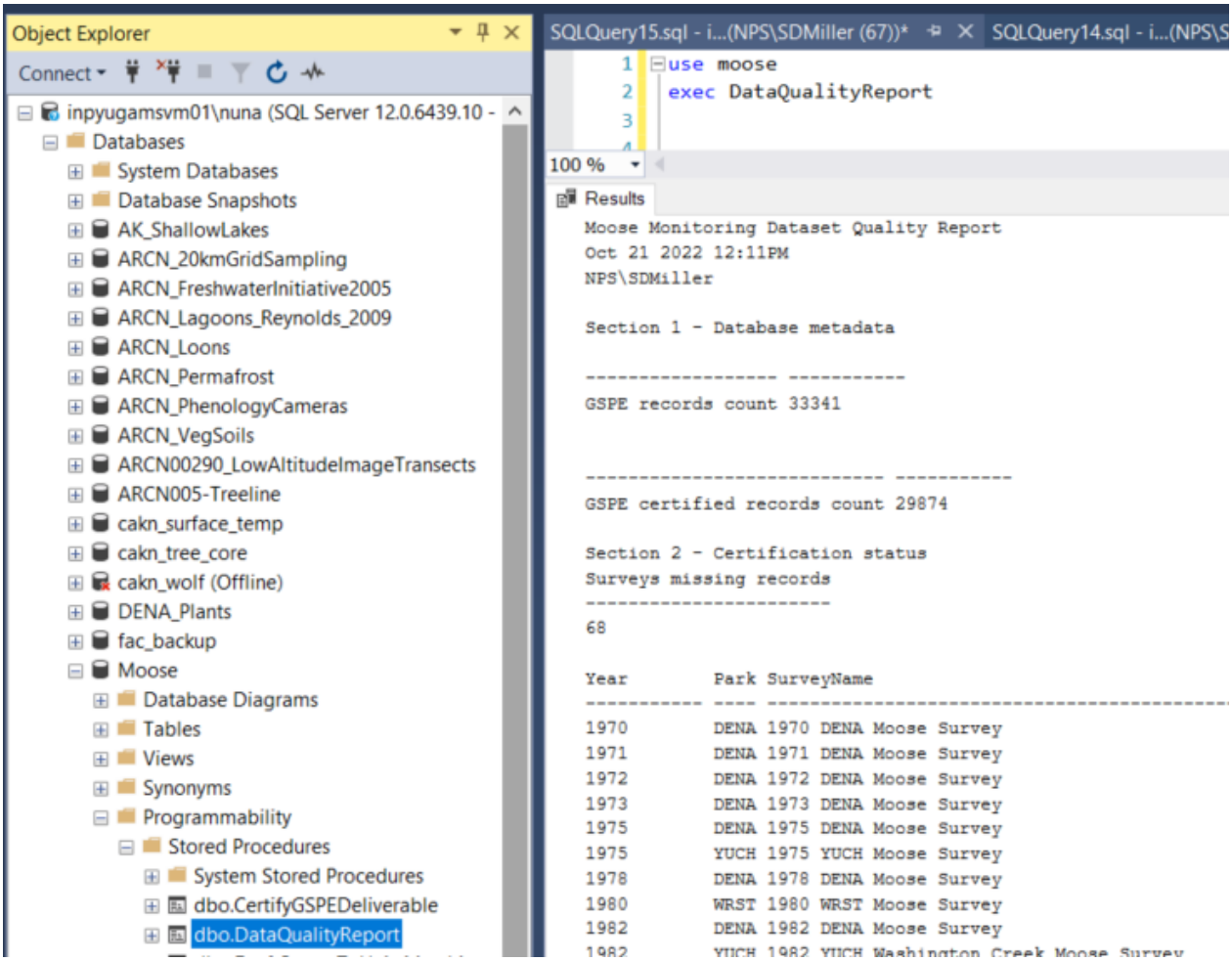
Primary QC methods involve visual record scanning, database interrogation via scripts, QC queries, and stored procedures. The primary tool for ensuring data quality is a stored procedure called DataQualityReport. A number of quality control checks are performed on the moose monitoring dataset and defects presented in a report. Execute DataQualityReport in SQL Server Management Studio (SSMS) as shown below. Carefully review the output and resolve any errors. Re-run the tool as needed.

Dashboard query

The Dashboard query provides a good summary of the database contents and quality. Run the Dashboard query through (SSMS) or the Moose Monitoring Database Application's Data Shaper tool.

DataQualityReport stored procedure

Note: for best results, right click the query pane and select **Results to -> Results to text**; otherwise the queries will be shown as grids and some context of what QC process is being run will be lost.



Dataset certification

Dataset certification

Data certification is an important tool to communicate data quality to data consumers. The moose monitoring dataset consists of data from many surveys done over many parks. In many cases datasets have been recovered many decades after institutional knowledge is gone, often without the benefit of a summary report or other forms of documentation. Three levels of certification are allowed:

Raw	Records that have not undergone any quality control procedures
Provisional	Records that have undergone rudimentary quality control or have been received directly from a park biologist or the Winfonet system, but have not been fully processed and approved for analytical purposes.
Certified	Data that have been approved for analysis. These records have been processed for quality, validated against a source dataset, a report or other summary, have defects documented, and are of known quality.

Certification process

GSPE records are certified at the record level. This means that each record in the GSPE data table has a column called CertificationLevel (and also CertifiedBy and CertificationDate).

CertificationDate	CertifiedBy	CertificationLevel
2021-10-29 13:21:45.070	NPS\SDMiller	Certified
2021-10-29 13:21:45.070	NPS\SDMiller	Certified
2021-10-29 13:21:45.070	NPS\SDMiller	Certified
2021-10-29 13:21:45.070	NPS\SDMiller	Certified
2021-10-29 13:21:45.070	NPS\SDMiller	Certified
2021-10-29 13:21:45.070	NPS\SDMiller	Certified

During data ingestion and quality control these columns should be null but CertificationLevel should be set to 'Raw' for raw data or 'Provisional' if the data are likely good enough for provisional analysis but not fully processed.

Verify the dataset ingested into the moose monitoring database exactly matches the GSPE deliverable that was submitted and analyzed in Winfonet. An alternative check for old datasets for which a report exists is to attempt to re-generate any summary tables in the report with the ingested data. Any data failing QC checks should have CertificationLevel set to 'Raw'. If the data all check out issue the following query, substituting the name of your survey for `<SurveyName>` exactly as stored in the parent GSPE_Surveys table.

```
USE Moose
Begin Transaction
UPDATE GSPE SET
CertificationLevel='Certified',CertifiedBy=suser_name(),CertificationDate=getd
ate() WHERE SurveyName = 'Your SurveyName goes here'
-- Execute COMMIT if all goes well, or else ROLLBACK if there are errors.
Warning: you must commit or rollback the query result above or the database
will be left in a hanging, unusable state.
```

ENSURE THE QUERY IS PROPERLY FILTERED to avoid contaminating records from other surveys. If you have any questions contact the ARCN/CAKN data manager.

sites

Data analysis

Data analysis

Winfonet is the primary tool for generating moose abundance estimates, calf/bull:cow ratios, etc., but the Moose Monitoring Database application has tools for accessing and summarizing survey data.

(Being fleshed out)
Dataset_GSPE and its naming variants are the major data product.

select Dataset_GSPE into an ArcMap query layer
it's probably best to let the data load and then save it to a local geodatabase to speed things up, going back and forth to the sql server will be quite slow unless you are in FAC

Data shaper

Dataset publication

Dataset publication

Periodically the moose monitoring dataset should be published so that people can use it. This should be done to the AK Region's GIS X drive and also to the IRMA Data Store.

Considerations for publishing sensitive data

Consider carefully the permissions of any dataset published to Data Store. Moose monitoring data falls under the category of 'species of commercial interest', which is one justification for tagging a reference as 'Internal'. Additionally the NPS and Alaska Department of Fish and Game may have one or more data sharing agreements in place which should not be violated. Generally a moose monitoring dataset should only be available internal to the NPS or to specific individuals within the Service. Consult the Data Store help for more information on permissions.

Generating a dataset for publication

Modify and execute the Python script below in ArcGIS to generate a shapefile of moose monitoring data. Notice that the SQL query is restricted to certified data only.

```
import arcpy

# Point to the database connection file (use Create Database Connection tool in
ArcToolbox)
database_connection_file_path = r"C:\Work\Code\ArcPy Scripts\Moose DB to
GeoDB\MooseDBLocalConnectionFile.sde"
```

```
# Get a reference to the main data frame; assumes it's the first and only one
df = arcpy.mapping.ListDataFrames(MXD)[0]

# The spatial query to submit to the QueryLayer
sql = "SELECT TOP 100 Percent SurveyName, Year, Network, Park, SubArea, Season,
Personnel, ReportReferenceCode, ReportLink, DeliverablesDatasetReferenceCode,
Methodology, ID, IntID, Rand, FiringOrder, Selected, Counted, Exclude, Strat,
StratName,                    StratMoose, StratTracks, StratHab, SearchMin,
DateCounted, Area_SqMi, CALF, COW, COW_W_0, COW_W_1, COW_W_2, COW_W_3, YBULL_ALL,
SM_BULL, MED_BULL, LG_BULL, BULL_ALL, ADULT, MOOSE,                    UNKNOWN,
BULL_30_40, BULL_30_50, BULL_30_60, BULL_41_50, BULL_ADULT, BULL_GT_50, BULL_GT_60,
BULL_GTE_50, BULL_LT_30, BULL_LT_50, CALF_LONE, MED_L_BULL, YBULL_GTSF, YBULL_SF,
Pilot, Observer, Density, SCF_Plot, Std, Int, Comments, CertificationLevel,
DataResourcesDirectory, DataSource, ValidatedDate, SourceFilename, SurveyUnitSet,
Feature,Convert(varchar(36),NEWID()) as UniqueID        FROM
Dataset_GSPE_WithSurveyUnits  WHERE  CertificationLevel='Certified'"

# Layer name
layer_name = "GSPE"

# Create the QueryLayer
arcpy.MakeQueryLayer_management(database_connection_file_path,  layer_name,  sql,
"[UniqueID]", "POLYGON", "4326")

# Create an MXD to hold the new QueryLayer
MXD = arcpy.mapping.MapDocument(r"C:\Work\Code\ArcPy Scripts\Moose DB to
GeoDB\MooseUnits.mxd")

# Get a reference to the created QueryLayer
output_layer = arcpy.mapping.Layer(layer_name)

# Add the layer to the main data frame
arcpy.mapping.AddLayer(df, output_layer, "AUTO_ARRANGE")

# Export the layer to a shapefile
try:
    arcpy.analysis.Select(layer_name, "C:/Temp/zMooseUnits/GSPE.shp","")
except:
    print("An exception occurred exporting the dataset " + layer_name)
```

Write metadata

Use the tools in ArcCatalog to write metadata for the shapefile generated above. Validate the metadata.

Publish

X drive

Build metadata and ask Kerry Shakarjian to replace the existing

Data Store

Update the existing Spatial Dataset reference with the latest revision of the shapefile generated above [Note 2022 this doesn't exist yet]

Data mining resources

Data mining resources

Mat Sorum put together a summary in 2016 for ARCN Moose Survey Data_3_23_2016.xlsx	N:\Natural\GAAR\Moose\GAAR Moose Surveys
DENA Moose drive	\\inpdenafiles02\teams\ResMgmt\Wildlife\Moose
CAKN Moose shared drive	J:\Monitoring\Moose\Data
ARCUS has old reports	

. He put data at

: \\inpdenafiles02\teams\ResMgmt

Hey Scott! Sounds like a big lift. You have all the YUCH surveys I have information on. I assume the red ones stem from some information you perhaps got from Shults or Lawler, but I don't have additional information on them. I tasked Mat Sorum with organizing all the ARCN survey data a few years back, which he did. So he may be in the best position to assess where we are at on those surveys. Attached is a spreadsheet that was developed. While I was on a bunch of the western surveys, I didn't end up with the data, Brad did. I assume Raime and Letty manage these datasets now.

For GAAR, we have 2 moose surveys: the Koyukuk and the upper Kobuk. So I would modify your naming conventions to reflect that. The upper Kobuk survey is really upstream of KOVA (ie it doesn't cover much of KOVA) and covers GAAR "boot". All the survey data for these 2 areas can be found here: N:\Natural\GAAR\Moose\GAAR Moose Surveys. Hopefully that will turn some of your reds green.

The Tagagawik (not Tagagawiki as spelled in first 2 lines) and the Tag are the same river, in case that helps you.

Kyle

An example data entry session using FileToSql

An example data entry session using FileToSql

There are many ways to enter data into the moose monitoring database. The example below shows the step I took to to enter the 2013 DENA Moose Survey data into the database.

There are a couple of resources that are ideally at hand for data entry of a moose survey in addition to the ones in the Data management resources table:

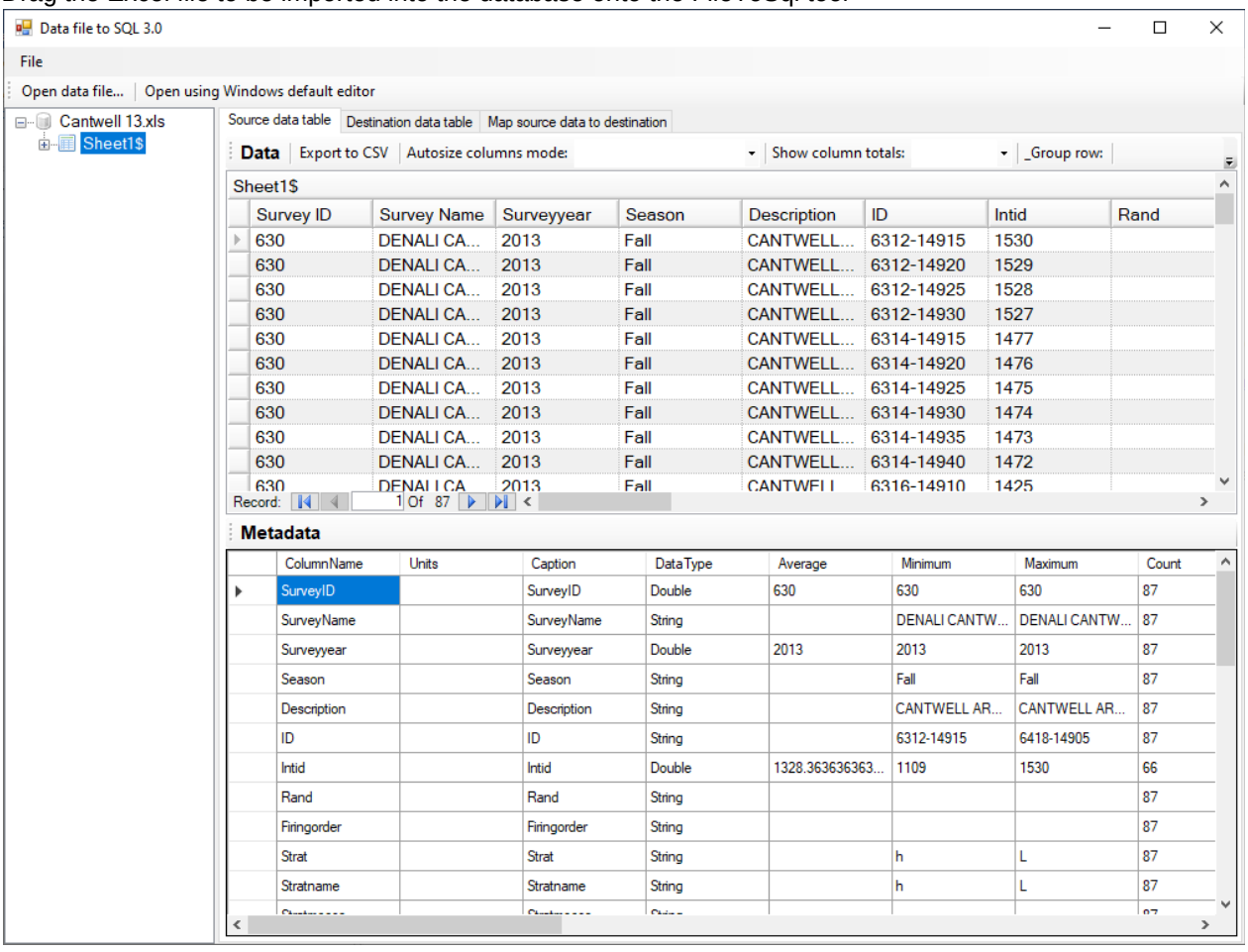
Resource	Location
Survey report	Not available as of writing. Probably not written. DENA biologist did not have it.

Data deliverable file (GSPE or older J:\Monitoring\Moose\Data\2013 DENA Moose Survey Gasaway format spreadsheet).

There are a number of ways to import data into a SQL Server database. These include:
1. SQL Server Import and Export Wizard
2. Writing INSERT queries as an Excel formula
3. Scripting with Python, R or other language
4. Using ARCN's FileToSql tool

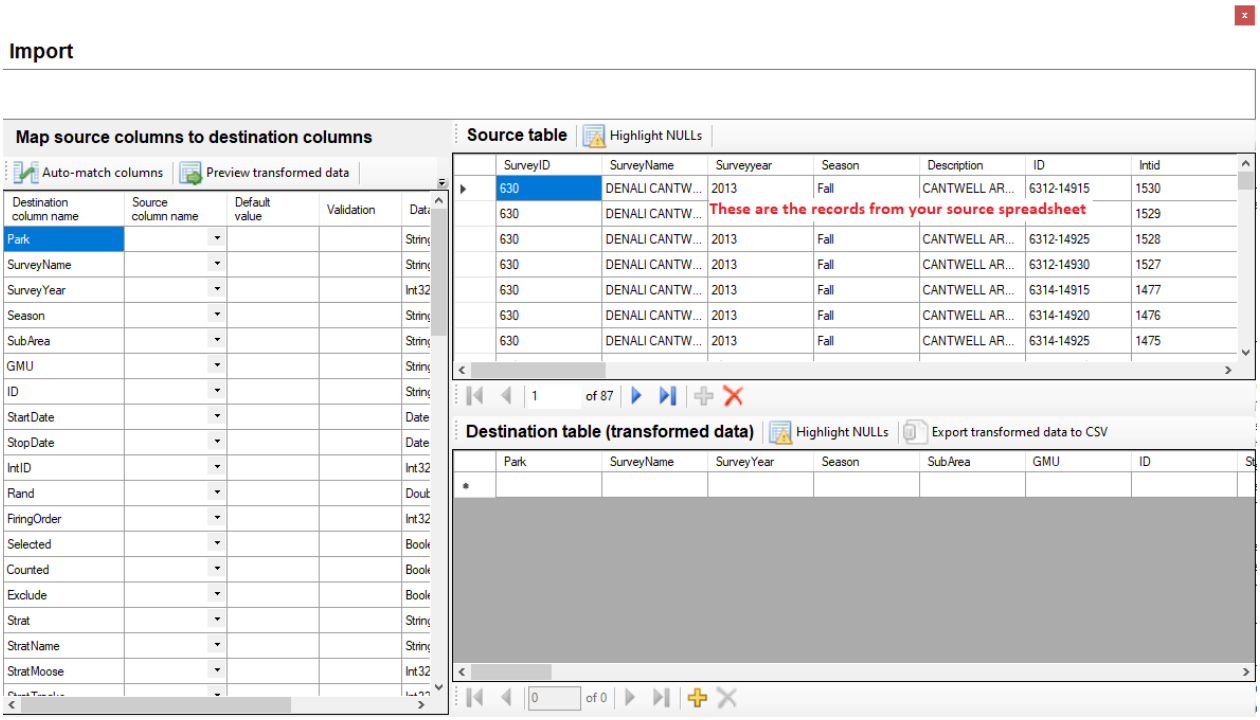
I used the FileToSql tool but the SQL Server Import and Export wizard would work just as well.

Procedure
Ensure the parent Survey record exists in the moose monitoring database for the data you wish to import. The Survey table must have a parent row before any child GSPE data records can be related to it. In this example I've ensured the parent '2013 DENA Moose Survey' records exists in the Surveys table.
Copy down or remember the unique SurveyName because you will need it later.
Start FileToSql
Drag the Excel file to be imported into the database onto the FileToSql tool

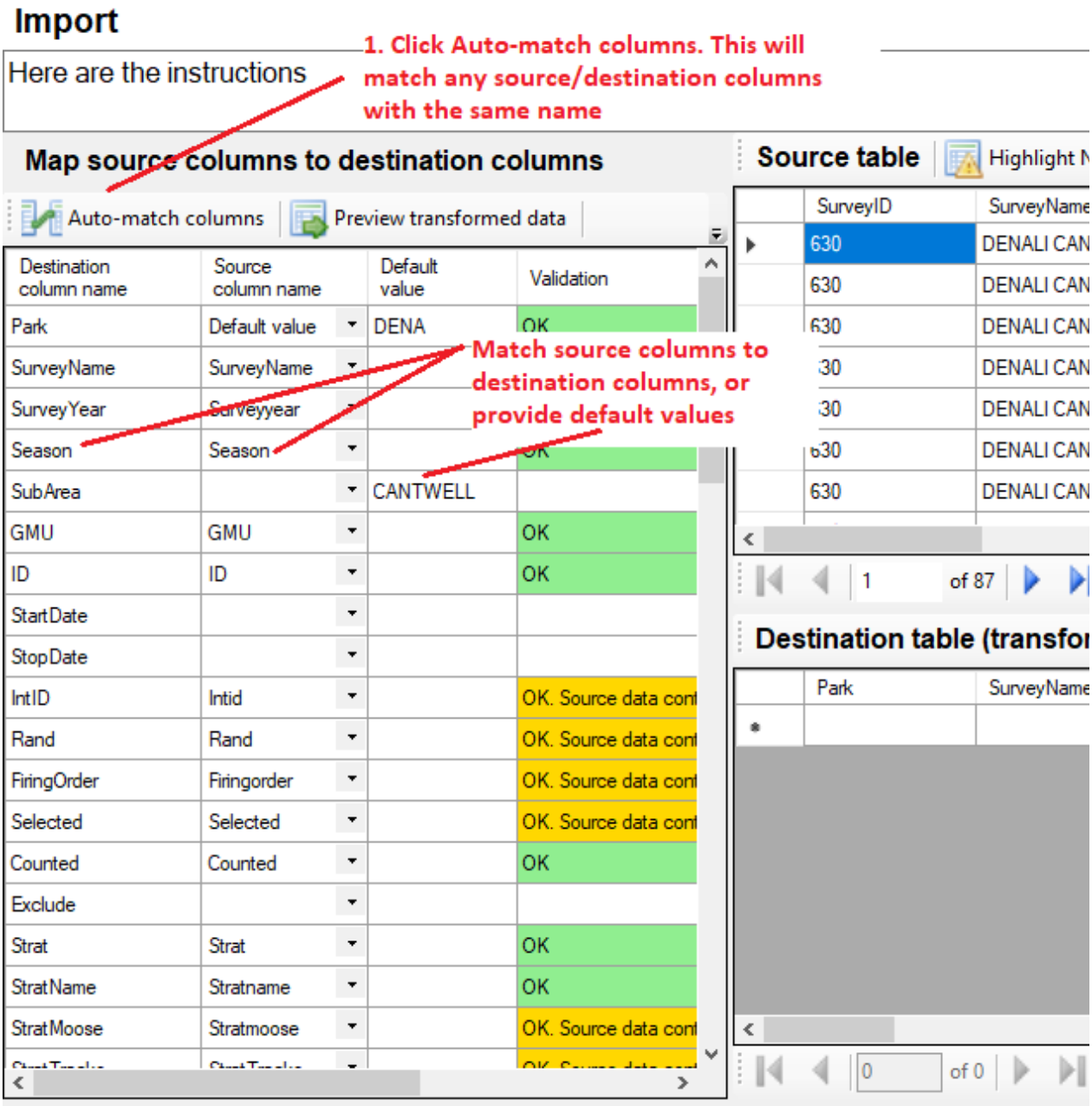


The spreadsheet and tabs show up in the tree at left. Ensure the data in the Data grid exactly represents what is in the source spreadsheet.The information in the Metadata grid below may help to elucidate any problems with the data. Look at the data types, max and min values.
If the source dataset looks good click on the Destination data table tab
Enter a connection string to the moose monitoring database. As of Oct. 2021 it is
Server=inpyugamsvm01\nuna_dev;Database=moose;Trusted_Connection=True;
Enter an SQL query that duplicates the data fields in the source data file you would like to import. In most cases the following query will work: **select top 10 * from GSPE**
Click Execute
If your permissions, connection string and query are all good then you should see some database records

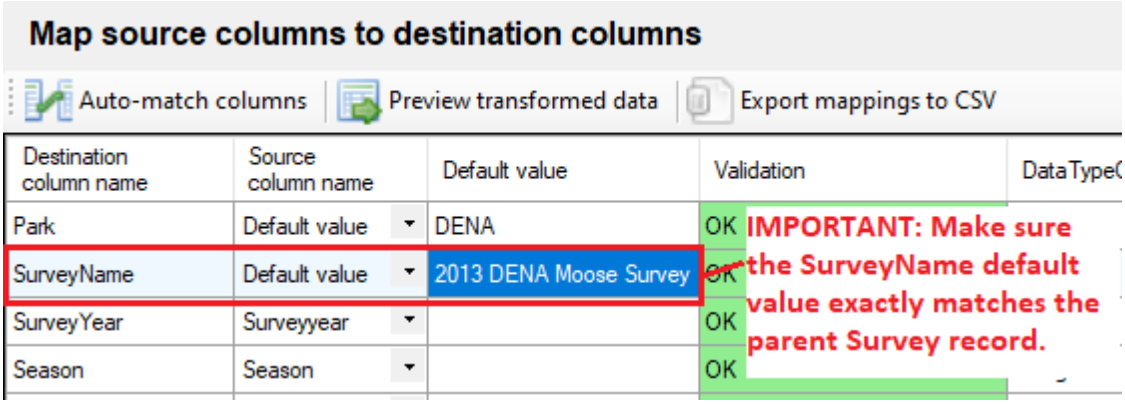
from the GSPE table of the Moose database and a columns mapping tool should appear. The source data should appear under the Source Table grid. On the left is a tool you will use to match source data columns to destination database table columns.



Click Auto-match columns

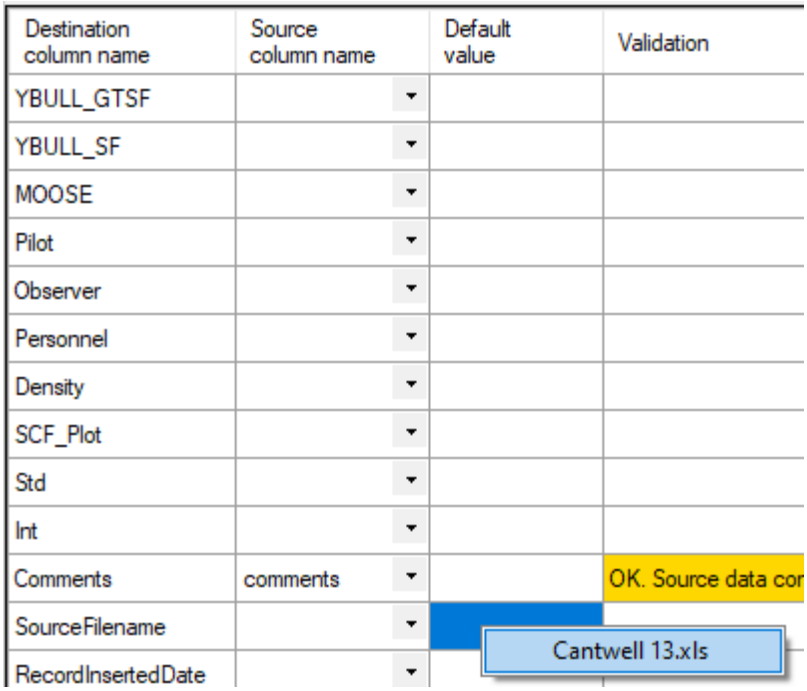


IMPORTANT: Ensure the default value for SurveyName exactly matches the SurveyName field of the parent Survey record in the database. If this is not done you will get referential integrity errors and the data import will fail. It is OK to override the existing SurveyName source column values.



Ensure the default value for SurveyName matches exactly the SurveyName of the parent Survey record; '2013 DENA Moose Survey', for example.
Match any remaining columns from the source spreadsheet to the database table columns, or provide default values.

Click in the **SourceFilename** row
Right click the Default value column for SourceFilename
The source filename appears, select it to auto-fill the SourceFilename default value to the file name.

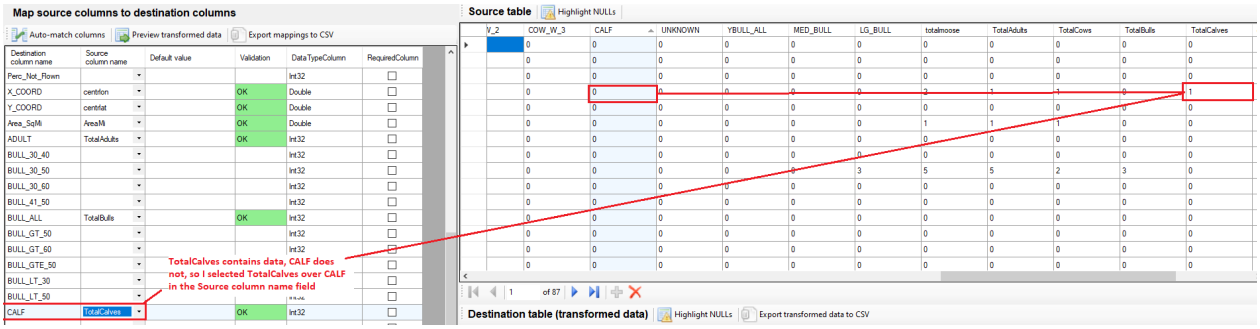


Select the **RecordInsertedDate** row
Right click the Source column name column and select Current datetime

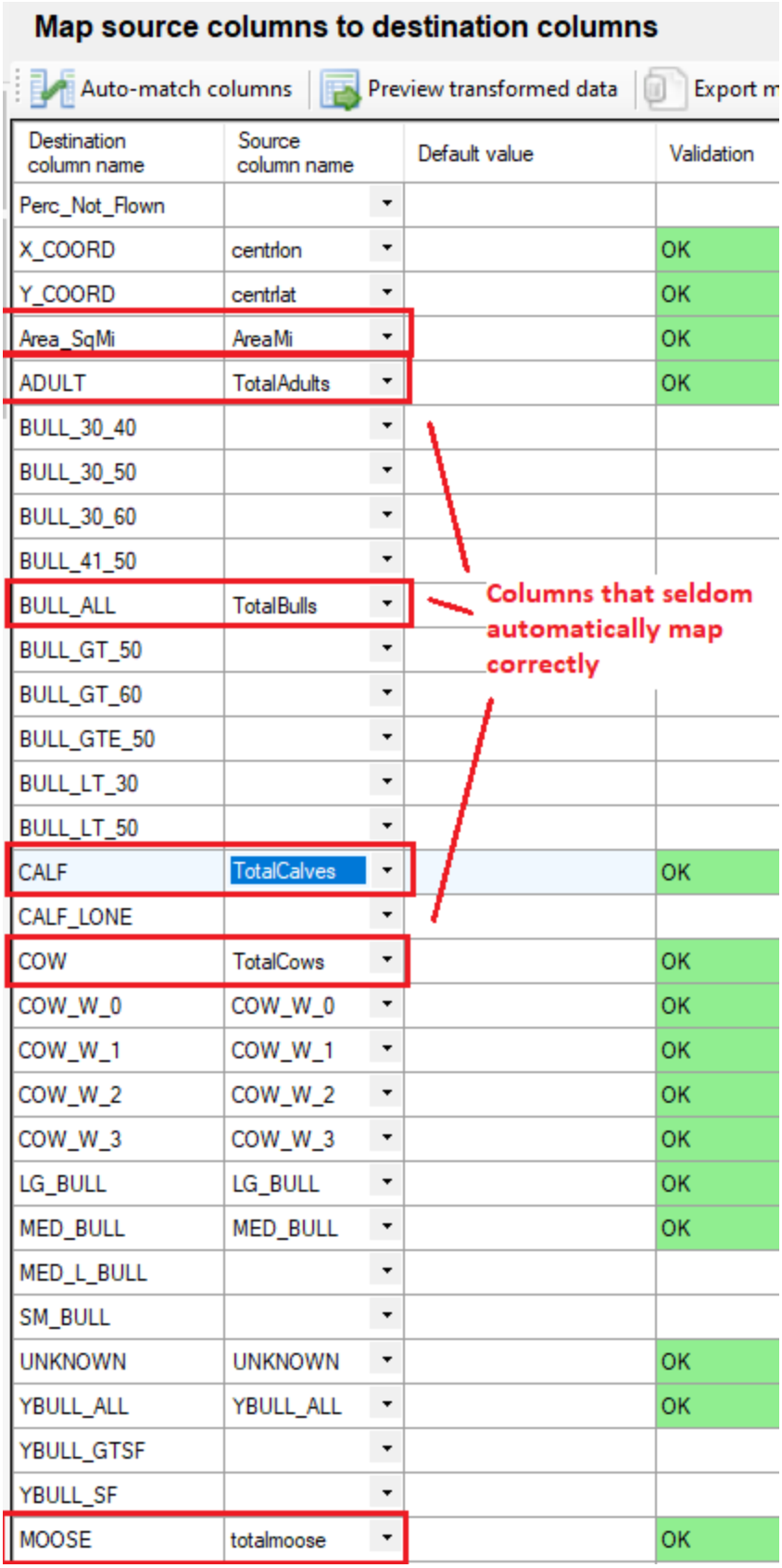
Select the RecordInsertedBy column and set its default value to Current username using the same steps as above.
Set CertificationLevel default value to 'Provisional'

Gotchas

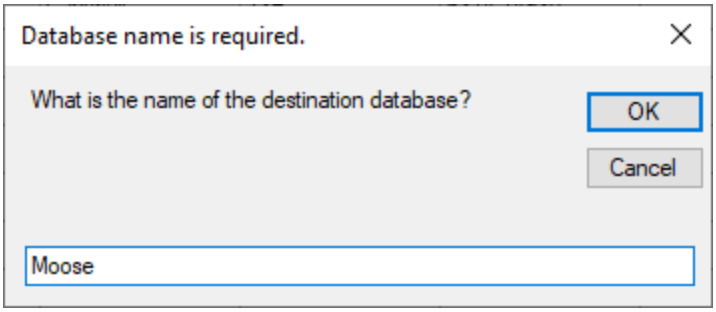
Notice that the source column CALF contains all zeroes where TotalCalves contains count data. Auto-match columns would have gotten this match wrong.



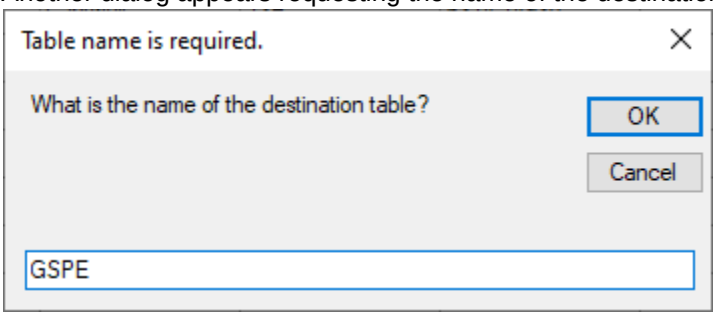
Other columns that seldom map correctly automatically appear below



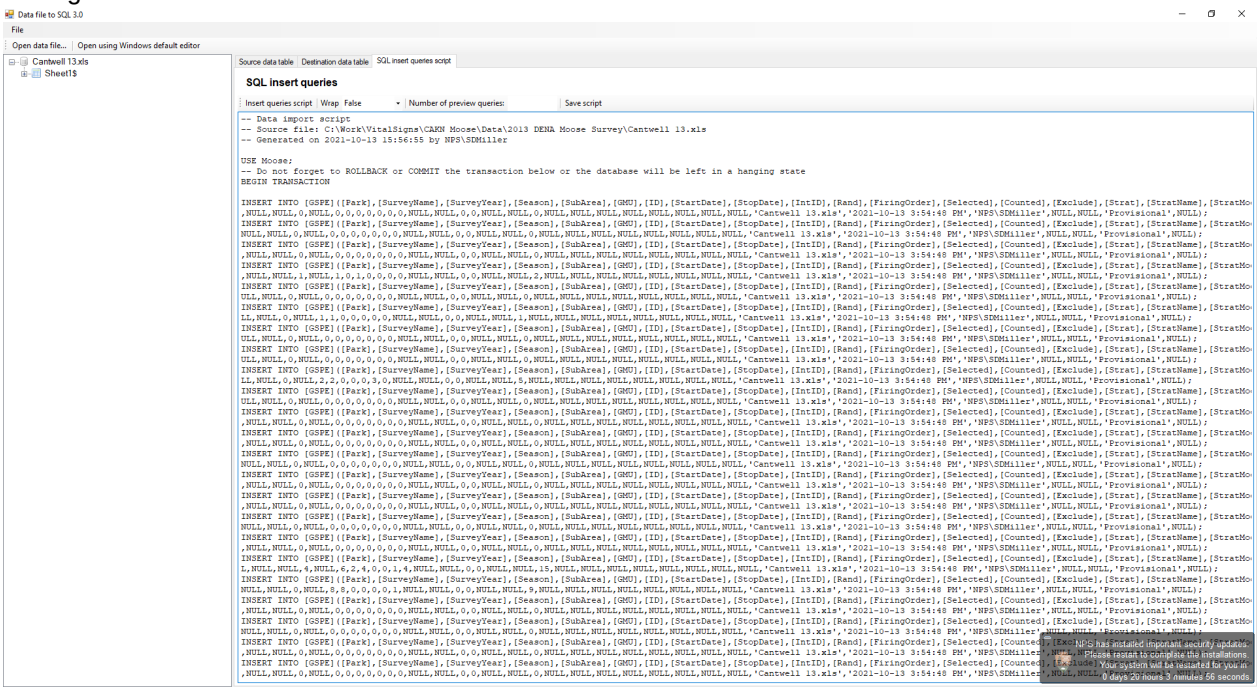
Carefully check your mappings and if all looks good click Import transformed data
A dialog appears requesting the database name. Enter 'Moose'.



Another dialog appears requesting the name of the destination database table. Enter 'GSPE'.



If all goes well the tool will generate a script of SQL insert queries that you may execute in SQL Server Management Studio to insert the records.



Copy the text of the insert queries script.
Open SSMS
Navigate to the moose monitoring database.
Open a new query window and copy the insert queries script into it.
Execute the script.
If all the queries executed with no failures then highlight the word COMMIT in the 6th line and execute the command to commit the new records to the database.
If SSMS reported any errors then highlight the work ROLLBACK to revert the insert queries. Fix any errors and try again.
YOU MUST COMMIT OR ROLLBACK or the database will be left in a hanging state with no access to others.
Issue an SQL query to ensure the records exist. Example: SELECT * FROM GSPE WHERE SurveyName='2013 DENA Moose Survey'.
Ensure the number of records inserted matches the number of rows in the source data table.
Sum the data in the source spreadsheet using a formula and compare the result to summed data in the database. Example SELECT Sum(CALF) as TotalCalves FROM GSPE WHERE SurveyName='2013 DENA Moose Survey'.

Data access

Data access

Most modern data consuming software can connect to the moose monitoring database. This section shows typical connection methods.

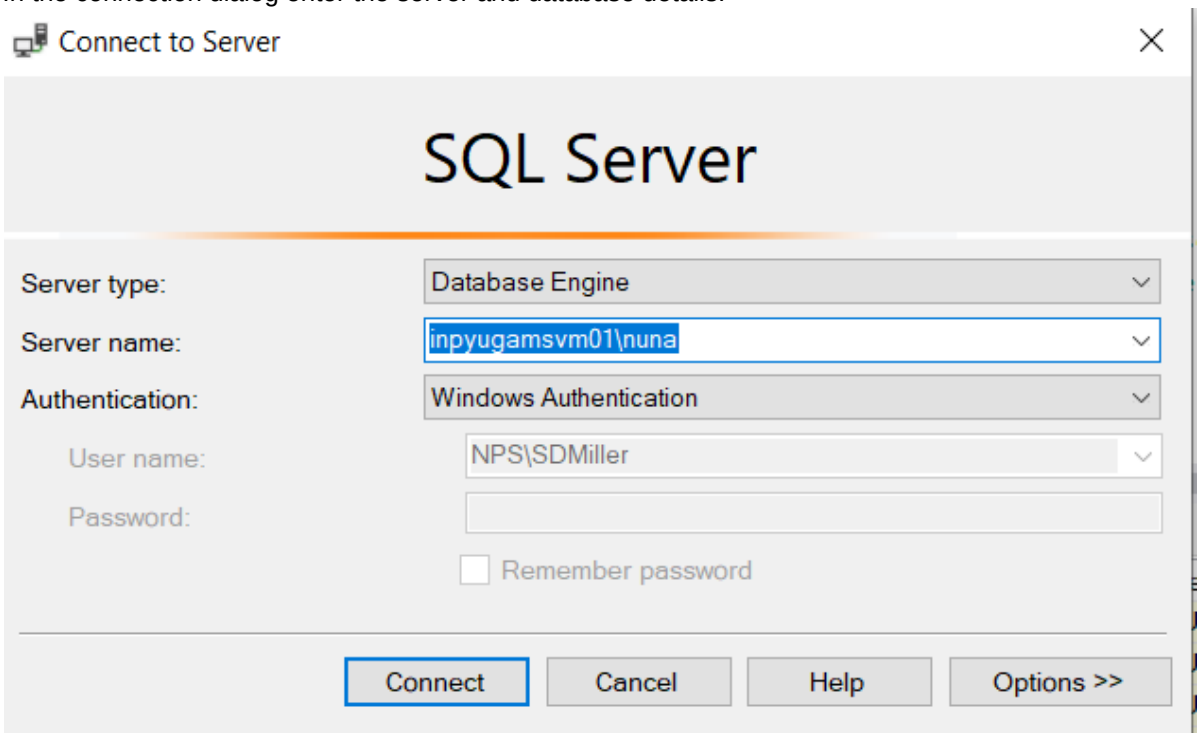
Microsoft Sql Server Management Studio

Microsoft Sql Server Management Studio

Microsoft Sql Server Management Studio (SSMS) is the best tool for administering the database and has good functionality for querying, analyzing and summarizing results. While the software is not difficult to use, it does require an advanced understanding of relational databases.

Process

1. Open SSMS.
2. In the Object Explorer click Connect
3. In the connection dialog enter the server and database details:



4.
5. Expand the server node.
6. Expand the Databases node and select Moose.
7. Consult the SSRS help and online documentation for further information on working with the Moose database.

Microsoft Excel

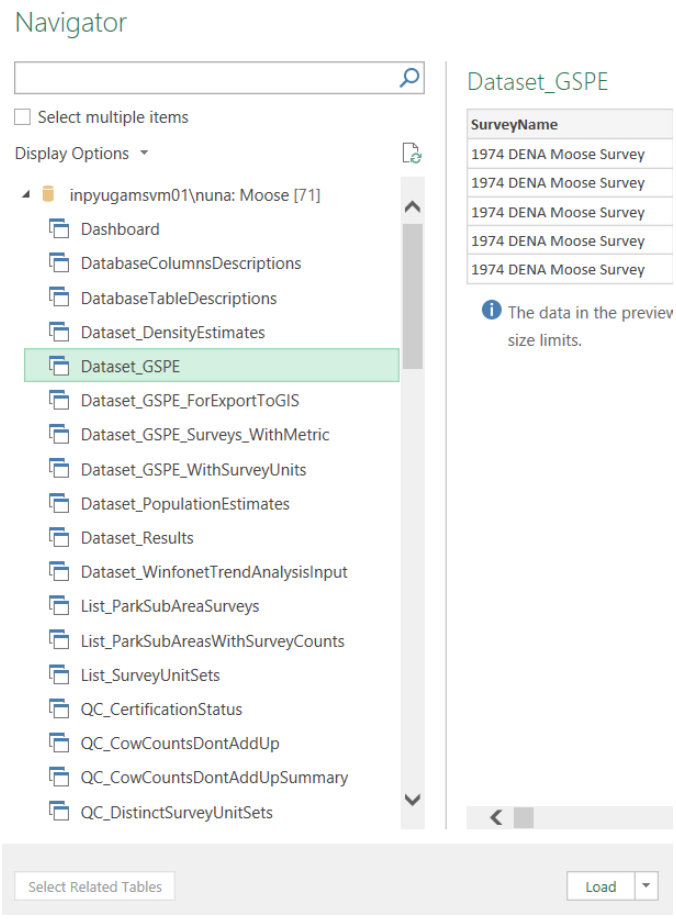
Accessing the Moose database with Microsoft Excel

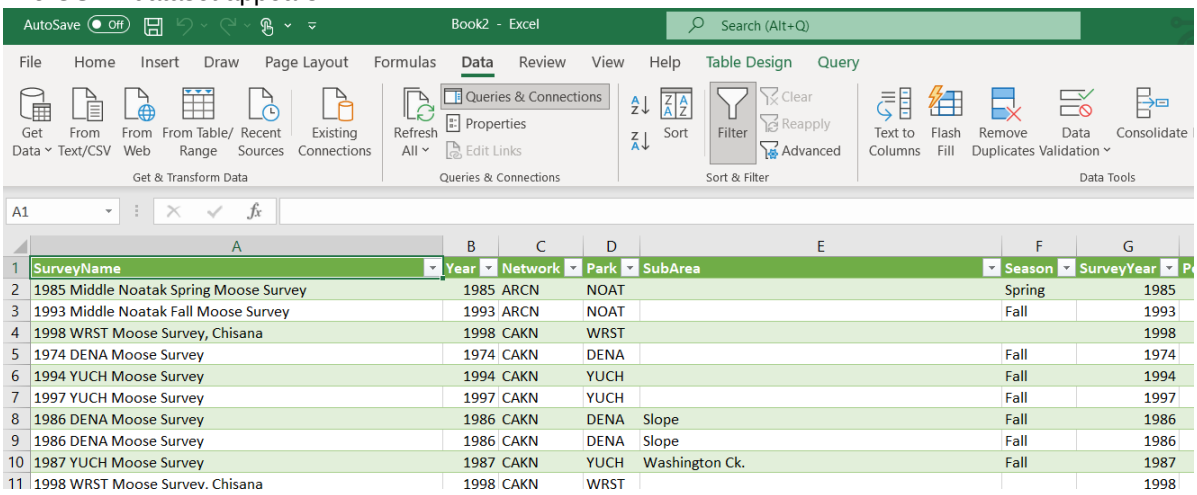
Prerequisite
At least datareader permissions. Contact your data manager.

Process

1. Open an Excel workbook
2. From the Data menu select **Get Data -> From Database -> From Sql Server Database**
3. Enter **inpyugamsvm01\nuna** for the Server and **Moose** for the database.
4. 

5. From the Navigator choose the query you would like to pull. Typically, you will request Dataset_GSPE to pull the whole dataset. Note: this dataset contains tens of thousands of records and may take a minute or so to retrieve depending on your network speed.

6. 

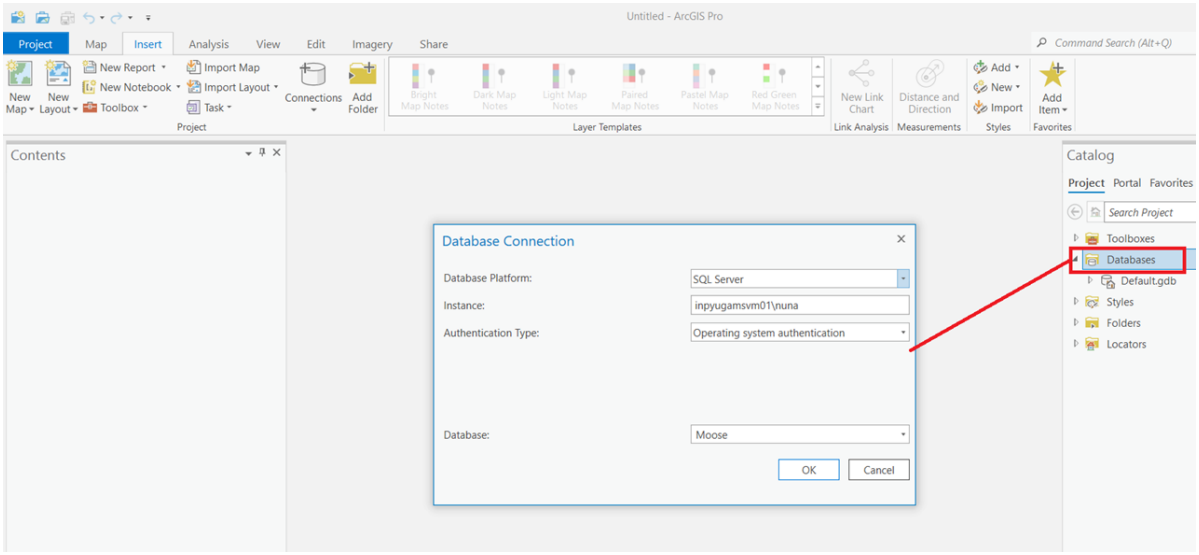
7. Click Load or Transform data.
8. The GSPE dataset appears.
9. 

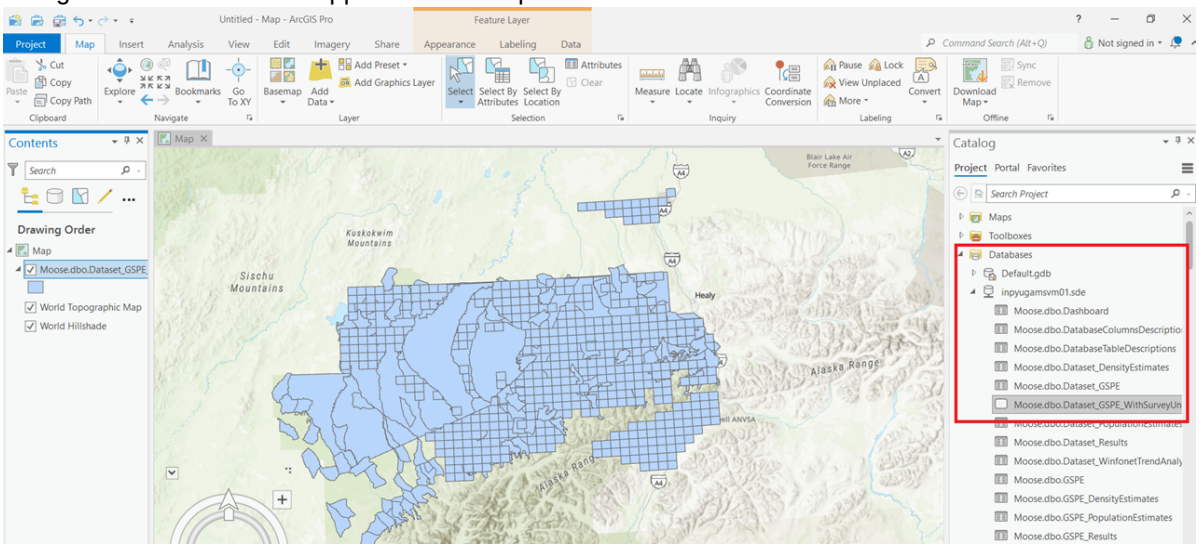
ArcGIS Pro

Accessing the Moose database with ArcGIS Pro

Prerequisite
At least datareader permissions. Contact your data manager.

Process
1. Open ArcGIS Pro

2. Locate the Catalog
3. Right click Databases -> New database connection
4. Enter the server/database information as shown in the screenshot below:
5. 

6. All the database tables and queries become available to you. Select Dataset_GSPE_WithSurveyUnits and drag it onto the map (there are tens of thousands of records which may take a while to load).
7. If all goes well the data will appear on the map
8. 

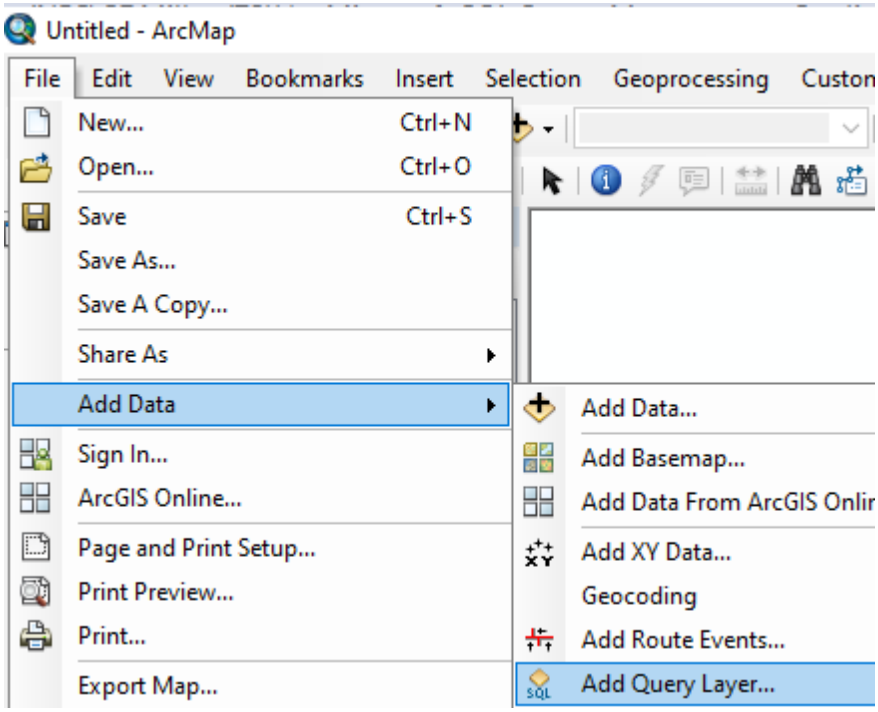
ArcMap

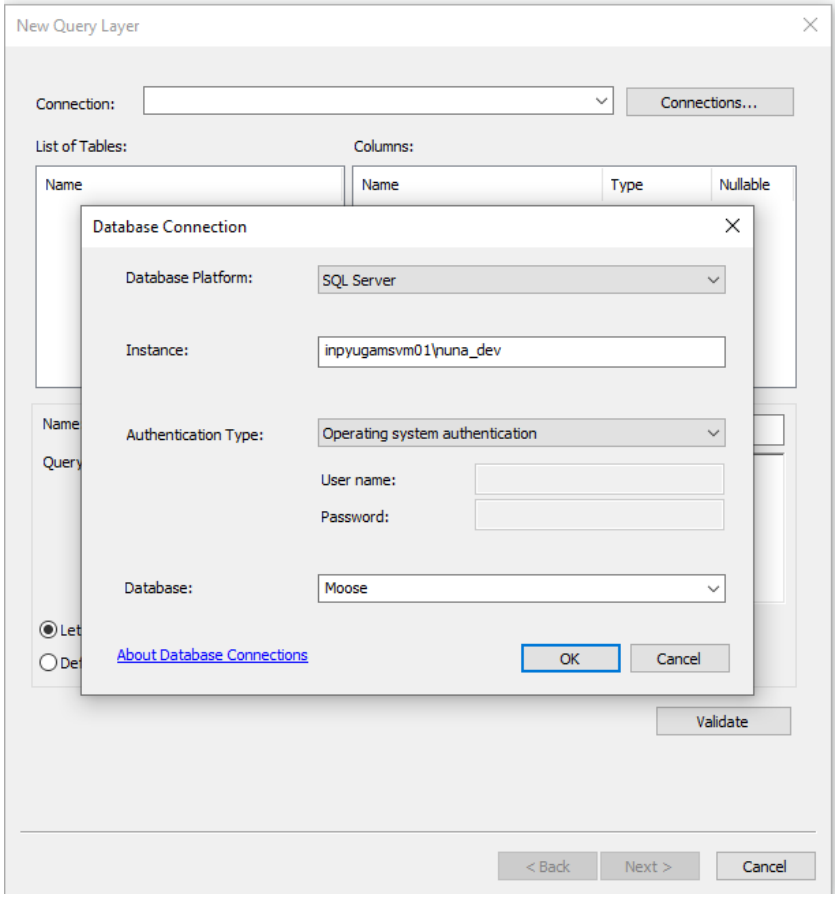
Accessing the Moose database with ArcMap

Using ArcMap to access data in the Moose Monitoring database is possible, but not recommended. It just doesn't work well. There is much better support for SQL Server database connections with ArcGIS Pro. See ArcGIS Pro.

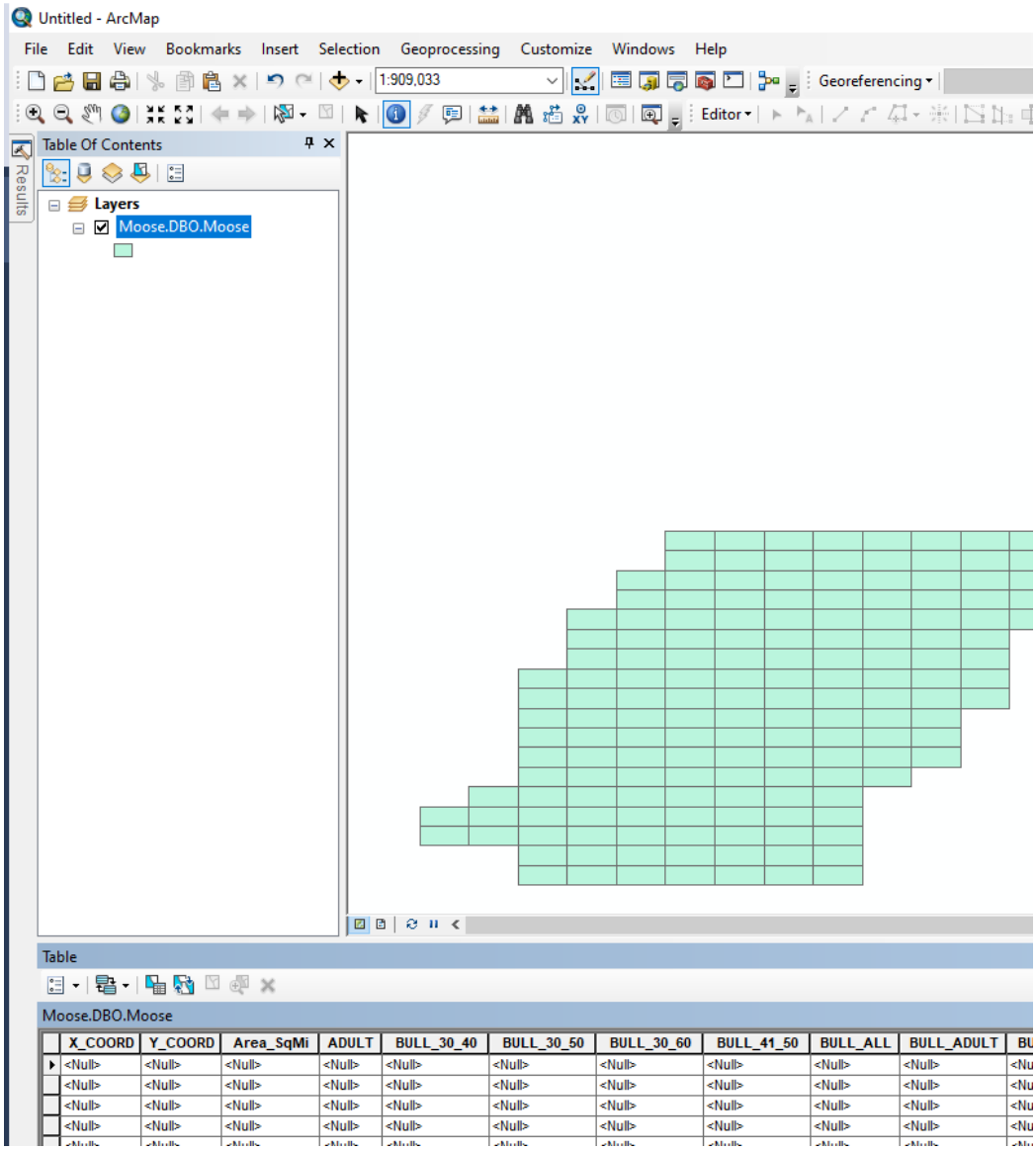
Prerequisite
At least datareader permissions. Contact your data manager.

Process
1. Open ArcMap
2. Add a Query Layer: File -> Add Data -> Add Query Layer

3. 

4. Make a new connection to the Moose database
5. 

6. Enter a query. The most likely query you will want is Dataset_GSPE_WithSurveyUnits. This is the same query as Dataset_GSPE but it includes the survey unit polygons with the GSPE data. Note: You will almost certainly want to filter your query with a WHERE clause to prevent overloading ArcMap with tens of thousands of polygons.
7. An example query: `SELECT * from Dataset_GSPE_WithSurveyUnits WHERE SurveyName='2015 DENA Moose Survey'`
8. Click Validate and move through the rest of the wizard.
9. The data should appear:

10. 

11. Consider exporting the data to a local shapefile or geodatabase to make analysis quicker by eliminating round trip passes over the network which will be slow. Otherwise analyze like any other GIS layer.
12. If you find errors please take the time to communicate them to the I&M data managers so they can be corrected for other users of the dataset!

R

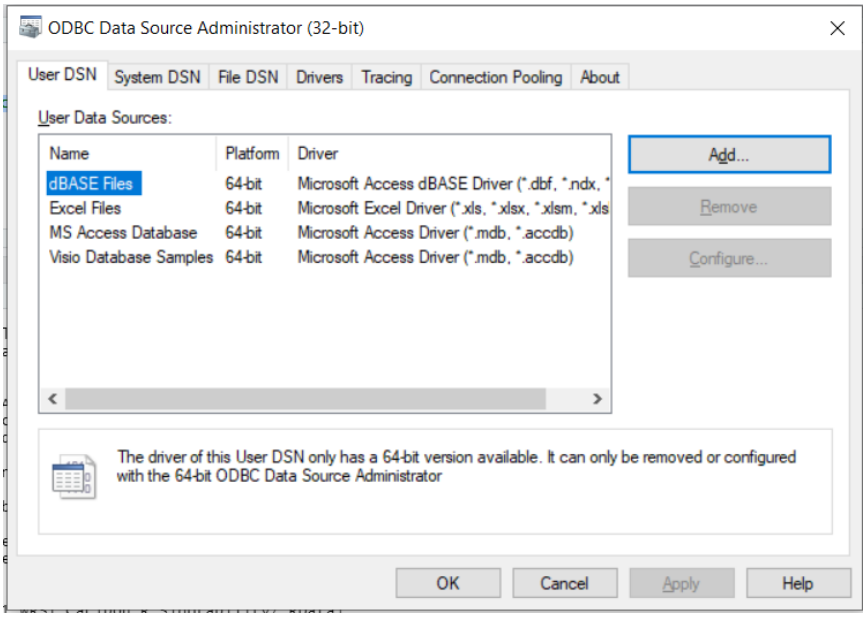
Accessing the Moose database with R

Prerequisite
At least datareader permissions. Contact your data manager.
An ODBC database connection to the Moose database.

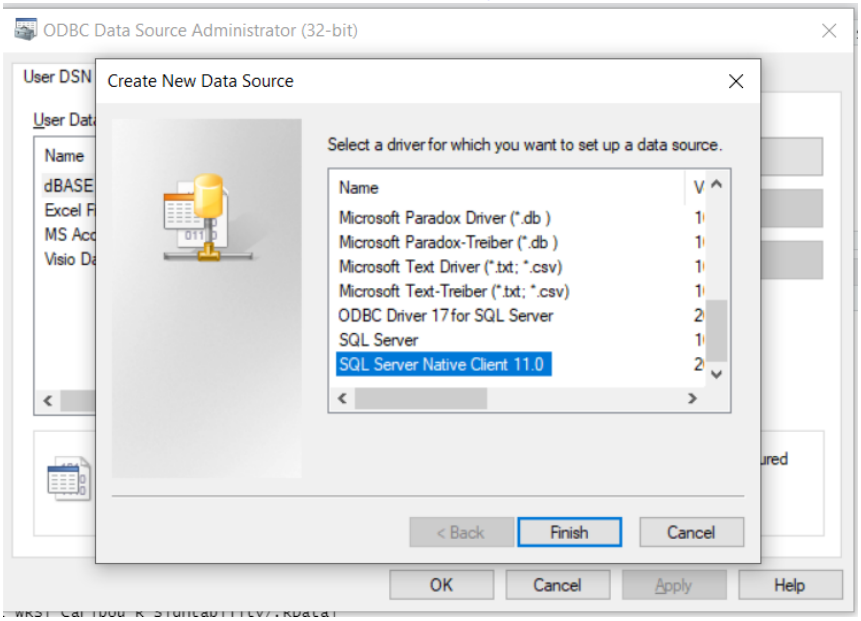
Create an ODBC connection

Process
1. Click the Windows key and search 'ODBC' to start the ODBC Data Source Administrator

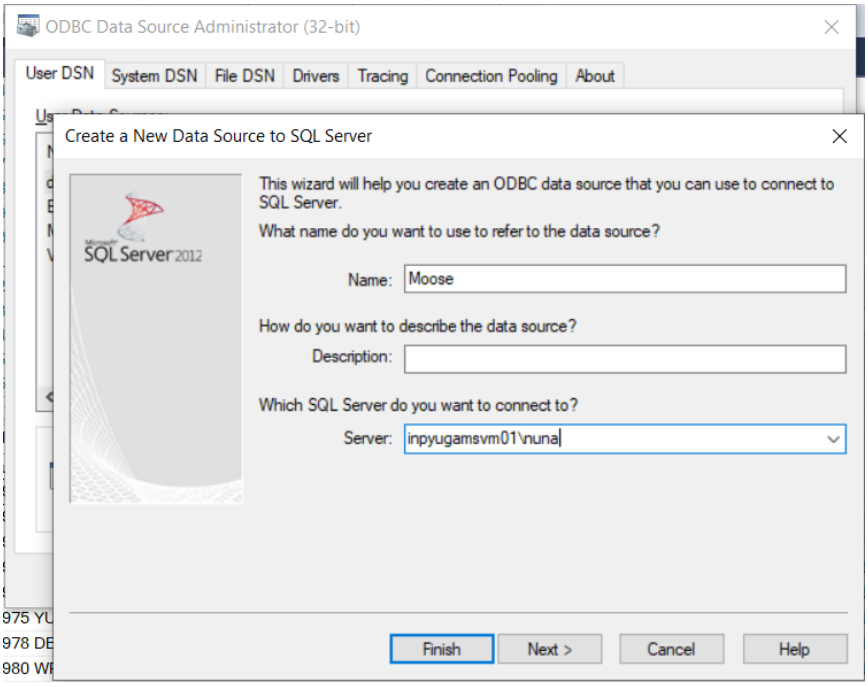
2. Click through the Wizard to create an ODBC connection to the Moose database
3. Click Add...

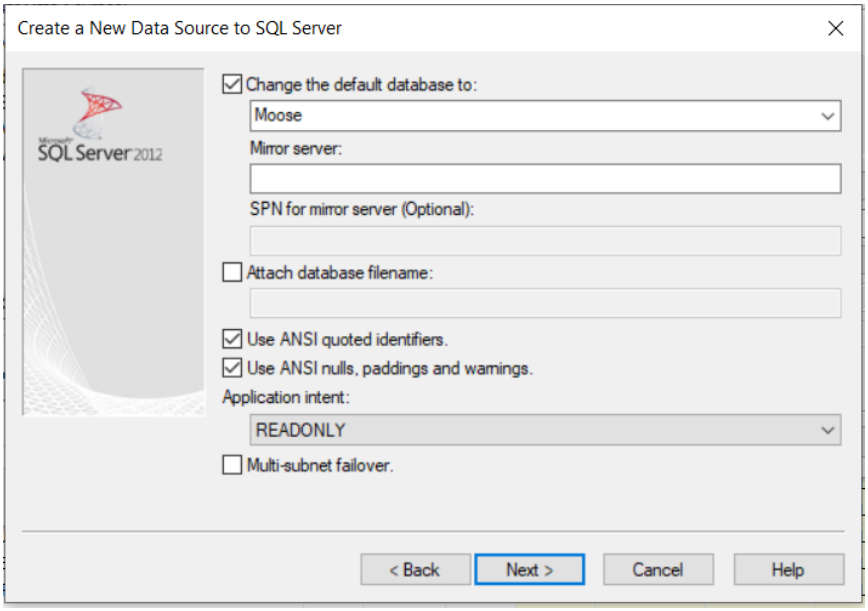


1.
2. Scroll down to Sql Server or Sql Server Native Client.If you don't have on ODBC driver for Sql Server see [Download ODBC Driver for SQL Server]



3.

4. 

5. 

4. Test the connection. If all goes well you can now connect to the database through this connection using R.

Example R script that connects to the Moose database

The example below shows how to connect to the Moose database and access the GSPE dataset. Modify the query as needed or omit the WHERE clause to pull the whole dataset (tens of thousands of records).

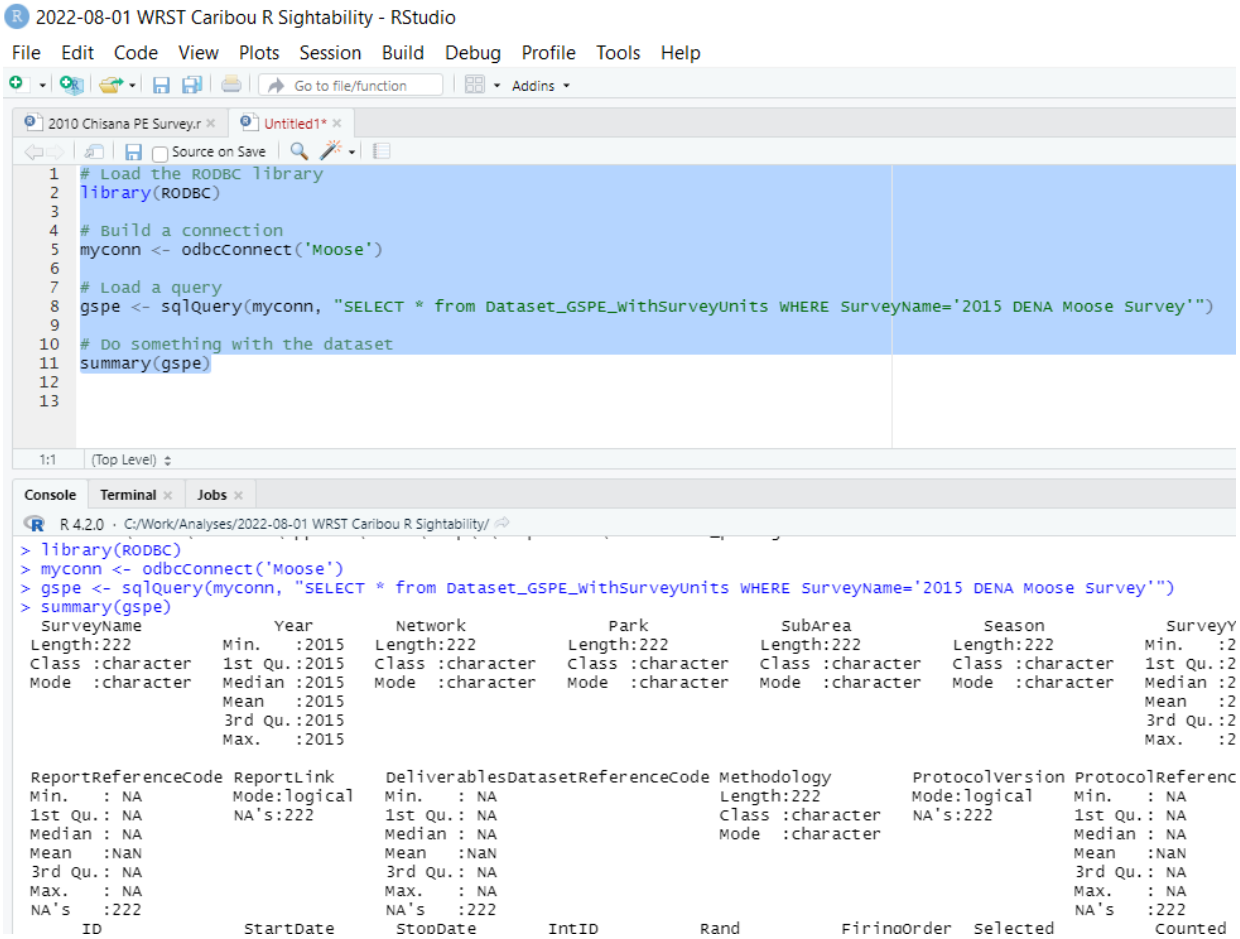
```
# Load the RODBC library
library(RODBC)

# Build a connection using the Moose ODBC connection
myconn <- odbcConnect('Moose')

# Load a query
gspe <- sqlQuery(myconn, "SELECT * from Dataset_GSPE_WithSurveyUnits WHERE
SurveyName='2015 DENA Moose Survey'")

# Do something with the dataset
```

```
summary(gspe)
```



Python (ArcPy)

Accessing the Moose database with Python (ArcPy)

Prerequisite
At least datareader permissions. Contact your data manager.

The most likely scenario for accessing the Moose database using Python is to extract some or all of the dataset for analysis. There is a Python script at Dataset publication that loads the GSPE data into ArcMap and exports it as a shapefile. This script can be modified to suit your needs.

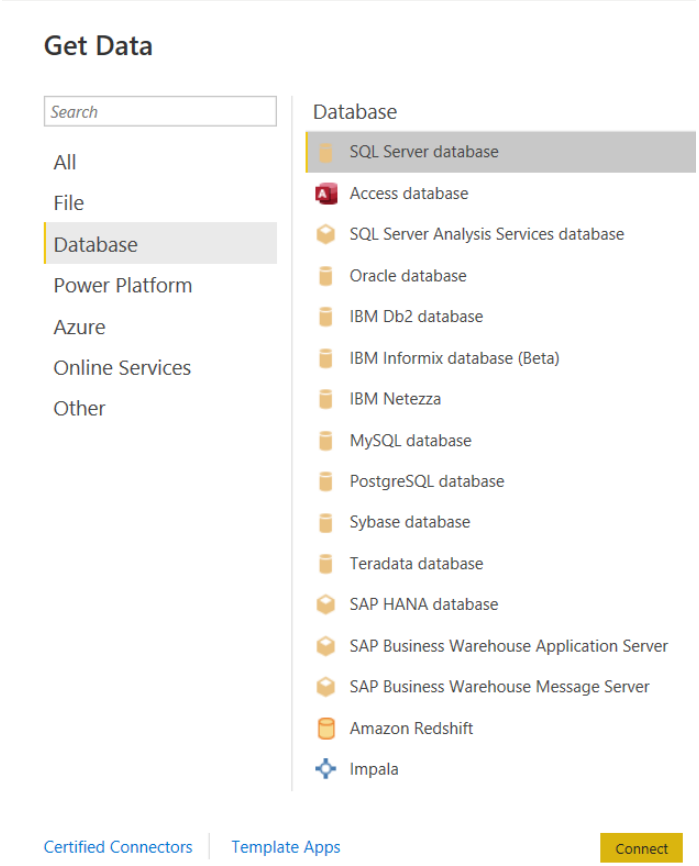
Power BI

Accessing the Moose database with Microsoft Power BI

Prerequisite
At least datareader permissions. Contact your data manager.
Power BI. Download here https://inp2300sqlsas02.nps.doi.net/PBIReports/browse/Development/CAKN

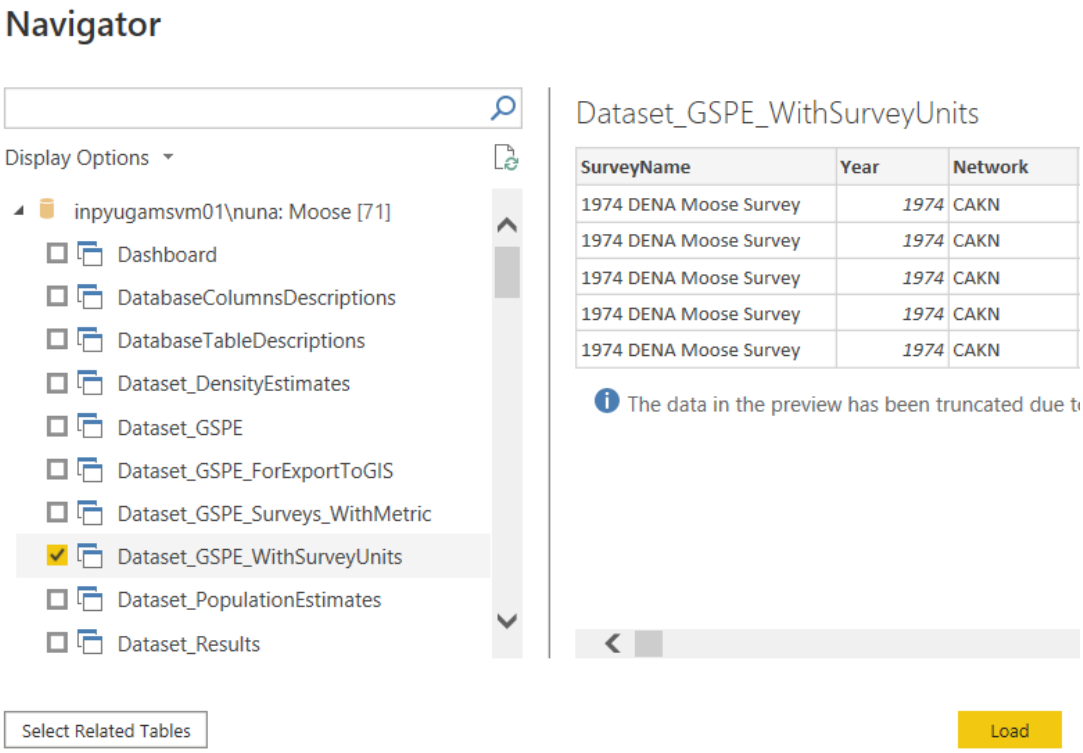
Process
1. Open Power BI

2. Click Get data
3. Select Database -> SQL Server database

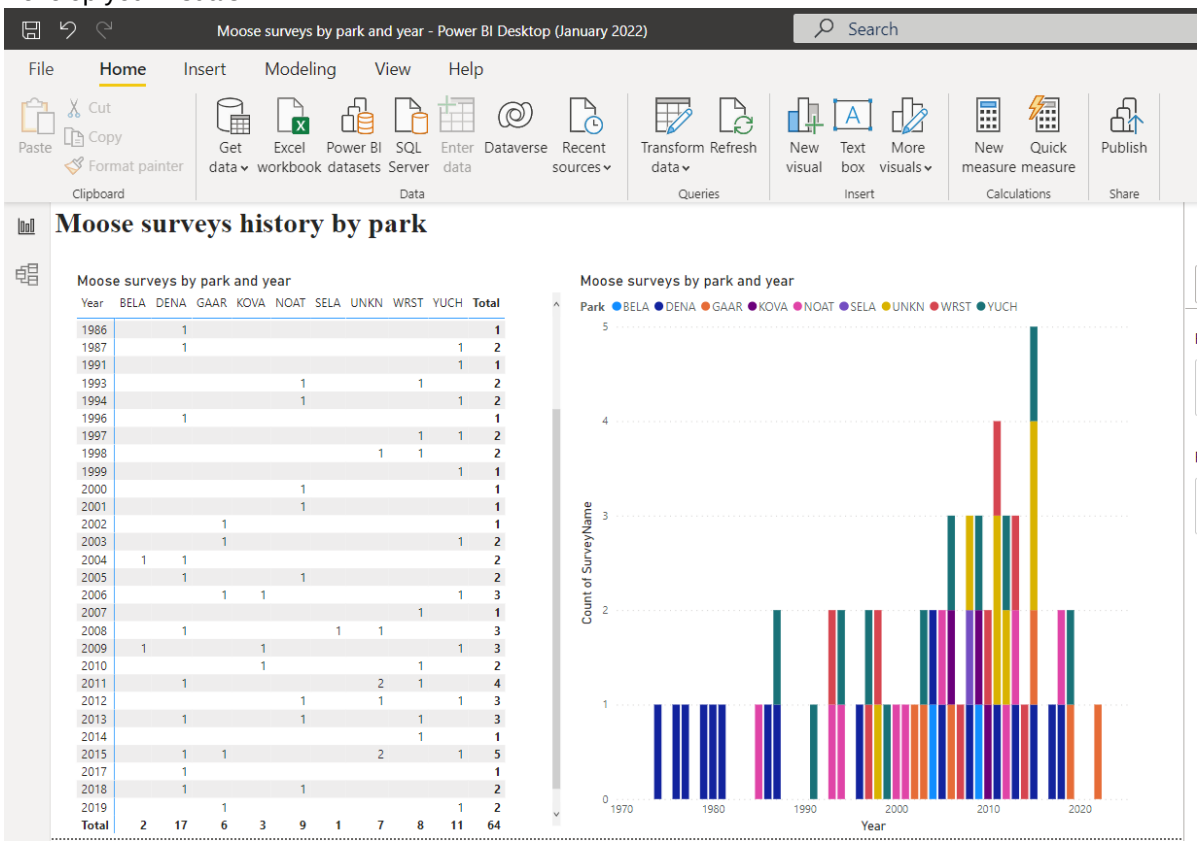


4.
5. Click Connect
6. Enter the server and database names

7.
8. The database Navigator opens:



9.
10. Select Dataset_GSPE_WithSurveyUnits
11. Develop your visuals



12.

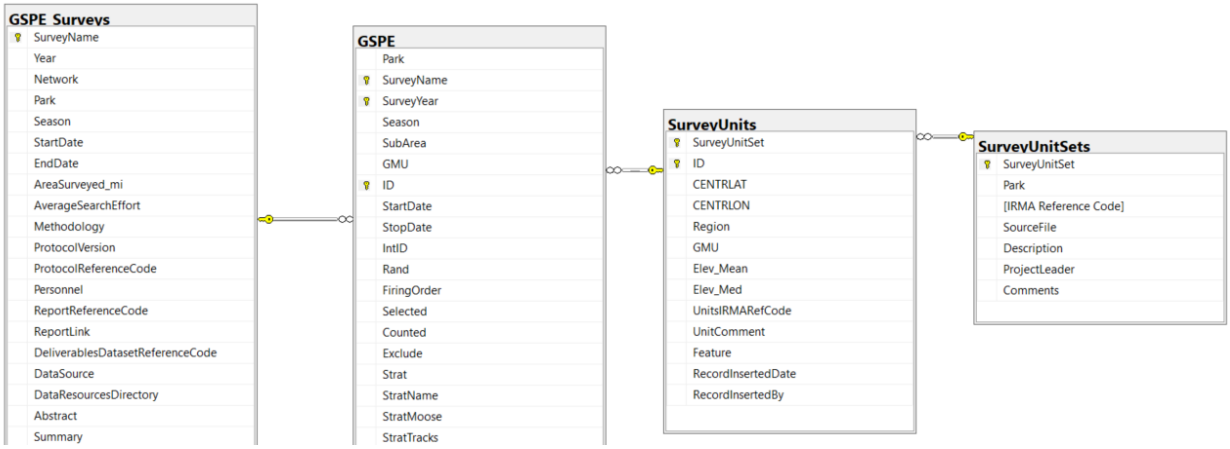
Database documentation

Database documentation

Database details

SQL Server: inpyugamsvm01\nuna
Database: Moose

Data model



Columns descriptions

Table	Column	DataType	Length	Precision	Scale	Nullabl
GSPE	ADULT	int	4	10	0	1
GSPE	Area_SqMi	float	8	53	NULL	1
GSPE	BULL_30_40	int	4	10	0	1
GSPE	BULL_30_50	int	4	10	0	1
GSPE	BULL_30_60	int	4	10	0	1
GSPE	BULL_41_50	int	4	10	0	1
GSPE	BULL_ADULT	int	4	10	0	1
GSPE	BULL_ALL	int	4	10	0	1
GSPE	BULL_GT_50	int	4	10	0	1
GSPE	BULL_GT_60	int	4	10	0	1
GSPE	BULL_GTE_50	int	4	10	0	1
GSPE	BULL_LT_30	int	4	10	0	1
GSPE	BULL_LT_50	int	4	10	0	1
GSPE	CALF	int	4	10	0	1
GSPE	CALF_LONE	int	4	10	0	1
GSPE	CertificationDate	datetime	8	23	3	1

GSPE	CertificationLevel	varchar	15	15	NULL	0
GSPE	CertifiedBy	varchar	50	50	NULL	1
GSPE	Comments	varchar	255	255	NULL	1
GSPE	Counted	bit	1	1	NULL	1
GSPE	COW	int	4	10	0	1
GSPE	COW_W_0	int	4	10	0	1
GSPE	COW_W_1	int	4	10	0	1
GSPE	COW_W_2	int	4	10	0	1
GSPE	COW_W_3	int	4	10	0	1
GSPE	DateCounted	date	3	10	0	1
GSPE	Density	float	8	53	NULL	1
GSPE	Exclude	bit	1	1	NULL	1
GSPE	FiringOrder	int	4	10	0	1
GSPE	GMU	varchar	20	20	NULL	1
GSPE	ID	varchar	10	10	NULL	0
GSPE	Int	int	4	10	0	1
GSPE	IntID	int	4	10	0	1
GSPE	IsIntensivePlot	bit	1	1	NULL	1
GSPE	IsStdPlot	bit	1	1	NULL	1
GSPE	LG_BULL	int	4	10	0	1
GSPE	MED_BULL	int	4	10	0	1
GSPE	MED_L_BULL	int	4	10	0	1
GSPE	MOOSE	int	4	10	0	1
GSPE	Observer	varchar	50	50	NULL	1

GSPE	Park	varchar	4	4	NULL	0
GSPE	Perc_Not_Flown	int	4	10	0	1
GSPE	Personnel	varchar	100	100	NULL	1
GSPE	Pilot	varchar	50	50	NULL	1
GSPE	Rand	float	8	53	NULL	1
GSPE	RecordInsertedBy	nvarchar	100	50	NULL	0
GSPE	RecordInsertedBy	sysname	100	50	NULL	0
GSPE	RecordInsertedDate	datetime	8	23	3	0
GSPE	SCF_Plot	bit	1	1	NULL	1
GSPE	SearchMin	float	8	53	NULL	1
GSPE	Season	varchar	20	20	NULL	1
GSPE	Selected	bit	1	1	NULL	1
GSPE	SM_BULL	int	4	10	0	1
GSPE	SourceFilename	varchar	255	255	NULL	1
GSPE	StartDate	datetime	8	23	3	1
GSPE	Std	int	4	10	0	1
GSPE	StopDate	datetime	8	23	3	1
GSPE	Strat	varchar	1	1	NULL	1
GSPE	StratHab	varchar	100	100	NULL	1
GSPE	StratMoose	int	4	10	0	1
GSPE	StratName	varchar	10	10	NULL	1
GSPE	StratTracks	int	4	10	0	1

GSPE	SubArea	varchar	50	50	NULL	1
GSPE	SurveyID	int	4	10	0	1
GSPE	SurveyName	varchar	100	100	NULL	0
GSPE	SurveyUnitSet	varchar	100	100	NULL	1
GSPE	SurveyYear	int	4	10	0	0
GSPE	UNKNOWN	int	4	10	0	1
GSPE	X_COORD	float	8	53	NULL	1
GSPE	Y_COORD	float	8	53	NULL	1
GSPE	YBULL_ALL	int	4	10	0	1
GSPE	YBULL_GTSF	int	4	10	0	1
GSPE	YBULL_SF	int	4	10	0	1
GSPE_Surveys	Abstract	varchar	-1	-1	NULL	1
GSPE_Surveys	AreaSurveyed_mi	float	8	53	NULL	1
GSPE_Surveys	AverageSearchEffort	float	8	53	NULL	1
GSPE_Surveys	Comments	varchar	-1	-1	NULL	1
GSPE_Surveys	DataResourcesDirectory	varchar	1000	1000	NULL	1

GSPE_Surveys	DatasetProcessingSteps	varchar	-1	-1	NULL	1
GSPE_Surveys	DataSource	varchar	2000	2000	NULL	1
GSPE_Surveys	DeliverablesDatasetReferenceCode	int	4	10	0	1
GSPE_Surveys	EndDate	date	3	10	0	1
GSPE_Surveys	Methodology	varchar	50	50	NULL	1
GSPE_Surveys	Network	char	4	4	NULL	1
GSPE_Surveys	Park	varchar	4	4	NULL	1
GSPE_Surveys	Personnel	varchar	200	200	NULL	1
GSPE_Surveys	ProtocolReferenceCode	int	4	10	0	1
GSPE_Surveys	ProtocolVersion	float	8	53	NULL	1
GSPE_Surveys	RecordInsertedBy	varchar	50	50	NULL	0
GSPE_Surveys	RecordInsertedDate	datetime	8	23	3	0
GSPE_Surveys	ReportLink	varchar	2000	2000	NULL	1
GSPE_Surveys	ReportReferenceCode	int	4	10	0	1
GSPE_Surveys	Season	varchar	10	10	NULL	1
GSPE_Surveys	StartDate	date	3	10	0	1
GSPE_Surveys	Summary	varchar	-1	-1	NULL	1
GSPE_Surveys	SurveyName	varchar	100	100	NULL	0
GSPE_Surveys	ValidatedBy	varchar	50	50	NULL	1
GSPE_Surveys	ValidatedDate	date	3	10	0	1

GSPE_Surveys	Year	int	4	10	0	0
SurveyUnits	CENTRLAT	float	8	53	NULL	1
SurveyUnits	CENTRLON	float	8	53	NULL	1
SurveyUnits	Elev_Mean	float	8	53	NULL	1
SurveyUnits	Elev_Med	float	8	53	NULL	1
SurveyUnits	Feature	hierarchyid	-1	-1	NULL	1
SurveyUnits	Feature	geometry	-1	-1	NULL	1
SurveyUnits	Feature	geography	-1	-1	NULL	1
SurveyUnits	GMU	varchar	5	5	NULL	1
SurveyUnits	ID	varchar	10	10	NULL	0
SurveyUnits	RecordInsertedBy	nvarchar	100	50	NULL	0
SurveyUnits	RecordInsertedBy	sysname	100	50	NULL	0
SurveyUnits	RecordInsertedDate	datetime	8	23	3	0
SurveyUnits	Region	int	4	10	0	1
SurveyUnits	SurveyUnitSet	varchar	100	100	NULL	0
SurveyUnits	UnitComment	varchar	1000	1000	NULL	1
SurveyUnits	UnitsIRMARefCode	int	4	10	0	1
SurveyUnitSets	Comments	varchar	-1	-1	NULL	1
SurveyUnitSets	Description	varchar	4000	4000	NULL	1
SurveyUnitSets	IRMA Reference Code	int	4	10	0	1
SurveyUnitSets	Park	varchar	4	4	NULL	1
SurveyUnitSets	ProjectLeader	varchar	50	50	NULL	1
SurveyUnitSets	SourceFile	varchar	255	255	NULL	1
SurveyUnitSets	SurveyUnitSet	varchar	100	100	NULL	0