MANAGING DBS

Tutorial

Managing Relational Databases

with

TNTmips®

TNTedit™

TNTview®

Before Getting Started

TNTmips®, TNTedit™, and TNTview® provide great flexibility when working with database materials. Databases can be in the internal format or in any format supported by Open Database Connectivity (ODBC). You can visualize information directly from databases (provided coordinate locations are included with each record) or the information can be related to elements in any object type for viewing and use in analysis operations.

Prerequisite Skills This booklet assumes you have completed the exercises in the *Displaying Geospatial Data* and *Navigating* tutorial booklets. Those exercises introduce essential skills and basic techniques that are not covered again here. Please consult these booklets for any review you need.

Sample Data The exercises presented in this booklet use sample data distributed with the TNT products. If you do not have access to a TNT products CD, you can download the data from MicroImages' web site. The first exercises in this booklet use the CB_SOILS Project File in the DB_MANAG directory of DATA. The objects in the CONSTRNT, PROPINFO, UNTDSTAT, and USA_OLD Project Files, as well as a text file and format files in this directory, are also used. Make a read-write copy of the sample data on your hard drive so changes can be saved when you use these objects.

More Documentation This booklet is intended only as an introduction to the database functions in TNTmips, TNTedit, and TNTview. Consult the TNTmips reference manual for more information.

TNTmips and TNTlite™ TNTmips comes in two versions: the professional version and the free TNTlite version. This booklet refers to both versions as "TNTmips." If you did not purchase the professional version (which requires a software license key), TNTmips operates in TNTlite mode, which limits the size of your project materials. All exercises in this booklet can be completed in TNTlite using the sample geodata provided.

Merri P. Skrdla, Ph.D., 4 October 2002

It may be difficult to identify the important points in some illustrations without a color copy of this booklet. You can print or read this booklet in color from MicroImages' web site. The web site is also your source of the newest Getting Started booklets on other topics. You can download an installation guide, sample data, and the latest version of TNTlite.

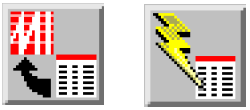
http://www.microimages.com

Welcome to Managing Databases

Attributes stored in databases can be visualized directly (pin mapping) or in conjunction with elements in vector, CAD, TIN, or even raster objects. The use of attributes is one of the key differences that separates geographic information system (GIS) software from simpler graphics packages.

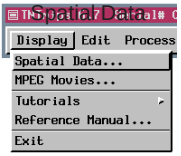
There are three booklets in this tutorial series that address different aspects of database / attribute utilization and management in TNTmips: this booklet, *Pin Mapping*, and *Managing Geoattributes*. This booklet focuses on the databases themselves with sections on viewing and manipulating the structure of a database (what tables it contains and how they are related), adding tables and making forms, and introducing statistics and computed fields to augment the information already present in your database. How to use databases in other formats that provide ODBC drivers is also described. The *Pin Mapping* booklet describes how to visualize databases directly when each record contains location information. The *Managing Geoattributes* booklet describes how to get the database information associated with elements in other object types. Obtaining histograms and XY plots from field values and saving tables in CAD format for inclusion in hardcopy layouts are also described in the *Managing Geoattributes* booklet.

All of the procedures described in this booklet are available in TNTmips, TNTedit, and TNTview. The primary database management utility is available in TNTmips as both a stand alone process and from within Spatial Data Display. The stand alone process is absent in TNTview, but all the features are available in the main display process (choose Edit Relations from the menu associated with the Make Table / Form icon).



STEPS

- ☑ launch TNT
- ☑ use Support / Maintenance / General File or your operating system to copy the needed .dbf and .txt files (see page 2) to your local drive
- ☑ use Support / Maintenance / Project File or your operating system to copy the RVC files in the DB_MANAG directory of DATA
- ☑ choose Display / Spatial Data...



Page 4 defines some key database terminology. Pages 5–14 provide instruction for using the features of the Database Editor. Pages 15–24 lead you through a variety of techniques, such as adding tables, using computed fields, and introducing statistics. Pages 25–27 describe database import and ODBC setup.

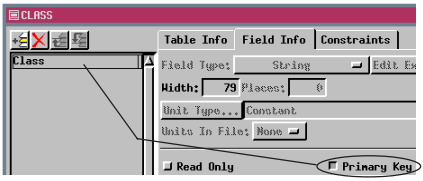
Database Terminology

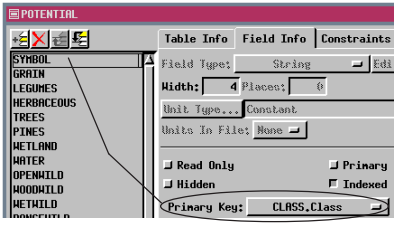
Vocabulary: a **primary key** field provides unique identification of related elements, such as the soil type of a polygon in a soil map. Once the records in a table containing a primary key field are attached to the elements in an object, other tables that contain this same information can be related to the same elements if the primary key in the directly attached table is referenced. When the primary key field in one table contains the same information and is referenced by a field in another table, it is called a **foreign key**.

In some cases in order to be unique, a primary key must be made from a number of field values, such as first, middle, and last name fields. The first of these fields is designated the primary key and the other fields are each referred to as a **next key** with the order established by the designation.

Database software often only allows one primary key in a table, but this is not strictly enforced in the TNT products since organization of materials into tables may make two primary keys a logical choice for reference by other tables (such as address and resident's name).

A database in TNTmips can be either a main level object used for pin mapping or a subobject of a vector, CAD, TIN, sketch, or raster object. Databases that are subobjects contain attributes associated with elements (vector, CAD, TIN, sketch) or cell values (raster). These attributes can be organized into one or more tables. Each table covers different aspects of the same common theme defined by the database.

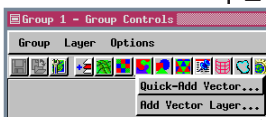




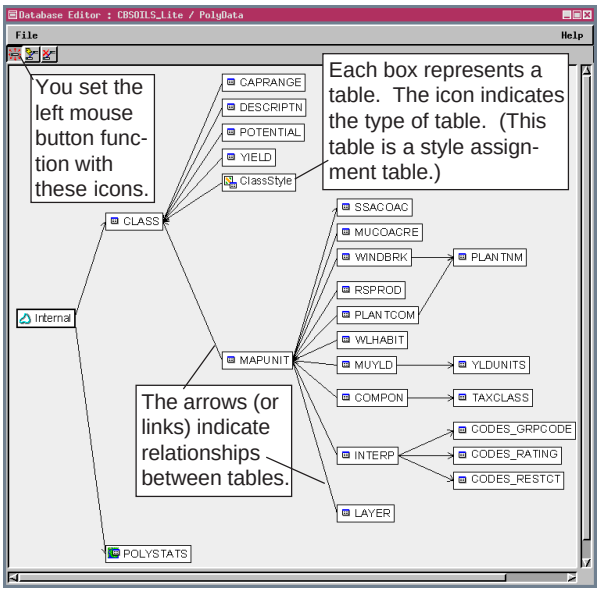
In order to retrieve attribute information for selected elements, individual records within a database table must be attached or related to those elements. (Table attachment types and establishing attachments between records and elements are described in the *Managing Geoattributes* booklet.) A relational database lets you define a primary key in one table that is directly attached to individual elements and then relate all other tables with the same attribute to the same elements by establishing a foreign key relationship. Thus, even a database as complex as that associated with the Crow Butte soil map (23 soil-related attribute tables) requires minimal effort to establish a retrievable relationship between polygons and the records in each of the tables. In the Crow Butte soils polygon database only two of the 23 tables are directly attached to the elements, and one of these (the polygon statistics table) had the attachments created automatically by the process that generated the table. The other 21 tables are indirectly related by foreign key fields.

Viewing Existing Database Structure

To view and edit database structure in the Spatial Data Display process or TNTview, select the object with which the database is associated for display, click on the Show Details icon, and choose Edit Relations from the Make Table / Form icon menu for the appropriate element type. You can use the Database Editor whether or not the element type with which the database is associated is active for selection. In TNTmips, you can also select Edit / Attribute Databases to run the process. You need to then specify the element type if a vector object is chosen.



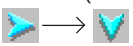
When you add a layer, you need to turn on the Show Details icon button if you want to list all database tables or view attributes in any but the first table, which opens automatically when you select an element if you have not shown object details.



STEPS

- ☑ choose New 2D Group from the Spatial Data Display toolbar

- ☑ choose Quick-Add Vector from the Add Vector icon menu and select CBSOILS_LITE from the CB_SOILS Project File
- ☑ click on the Show Details icon for the vector layer you added (CBSOILS_LITE)



- ☑ scroll down or enlarge the window as needed to reveal the polygon element selection icon row, and select Edit Relations from the Make Table / Form icon menu
- ☑ click on the Move Table icon if it is not currently the active icon on the Database Editor toolbar

Move Table

Relate

Unrelate

Keep this database open in the Database Editor through the exercise on page 9.

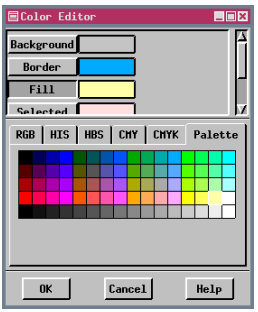
Colors and Text Style

STEPS

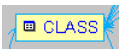
- ☑ right-click over the background of the Database Editor window and select Text Style from the menu

- ☑ click on [Font] to select a different font, choose one from the list, and click [OK] in the Scalable Font Selection window
- ☑ click on a tile in the Foreground Color palette, then click [OK]
- ☑ right-click over the background of the Database Editor window and select Colors from the menu

- ☑ choose a new Fill color from the palette for the table rectangles then click on [Border] and select a contrasting color for the rectangle outlines and links



- ☑ click [OK] in this Color Editor window

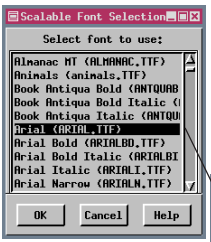


- ☑ redo any of the steps on this page you want to further adjust color and style

All the features available for styling text in TNTmips are available for table names in the Database Editor. The font initially selected provides good readability at relatively small size. The table rectangles are resized if necessary to accommodate changes in text width and height.

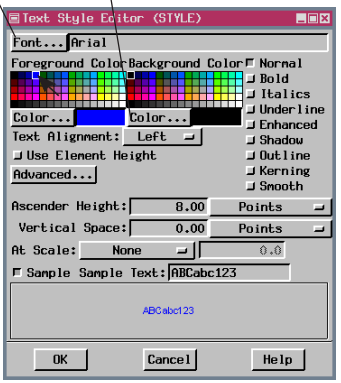
The text style settings chosen for the sample shown toward the bottom of this page are 10-point Arial in blue. A change in size and font may increase the table rectangle sizes so the database no longer fits entirely within the Database Editor window (shown on page 5). You can resize the window, scroll to view all the tables, or move the tables so they once again all fit within the window if you find a larger font is to your liking. You get a preview of your text style adjustments if the Sample toggle is on.

You can change the fill and outline colors for the table rectangles. The outline color is also the color in which the links are drawn, so red is not a recommended choice (selected links are highlighted in red). A "missing" color can also be set to distinguish linked tables that cannot be found from other tables.



The default font is one of TNTmips' original outline fonts (Stork), which are listed after all the TrueType fonts on your system. This font is for the Database Editor interface only—it does not affect how the table itself is displayed.

This background color is used in addition to the foreground color with Enhanced and Shadow styles.



Mouse Functions in Move Table Mode

You can use the mouse three different ways to get at features in the Database Editor: click left, click right, and double-click (left). On a Macintosh with a single-button mouse, the right mouse button is simulated by holding the Command key (Open Apple) and clicking. The function of the left mouse button is determined by the icon button that is toggled on in the toolbar. The three available left mouse button functions are: move table, relate, and unrelate. The right mouse button opens a context sensitive menu.

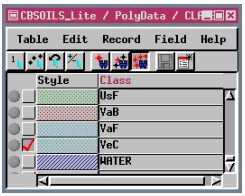
Double-clicking on any table name in the Database Editor opens the table. You can select elements from a table opened in this manner if you opened the Database Editor from Spatial Data Display, as you did on page 5.

The right mouse button menus are available in all three left mouse button modes (Move Table, Relate, and Unrelate). You can make your selection from a right mouse button menu using either the right or the left mouse button.

Moving tables with the left mouse button is, for the most part, a cosmetic adjustment for display purposes. It may also serve to make the endpoints of the links more apparent in a particularly complex database structure.

STEPS

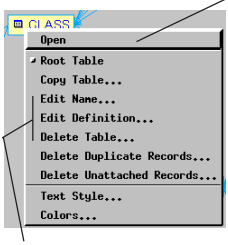
- ☑ with CBSOILS_LITE still displayed in a 2D group, double-click on the CLASS rectangle in the Database Editor window
- ☑ scroll to the bottom of the tabular view that opens and click in the Select Record box for the VeC class



- ☑ click on the Exclusive icon on the table's toolbar and view all the polygons with soil type VeC

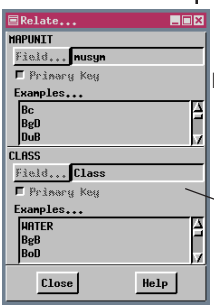
- ☑ click and hold the left mouse button on the CLASS table box (Database Editor window), then drag the mouse to reposition the box
- ☑ click the right mouse button on the CLASS table box and note the menu selections
- ☑ right-click on the link between the CLASS and MAPUNIT tables and choose Show Link
- ☑ note information about the fields that relate the two tables is presented, then click on [Close] in the Relate window

Table right-click menu



Opens table, just like double-clicking on box.

These three functions are also available when you right-click on a table in the Layout Controls or Group Controls window list.



Link right-click menu

Breaking Links

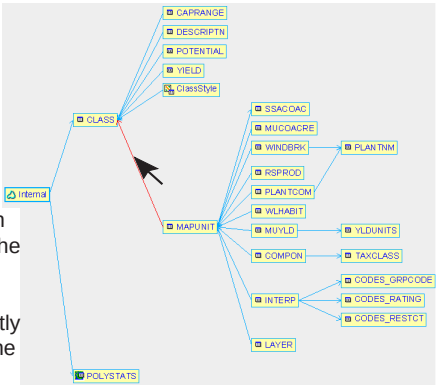
Vocabulary: The relationship between database records and elements in an object may be either **direct** or **indirect**. Direct relationships are made at the element level while indirect relationships are established as part of the table definition. Directly related records are referred to as **attached records**.

STEPS

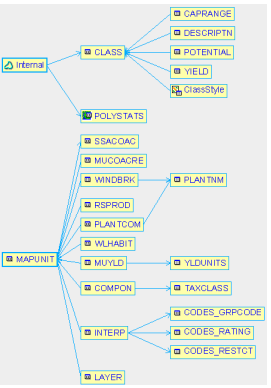
- ☑ click on the Unrelate icon in the toolbar 
- ☑ click on the link between the CLASS table and the MAPUNIT table
- ☑ click on [Yes] in the Verify window
- ☑ if you feel the display is too disorganized after breaking the link, right-click on the MAPUNIT box and turn on the Root Table toggle

Because all the tables in the CBSOILS_LITE polygon database are either directly attached or related through key fields, we will break one of the links and then reestablish it so you can see how relating tables in the Database Editor works. You could have tables in the database that weren't either attached to the elements or related to another table if you'd copied a table from another database (Process / Database / Copy) or if you'd added a table in the display process without establishing a relationship at that time. (Adding tables in the display process is described in a later exercise.)

The table boxes will be rearranged when the link is broken. Note that the MAPUNIT table is still approximately in the center of the window horizontally. You can move the table box to the left of the window by turning on the Root Table toggle in the table's right mouse menu (result shown below, right). Beside the simple left click with the Unrelate icon selected, you can also break a link by clicking the right mouse button on the link and choosing Break Link from the pop up menu in any of the editing modes.



Tables shown with links to the Internal table have records that are directly attached to the elements.



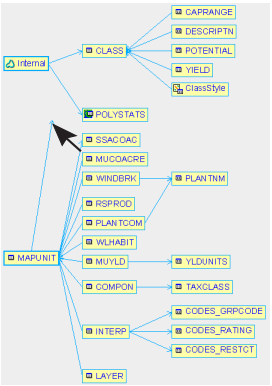
DO NOT break links that represent direct attachments to elements without first verifying that a key field relationship can be established. Links without key field relationships cannot be reestablished in the simple procedure described on the next page—you must select each element and attach the appropriate record to recover from such an error.

Relating Tables

There must be at least one table with direct attachments to the elements in an object for associated database information to be shown when elements are selected. The time required to associate database information with the corresponding elements is minimized when you can make use of indirect relationships to tables that have direct attachments.

You can relate any two tables, but for the relationship to be meaningful the two tables need to each have a field that contains the same kind of information expressed in the same way. For example, the CLASS table and MUCOACRE table both contain soil type information, but a meaningful relationship between the two can't be directly established because the soil types are appended to the county code in the latter table (BgB in one is 045BgB in the other). A sampling of field values is provided in the Relate window so you can tell if the two selected fields express the information similarly.

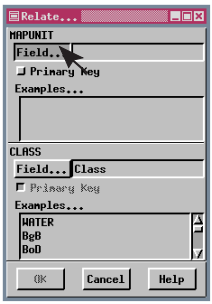
The primary key (if there is one) is provided as the best guess for the field you want to establish the relationship between tables. You just need to select the corresponding field in the table you are relating. You can of course, also change the field you are relating to if the designated primary key is not the one desired.



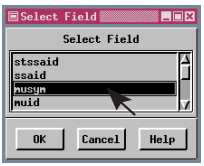
Click anywhere on the table you want to relate. As you drag the mouse a link appears.

STEPS

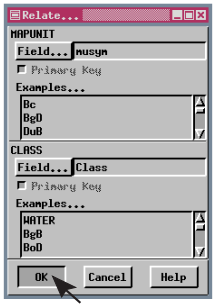
- ☑ click on the Relate icon in the toolbar 
- ☑ click and hold on the MAPUNIT table then drag the mouse until the cursor reaches the CLASS table
- ☑ release the mouse
- ☑ click on [Field] for the MAPUNIT table (upper) in the Relate window



- ☑ double click on musym in the Select Field window



- ☑ click on [OK]

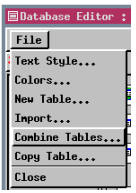


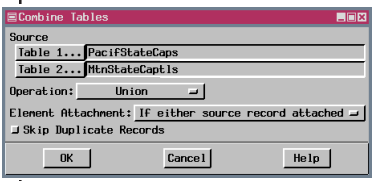
- ☑ turn off the Root Table designation for the MAPUNIT table (if you turned it on)

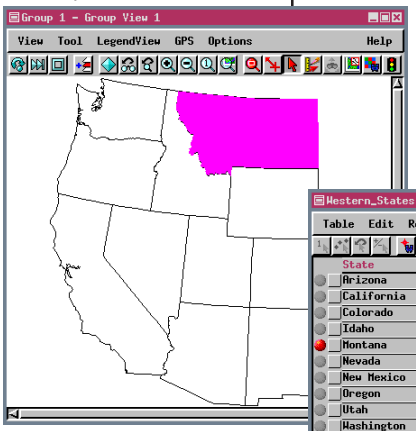
Combining Tables (Unions)

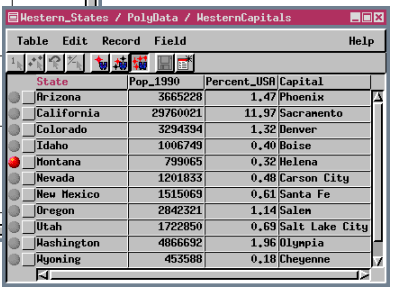
STEPS

- ☑ open a new 2D group, add the WESTERN_STATES object in the UNTDSTAT Project File, and open the Database Editor from the Polygon Make Table / Form icon (see page 5 if you don't remember how)
- ☑ choose File / Combine Tables in the Database Editor window
- ☑ click on [Table 1] and select PacifStateCaps
- ☑ click on [Table 2] and select MtnStateCaptls
- ☑ set the Operation to Union, Element Attachment to If either source record attached, and click [OK]
- ☑ name the output table WesternCapitals and enter a description if desired
- ☑ open each input table and the output table to compare contents









State	Pop_1990	Percent_USA	Capital
Arizona	3665228	1.47	Phoenix
California	29760021	11.97	Sacramento
Colorado	3294394	1.32	Denver
Idaho	1006749	0.40	Boise
Montana	799065	0.32	Helena
Nevada	1201833	0.48	Carson City
New Mexico	1515069	0.61	Santa Fe
Oregon	2842321	1.14	Salem
Utah	1722850	0.69	Salt Lake City
Washington	4866692	1.96	Olympia
Wyoming	453588	0.18	Cheyenne

Keep the Database Editor open for the next exercise.

When tables are combined, records from the two selected tables are copied to a new table, and the original tables remain. Once you determine the new table contains the expected data, you can delete the input tables if desired.

There are six methods available for combining tables: two unions (Union, Outer Union) and four joins (see next page). Outer Union is the safer Union method because it retains all records and fields from the input tables (except duplicate records if you elect to skip them). The Union method assumes that the contents of the first field in each table are equivalent, as are the contents of the second field and so on, regardless of field names. This assumption is desirable when separate tables have equivalent data in fields with different names, such as State_Name and State. If the second field is Population in one table and Average Temperature in the other, the combined results will be incorrectly labeled (and quite misleading) for some of the records. Such data is appropriately handled by join operations, provided a relationship between tables exists. The tables selected for this exercise are such that you get the same results using either union method because the records in the two tables have the same fields.

Combining Tables (Joins)

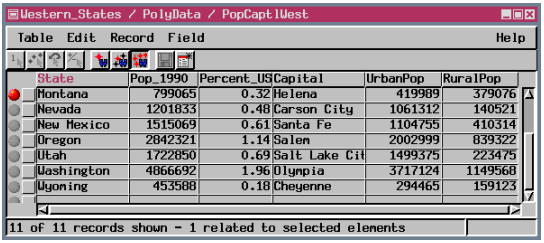
Union operations are intended to add the records from one table to the records in another table. Join operations are intended to add fields from one table to existing, related records in another table. The "outer" options take care of the less than ideal cases for both unions and joins.

The tables selected for joining in this exercise illustrate some of the differences between a union and a join. One of the tables has records only for the western states while the other has records for all 50 states and the District of Columbia. The combined table has records for only the eleven western states. Each of the input tables has four fields, but the combined table has six fields because two of the fields are held in common (the state name and its 1990 population). The combined table has the four original fields of Table 1 plus the two unique fields from Table 2.

The Element Attachment options do not affect the information in the output table. They only affect how those records are attached to the elements based on the original tables' attachments.

STEPS

- ☑ choose File / Combine Tables in the Database Editor window
- ☑ click on [Table 1] and select WESTERNCAPITALS (table created in previous exercise)
- ☑ click on [Table 2] and select US_POPULATION
- ☑ set the Operation to Natural Join, Element Attachment to If either source record attached, and click [OK]
- ☑ name the output table PopCaptlWest and enter a description if desired
- ☑ open each of the input tables and the output table to compare contents



combination method	number of records	number of fields	no values for some fields
Outer Union	62	6	yes
Union	62	4	no
Natural Join	11	6	no
Left Outer Join	11	6	no
Right Outer Join	51	6	yes
Outer Join	51	6	yes

Combination Results with WesternCapitals as Table 1 and US_Population as Table 2. All key field attribute values for Table 1 are found in Table 2. Table 1 has 11 records; Table 2 has 51 records. Each input table has four fields, two of which are held in common. This table is intended to test and extend your understanding of table combinations—*not all results are desirable.*

You might also try other combination methods to see the results. Natural join provides the most useful results for this situation, but other methods may be more appropriate with other data.

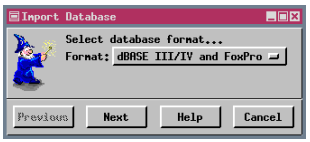
Note: The Skip Duplicate Records option doesn't eliminate duplicate records found in the original tables, it just doesn't copy records that would become duplicate records to the new table. This option applies only to unions since joining does not create duplicate records.

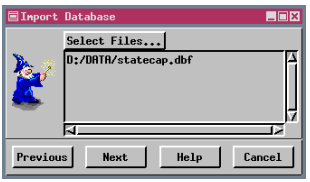
Import Table Wizard

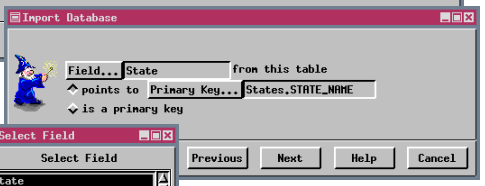
STEPS

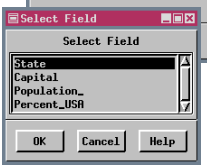
- ☑ open a new 2D Group, add the USSTATES object from the UNTD STAT Project File, and open the Database Editor for the polygon database
- ☑ choose File / Import in the Database Editor
- ☑ set the Format to dBASE III/IV and FoxPro, then click [Next]
- ☑ click on [Select Files], select STATECAP.DBF, click [Next], turn the Copy toggle on, and click[Next]
- ☑ set the String Encoding to ASCII and the Attachment Type to Related Only, then click [Next]
- ☑ click on [Field] in the first line and select State; turn on the *points to Primary Key* toggle, click on [Primary Key] in the same line and select States.State_Name, then click on [Next]
- ☑ click on [Finish]

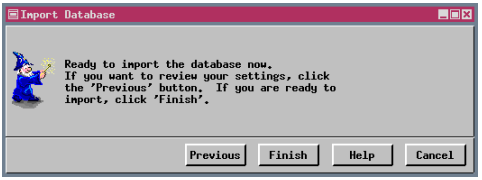
When you choose Import from the File menu in the Database Editor, a wizard takes you through the necessary steps rather than simply opening the Import / Export window. The database opened by choosing Edit Relations (or Edit / Attribute Databases) is the destination for the table(s) from the database you select for import.











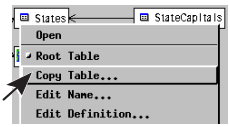
You can choose to copy the database table(s) to internal format on import or link to tables in their native format so the same file can be used in other software packages, as well as in the TNT products. Only those database formats that can be linked to are available for import using the import wizard in the Database Editor (choose Process / Import/ Export for other formats).

The unique feature of import within the Database Editor is the ability to set up primary key fields and the relationship between tables during import. You can even establish the identity of a foreign key during this import so that it is not necessary to edit the table's definition before viewing attributes for related elements.

Renaming and Copying Tables

The import wizard automatically names the table the same as the file imported. You may want to change this name and / or add a description. The description is not evident in the Database Editor but is shown when tables are listed in the Group or Layout Controls window. You can also edit a table's name and description by right-clicking on the entry for that table when tables are shown in the Group or Layout Controls window.

The Copy Table option on the right mouse button menu for a table in the Database Editor creates a duplicate of the table the option is selected from. The attachments in the copy are also as in the original. This utility is useful when you want to create another table that contains at least one of the same fields with the same field values as in the original. For example, you could create a second table that contained more information about each state without having to enter all the state names again. A similar feature is available in Process / Database / Copy, which also lets you copy a table to a different database. However, direct attachments to elements are not maintained by this latter process as they are with the Copy Table function in the Database Editor.

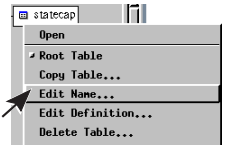


You could also use this function to make a table with many fields more readily readable in tabular view by creating two (or more) tables with fewer fields to eliminate horizontal scrolling. You would simply copy the table, then use the Edit Definition option to delete fields from each table, being sure to maintain the primary key field in both cases.

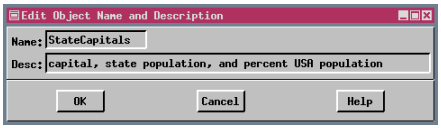
The delete table function is available from the right mouse button menu over a table entry in the Group or Layout Controls window and from the Table menu when viewing the table.

STEPS

- ☑ click the right mouse button on the box for the table just imported, select Edit Name and change the name to StateCapitals



- ☑ type in *capital, state population, and percent USA population* in the Description field, then click [OK]

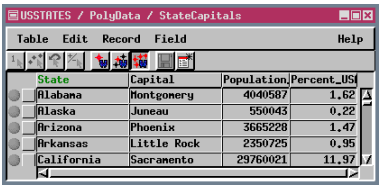


- ☑ click the right mouse button on the States box, choose Copy Table, accept the default name, and click [OK]; note the attachment of this table directly to elements as with the States table
- ☑ click the right mouse button on the copied table (States1), choose Delete Table, and click [Yes] in the Verify window

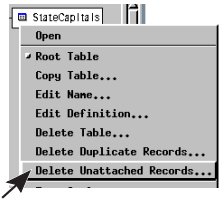
Additional Database Editor Functions

STEPS

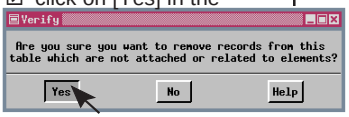
- ☑ double-click on the StateCapitals box or right-click on the box and choose Open
- ☑ click on the Show All Records icon in the table and note the Alaska record, which is not represented in the vector object (also Hawaii)



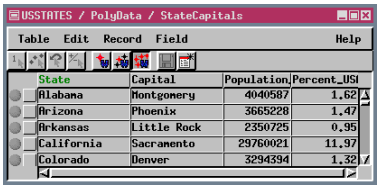
- ☑ right-click on the StateCapitals box, choose Delete Unattached Records



- ☑ click on [Yes] in the Verify window



- ☑ note the absence of the record for Alaska (and Hawaii)



There are a number of other database management features available from the File or right mouse button menus in the Database Editor. Some of these are also available in the table / selection icon row (Group or Layout Controls window) or on the Table menu (tabular and single record view), and some are available only in the Database Editor window or as part of some other process, such as Vector Extract, Vector Combine, or the Spatial Data Editor.

In addition to the Delete Unattached Records option, which is demonstrated in this exercise, you can choose to delete duplicate records. Duplicate records have identical attribute values. When you remove duplicate records, all elements previously attached to any one of the records are attached to the remaining record. If the attachment type for the table is One Element per Record, the Delete Duplicate Records option is unavailable because the table's attachment type prevents the elements attached to the duplicate records from being attached to the remaining record. You will need to change the attachment type before duplicate records can be deleted. You can also create new tables while using the Database Editor (File menu). This last feature is available in the Group Controls.

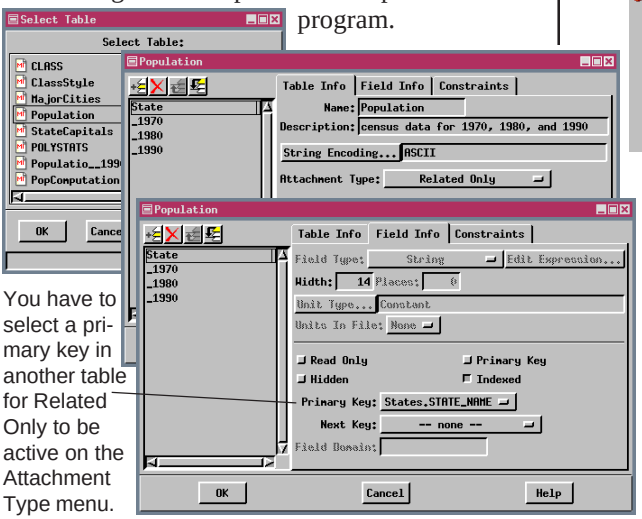
Recall there are two ways to get to the Database Editor: Edit / Attribute Databases and from the table / selection icon row of a Group or Layout Controls window in any process that displays vector objects. The features are the same regardless of which way you open the Database Editor, but only two icons are present on the icon bar in the standalone Database Editor process because the other icons are related to element selection (and there's no element display).

Adding Tables

There are a variety of reasons to copy a table from one database to another. In this example, an old vector object (USA state outlines created by raster-to-vector conversion of a low resolution, poorly georeferenced scanned map) has a number of associated tables with population and cultural information. A higher resolution outline map with good georeferencing has since been acquired, but the only originally associated polygon attributes are the state codes and names (standard attributes tables for the lines and polygons have also been calculated).

Adding tables is initiated from the Make Table / Form icon in the Group or Layout Controls window. Bringing in tables with computed fields (as we do in the last step of this exercise) is generally not recommended unless you are sure that all referenced tables and fields also exist in the recipient database. The order in which tables are added in this exercise ensures that the necessary tables and fields already are found in this database.

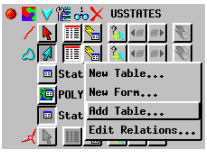
When you add a table from a linked database, a copy in TNTmips' internal format is created. The copy can no longer be manipulated in the parent database program.



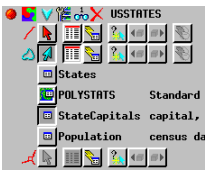
You have to select a primary key in another table for Related Only to be active on the Attachment Type menu.

STEPS

- ☑ with USSTATES from the UNTDSTAT Project File still open from the previous exercise, click on the Make Table / Form icon for polygons and choose Add Table from the menu



- ☑ navigate to and double-click on USA in the USA_OLD Project File, then select PolyData
- ☑ select Population in the Select Table window
- ☑ choose States.STATE_NAME from the Primary Key option menu (Field Info panel)
- ☑ set the Attachment Type option to Related Only (Table Info panel)



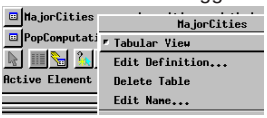
- ☑ repeat steps 1–5 substituting Major Cities in step 3
- ☑ repeat steps 1–5 substituting PopComputation in step 3

Not all tables are brought forward from this old vector object because they may be incomplete or object specific.

Database Viewing

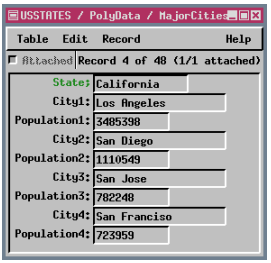
STEPS

☑ right-click on the Major Cities table and turn off the Tabular View toggle



☑ click on the Select icon in the Group View window's toolbar

☑ click on the View Table icon for the MajorCities table



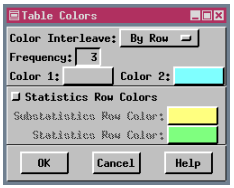
☑ click on California in the Group View

☑ review the information presented, then choose Switch to Tabular View from the table's File menu

☑ click on the View All Records icon

☑ choose Colors from the Table menu

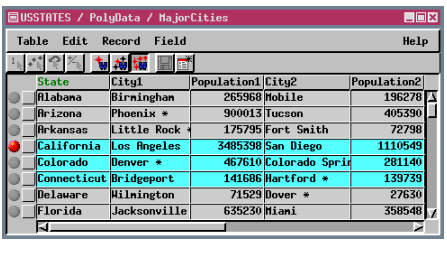
☑ set the Color Interleave to By Row and the Frequency to 3, then click [OK]



TNTmips provides two methods of database viewing: tabular and single record views. Tabular view presents all the records, or only records attached to selected elements, as rows in a table in which the columns are the fields. Single record view shows the fields for a single record. You can switch between view types from the Table menu in either view or from the tables' right mouse button menu in the Group or Layout Controls window.

Single record view is designed for viewing all information about one element at a time and for data entry. Single record view works best for data entry because a new record is created automatically whenever you click on an element that lacks a related record. It also avoids the problem of accidentally entering information that actually belongs in the row above or below that is often encountered when entering information in a table.

Tabular view has a number of features not found in single record view because they are related to viewing multiple records. You have the ability to add statistics and substatistics rows to the table. You can assign separate colors to records and statistics rows and interleave two colors for easier record identification. The ability to save selected records or statistics records as a new table is also available in tabular view. You can view all records in a tabular view or just those records associated with the active element or with all selected elements. You can also select elements by their attributes from a tabular view (as described on page 7).



Note that you need to scroll horizontally to see all fields in this table.

Computed Fields

Computed fields let you do computations with the information in a table and also bring in information from other tables. You added a table with computed fields in the exercise on page 15. We'll examine the expressions for those fields here. You create a new table with computed fields in the next exercise.

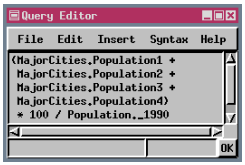
There are two types of computed fields: those that incorporate string fields, which are referred to more specifically as *string expression fields*, and those that use numeric fields, which are referred to by the more general term *computed fields*. The PopComputation table also contains one field with directly entered data, the State field. The PopComputation table was created before the Implied One-To-One option existed. This option eliminates the need for any direct data entry as demonstrated in the next exercise. Names for computed fields are shown in blue, which makes it easy to distinguish them from fields for direct data entry.

String expression fields may simply supply the information from a string field in another table or can be a concatenation of other string fields, such as LastName, FirstName from another table that contains first and last names in separate fields. Any operations available in database query can be used on numeric fields to generate the desired results.

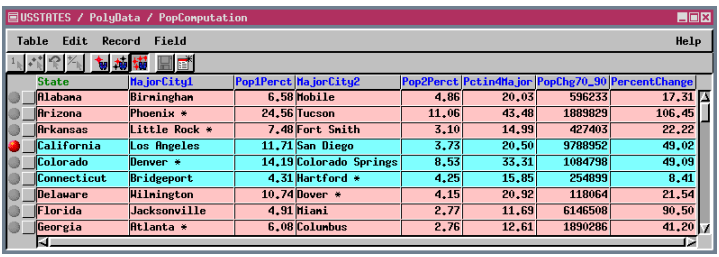
You can also edit the expressions for individual computed fields by right-clicking on the field in tabular view, selecting Field Options from the popup menu, and clicking on [Edit Expression].

STEPS

- ☑ click on the View Table icon for the PopComputation table 
- ☑ choose Table / Edit Definition, click on MajorCity1 in the field list, note that the Field Type is String Expression, and click on [Edit Expression]
- ☑ note that the expression simply pulls the City1 field information for the related polygon from the MajorCities table, then click [OK] in the Query Editor window
- ☑ click on Pctin4Major in the field list, note that the Field Type is Computed, and click on [Edit Expression]
- ☑ note the arithmetic combination of fields from two different tables to derive the percentage of a state's population living in its 4 largest cities



- ☑ click [OK] in the Query Editor and [Cancel] in the Table Definition window



State	MajorCity1	Pop1Perct	MajorCity2	Pop2Perct	Pctin4Major	PopChg70_90	PercentChange
Alabama	Birmingham	6.58	Mobile	4.86	20.03	596233	17.31
Arizona	Phoenix *	24.56	Tucson	11.06	43.48	1889829	106.45
Arkansas	Little Rock *	7.48	Fort Smith	3.10	14.99	427403	22.22
California	Los Angeles	11.71	San Diego	3.73	20.50	9788952	49.02
Colorado	Denver *	14.19	Colorado Springs	8.53	33.31	1084798	49.09
Connecticut	Bridgeport	4.31	Hartford *	4.25	15.85	254899	8.41
Delaware	Wilmington	10.74	Dover *	4.15	20.92	118064	21.54
Florida	Jacksonville	4.91	Miami	2.77	11.69	6146508	90.50
Georgia	Atlanta *	6.08	Columbus	2.76	12.61	1890286	41.20

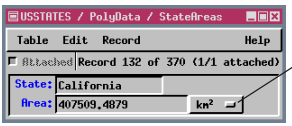
Creating New Tables

STEPS

- ☑ click on the Make Table / Form icon in the polygon table / selection icon row and choose New Table from the drop down menu
- ☑ enter StateAreas for the table name, and click [OK]
- ☑ choose Implied One-To-One from the Attachment Type option menu (Table Info panel)

- ☑ click on the Add Field icon in the Edit Definition window, change the field name to State, set the Field Type to String Expression and Width to 16 (Field Info panel)
- ☑ click on [Edit Expression], choose Insert / Field, click on States in the Table column and STATE_NAME in the Field column, then on [Insert]
- ☑ click [OK] in the Query Editor window
- ☑ click on the Add Field icon, change the field name to Area, Field Type to Computed, Width to 20, Places to 4, and Unit Type to Area with Units in File as square meters
- ☑ repeat steps 4 and 5 but choose the POLYSTATS table and the Area field, and click [OK]

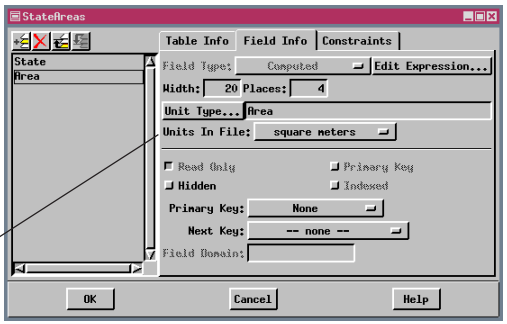
You can change the units for viewing if the units for a field are set in the table definition.



Choosing Make New Table creates a new, blank table. After you add and define the fields, there are two tasks to complete before the table can be useful in attribute retrieval for selected elements: data entry and establishment of a relationship between elements and records. Your job is greatly simplified if another table in the database contains a field that can be identified as a foreign key; you just set the Attachment type menu option to Related Only and the second task is completed. You still, however, have to enter the attribute values for the foreign key field in the new table and for any other fields that are not computed fields. You can get around data entry altogether if the Implied One-To-One option is appropriate for the record attachment type and the table is comprised entirely of computed fields as in this example.

When you add a field to a table, it gets a default name, which you change by selecting the name and editing it directly in the field list. The field list and associated icons are visible in the Table Definition window regardless of the active panel. Most of the work for this exercise is done from the Field Info panel, but the Attachment Type is set on the Table Info panel.

A units option menu is provided in single record view next to each field that has units set. The units used in tabular view can be changed by selecting the field then choosing Options from the Field menu.



Statistics and Substatistics

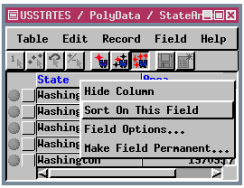
The USSTATES vector object does not have a one-to-one correspondence between states and polygons; 48 states and the District of Columbia are represented by 370 polygons. About half the states are a single polygon, but states with intricate coastlines may be made of many polygons. (Florida has the most with 57 polygons.) The purpose of generating a table with the state name and area drawn from other tables is to get areas for states as a whole, then use these areas to compute population densities.

TNTmips offers six mathematical functions that can be applied to all records in a table (statistics) or to all records with the same attribute value in the field used for sorting (substatistics). These six functions are count, sum, mean, standard deviation, minimum, and maximum. Including a sum sub-statistic row in the table you made in the last exercise will provide total areas for each state.

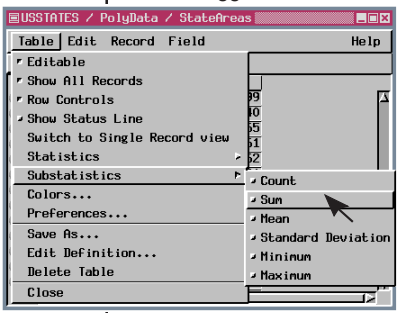
A symbol identifying the type of statistic is placed at the left of the table in the position of the selection indicator for data records. The Select Record box is active for substatistics rows and selects all records (and attached elements) included in calculation of the selected substatistic value. Turning on color for the substatistics rows helps to distinguish these rows from the data. Statistics and substatistics are turned off by returning to the menu and turning off the corresponding toggle.

STEPS

þ choose Table / Switch to Tabular View if not in tabular view for the State Areas table already, click on the Show All Records icon, then right-click on the State column heading, and choose Sort On This Field

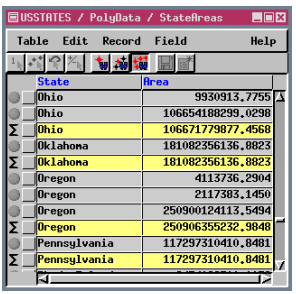


☑ choose Table /Substatistics and turn on the Sum toggle button

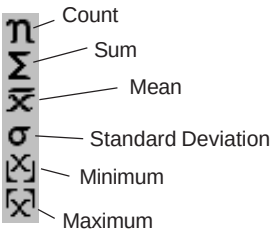


☑ choose Table / Colors and turn on the Statistics Row Colors toggle button

☑ click [OK] in the Table Colors window



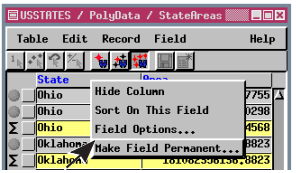
Substatistics occur each time the attribute value in the sorted field changes. You could also add a count substatistic and easily see how many polygons make up each state.



New Tables from Summary Statistics

STEPS

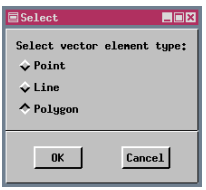
- ☑ right-click on the State field heading in Tabular View and choose Make Field Permanent, then click on [Yes] in the Verify window
- ☑ repeat step 1 except right-click on the Area field heading
- ☑ choose Table / Edit Definition and select States.STATE_NAME on the Primary Key option menu (Field Info panel)
- ☑ set the Attachment Type to Related Only (Table Info panel), and click [OK]
- ☑ choose Table / Save As
- ☑ check that the Format and Records option menus are set to Internal and All Records, respectively
- ☑ set the Values option menu to Sum, the Units to As Stored, and click [OK]
- ☑ navigate to the USSTATES object you have selected for display and select it for the destination object, turn on the Polygon toggle in the Select window, accept the default name (StateAreas1), and click [OK]
- ☑ click on the View Table icon for StateAreas1



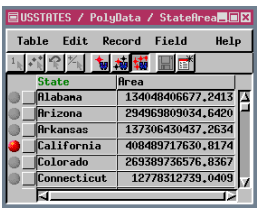
The Save As feature for database tables is useful in a variety of contexts. You can choose to save the data or the statistics for all records, selected records, or records attached to selected elements*. We will use this feature to create a table that contains state names and area totals. Each record in such a table is attached to all the elements that contributed to the substatistic calculated. The table created can be in internal format, in dBASE III format, in a format you've identified by ODBC, or as comma separated values in a text file if you are running TNTmips or TNTedit. Internal database format is the only option if you are running TNTview or TNTlite.

Because the table you are going to save substatistics from has an Implied One-To-One Attachment Type, which will no longer be the case after the Save As, you need to do a little database manipulation in preparation for the Save As. The two computed fields need to be made permanent, which poses no problem for future flexibility since neither the state name or area are likely to change. The Attachment Type also needs to be changed.

You specify the format, which records, and what values to save.



When the table is saved to an existing vector object, you need to specify which database should include the table.



The resulting table has one record per state, which is related to all the state's polygons and provides the state's total area instead of just the area for the selected polygon.

* You can also create a CAD snapshot, which is discussed in the *Managing Geoattributes* and *Making Map Layouts* booklets.

Modifying An Existing Table

Now that we have the total area for each state in data form, we can calculate population density. (Statistics and substatistics fields cannot be referenced in a computed field expression because their values change depending on the records currently viewed.) The query shown will give population per square kilometer. Add an additional term to the query that divides by 0.3861 if you want the population density expressed as people per square mile (also change Units in File).

Since the computed field expresses population density in square kilometers, it makes sense to also display the area in square kilometers. Recall that in the exercise on page 18 in which the initial StateAreas table was created, we specified that the area field was expressed in square meters (as it is in all standard attributes tables calculated in TNTmips), which enables you to change units for display. In single record view, the units option button is next to the field. You set the units in tabular view by first selecting the field (click on the field name), then choosing Options from the Field menu (or right click and select Field Options). You can also set whether or not to calculate statistics for the field if statistics are included in the table. If the field selected is a computed field, there is also an Edit Expression button that opens the Query Editor window with the current expression shown.

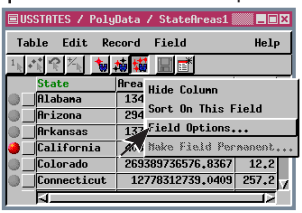
```
Population._1990 /
StateAreas1.Area *
1000000
```

The District of Columbia does not have an entry in the PopDensity field because the Population table does not have an entry for the District of Columbia.

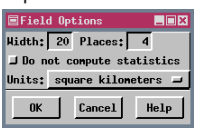
Note that areas are shown in square kilometers on this page and in square meters in the table on the preceding page, but are stored as square meters for both.

STEPS

- ☑ choose Table / Edit Definition in the StateAreas1 tabular view
- ☑ highlight Area in the fields list then click on [Add Field]
- ☑ change the new field name to PopDensity
- ☑ choose Computed from the Field Type option menu
- ☑ set the Width to 6 and Places to 1
- ☑ set the Unit Type to Constant
- ☑ click on [Edit Expression] and enter the expression as shown at the left (you can use Insert / Field to get the names correct without typing if desired)
- ☑ click [OK] in the Query Editor window and [OK] in the Table Definition window
- ☑ right-click on the Area field name in tabular view then select Field Options



- ☑ set the units menu to square kilometers



- ☑ click [OK] in the Field Options window

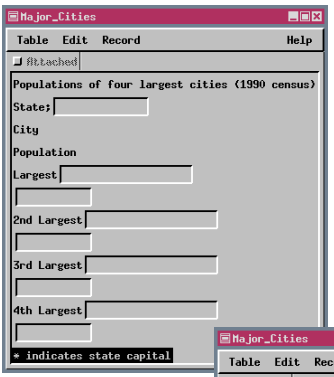
Creating Forms

STEPS

- ☑ click on the Make Table / Form icon for polygons and choose New Form (still USSTATES)
- ☑ enter *Major_Cities* as the name and *Populations of four largest cities (1990 census)* as the description for the new object
- ☑ choose Edit Form from the Table menu in the Major_Cities form
- ☑ click on [Table] and choose MajorCities as the table the form is based on
- ☑ click on [Add Field] and choose State
- ☑ click on [Add Label], type *City* in the Label field, then click on [Add Label], and change it to *Population*
- ☑ click on [Add Field], choose City1, and change its label to *Largest*
- ☑ click on [Add Field], choose Population1, and delete its label text
- ☑ click on [Add Field], choose City2, and change its label to *2nd Largest*
- ☑ click on [Add Field], choose Population2, and delete its label text
- ☑ repeat the last two steps substituting the fields for the 3rd (City3, Population3) and 4th largest cities, changing the city labels appropriately
- ☑ click on [Add Label] and type in ** indicates state capital*

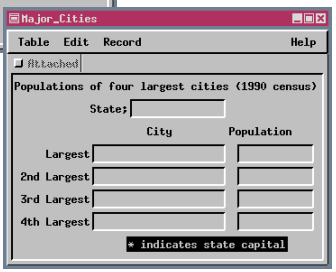
A form is simply another way of presenting the information in a database table. You can include all the fields or leave some out, insert labels not associated with any particular field, change field names to include spaces or special characters, and drag fields into new positions. A form is stored as a subobject of the database it is associated with.

When you create a new form, the name entered becomes the title of the form window and the description becomes the first label in the form. A label can be a field name or a heading or other text not associated with an individual field. The default label for a field is the field name including the punctuation (colon or semicolon) that appears in single record view. You can remove the punctuation and further change field names by editing the text in the Label text box.



Items are placed at the left margin of the form as they are added. You can click on any item in the form window and drag it to a new position while the Form Editor window is also open.

- ☑ click on each item in the form window and drag to the desired position
- ☑ click [OK] in the Form Editor window



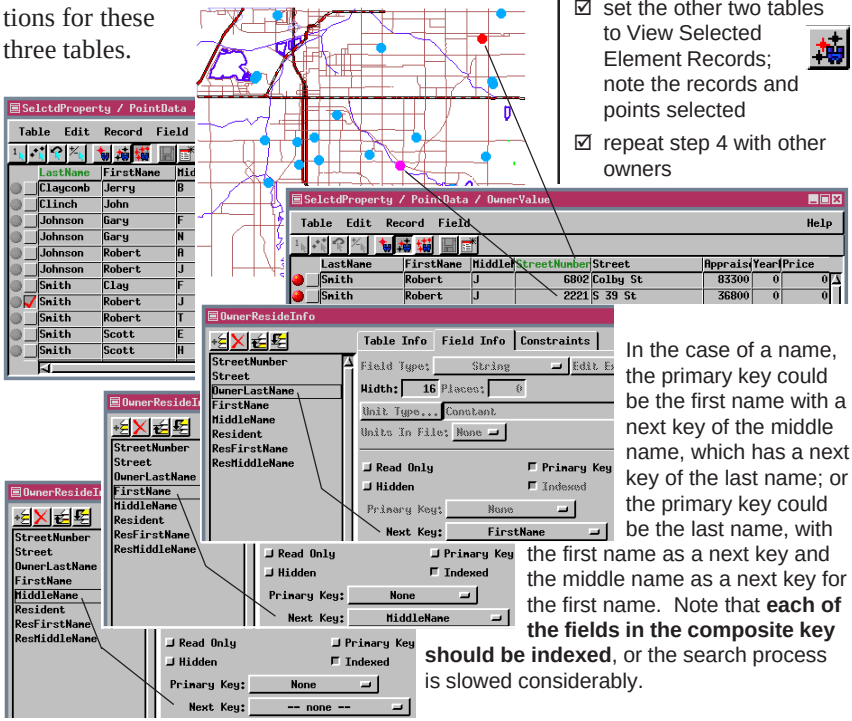
Composite Fields

Composite key fields allow unique identification of records that may not have any individual unique fields but have unique combinations of fields. In the property information database used in this exercise, property owner, resident, and street address are represented by composite fields. So while Smith is the most popular last name in this sample database, you can readily select only the properties owned by Robert J. Smith and not those owned by Robert T. Smith or Robert J. Johnson.

The tables in the point database use other important relational database features, such as multiple primary keys so related elements can be selected by owner name or by address and sorting on multiple fields so that addresses can appear in numerical order after sorting alphabetically on the street name. You should spend some time looking at the definitions for these three tables.

STEPS

- ☑ open a new 2D group; quick-add the two vectors in the PROPINFO Project File
- ☑ click on the Show Details icon for the SELCTD PROPERTY object, then on the Show Tables icon for points
- ☑ click on the View Table icons for the OWNERRESIDEINFO, OWNER VALUE, and OWNER tables
- ☑ click on the Select Record box for Robert J. Smith in the Owner table (View All Records), then click on Exclusive
- ☑ set the other two tables to View Selected Element Records; note the records and points selected
- ☑ repeat step 4 with other owners



In the case of a name, the primary key could be the first name with a next key of the middle name, which has a next key of the last name; or the primary key could be the last name, with the first name as a next key and the middle name as a next key for the first name. Note that **each of the fields in the composite key should be indexed**, or the search process is slowed considerably.

Field Constraints

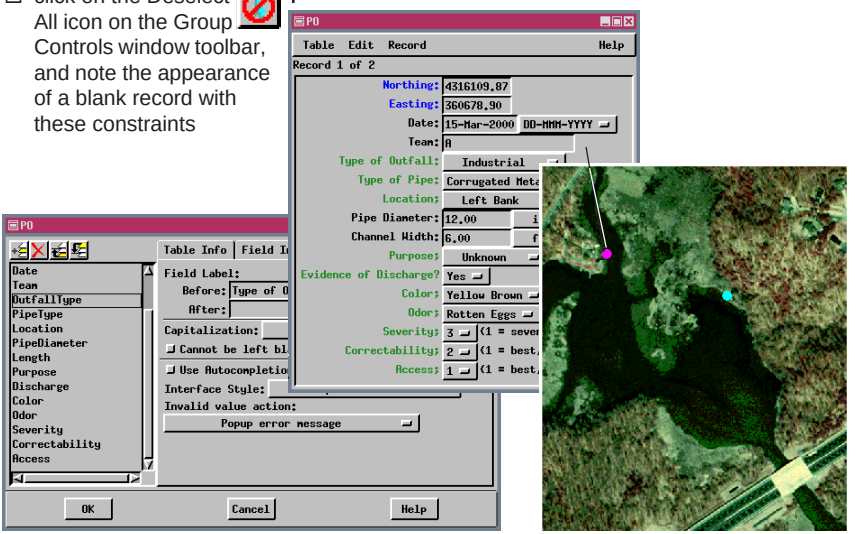
STEPS

- ☑ click the Open icon on the Display Spatial Data toolbar, choose Open Group, and select DATALOGGER from the CONSTRNT Project File
- ☑ click on the Select icon in the View window then click on one of the two points in the pinmap overlay
- ☑ choose Edit Definition from the Table menu in the PO table window
- ☑ select the OutfallType field, then click on the Constraints tab and note the options
- ☑ click on the Field Info tab and note that this field points to a primary key in another table
- ☑ check the constraints for other fields
- ☑ click on [Cancel]
- ☑ click on the Deselect All icon on the Group Controls window toolbar, and note the appearance of a blank record with these constraints

You may anticipate untrained personnel, such as temporary help, entering values into a database you have designed. You may also need to enter values without a keyboard under field conditions. Placing constraints on the values allowable for database fields lets you guide anyone entering data values to the choices acceptable to the database designer, which increases the accuracy and reproducibility of observations.

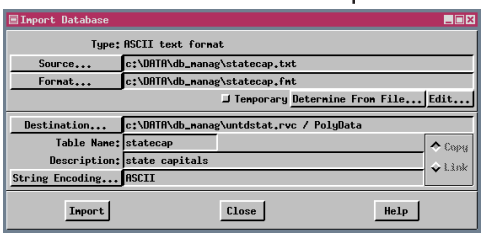
If all field values are constrained to be choices from a list, the need for a keyboard to fill in observations is eliminated. Other tables that contain all of the list choices must be in the database so you can pick from a list. There are eight tables that supply the list options for the PO (pipe outfall) table. The values you want to choose from need to be in a primary key field so it can be selected as a foreign key for the field that is constrained.

Constraints are a part of the table definition. However, just as tabular view has unique features not available in single record view, constraints apply only to data entry in single record view.



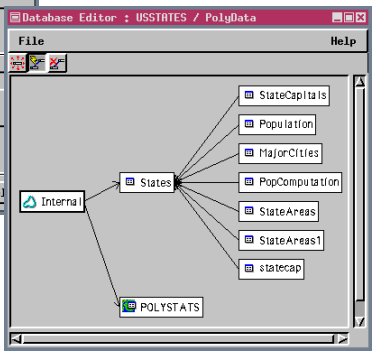
Importing or Linking to Other Formats

We are going to import a text file that lists states and their capitals, separated by commas. When you import from text you need to specify the format of the text file in relation to the information it contains. You can either save this information as a format file that can be loaded and used again or specify one for temporary use. One has been created for your use in this exercise (you can see how it is set up by clicking on the Edit button to the right of the temporary toggle, but leave the toggle off). Some database formats can also be linked to (Link button will be active), which means they are maintained in their native format and can be used by TNT products and the program that created them.



Imported tables initially appear at the left edge of the Database Editor window with no attachments indicated. Once you set up the foreign key relationship, the table will be aligned with the others that use STATES.STATE_NAME as a foreign key.

You need to select the fields in both tables because no prior relationships have been established.



Your database tree should look like this when you've finished the exercise.

STEPS

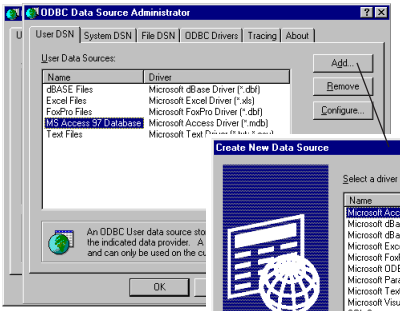
- ☑ choose Process / Import/ Export in TNTmips or click on the Import icon in TNTview 
- ☑ set the Object Type option menu to Database, then double-click on ASCII
- ☑ click on [Source], select STATECAP.TXT from the DB_MANAG data collection*
- ☑ click on [Format] and select STATECAP.FMT
- ☑ click on [Destination], choose the USSTATES object, specify polygons as the element type, then click on [Import]
- ☑ choose Edit / Attribute Databases and select USSTATES and polygons (follow the path on page 5 if using TNTview)
- ☑ click on the Relate icon, place the cursor on the STATECAP box, drag to the States box, and release 
- ☑ click on [Field] for STATECAP and select State
- ☑ click on [Field] for States and select STATE_NAME, then click [OK]

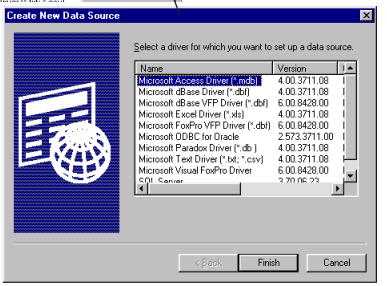
* This text file is an ASCII version of the .dbf file you imported in the exercise on page 12. This duplication allows direct comparison of methods and results.

Setting Up ODBC on Your System

STEPS

☑ select Settings / Control Panel from your Windows 95, 98, or NT Start menu and double-click on the 32bit ODBC icon 

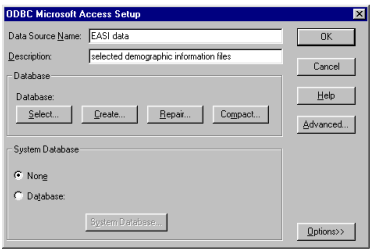




[for Windows 2000 the Data Sources (ODBC) icon is in the Administrative Tools folder of the Control Panel]

☑ click on [Add] in the ODBC Data Source Administrator window

☑ double-click on the Microsoft Access driver or other driver of your choice in the Create New Data Source window



☑ type in the name and description for your data source, and click [OK] and [OK] again

Open Database Connectivity (ODBC) is a cross platform interface specification that gives application software, such as TNTmips, direct access to database tables in many different formats. Vendors of host DBMS software (such as dBASE, Microsoft Access, and Oracle) provide ODBC drivers with their software. You need to define one or more ODBC Data Sources at the Operating System level before you can use ODBC data sources in the TNT products.

Windows 95/98 and Windows NT/2000 give you ODBC configuration control through an ODBC control panel. Your Control Panel folder contains an ODBC entry if ODBC drivers have been installed on your computer. Go back to the setup program for your DBMS software and look for the ODBC installation option if you don't have this control panel.

There is little point in setting up an ODBC data source for a dBASE IV or FoxPro database since TNTmips can link directly to these formats. The procedure is essentially the same for all ODBC drivers. It is likely your DBMS software has a sample data file you can use to run through this exercise, such as the Northwind Sample Database that comes with Microsoft Access, if you haven't already created one of your own.

You can use the Select button in the Setup window to have your choice point you to an individual file, however you will have to define a source for every file if you do.

Using ODBC in TNT Products

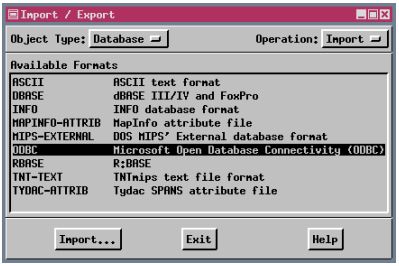
After you complete the preceding exercise, you can import or link to the data source you created. If your setup did not include selection of a specific file, you will be prompted to select one after you select your data source. You can also add a data source directly in the Import process by choosing to Import an ODBC database, clicking on the Source button in the Import Database window and next clicking on the ODBC Control Panel button. You then add a data source and follow the procedure just as though starting at step 2 on the previous page.

You can copy or link to one or more tables from any ODBC data source. Linking lets you continue to manage the data in your external DBMS so that changes can be made in the external master file and viewed in the TNT products without having to import the data again or make duplicate changes in the database you copied.

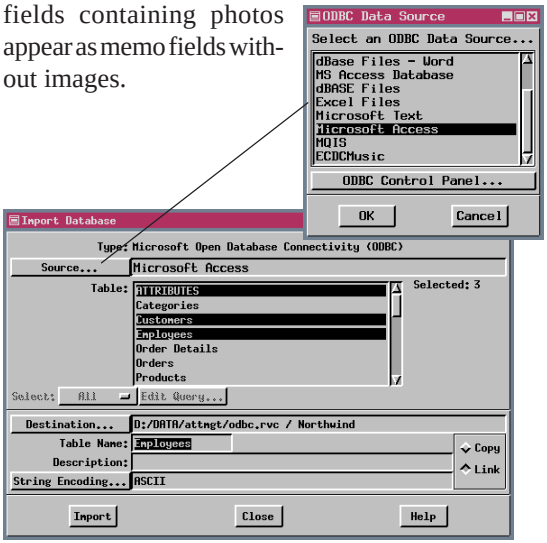
Some external database programs support field types not supported in TNTmips. The data in such fields will not be available in TNTmips. For example, fields containing photos appear as memo fields without images.

STEPS

- ☑ click on the Import icon (TNTview or TNTedit toolbar) or choose Process / Import/ Export from the main menu
- ☑ set the Object Type option menu to Database, then double-click on ODBC in the available formats list



- ☑ click on [Source] in the Import Database window and choose the ODBC Data Source you defined in the previous exercise (or some other data source of interest), then select the desired file
- ☑ click on [Destination] and create a new object or select an existing vector object then specify whether the tables will go in the point, line, or polygon database
- ☑ choose one or more tables by clicking on each in the table list
- ☑ turn on the Link radio button at the lower right of the Import Database window
- ☑ click on [Import]



Advanced Software for Geospatial Analysis

MicroImages, Inc. publishes a complete line of professional software for advanced geospatial data visualization, analysis, and publishing. Contact us or visit our web site for detailed product information.

TNTmips TNTmips is a professional system for fully integrated GIS, image analysis, CAD, TIN, desktop cartography, and geospatial database management.

TNTedit TNTedit provides interactive tools to create, georeference, and edit vector, image, CAD, TIN, and relational database project materials in a wide variety of formats.

TNTview TNTview has the same powerful display features as TNTmips and is perfect for those who do not need the technical processing and preparation features of TNTmips.

TNTatlas TNTatlas lets you publish and distribute your spatial project materials on CD-ROM at low cost. TNTatlas CDs can be used on any popular computing platform.

TNTserver TNTserver lets you publish TNTatlases on the Internet or on your intranet. Navigate through geodata atlases with your web browser and the TNTclient Java applet.

TNTlite TNTlite is a free version of TNTmips for students and professionals with small projects. You can download TNTlite from MicroImagesí web site, or you can order TNTlite on CD-ROM.

Index

MicroImages, Inc.

11th Floor – Sharp Tower
206 South 13th Street
Lincoln, Nebraska 68508-2010 USA

Voice: (402)477-9554
FAX: (402)477-9559

email: info@microimages.com
Internet: www.microimages.com