

TEKDB GIS User Guide

A live version of this document is available at: https://github.com/Ecotrust/TEKDB/wiki/gis_guide

This guide is intended to assist GIS Technicians, Analysts, and other professionals connect to the spatial data in TEKDB's PostgreSQL+PostGIS database. It is not intended to function as a guide for how to interpret and use the data - it instead assumes users are familiar with the features of the tools they are using to connect to the database, and they know why they are accessing the data.

This document assumes the TEK DB tool was installed on a Linux server. If it was installed on a Windows server, GIS connectivity **MIGHT** not be possible.

Connecting to the PostgreSQL/PostGIS database with ArcMap is possible, but you will only have 'read only' access: You can pull data out, and do whatever you want with it, but you can't write to or edit records inside of the database. If you need this level of access, it is recommended you look into using another tool such as QGIS.

Connecting ArcGIS to TEKDB's Spatial Data

Based on steps from [this guide for ArcMap 10.4](#)

Adding the connection to your catalog

To connect to the database, a connection record will need to be added to the catalog, either via ArcMap or ArcCatalog. These steps walk you through this using ArcMap:

- Collect the following information from your Systems Administrator or IT department (determined during [Server Installation](#)):
 - The TEK DB IP address
The format (IPv4) should be 4 numbers between 0 and 255 separated by a period, such as 192.168.0.1
 - The TEK PostgreSQL DB user name
 - The TEK PostgreSQL DB user password
 - The TEK PostgreSQL DB name
- Open ArcMap
- Open Catalog
 - This can be found under the 'Windows' menu in the top bar
- Expand the selection menu in the location bar and select 'Database Connections'
- Under the 'Database Connections' folder, select 'Add Database Connection'
- Fill out the following fields in the resulting Dialog Box:
 - Database Platform: 'PostgreSQL'
 - Instance: The IP address of the server collected above
 - Authentication Type: 'Database authentication'
 - User name: the PostgreSQL user name collected above
 - Password: the PostgreSQL user password collected above
 - Save user name and password
 - Database: A database of the name collected above should be available in the drop-down list once ArcGIS has had a moment to connect after entering the user's password.
NOTE: On ArcMap 10.5 the system will claim that it could not connect to the database, however if you type the database name in correctly, and everything else is installed correctly it will let you in anyway.
- Click 'OK'

Adding The Layers To Your Map

Spatial data is stored in PostgreSQL/PostGIS differently than most spatial sources: features of all types may be stored in the same layer or table. TEK allows Points, Lines, and Polygons to all exist in it's primary spatial table 'places'. ArcMap requires that all features on a layer be of the same type, so to see all features, the layer will need to be added three times.

- Under your connection to the TEK DB in your catalog you will see a great number of tables, all prefixed with the database name, followed by '.public.'.
- Two spatial tables exist: 'locality' and 'places' - please consult with your TEK Application Administrator as to whether your instance uses 'locality' (if unsure, assume "no"). This document only specifically covers 'places', but managing 'locality' would be much the same.
- Add '(db name).public.places' to your 'Table Of Contents' three times.
- Note which feature type (Point, Line, Polygon) is represented in all three.
- Change the other two so that all three are represented:
 - Right Click on the layer name
 - Select 'Properties'
 - Under the 'Source' tab:
 - Click the 'Change Query' button
 - Under 'List of Tables, select tekdb.public.places
 - Leave the 'Query' alone
 - Select 'Let ArcGIS discover spatial properties for the layer'
 - You may define your own: the SRID is 3857
 - Click 'Next'
 - 'placeid' should already be selected as the 'Unique Identifier Field'
 - Change 'Geometry Type' to 'Point', 'Line' or 'Polygon'
 - Depending on which feature type you wish this layer to present
 - Click 'Finish'
 - Click 'Apply'

Layer Information

The 'places' table contains the following attributes. Notes on fields that require a join are included as well:

- **enteredbyname** - the name of the user who first created the data in the database, if available
- **enteredbytribe** - the name of the tribe that the user who first created the data in the database belongs to
- **enteredbytitle** - the title of the user who first created the data in the database
- **enteredbydate** - the date that this record was first created
- **modifiedbyname** - the name of the last user to modify this record
- **modifiedbytitle** - the title of the last user to modify this record
- **modifiedbytribe** - the tribe of the last user to modify this record
- **modifiedbydate** - the date this record was last edited
- **placeid** - the primary key of the table
- **indigenousplacename** - the place's traditional indigenous name
- **indigenouosplacenamemeaning** - a translation of the place's indigenous name to English
- **englishplacename** - the English name used for the place
- **islocked** - a leftover remnant from the MTKEDB database (deprecated)
- **geometry** - the spatial data field, possibly represented as the geometry type
- **source** - the name of the source of this data
- **digitizedby** - the name of the user who digitized this record
- **digitizeddate** - the date this record was digitized
- **planningunitid** - the ID of the planning unit this place is associated with
 - Join with '(db name).public.lookupplanningunit' to get the name
- **primaryhabitat** - the ID of the habitat type
 - Join with '(db name).public.lookuphabitat' to get the name
- **tribeid** - the ID of the tribe
 - Join with '(db name).public.lookuptribe' to get the name

For More information on what data is available in the remaining layers and how they are joined, please see the [Layer Definitions](#) documentation.

Other Ways To Connect

This spatial data is made available to a wide assortment of tools for viewing, access, and analysis. Here are a few options:

- [QGIS](#) - An Open Source alternative to ArcMap. Good for when you are on a computer that doesn't have an ArcGIS license.
 - **Added bonus:** With QGIS you can not only read the spatial data from the TEK DB, but you can **WRITE TO IT**, TOO!

- **Web Browser** - The TEK application displays spatial data on maps within most web browsers. Administrators even have the ability to create, edit, and delete this data.

Differences between TEKDB and MTKEDB

The MTKEDB was an AccessDB where a separate spatially-enabled Access database was made available so both the MTKEDB Application users and the GIS crew on ArcGIS could enter info and sync data about the places in the database. Also, it is intended to be installed on Windows servers within a network and run via Microsoft Access.

The new TEKDB is a web application built on a PostgreSQL database. Spatially-enabled with PostGIS, all of the data can coincide in the same database at once without having to synchronize on an ID field. It is installed on a Linux server and run through a web-browser.

Connecting to the old tool was somewhat automated - install and run the add-in, connect to the spatial-database to the places and localities layers, point the application toolbar at the non-spatial database to define the joining ID between the two databases. It came pre-zoomed in to the Tolowa region with a number of helpful layers: NatGeo_World_Map, Populated Places, Water Features, Streams, Water Bodies, Township and Range, Counties, and Quads.

The convenience of the add-in is commendable, and it is recommended that if you are used to it, keep the template it created for MTKEDB projects, but going forward, you will want to replace the old Places and Localities layers with the new ones (by connecting to the PostgreSQL database, as described below).

Since the new DB is consolidated, the old styling on places will not carry over and will have to be redone. Also, a few fields (such as FeatType) now have new names (such as primaryhabitat) that match the non-spatial Access Database rather than the spatial one integer values, representing an ID in the lookuphabitat table.

Despite this inconvenience, there are a number of advantages to the new database. The largest of which is that data browsers can see spatial data within the tool (in fact, editors can even edit this spatial data), and in the meantime, GIS professionals can access all of the data directly through ArcGIS. Another major difference, in terms of maintaining database integrity, is the consolidation of the Places and Localities tables - data no longer has to be duplicated between the spatial and non-spatial tables and kept in sync. All parties are looking at the same live data.