

PISCES mapping is a flexible automation system for generating large numbers of maps in the same style, but with different underlying data. While the most common map outputs from PISCES are species range maps, these maps are not hard coded into PISCES - they are instead created through map configurations called Map Sets.

These map sets are not limited to generating species range maps. They can combine and mash up any data in the PISCES database, including species status information, connections between range information, or HUC 12 characteristics, in order to generate maps with different thematic content.

## Generating Maps

### Generate Maps Tool

The **Generate Maps Tool** is a part of the PISCES toolbox. It can be used to generate preconfigured maps set in ArcMap and ArcCatalog.

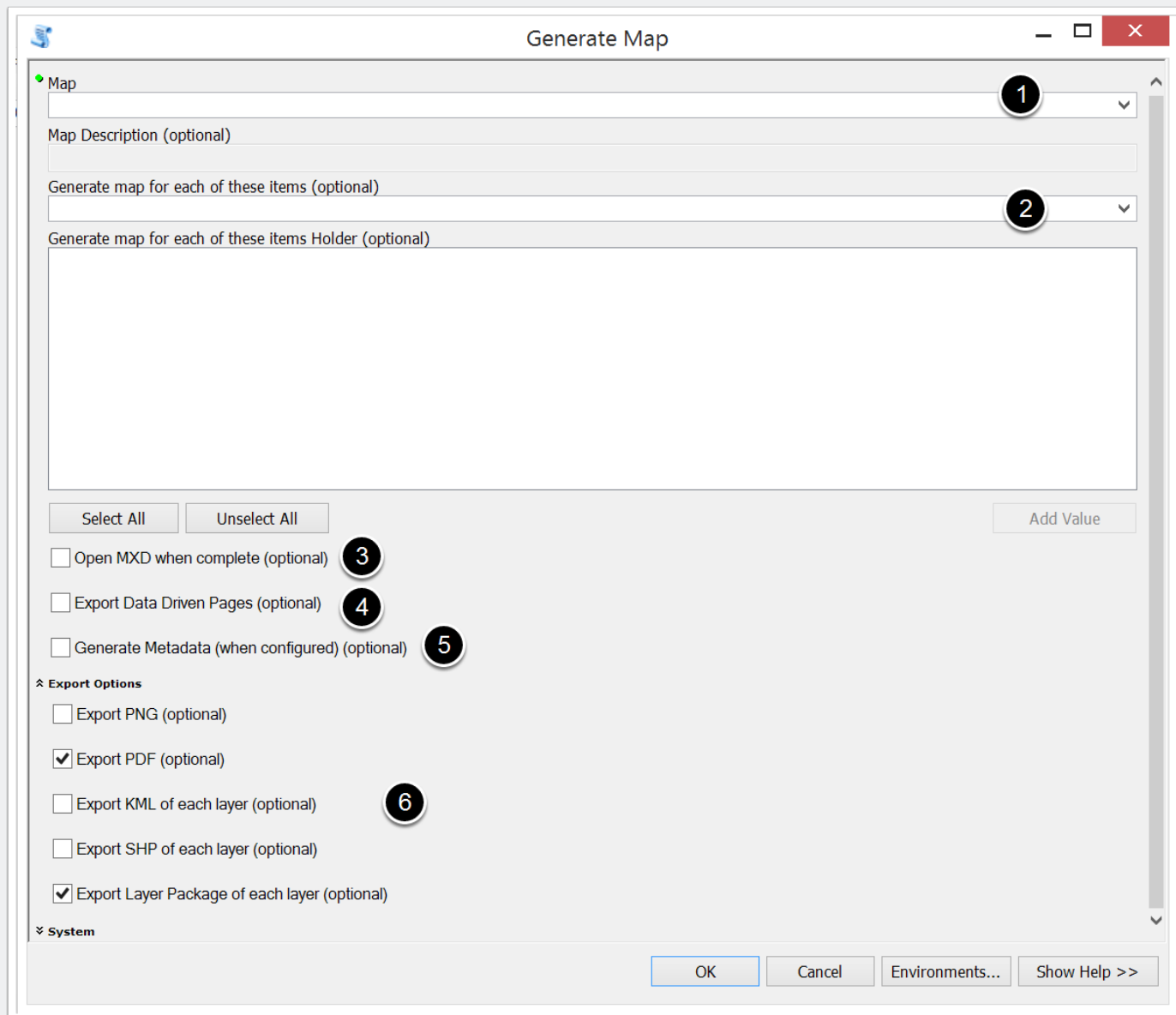
1. Select desired map set from the drop down menu. If the map set you want to run is not visible in the list, double check that it is enabled in your user profile (See **Users Accounts - Access to Mapsets**).
2. Select items to generate. This can be a species, a list of species or a species group. Items selected from the drop down list will be added to the holder box. Map sets that are already configured to run for certain groups (ie: Native Richness is set up to run only native fish species), this option will appear grayed out.
3. Open MXD when tool finishes. This option will only work if you are generating a map with a single item. ArcGIS is unable to open multiple map documents at the same time. The map documents will be saved to the MXDs folder in the PISCES install location.

# Maps

4. Data driven pages gives you the ability to generate a set of output pages by taking a single layout and iterating over a set of map extents. It is set up for **Main Range Maps** and **Unlimited Range Maps** to iterate over forest service regions.
5. Generate metadata for the layers in the map. This will add some processing time to generating the maps.
6. Select the desired output options.

Tool will run. Outputs will be saved in the MXD subfolder and the Maps subfolder in the PISCES installed location.

# Maps



## Error while using Generate Maps

If you get an error message like "Error removing the previous dataset for this query. Skipping this Map.....", please check if you have a previously generated map for that

# Maps

query already open. PISCES is unable to overwrite the layers if they are currently open. Close the open map documents and try running the tool again.



## Generate Maps Command Line

Maps can also be generated using the PISCES command line. Please see the full documentation for more information.

## Map Sets

PISCES mapping is a flexible automation system for generating large numbers of maps in the same style, but with different underlying data. Map outputs are not hard coded into PISCES - they are instead created through map configurations called Map Sets. These map sets are not limited to generating species range maps. They can combine and mash up any data in the PISCES database, including species status information, connections between range information, or HUC 12 characteristics, in order to generate maps with different thematic content.

The most critical part of PISCES mapping to understand are that each map set has a template ArcGIS map document that's used as the base map. Into this map document, PISCES injects the layers that are generated for the map. Each map has one or more layers, which are produced by executing a custom SQL query against the PISCES database.

### Creating a new map set

To create a new map set:

1. Set up new record in `defs_query_sets`
2. Create layer queries to be included
3. Add map set to user profile

### Define map set in `defs_query_sets`

To create a new map set in `defs_query_sets`:

# Maps

1. Add a row to the **defs\_query\_sets** table in the PISCES database. Please see the **Database Tutorial** for more information if you don't know how to do this.
2. Give your new map set a **set\_name** (this will show up in the tool drop down menu), a **short\_name** (internal - so no spaces), and a **set\_description**.
3. Fill the **map\_title** with information including map bind variables ( curly braces { } that swap in information about the species to allow maps to be run in batch with customized titles)
4. **base\_mxd** - name of the map document base template in mxds\base that will be used to base for the generated maps. See below for more information about creating base template mxds.
5. Optional - **ddp\_mxd** - name of the map document base template in mxds\base used for data driven pages (must be enabled in base map).
6. Remember to commit changes in the data base to save the row.

See full documentation for more information

SQLiteStudio (v2.1.5) [defs\_query\_sets (pisc.es.sqlite)]

Databases Tables Indexes Triggers Views Window Tools Help

Structure Data Indexes Triggers DDL

Grid view Form view

Total rows: 25

#	id	set_name	map_title	short_name	set_description	base_mxd	ddp_mxd	iterator	active	callback
1	10	Records	Set only modify recor	NULL	Internal - don't use	NULL	NULL	NULL	0	NULL
2	19	All Diversity	Multiple Diversity	all_div	Runs all the major diversity for richness_lite.mxd				0	
3	18	Beta Diversity	Beta Diversities	beta_div	Outputs various beta diversity blank_graphic_leg				0	
4	12	Connectivity	Connectivity	connectivity	Determines the number of ups: blank.mxd				0	
5	17	Data Gaps	Empirical Observations by	data_gap	Outputs whether or not a huc: blank_graphic_leg				0	
6	26	Downstream Diver	Downstream Diversity	downstream	Outputs data on the assemble richness_lite.mxd				0	
7	36	Inset - Range Map (Species)		inset	Range maps with California in blank_graphic_leg			Species_Groups.FID	0	
8	23	Meadows_Richness	Richness of Meadow Ind	meadows_richne	Outputs richness, but only for meadows_richness				0	
9	4	Native Richness	Native Species Richness	native_richness	A basic alpha richness map ru richness_lite_native	NULL			0	
10	14	Network Distance	Network Distance	network	Determines the HUC network: blank.mxd			HUC12FullState:HUC_12	0	
11	7	NonNative Richness	Non-Native Species Rich	nonnative_richne	Nonnative alpha richness.	richness_nonnative			0	
12	6	QA Dist	{Species} Dist for QA	qa_dist	A map set that outputs color: bw_range.mxd	bw_ddp.mxd		Species_Groups.FID	0	
13	2	Main Range Maps (Species)	Distribution by range		Outputs all quality controlled d: blank_graphic_leg: fs_ddp_graphic_legen			Species_Groups.FID	0	
14	16	Unlimited Range h	{Species} Distribution by range_uni		This set is identical to Main R: blank_graphic_leg: fs_ddp_graphic_legen			Species_Groups.FID	1	
15	38	Raster	Raster	raster	Map set for exporting ranges : blank.mxd			Species_Groups.FID	0	
16	21	Streamlines Repre	{Species} Distribution as representation		Uses the representation call: blank_streamline_			Species_Groups.FID	0	
17	34	HUC8 Richness	Richness By HUC 8	rich_8	Alpha richness aggregated by blank_graphic_leg				0	
18	3	Species Richness	Species Richness by HU	richness	A basic alpha richness map to richness_all_base.r				0	
19	22	Richness Differen	Richness Difference (Cur	richness_diff	Calculates the difference betw richness_difference				0	
20	9	Sensitive Native T	Sensitive Native Taxa by	sensitive_taxa	Alpha richness of sensitive na richness_native_be				0	
21	28	Sensitivity_Metrics	Species Sensitivity By HL	sensitivity	Provides a measure of sensit: richness_lite.mxd			Species_Groups.FID	0	
22	30	Status_scores_av	Status Scores	status	Average fish status score of s blank_graphic_leg				0	

## Set layers in map\_queries

Each layer in the generated map must be defined by a query in the **map\_queries** table. This is not trivial. Please use the established layer queries as a guide for proper syntax and expression. If the query is not correct, the map set will not work.

1. For each layer to be included in the map, add a row to the **map\_queries** table.
2. **custom\_query** - SQL query for the layer. Each query should select HUC12s as zone\_id
3. **layer\_name** - Name for the layer. Also see **name\_formula** for naming exported layers with species information.
4. **query\_set** - Find ID number for query set in **defs\_query\_sets**
5. **query\_rank** - The order the the queries will be processed when multiple layers are in a map set (rank 1 will be the top layer in the map)
6. **layer\_file** - the .lyr file in mxds/base folder used for symbology for the layer
7. **callback\_function** and **callback\_args** - please see the **Callbacks** section in the main documentation for details.

The screenshot shows the SQLiteStudio interface with the **map\_queries** table selected. The table has 39 rows and 6 columns. Annotations 1 through 5 are placed over the following fields:

- 1: **view** button in the toolbar
- 2: **custom\_query** column
- 3: **layer\_name** column
- 4: **query\_set** column
- 5: **query\_rank** column

#	id	custom_query	layer_name	query_set	query_rank	description
1	1	select observations.zone_id, Observations.Set_ID, Observations.IF_Method, Obser...	Extant Range - Expert Opinion	2	6	
2	2	select distinct observations.zone_id, Observations.Set_ID, Observations.IF_Metho...	Observed	2	4	
3	3	select HUC_12 as zone_id from HUC12FullState	All Species Richness	3	1	
4	6	select HUC_12 as zone_id from HUC12FullState	Native Species Richness	4	1	
5	9	select HUC_12 as zone_id from HUC12FullState	NonNative Species Richness	7	1	
6	12	select HUC_12 as zone_id from HUC12FullState	Sensitive Species (Endangered or Vulnerable)	9	1	
7	13	select distinct zone_id from Observations where Species_ID = ? And Presence_Typ...	New Distribution	10	1	Only used by modify...
8	16	select distinct observations.zone_id, Observations.Set_ID, Observations.IF_Metho...	Historical Range - Expert Opinion	2	2	
9	17	select distinct observations.zone_id, Observations.Set_ID, Observations.IF_Metho...	Translocated - Expert Opinion	2	5	
10	18	select distinct observations.zone_id, Observations.Set_ID, Observations.IF_Metho...	Translocated - Observed	2	3	
11	19	select HUC_12 as zone_id from HUC12FullState	Connectivity Info	12	1	
12	20	select ZoneU as zone_id from Connectivity where ZoneD = ?	Upstream Zones	13	1	
13	21	Select ZoneD as zone_id from Connectivity where ZoneU = ?	Downstream Zones	13	2	
14	22	select HUC_12 as zone_id from HUC12FullState where HUC_12 = ?	Selected Zone	13	3	A hack to make sure
15	23	select HUC_12 as zone_id from HUC12FullState	Network Distance	14	1	
16	32	select distinct zone_id from Observations where Species_ID = ? And Presence_Typ...	Extant Range - Expert Opinion - Unedited	16	6	
17	33	select distinct zone_id from Observations where Species_ID = ? And (Presence_Ty...	Observed - Unedited	16	4	
18	34	select distinct zone_id from Observations where Species_ID = ? And (Presence_Ty...	Historical Range - Expert Opinion - Unedited	16	2	
19	35	select distinct zone_id from Observations where Species_ID = ? And Presence_Typ...	Translocated - Expert Opinion - Unedited	16	5	
20	36	select distinct zone_id from Observations where Species_ID = ? And Presence_Typ...	Translocated - Observed - Unedited	16	3	
21	37	select distinct zone_id from Observations where Presence_Type in (1,7,9,10)	Observations - Boolean	17	1	
22	38	select HUC_12 as zone_id from HUC12FullState	Native Beta Diversity	18	1	
23	39	select HUC_12 as zone_id from HUC12FullState	NonNative Beta Diversity	18	2	

## Add map set to map\_users

In order to have the new map set show up in the **Generate Maps** tools, you will need to add the map set ID to the **map\_users** table in the database. Please see the **User Accounts tutorial** for more information.

## Base Maps

Base maps are the ArcGIS map documents files that are used as templates to automatically generate map sets. Base maps and layer symbology are both saved in the mxds\base folder. Each map set has a template file that it uses by default that is defined in the **defs\_queries\_sets** table. Changing the template file will change all of the maps that use that map document file as a base.

## Placeholder Layer

The most important feature of a template files is a empty placeholder layer called "Distribution". This placeholder feature that sits at the level in the document where we wish to place new features. It allows the layers that are set in map\_queries be passed to the output map documents.

The empty layer must have its data source as PISCES/data/layer\_cache.gdb/blank\_feature.



## Bind Variables

Base maps use mapping bind variables to pass specific information from the map query to text boxes in the map document. Bind variables are curly braces {text} that are exchanged with appropriate data from the query. Bind variables allow flexibility and customization in automatic map generation.

Supported bind variables include:

{Title}, {Scientific Name}, {Species}, {Bind}, {Date}, {Version}, {Sources}

Bind variables are case sensitive but can be used in any text box in base map templates.