

Tutoriales



CONSULTAS



con
TNTmips®
TNTedit™
TNTview®

Antes de Tutorial

Este folleto introduce en el uso de consultas a bases de datos en los productos TNT, y muestra como construir consultas para utilizar la información de atributos enlazada a los objetos Vector, CAD, y TIN. El proceso de consultas puede parecer complejo al inicio, pero la serie de ejercicios paso a paso le llevan a través de la estructura y sintaxis requerida para las consultas, progresando desde ejemplos de una sola línea a consultas con múltiples declaraciones de condición y bucles de procesamiento.

Requisitos Previos Este folleto asume que usted a completado los ejercicios en el *Tutorial: Desplegando Datos Geoespaciales* y *Tutorial: Navegando*. Esos ejercicios le proporcionarán los conceptos y habilidades que no son cubiertos nuevamente en este folleto. Por favor consulte esos folletos y el manual de referencia de TNTmips para cualquier revisión que necesite.

Datos de Ejemplo Los ejercicios presentados en este folleto utilizan datos de ejemplo distribuidos con los productos TNT. Si no tiene acceso al CD de productos TNT, usted puede bajar los datos desde el sitio web de MicroImages. En particular este folleto usa los archivos de ejemplo en la colección de datos QUERY. Asegúrese de disponer de una copia de los datos de ejemplo en su disco duro, de forma que los cambios puedan ser almacenados cuando utilice los objetos en estos archivos.

Más Documentación Este folleto solo intenta ser una introducción al uso de consultas de la base de datos. Consulte la sección titulada “Display Vector, CAD, and TIN by Query” en el volumen Display del Manual de referencia de TNTmips para mayor información sobre el Query Editor, y el Apéndice 2: Database Queries, para una completa referencia en los comandos de consulta y funciones.

TNTmips y TNTlite® TNTmips viene en dos versiones: la versión profesional y la versión libre TNTlite. Este folleto se refiere a las dos versiones como “TNTmips.” Si usted no compra la versión profesional (la cual requiere de una llave de licencia de software), TNTmips opera en modo TNTlite, el cual limita el tamaño de sus materiales de proyecto y activa el compartir de datos únicamente con otras copias de TNTlite.

Las consultas a bases de datos pueden también ser usadas para controlar el despliegue de objetos geoespaciales en TNTview, y para seleccionar elementos para edición en TNTedit. Todos los ejercicios en este folleto pueden completarse en TNTlite usando los ejemplos de geodatos proporcionados.

Randall B. Smith, Ph.D., 27 July 2000

Sin una copia a color de este folleto podría ser difícil identificar algunos puntos importantes en algunas ilustraciones. Usted puede imprimir o leer este folleto a color en el sitio Web de MicroImages. Este sitio Web es también su fuente de nuevos Tutoriales sobre otros temas. Usted puede descargar una guía de instalación, datos de ejemplo y la última versión de TNTlite.

<http://www.microimages.com>

Bienvenido a Construyendo y Usando Consultas

TNTmips le proporcionan una gran flexibilidad para usar atributos de bases de datos de objetos Vector, CAD, y TIN para controlar el despliegue e impresión de los objetos, o para seleccionar elementos para su uso en varios procesos. Las Consultas a la Base de Datos proporcionan el medio más completo y versátil para utilizar esta información de atributos.

Una **Consulta a la Base de Datos** (database query) es un grupo de instrucciones que definen el criterio de atributos que es usado para seleccionar registros de una base de datos. Los elementos espaciales específicos (tales como líneas o polígonos) a los cuales dichos registros están enlazados, son automáticamente seleccionados por el actual proceso. Una consulta se aplica a un tipo de elemento específico, y puede usar simultáneamente consultas separadas para diferentes tipos de elementos en un objeto. En el Despliegue de Datos Espaciales, la opción de Style By Script, le permite especificar los parámetros de despliegue para los elementos en base de sus atributos. La información de atributos que se refiere en las consultas puede ser cualitativa (tal como un nombre de clase), o cuantitativa (tal como valores de producción de un cultivo).

Las consultas deben usar una “gramática y sintaxis” estándar que entienda TNTmips. El lenguaje de consulta usado es un subgrupo del Lenguaje de Manipulación Espacial (SML) de TNTmips. Usted compone consultas en la ventana Query Editor, la cual dispone de menús que simplifican la construcción de consultas válidas, permitiendo escoger campos desde las tablas disponibles en las bases de datos, e insertar símbolos y funciones desde las ventanas de listas que muestran la sintaxis correcta. El editor de consultas también proporciona un verificador de sintaxis para ayudarle a encontrar errores antes de aplicar la consulta.

Los ejercicios en este folleto usan consultas para seleccionar o asignar estilos a los elementos de un objeto vector para su despliegue. Tenga presente sin embargo que las consultas pueden ser usadas en cualquier proceso que seleccione elementos componentes en objetos Vector, CAD, o TIN. Adicionalmente se puede usar consultas en algunos procesos raster para seleccionar valores de celdas para el procesamiento.



PASOS

- asegúrese que dispone de una copia de los datos de ejemplo en la colección de datos QUERY en su disco duro
- inicie TNTmips
- inicie el proceso Spatial Data Display y seleccione el Grupo New 2D de la barra de herramientas

Los ejercicios de las páginas 4-9 introducen la estructura de las declaraciones de consultas simples, operadores de comparación, y algunas herramientas útiles para construir y verificar consultas. Las páginas 10-17 cubren las estructuras de consultas que involucran declaraciones compuestas, el uso de variables y comentarios, y el uso de consultas para verificar los enlaces de registros de la base de datos. Generación de estilos por guiones se trata en las páginas 18-19, conjuntamente con el uso de las construcciones condicionales “if-then-else”. En las páginas 20-23 se encuentran ejemplos de consultas basadas en los atributos espaciales y topológicos de los objetos vector. Las páginas 24-25 muestran guiones para crear campos calculados en las tablas de la base de datos, mientras que las páginas 26-27 proveen ejemplos de consultas que son útiles en la edición de objetos vector.

Seleccionar por Consulta a un Campo Unico

PASOS

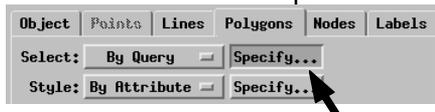
- clic en el botón del icono Add Vector y seleccione Add Vector Layer del menú desplegable
- seleccione el objeto CBSOILS_LITE del Archivo de Proyecto CB_SOILQ
- en la ventana Vector Object Display Controls, verifique que la opción Select en el panel de tarjeta Lines esté fijado a All
- fije a None la opción Select del panel Labels
- verifique que la opción Style en el panel Polygons este fijada a By Attribute



La forma más simple de consulta selecciona un tipo específico de elemento espacial (tal como polígonos, líneas, o puntos en un objeto vector) en base de los valores de un atributo único de la base de datos. En este ejercicio usted ingresa una consulta simple que selecciona para el despliegue, los polígonos del mapa de suelos en un objeto vector. Cada tipo de suelo tiene asociado valores del máximo potencial de producción para varios tipos de cultivos. La consulta selecciona los polígonos para los cuales la potencial producción del cultivo de trigo es mayor que 35 fanegas por acre. La declaración de la consulta tiene la forma:

Atributo	Operador de Comparación	Valor
----------	-------------------------	-------

La consulta debe especificar que tabla de la base de datos contiene la información del atributo, y en que campo se halla éste. Esta información de "ubicación de atributo", debe ser ingresada en la forma TABLE.FIELD. El valor en este ejemplo es un simple valor numérico, y el operador de comparación es el operador "Mayor que" (>).

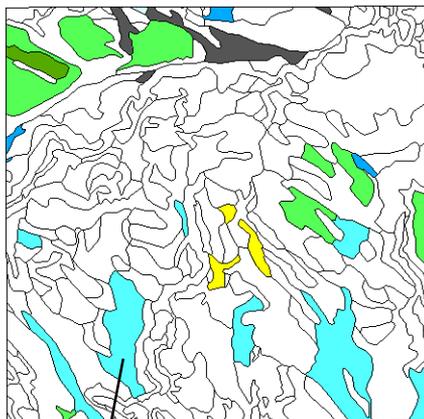


- fije la opción Select en el panel Polygons a By Query y clic en el botón Specify adyacente
- en la ventana Query Editor, escriba exactamente el siguiente texto (incluyendo las mayúsculas):

YIELD.WHEAT > 35



- clic [OK] en la ventana Query Editor
- clic [OK] en la ventana Vector Object Display Controls para aceptar las condiciones mostradas y despliegue el objeto vector



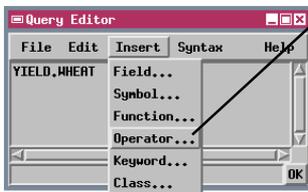
Dado que la selección de línea fue fijada a All, se dibujan los contornos de todos los polígonos de suelos. Los polígonos seleccionados por la consulta son rellenos con colores y patrones de relleno que están basados en el tipo de suelo que fueron previamente establecidos para el despliegue By Attribute. Varios tipos de suelos cumplen el criterio de selección por producción de trigo. Los polígonos no seleccionados se mantienen sin relleno.

Usando la Opción Insert Operator

La consulta previa seleccionó los polígonos de suelos pertenecientes a varios tipos de clases de suelos. Refinemos el criterio de selección de forma que la consulta seleccione únicamente aquellos polígonos con un potencial de producción de trigo exactamente igual a 38 fanegas por acre. Para especificar este criterio de selección, use el operador "Igual a" (==, doble signo igual) en la declaración de la consulta. La opción del menú Insert / Operator en el Query Editor abre la ventana Insert Operator, la cual le permite escoger el operador desde una lista desplegable e insertarlo en la declaración de la consulta, en la ubicación actual del cursor.

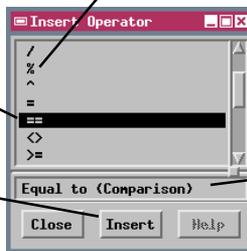
PASOS

- clic en el icono  Vector de la fila del icono Layer para abrir la ventana Vector Object Display Controls
- clic en Select: [Specify] en el panel Polygon Options
- resalte "> 35" en la ventana Query Editor y presione <Delete>
- abra el menú Insert y seleccione Operator
- en la ventana Insert Operator, despliegue hacia abajo y seleccione el operador "=="; clic [Insert], luego [Close]
- en la ventana Query Editor, escriba 38 a la derecha del operador, luego clic [OK]
- clic [OK] en la ventana Vector Object Display Controls para aceptar las condiciones mostradas y despliegue el objeto vector



Seleccione Operator del menú Insert para abrir la ventana Insert Operator ...

.... La cual lista los operadores de consulta disponibles



Seleccione el operador "Equal to".

Clic [Insert]...

...para ingresar el operador en la declaración de la consulta

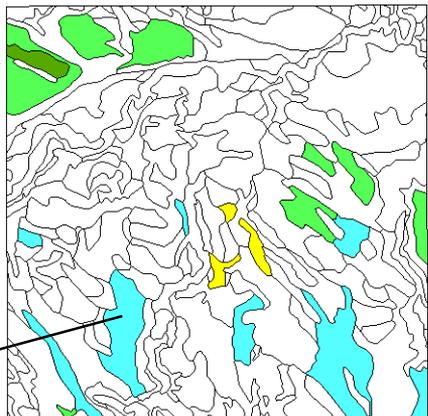
El campo de texto describe la función del operador seleccionado



Escriba el valor a ser igualado.



Menos clases de suelos cumplen el criterio de selección más restrictivo en la declaración revisada de la consulta.



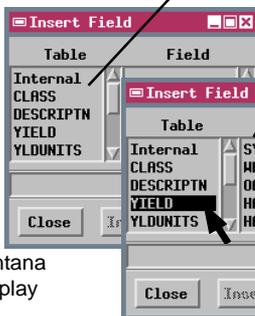
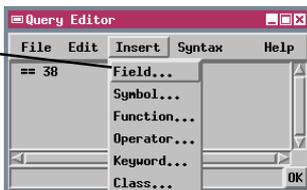
Usando la Opción Insert Field

PASOS

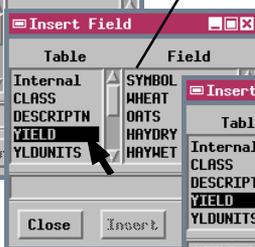
- ☑ abrir la ventana Vector Object Display Controls y la ventana Query Editor
- ☑ seleccionar `YIELD.WHEAT` en la consulta existente y presionar <Delete>
- ☑ abrir el Menú Insert y seleccionar Field
- ☑ en la ventana Insert Field que se abre, clic en `YIELD` de la lista Table
- ☑ clic en `OATS` en la lista Field, luego clic [Insert]
- ☑ clic [Close] en la ventana Insert Field
- ☑ cambiar el valor al lado derecho de la declaración de la consulta a 43, luego clic [OK]
- ☑ clic [OK] en la ventana Vector Object Display Controls

Usted también puede usar la opción del menú Insert/Field en el Query Editor para ayudar a construir o modificar las consultas. Esta opción abre la ventana Insert Field, desde la cual puede escoger la tabla y campo y automáticamente insertar en forma correcta la información de ubicación del atributo en su declaración de la consulta.

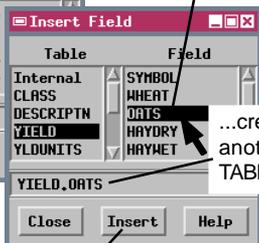
Escoja Field del menú Insert para abrir la ventana Insert Field...
...la cual lista todas las tablas disponibles en la base de datos para el tipo de elemento seleccionado.



Seleccionando una tabla también se despliega una lista de sus campos

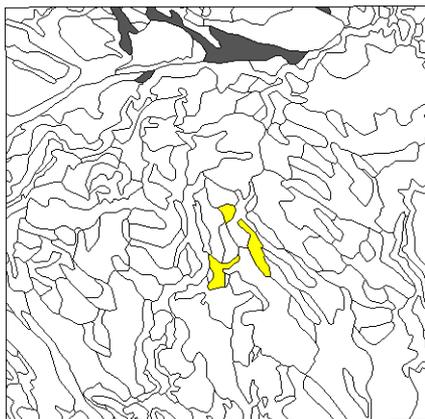


Seleccionando un campo...



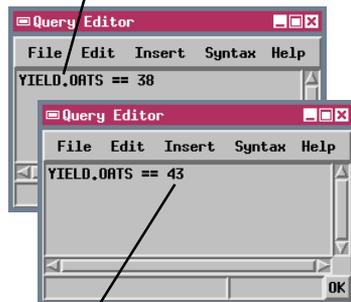
...crea la anotación TABLE.FIELD

Esta consulta selecciona varias de las mismas clases de suelo seleccionadas por la primera consulta (página 4)



Clic el botón Insert...

... para insertar la anotación TABLE.FIELD en la declaración de su consulta



Cambie el valor en el lado derecho de la declaración a 43.

Consultando un Campo String

El lenguaje de consulta usado en TNTmips es sensitivo a mayúsculas y minúsculas. Si la tabla CLASS contiene un campo llamado Class, la entrada TABLE.FIELD debe leerse CLASS.Class; si usted ingresa este como CLASS.CLASS, el proceso de consulta no encontrará el campo e indicará que existe un error en la consulta. Usando el procedimiento Insert Field le ayuda para evitar este tipo de problema.

Los campos de la base de datos que ha usado en sus consultas hasta ahora contenían datos numéricos. La tabla YIELD para CBSOILS_LITE también contiene un campo llamado SYMBOL con los símbolos de tipos de suelo en formato String (cadena de caracteres). El término "string" es una abreviación de "character String" que significa que el campo no es evaluado numéricamente, y puede contener texto y otros caracteres no numéricos. Campos string pueden contener números (por ejemplo, CLASS1), pero ellos son leídos como caracteres en lugar de números. Valores String en declaraciones de consultas deben ser encerradas en comillas, y son también sensitivas a las mayúsculas.

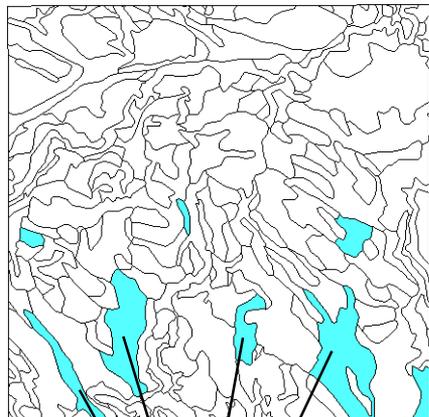
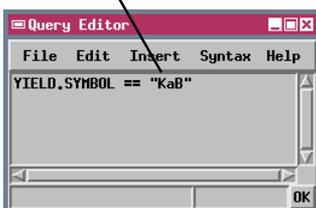
PASOS

- abrir la ventana Vector Object Display Controls y la ventana Query Editor
- seleccionar YIELD.OATS en la consulta existente y presionar <Delete>
- use el procedimiento Insert Field para insertar YIELD.SYMBOL en el lado izquierdo de la declaración de la consulta
- cambie el valor en el lado derecho de la declaración de la consulta a "KaB" (incluyendo las comillas), luego clic [OK]
- clic [OK] en la ventana Vector Object Display Controls



El campo SYMBOL contiene valores string.

Encierre un valor string en comillas.



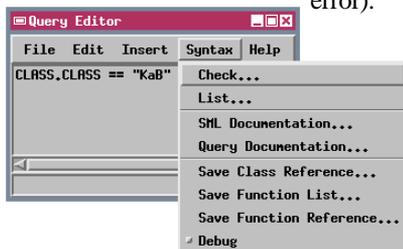
Polígonos de suelo seleccionados pertenecientes a la clase KaB.

Verificando la Sintaxis de la Consulta

PASOS

- ☑ abrir la ventana Vector Object Display Controls y la ventana Query Editor
- ☑ cambiar manualmente el lado izquierdo de la declaración de la consulta existente a CLASS.CLASS (todas mayúsculas)

Las reglas concernientes al uso de mayúsculas y comillas para los valores string son ejemplos de la sintaxis del lenguaje de consulta de TNTmips. La sintaxis de la consulta es verificada automáticamente cuando hace clic en [OK] para ejecutar la consulta, y cuando escoge Close del menú File para cerrar el Query Editor. (Si la consulta tiene un error de sintaxis cuando trata de cerrar el Query Editor, la ventana permanece abierta y se despliega un mensaje de error).



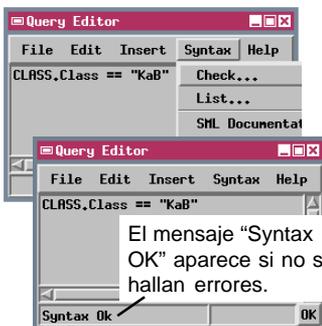
- ☑ escoger Check desde el menú Syntax
- ☑ note el mensaje de error, luego clic [OK] en la ventana de mensajes.
- ☑ cambie el lado izquierdo de la declaración de la consulta existente a CLASS.Class
- ☑ escoger Check desde el menú Syntax
- ☑ note el mensaje "Syntax OK"

Usted puede verificar la sintaxis de una consulta antes de ejecutarla, seleccionando Check del menú en el Query Editor. El proceso puede encontrar errores de ortografía, falta de paréntesis u otros símbolos, o referencias a campos de la base de datos no existentes. El proceso inicia la verificación al inicio de la consulta. Si no se hallan errores de sintaxis, la línea de mensajes al final de la ventana Query Editor, indica "Syntax OK.". Si un error de sintaxis es detectado, el cursor de texto es localizado al final del primer componente de la declaración que la computadora no puede interpretar, y un mensaje de error aparece en la línea de mensaje. En este ejemplo, el verificador de consulta, detecta que no existe en la base de datos un campo llamado CLASS en la tabla CLASS, por lo tanto el cursor es ubicado al final de la entrada CLASS.CLASS. Después de corregir el error de sintaxis, puede usar la opción de Syntax nuevamente para verificar los errores en el resto de la consulta.

En este ejemplo el cursor es ubicado al final de la entrada inválida TABLE.FIELD.



En consultas más complejas, el error de sintaxis puede actualmente ser el resultado de algo que falta después del punto donde el cursor es ubicado, tal como un paréntesis de cierre o la declaración "end" de un bucle begin / end.



El mensaje "Syntax OK" aparece si no se hallan errores.

Usando Cálculos en las Consultas

El valor en el lado derecho de la declaración de una consulta puede también ser provisto por un campo de la base de datos o un cálculo que involucre un campo de la base de datos. Los cálculos en las consultas pueden usar operaciones aritméticas estándar: suma (+), resta (-), multiplicación (*), y división (/). Si lo desea, puede ingresar los símbolos de operación desde la ventana Insert Operator. La opción Insert / Function provee acceso a otras funciones trigonométricas y matemáticas que también se pueden usar en las declaraciones de consultas. El ejemplo de consulta para este ejercicio selecciona polígonos de suelos para los cuales el potencial de producción de avena es exactamente 5 fanegas por acre más que la producción de trigo.

A este momento usted probablemente ha notado que la última consulta usada para un objeto particular y tipo de elemento es almacenada automáticamente con el objeto y ésta se abre la siguiente vez que selecciona la misma opción By Query. Si desea almacenar varias consultas para el mismo objeto para su uso futuro, puede usar las opciones Save y Save As del menú File en el Query Editor. Estas opciones le permiten guardar la consulta actualmente desplegada en la ventana Query Editor como un archivo de texto con la extensión **.QRY**, o como un objeto texto en el Archivo de Proyecto. Para volver a llamar una consulta almacenada, use la opción Open en el menú File.

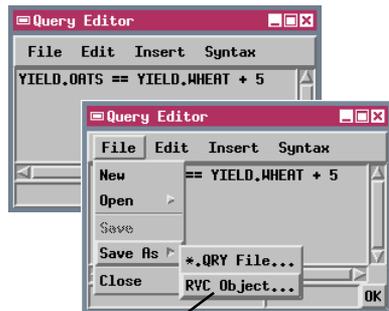
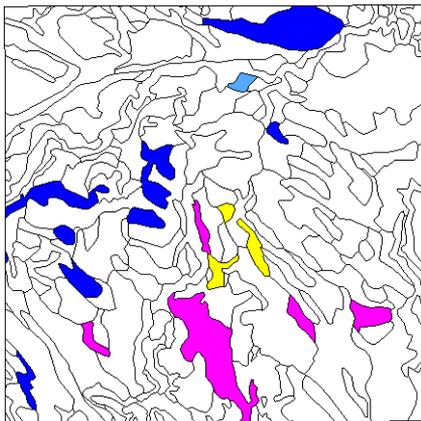
PASOS

- escoja New desde el menú File
- use los procedimientos de Insert y/o ingreso manual para crear la siguiente declaración de consulta:

```
YIELD.OATS == YIELD.WHEAT + 5
```

- seleccione Save As del menú File, luego *.QRY File del menú que se abre
- use la ventana estándar de File Selection para nominar al nuevo archivo que contiene la consulta
- clic [OK] en la ventana Query Editor
- clic [OK] en la ventana Vector Object Display Controls

Usualmente una consulta trabajará únicamente con un objeto específico debido a la referencia a un campo único de la base de datos. Si dispone de una serie de objetos con formatos idénticos de bases de datos, o las consultas se refieren únicamente a campos en tablas estándar creadas por TNTmips, entonces puede usar la misma consulta para cualquiera de los objetos.



Escoja RVC Object para almacenar la consulta como un objeto texto en un Archivo de Proyecto.

Consultas Compuestas

PASOS

- abrir la ventana Vector Object Display Controls y la ventana Query Editor
- escoger New del menú File
- use el procedimiento Insert y/o ingreso manual para crear la siguiente consulta:

```
YIELD.WHEAT > 34 and YIELD.OATS > 40
```

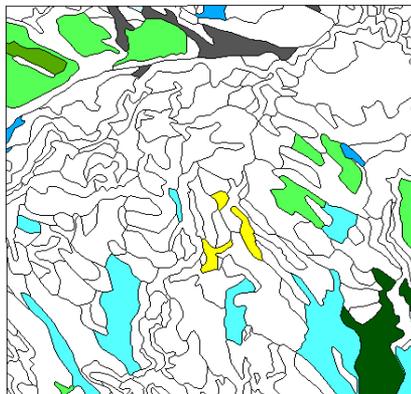
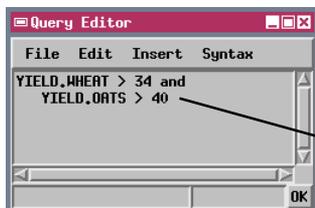
- clic [OK] en la ventana Query Editor y en la ventana Vector Object Display Controls
- repita los pasos de arriba sustituyendo "or" por "and" en la declaración de consulta

Cada una de las consultas usadas en los ejercicios previos emplean una sola comparación al escoger polígonos para el despliegue. En muchos casos podría desear seleccionar elementos usando una combinación de varios criterios. Una serie de selecciones de comparación en una declaración de consulta deben estar relacionadas entre si por uno o más operadores lógicos de la teoría de conjuntos, tales como "and," "or," y "not.". Estos operadores deben ser ingresados todos en minúsculas. Usted

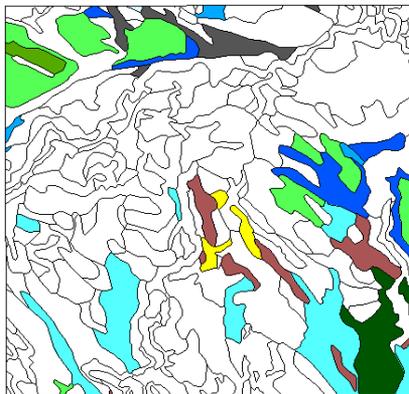
puede insertar operadores lógicos en declaraciones de consulta, usando el procedimiento de Insert Operator.

Cuando dos comparaciones están enlazadas por el operador lógico "and", ambas comparaciones deben ser verdaderas para hacer que toda la declaración de consulta sea verdadera y se seleccione el elemento. Cuando dos comparaciones están enlazadas por el operador lógico "or", la declaración de consulta es verdadera si cualquiera de las comparaciones es verdadera. Los elementos que cumplen cualquiera de los criterios son seleccionados.

Una declaración de consulta larga puede continuar en líneas adicionales, sin embargo usted puede desear poner una sangría a las líneas subsiguientes para hacer claro que ellas son parte de una declaración.



Polígonos para los cuales el potencial de producción de trigo es mayor a 34 fanegas por acre, y la potencial producción de avena es mayor a 40 fanegas por acre.



Polígonos para los cuales bien el potencial de producción de trigo es mayor a 30 fanegas por acre, o bien el potencial de producción de avena es mayor a 40 fanegas por acre.

Usando el Operador “not equal to”

La mayoría de tipos de suelo en el área de Crow Butte tienen un mayor potencial para producción de avena que para trigo, pero el trigo normalmente rinde un mayor precio que la avena cuando se vende la cosecha. Asumamos que los precios de la cosecha por fanega son \$3.25 para la avena y \$4.00 para el trigo. Este ejemplo de consulta es usado para identificar los tipos de suelo para los que el potencial total del precio del cultivo de avena por acre (producción potencial en fanegas por acre multiplicado por el precio por fanega) es mayor o igual que aquel para el trigo.

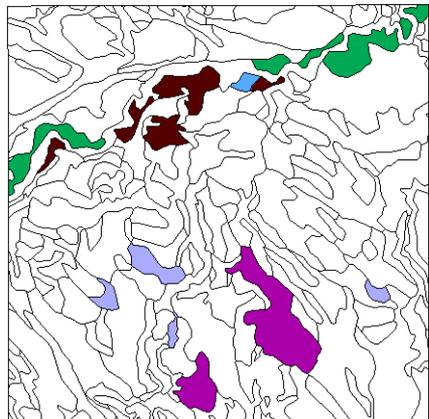
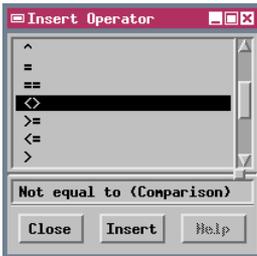
```
YIELD.WHEAT <> 0 and
YIELD.OATS * 3.25 >= YIELD.WHEAT *
```

Esta consulta es complicada por el hecho que el potencial de producción del cultivo para tipos de suelos que no pueden ser cultivados es 0, y tales tipos de suelos satisfacen la selección de comparación en la segunda línea de la consulta. La primera línea de la consulta excluye los tipos de suelo cero-producción, e ilustra el uso del operador “not equal to” (<> or !=). Esta declaración selecciona los polígonos para los que la potencial producción de trigo no es igual a 0. Solo estos polígonos están sujetos a la comparación de precios en la segunda línea.

PASOS

- abrir la ventana Vector Object Display Controls y la ventana Query Editor
- escoja New desde el menú File
- use los procedimientos de Insert y/o ingreso manual para crear la siguiente consulta:

- clic [OK] en la ventana Query Editor
- clic [OK] en la ventana Vector Object Display Controls



Polígonos para los cuales un cultivo de avena produce mayor precio por acre que los de trigo, asumiendo la producción potencial del cultivo y el precio fijado por fanega.

Usando Comentarios y Variables

PASOS

- abrir la ventana Vector Object Display Controls y la ventana Query Editor
- escoja New del menú File
- use el procedimiento de Insert y/o ingreso manual para crear la siguiente consulta:

```
dollars = 129 # define variable for
              # required crop
              # price per acre

# select polygons based on crop price
YIELD.OATS * 3.25 > dollars or
YIELD.WHEAT * 4.00 > dollars
```

- clic [OK] en la ventana Query Editor
- clic [OK] en la ventana Vector Object Display Controls

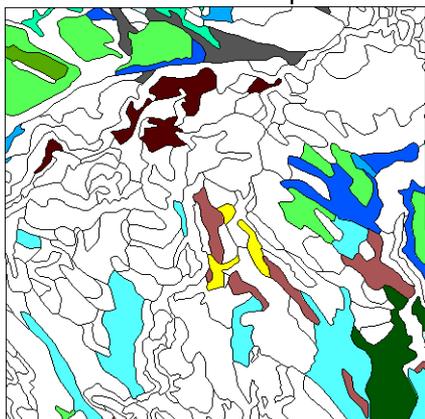
Usted puede mejorar la facilidad de lectura y posterior utilidad de una consulta incluyendo comentarios.

Un comentario inicia con el símbolo “#”, y puede estar en una línea por sí solo o al final de una declaración. Usted puede usar comentarios dentro de una consulta para explicar declaraciones individuales, y un comentario introductorio para proporcionar una explicación de la intención de uso de la consulta y a que objetos se aplica la misma.

El proceso de consultas en TNTmips también le permite nominar y asignar valores a variables para

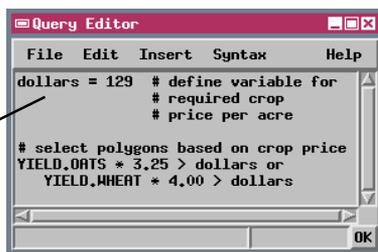
uso en una consulta. Este ejemplo de consulta selecciona polígonos de suelos que exceden el precio potencial de cultivo por acre requerido, tanto para avena como para trigo. La primera

línea de la consulta, es una declaración de asignación, que define una variable numérica llamada “dollars” para almacenar el precio requerido, y se le asigna el valor de 129. El símbolo “=” es usado para asignar un valor a la variable (por lo que debe usarse “==” para el operador “Equal to”).



Los nombres de variables deben estar siempre en minúsculas, y deben empezar con una letra. Un nombre de variable no puede ser igual al nombre de un comando de consulta, tabla de base de datos o de campo.

Las variables son útiles cuando el mismo valor es usado mas de una vez en la consulta. Si desea ejecutar esta consulta nuevamente con un precio requerido diferente, solo necesita cambiar el único valor asignado a la variable “dollars”. Si la consulta hubiese sido escrita sin la variable, necesitaría cambiar dos valores numéricos en la declaración de selección.



Usando Variables String

You Usted también puede definir variables que contengan valores string. El nombre que crea para una variable string debe terminar en \$. La consulta en este ejercicio define una variable string name\$, a la cual es asignado el valor "Glenberg". La consulta selecciona un subconjunto de polígonos de suelo pertenecientes a la serie de suelos Glenberg, la cual incluye dos tipos de suelo en el área de Crow Butte. En lugar de usar las dos clases de símbolos para seleccionar los polígonos, esta consulta toma ventaja del hecho que el campo NAME en la tabla DESCRIPTN provee de una descripción del suelo que empieza con el nombre "Glenberg" para ambas clases. La consulta usa el operador "contains", el cual selecciona elementos para los cuales un carácter string especificado es igual a todo o parte de un campo string. En este caso el carácter string a ser igualado ("Glenberg") es almacenado en la variable name\$.

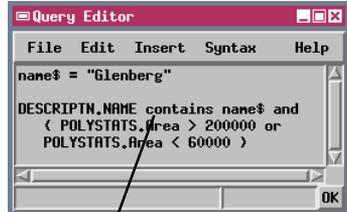
PASOS

- abrir la ventana Vector Object Display Controls y la ventana Query Editor
- escoja New del menú File
- use el procedimiento Insert y/o ingreso manual para crear la siguiente consulta:

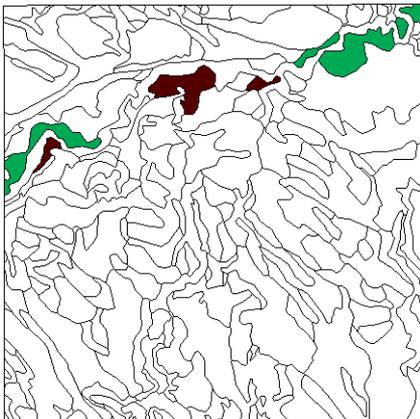
```
name$ = "Glenberg"
DESCRIPTN.NAME contains name$ and
( POLYSTATS.Area < 60000 or
POLYSTATS.Area > 200000 )
```

- clic [OK] en la ventana Query Editor
- clic [OK] en la ventana Vector Object Display Controls

Los polígonos que cumplen la selección de comparación son luego presentados en pantalla en función de su área (en metros cuadrados), lo cual es almacenado en el campo Area en la tabla estándar POLYSTATS. (Una tabla POLYSTATS se presenta solamente si los atributos estándar han sido calculados para el objeto vector).



El operador "contains" selecciona polígonos para los cuales el carácter string en la variable name\$ es igual a cualquier parte del campo string DESCRIPTN.NAME



Style	SYMBOL	NAME
DuB		Duroc very fine sandy loam,
EpF		Epping silt loam, 3 to 30 pe
GbB		Glenberg loamy very fine sar
GoB		Glenberg loamy very fine sar
JnL		Jagen loamy very fine sand,

El texto string "Glenberg" es incluido en el campo DESCRIPTN.NAME para los dos tipos de suelo pertenecientes a la serie Glenberg.

Usando el Operador Lógico “not”

PASOS

- abrir la ventana Vector Object Display Controls y la ventana Query Editor
- escoja New del menú File
- use el procedimiento de Insert y/o ingreso manual para crear la siguiente consulta:

```
YIELD.WHEAT >= 27 and  
YIELD.WHEAT <= 32 and  
!( DESCRIPTN.NAME contains "Glenberg" )
```

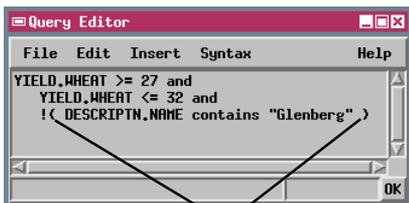
- clic [OK] en la ventana Query Editor
- clic [OK] en la ventana Vector Object Display Controls

Un número de suelos en el área de Crow Butte, tienen un potencial de producción de trigo comparable a aquellos dos suelos de Glenberg (27 y 32 fanegas por acre). La consulta en este ejercicio selecciona todos los tipos de suelo dentro de este rango de valores de producción de trigo *excepto* los suelos de Glenberg.

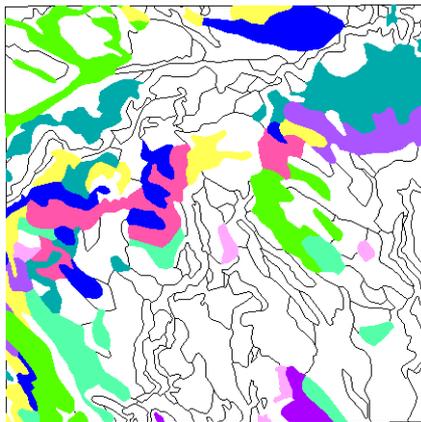
Las dos primeras líneas de la consulta seleccionan los polígonos para los que el potencial de producción de trigo está

dentro del rango designado. La tercera línea de la consulta empieza con el operador lógico “not” (!), el cual reversa el resultado de la variable, operador, o expresión que le sigue. En este caso la expresión siguiente al operador “not” solamente seleccionará los polígonos pertenecientes a los dos suelos Glenberg. El operador “not” reversa este resultado y selecciona todos los polígonos que cumplen los requerimientos previos de producción excepto los polígonos de suelos Glenberg.

El operador “not” es especialmente útil cuando existe un gran grupo de valores que desea seleccionar y un grupo pequeño de valores mas fácilmente especificable que no desea seleccionar.



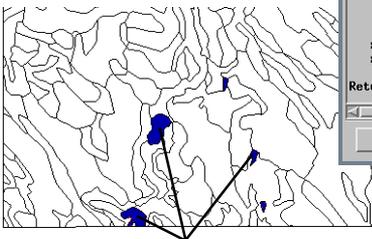
El operador 'not' reversa el siguiente elemento que lo sigue (incluyendo variables u otros operadores). Si usted desea aplicar el operador 'not' a una expresión completa (como en este ejemplo), la expresión debe estar encerrada en paréntesis.



Verificando Enlaces de Registros

La consulta usada en este ejercicio difiere de aquellas usadas en todos los ejercicios previos. En lugar de consultar valores específicos de atributos, ésta es basada en el número de registros en una tabla particular de la base de datos, que están enlazados a un polígono. La consulta emplea la función de grupo `SetNum()`, la cual retorna el número de ítem en un grupo. En este caso el conjunto es proporcionado por una expresión que tiene la forma `Table[*]`, la que indica a todos los registros enlazados a un elemento en la tabla mencionada.

Esta consulta identifica todos los polígonos de clases de suelos que no tienen registros enlazados en la tabla `YIELD`. La consulta inversa `SetNum(YIELD[*]) > 1` identificará todos los elementos con más de un registro enlazado. Usted puede usar consultas con esta forma para identificar polígonos que no han sido aún asignados atributos, o que de alguna manera tienen registros extras enlazados. En este ejemplo, todos los polígonos sin registros enlazados en la tabla `YIELD`, pertenecen a la clase `WATER`. Ellos representan pequeños lagos y estanques, y consecuentemente no tienen potencial producción de cultivo.



Los polígonos sin registros enlazados en la tabla `YIELD` de la base de datos, todos pertenecen a la clase `WATER`. Usted también puede seleccionar elementos sin enlace haciendo un clic en el botón derecho sobre el icono de la tabla correspondiente en el `Group Controls` o en la ventana `Layout Controls` y seleccionando desde el menú desplegable `Select All Unattached Elements`.

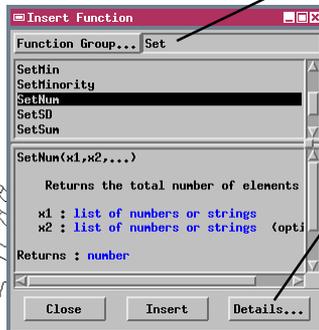
PASOS

- abrir la ventana `Vector Object Display Controls` y la ventana `Query Editor`
- escoja `New` del menú `File`
- use el procedimiento de `Insert` y/o ingreso manual para crear la siguiente consulta:

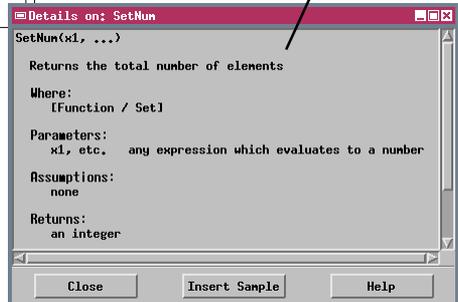
```
SetNum(YIELD[*]) < 1
```

- clic [`OK`] en la ventana `Query Editor`
- clic [`OK`] en la ventana `Vector Object Display Controls`

Use la opción `Function Group` en la ventana `Insert Function` para seleccionar cual grupo de funciones listar en la ventana.



Clic [`Details`] en la ventana `Insert Function` para ver más información sobre la función actualmente seleccionada y un ejemplo de su uso.



Seleccionar usando Múltiples Registros Enlazados

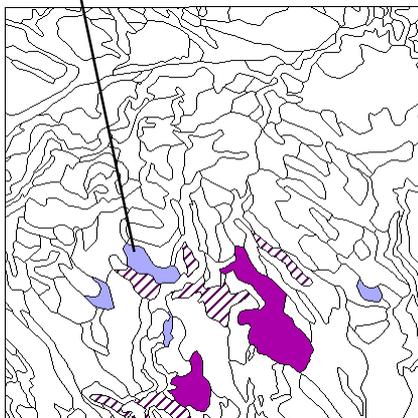
PASOS

- abrir la ventana Vector Object Display Controls y la ventana Query Editor
- escoja New del menú File
- use el procedimiento de Insert y/o ingreso manual para crear la siguiente consulta:

```
"WB" in
LAYER[*].texture
```

- clic [OK] en la ventana Query Editor
- clic [OK] en la ventana Vector Object Display Controls

Polígonos de clases de suelos que tienen estratos rocosos alterados como parte de su perfil típico de suelo.



La tabla LAYER de los polígonos de clases de suelos en el objeto CBSOILSQ contiene información de los diferentes niveles en un perfil típico de suelo para uno de ellos. Existe un registro separado para cada nivel en el perfil, y de este modo hay múltiples registros enlazados para cada polígono de suelo. Seleccionar elementos en base de atributos entre múltiple registros enlazados, requiere de una consulta con una sintaxis especial.

En este ejercicio por ejemplo, deseamos seleccionar los tipos de suelo que incluyen estrato rocoso alterado en cualquier parte del perfil. Este atributo es codificado por el string "WB" en el campo texture. Si usted intenta usar la consulta de selección convencional LAYER.texture == "WB", encontrará que no se selecciona ningún polígono, sin embargo de que algunos suelos tienen estrato rocoso alterado en la parte baja del perfil. Esta estructura de consulta solamente verifica el *primer* registro enlazado en la tabla para cada polígono, el cual en este caso es normalmente el registro del primer nivel, conteniendo atributos del nivel de suelo más superior. Los registros subsiguientes para las capas de suelo más profundas son ignorados.

Para consultar el campo texture de *todos* los registros enlazados, debemos usar la expresión LAYER[*].texture, la cual nos retorna un conjunto que lista el contenido del campo texture de cada registro enlazado al actual polígono. Luego necesitamos determinar si cualquiera de los miembros del conjunto corresponde al atributo "WB" deseado. La forma más fácil para hacerlo es usar la palabra clave "in" como un operador lógico. La consulta es verdad si la variable precedente al operador es una comparación exacta a cualquiera de los elementos en el conjunto producido por la expresión siguiente al operador. Esta construcción puede ser utilizada bien sea con campos string o numéricos.

Encontrar Polígonos Isla

Un polígono vector que está totalmente encerrado dentro de un polígono más grande es denominado **polígono isla**. Debido a que a menudo los polígonos isla tienen atributos diferentes de aquellos que los encierran, los procesos que alteran la topología o asignación de atributos de un objeto vector deben mantener un registro de las relaciones de los polígonos isla.

La tabla interna de los polígonos incluye varios campos que contienen información pertinente a los polígonos isla. Usted puede consultar estos campos para seleccionar los polígonos isla o aquellos que contienen islas. El campo `Internal.inside` contiene el número de elementos de polígonos incluidos si existe alguno. Todos los polígonos isla tendrán un valor diferente de cero en este campo. La primera consulta por lo tanto selecciona todos los polígonos isla. El campo `NumIslands` muestra el número de islas contenidas por cada polígono. La segunda consulta en este ejercicio selecciona los polígonos que tienen en `NumIslands` un valor mayor a cero, correspondiendo a todos los polígonos que contienen islas.

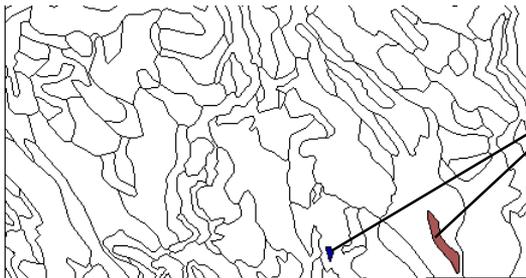
PASOS

- abrir la ventana Vector Object Display Controls y la ventana Query Editor
- escoja New del menú File
- use el procedimiento de Insert y/o ingreso manual para crear la siguiente consulta:

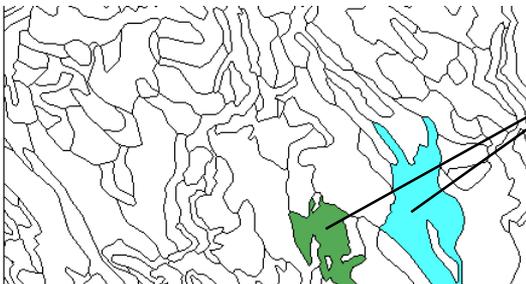
```
Internal.Inside > 0
```

- clic [OK] en la ventana Query Editor
- clic [OK] en la ventana Vector Object Display Controls
- repetir los pasos anteriores usando la siguiente consulta:

```
Internal.NumIslands > 0
```



Polígonos isla seleccionados por la primera consulta. Cada isla pertenece a una clase diferente de suelo que el polígono que la encierra.



Dos polígonos en el objeto vector CBSOILS_LITE incluyen polígonos isla, y fueron por lo tanto seleccionados por la segunda consulta..

Asignando Estilos por Guión

PASOS

- abrir la ventana Vector Object Display Controls
- en el panel tipo tarjeta Lines, prender el botón conmutador Draw Lines Before Polygons

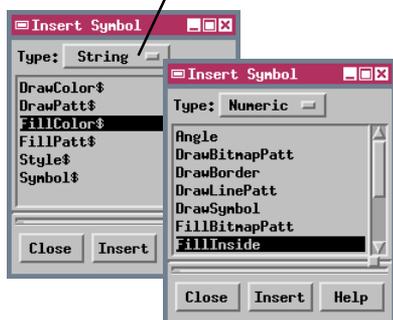


- en el panel tarjeta Polygons, fijar la opción Style a By Script, y clic el botón adyacente Specify
- use el procedimiento Insert Symbol (opciones Numeric y String) para crear el siguiente guión de estilo:

```
FillInside = 1
FillColor$ = "100 50 0"
DrawBorder = 1
DrawColor$ = "red"
```

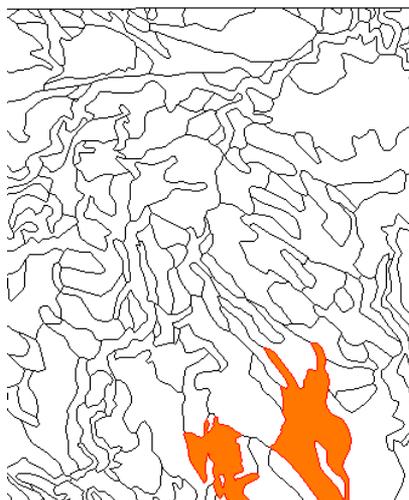
- clic [OK] en la ventana Query Editor
- clic [OK] en la ventana Vector Object Display Controls

Use el botón de opción en la ventana Insert Symbol para acceder a las diferentes listas de símbolos: constantes o los diferentes tipos de variables (incluyendo numéricas, string y vector entre otras).



La opción Estilo por Guión (Style by Script) le permite en base de sus atributos, especificar las características de despliegue para subgrupos de elementos seleccionados. Para ingresar las opciones de estilo, este ejercicio retiene la consulta de selección previa, pero utiliza un guión de estilo para fijar los nuevos parámetros de despliegue para todos los polígonos seleccionados. (Normalmente usted usaría la opción de estilo All Same para llevar a cabo esto).

Cuando esta determinando estilos por guión, el menú Insert en la ventana Query Editor proporciona acceso a variables adicionales que son usadas para fijar las características de despliegue. FillInside y DrawBorder son variables numéricas que están asignadas un valor de 1 para rellenar los polígonos seleccionados y dibujar un borde alrededor de ellos. FillColor\$ y DrawColor\$ son variables string que son utilizadas para fijar el color para el relleno y el borde del polígono respectivamente. El valor asignado a ellos (encerrado en comillas) puede ser tanto un nombre de color (red, green, blue, black, white, orange, brown, cyan, magenta, o gray), como un grupo de valores RGB (cada uno de 0 a 100%).



Guiones para Estilo Compuesto

En este ejercicio todas los polígonos de clases de suelo son seleccionados para el despliegue, y un guión de estilo es usado para definir dos grupos distintos de parámetros de despliegue de los polígonos en función del área del polígono.

Cuando usted desea especificar acciones alternativas en una consulta o guión de estilo, debe usar los comandos “if-then-else”, para explícitamente definir la lógica. Las declaraciones en este guión se traducen a “**Sí (if)** un polígono tiene un área mayor a 200000 metros cuadrados, **luego (then)** rellénelos con amarillo, **Caso contrario (else)** rellénelos con un patrón de mapa de bits (‘BitmapPatt4’)”. Cuando más de una declaración relacionada sigue al comando “then” o “else” (como en este ejemplo), el grupo de declaraciones debe encerrarse dentro de los comandos begin/end. Omitiendo los comandos begin/end después de “else” en esta consulta podría no producir un error de sintaxis. Sin embargo en este caso solamente la primera declaración se aplicaría como alternativa a la acción “then”; las declaraciones restantes serían interpretadas como un apliación global a **todos** los polígonos seleccionados (igual que las dos primeras líneas de la consulta), sobrescribiendo los parámetros de estilo definidos anteriormente.

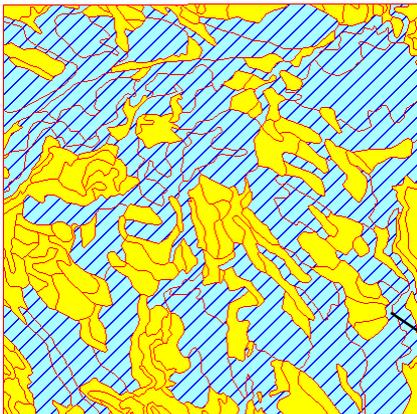
PASOS

- abrir la ventana Vector Object Display Controls
- en el panel de tarjeta Polygons, fije la opción Select a All, deje la opción Style fija a By Script, y clic el botón adyacente Specify
- escoja New del menú File
- use el procedimiento Insert y/o ingreso manual para crear la consulta que se muestra abajo
- clic [OK] en la ventana Query Editor
- clic [OK] en la ventana Vector Object Display Controls

```

DrawBorder = 1
DrawColor$ = "red"
if ( POLYSTATS.Area < 200000 )
then
  begin
    FillInside = 1
    FillColor$ = "yellow"
  end
else
  begin
    FillInside = 1
    FillBitmapPatt = 1
    FillPatt$ = "BitmapPatt2"
  end

```



Para usar un patrón de relleno bitmap en un guión, el patrón asignado a la variable FillPatt\$ debe estar en el User Set de los patrones definidos para el objeto. Para que se dibuje el patrón, las variables FillInside y FillBitmapPatt deben las dos ser fijadas a 1. Mire el folleto *Tutorial: Creando y Usando estilos* para información en la creación de patrones de relleno y otros estilos.

Los polígonos con un área de 200 000 metros cuadrados o mayor son rellenos con un patrón de rayas definido en BitmapPatt2. Los polígonos menores son rellenos con amarillo.

Encontrar el Polígono que Encierra un Punto

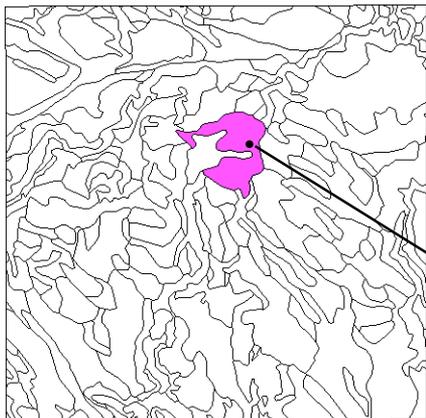
PASOS

- abrir la ventana Vector Object Display Controls
- en el panel de tarjeta Polygons, fijar la opción Style a By Attribute
- fijar la opción Select a By Query, y clic el botón adyacente Specify
- escoger New desde el menú File en la ventana Query Editor
- use los procedimientos Insert y/o ingreso manual para crear la consulta que se indica abajo
- clic [OK] en la ventana Query Editor
- clic [OK] en la ventana Vector Object Display Controls

El proceso de consultas incorpora una serie de funciones espaciales que pueden ser utilizadas para seleccionar elementos. La función FindClosestPoly() (ubicada en el grupo de funciones Vector), nos entrega el número de elemento del polígono que encierra un punto con las coordenadas especificadas X y Y. Los parámetros requeridos para la función incluyen el objeto vector a consultar, los valores X y Y de coordenadas, y el número de objeto del subobjeto de georeferencia usado en el procesamiento de los valores de las coordenadas. El número de objeto de georeferencia es proporcionado por la función GetLastUsed-GeorefObject() (ubicada en el grupo de funciones Georeference), el cual es usado en la declaración de asignación para la variable “georef”. La variable predefinida “Vect” (ubicada en la lista Vector en la ventana Insert Symbol) es usada para indicar el actual objeto vector. Las dos primeras líneas en la consulta definen las variables que contienen los valores de las coordenadas X y Y del punto.

```
xvar = -103.33861
yvar = 42.73583
georef = GetLastUsedGeorefObject( Vect )
c_poly = FindClosestPoly( Vect, xvar,
yvar, georef )
Internal.ElemNum == c_poly
```

La función FindClosestPoly() es usada en una declaración de asignación que almacena en una variable numérica (c_poly en este ejemplo) el número de elemento del polígono que circunscribe al punto. La declaración final de la consulta compara el número de elemento de cada polígono en el objeto (Internal.ElemNum) con el número almacenado en c_poly para encontrar el polígono que se iguala para su despliegue.



Ubicación del punto especificado por las coordenadas xvar (longitud) y yvar (latitud). Los valores usados en la consulta deben estar en el mismo sistema de coordenadas al especificado en el subobjeto de georeferencia. Las coordenadas de Latitud / Longitud deben ser expresadas en grados y fracciones decimales.

Consulta de Vecindad de Polígonos: Lógica

Una consulta de selección también puede usar la información topológica asociada con un objeto vector. Cada línea en un objeto vector tiene un nodo de inicio y uno final, lo que definen un lado izquierdo y uno derecho de la línea. Cada polígono está conformado de elementos lineales específicos y la tabla interna para líneas incluye campos que contienen los números de los elementos de los polígonos que se hallan a los lados de la línea. La función `GetVectorPolyAdjacentPolyList()` (en la lista Vector de la ventana Insert Function) usa esta información para determinar cuáles polígonos son adyacentes al polígono actual. Esta función puede ser usada en una consulta para seleccionar polígonos que están adyacentes a clases específicas de polígonos.

Como un ejemplo examinemos una consulta al objeto vector `CBSOILS_LITE` que selecciona los polígonos pertenecientes a la clase de suelo “SrD”, que también son adyacentes a los polígonos de clase “Sa”. Para ser considerado adyacente, los polígonos deben compartir una línea límite común, no solamente un nodo común. La estrategia general usada en tal tipo de consulta es la siguiente:

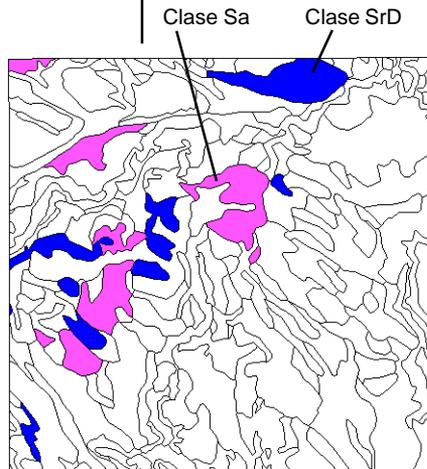
- 1) Definir la clase a seleccionar.
- 2) Si un polígono pertenece a la clase seleccionada ejecutar los subsiguientes pasos (prueba de vecindad – adyacencia), caso contrario descartarlo.
- 3) Obtener la lista de polígonos que son adyacentes al polígono actual.
- 4) Verificar la asignación de clase para cada polígono adyacente. Si cualquiera de ellos se iguala a la clase adyacente definida seleccionar el polígono para el despliegue. Si ninguno se iguala, descartarlos.

La sintaxis para esta consulta se muestra y explica en la siguiente página.

PASOS

- abrir la ventana Vector Object Display Controls y la ventana Query Editor
- seleccione New desde el menú File
- use los procedimientos Insert y/o ingreso manual para generar la consulta que se muestra en la siguiente página
- clic [OK] en la ventana Query Editor
- clic [OK] en la ventana Vector Object Display Controls

El objeto vector `CBSOILS_LITE` con todos los polígonos de clases SrD y Sa seleccionados (para comparación con la ilustración en la siguiente página)



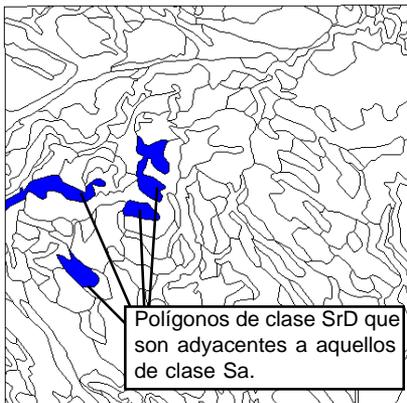
Consulta de Vecindad de Polígonos: Sintaxis

```

1  if (CLASS.Class == "SrD") then
2  begin
3  array polylist [10]
4  numpolys = GetVectorPolyAdjacentPolyList(Vect, polylist)
5  for i = 1 to numpolys begin
6  polynum = polylist[i]
7  if (Vect.poly[polynum].CLASS.Class$ == "Sa")then
8  return 1
9  end
10 return 0
11 end
12 else return 0

```

1. Selección condicional de los polígonos de clase SrD para verificación subsiguiente.
2. Inicia el bucle de procesamiento para verificar la clase de los polígonos adyacentes.
3. Define un arreglo uni-dimensional llamado "polylist" que contendrá una lista de los números de los elementos que son adyacentes al polígono actual. Inicializa el tamaño del arreglo a 10 elementos (el tamaño es automáticamente actualizado por la función en la siguiente declaración).
4. Define una variable "numpolys" a la que se asigna un valor igual al número de polígonos adyacentes al polígono actual. Este valor es devuelto por la función GetVectorPolyAdjacentPolyList, la cual también encuentra los números de los elementos de los polígonos adyacentes y los almacena en el arreglo "polylist". La variable predefinida "Vect" es usada para indicar el objeto vector actual.
5. Inicia un bucle de procesamiento para examinar la clase de cada polígono en el arreglo. El bucle se corre una vez para cada elemento en el arreglo, empezando con la primera posición (índice 1 del arreglo) y continuando hasta la última posición (especificada por el valor actual de la variable "numpolys"). En cada bucle la variable "i" se asigna el valor del índice actual del arreglo para su uso en la siguiente declaración.
6. Asigna el número del polígono adyacente actual (especificado por su índice del arreglo) a la variable "polynum".
7. Averigua la clase del polígono adyacente actual y la compara con la clase adyacente especificada. La especificación de la base de datos es de la forma "Object.database[record#].table.field". (El "\$" al final de la especificación de la base de datos indica que el campo objetivo es un campo string). Si las clases coinciden, entonces....
8. La declaración "return 1" explícitamente indica que la consulta es verdadera para un polígono satisfaciendo la condición de arriba, por lo tanto el polígono será seleccionado para el despliegue.
9. Fin del bucle de procesamiento del arreglo.
10. Si todos los polígonos adyacentes fallan la verificación de clases de arriba, la declaración "return 0" indica que la consulta es falsa.
11. Fin del bucle de vecindad de polígonos.
12. Indica que la consulta es falsa para un polígono que no cumple la condición inicial de selección de clases en la declaración 1.



Fronteras Censales en Datos TIGER

Los objetos vector importados de los archivos TIGER / Line de la Oficina de Censos de los USA, están conformados por segmentos de líneas que representan las características físicas naturales y hechas por el hombre, al igual que las fronteras gubernamentales y de los censos. Las fronteras de las zonas de los censos (y sus equivalentes Areas de Bloques Numerados. o BNA's) y sus componentes bloques censales, usualmente coinciden con otros detalles del mapa y no son identificados explícitamente por una Característica de Código de Clase (CFCC) como las características básicas de un mapa.

Las líneas de límite de un bloque censal pueden ser seleccionadas para el despliegue o extracción usando una consulta que seleccione líneas para las que los números de bloque al lado izquierdo y derecho no son los mismos. Los bloques que han sido subdivididos retienen el mismo número de bloque, pero son identificados por diferentes letras en los campos sufijos de los bloques de la izquierda y derecha; el segundo grupo de declaraciones en la consulta de ejemplo, selecciona estos límites. Finalmente, los bloques en BNA's adyacentes pueden tener el mismo número, de forma que la declaración final selecciona las líneas que separan diferentes BNA's.

PASOS

- remover el layer usado en el ejercicio previo 
- clic el icono Add Vector y seleccionar Add Vector 
- Layer del menú desplegable
- seleccionar el objeto ALAMEDA del Archivo de Proyecto TIGER
- en la ventana Vector Object Display controls, fije la opción Style a All Same en el panel Lines y la opción Select a By Query
- abra la ventana Query Editor y escoja New desde el menú File
- use el procedimiento Insert y/o ingreso manual para crear la consulta que se muestra abajo
- clic [OK] en la ventana Query Editor
- clic [OK] en la

```
Basic_Data.Block_Left <> Basic_Data.Block_Right
or ( Basic_Data.Block_Left ==
Basic_Data.Block_Right
and Basic_Data.BlockSuff_Left <>
Basic_Data.BlockSuff_Right )
or Basic_Data.BNANum_Left <>
Basic_Data.BNANum_Right
```

ventana
Vector
Object
Display
Controls



Campos Calculados de Registros Múltiples

PASOS

- ☑ minimice la ventana Display del ejercicio previo
- ☑ escoja Edit / Attribute Databases del menú principal de TNTmips
- ☑ navegar y seleccionar el objeto CBSOILS_LITE en el Archivo de Proyecto CBSOILSQ
- ☑ encender en la ventana Select el botón Polygon
- ☑ en la ventana Database Editor, clic derecho en la caja de la tabla Class y seleccione Edit Definition
- ☑ en la ventana de definición para la tabla Class, clic el icono del botón Add Field 
- ☑ en la lista field, resalte el nombre por defecto para el nuevo campo e ingrese por teclado ClassArea
- ☑ en el panel Field info, seleccione Computed desde el menú Field Type y clic Edit Expression
- ☑ ingrese en la ventana Query Editor la consulta indicada abajo

```
sum = 0.0
num = SetNum( POLYSTATS[*] )
for i = 1 to num begin
    sum = sum +
POLYSTATS[i].Area
end
return sum
```

- ☑ clic [OK] en la ventana Query Editor
- ☑ ingrese 12 en la caja de texto Width y 2 en la caja de texto Places
- ☑ clic [OK] en la ventana de definition
- ☑ doble-clic en la caja de la tabla Class para abrir la tabla
- ☑ escoja Table / Close para cerrar la tabla Class

Los guiones se pueden usar también para definir los valores de campos calculados en las tablas de la base de datos. En muchos casos estos guiones necesitan solo crear combinaciones aritméticas simples de otros campos en el mismo registro. La tarea en este ejercicio es más compleja: crear un campo calculado en la tabla Class del polígono de CBSOILS_LITE que indique el total del área para cada tipo de suelo.

Las áreas de los polígonos se almacenan en la tabla POLYSTATS para los polígonos individuales, pero estamos creando el campo calculado en la tabla Class, la que tiene un registro por cada tipo de suelo. El guión mostrado aquí está designado para sumar las áreas de los polígonos de cada clase de suelo y entregar esa suma como el valor del campo calculado.

El guión define una variable numérica “sum” que es usada para sumar las áreas en POLYSTATS.Area field. Esta variable debe inicialmente fijarse a un valor de 0.0 para cada clase. A la variable “num” está asignado el valor (para la clase actual de suelo) igual al número de registros enlazados en la tabla POLYSTATS. Esta variable es usada para fijar el número de iteraciones para el bucle que suma las áreas.

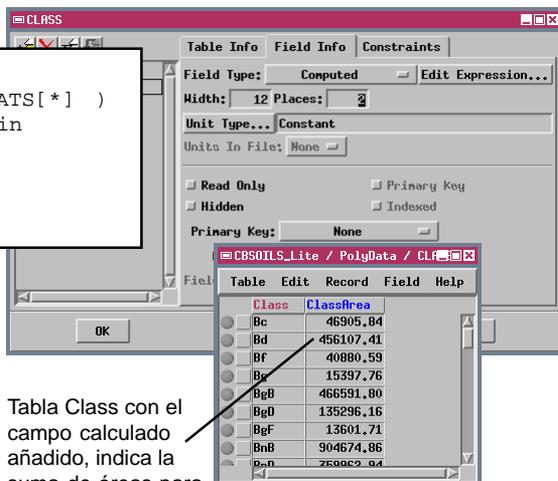


Tabla Class con el campo calculado añadido, indica la suma de áreas para cada tipo de suelo.

Campos de Expresiones String

Un campo de expresión string es un tipo especial de campo calculado en una tabla de la base de datos. El uso más simple de un campo de expresión string es para copiar el contenido de un campo string a la tabla actual, desde otra tabla enlazada. La expresión en este caso es simplemente la referencia apropiada TABLE.FIELD. También puede usar las expresiones string para unir el contenido de varios campos string en un nuevo campo. Por ejemplo una tabla llamada NAME podría contener campos separados para los nombres y apellidos. Usted puede usar el operador “+” (suma) para unir estos string. La expresión NAME.FIRST + “ ” + NAME.LAST producirá ingresos de la forma “John Doe”. La expresión debe incluir cualquier carácter de separación (espacios, comas) en comillas como se muestra. Usted puede usar campos de expresiones string unidas para proveer textos para Data Tips más informativos o etiquetas.

La expresión que usará en este ejercicio emplea la función sprintf(), la que le permite formatear complejas expresiones string más fácilmente. El primer argumento de la función es un string de control (en comillas), el que es seguido por las referencias del campo string. Cada uno de los ingresos “%” en el string de control, corresponde a cada una de las referencias de campos string listados. El string de control puede también incorporar texto insertado, espacios, o puntuaciones.

PASOS

- escoja File / Open Database en la ventana Database Editor
- navegar hasta el objeto CBSECT y seleccionarlo del Archivo de Proyecto CB_SECT
- activar el botón Polygon en la ventana Select
- en la ventana Database Editor, clic en el botón derecho sobre la caja de la tabla Sections y seleccione Edit Definition
- en la ventana definition para la tabla Sections, clic en el campo Range de la lista y clic el botón del icono  Add Field
- en el listado de campos, resaltar el nombre por defecto para el campo new y escriba SecTwpRng
- seleccione String Expression del menú Field Type y clic Edit Expression
- ingrese la consulta indicada abajo en la ventana Query Editor window

```
printf( "Sec %s Twp %s Rng %s",
Sections.Section, Sections.Township,
Sections.Range )
```



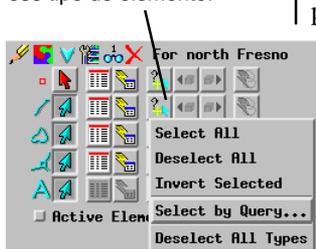
Section	Township	Range	SecTwpRng
34	31N	51W	Sec 34 Twp 31N Rng 51W
35	31N	51W	Sec 35 Twp 31N Rng 51W
36	31N	51W	Sec 36 Twp 31N Rng 51W
13	32N	51W	Sec 13 Twp 32N Rng 51W
14	32N	51W	Sec 14 Twp 32N Rng 51W
15	32N	51W	Sec 15 Twp 32N Rng 51W
16	32N	51W	Sec 16 Twp 32N Rng 51W

Texto formateado creado por la expresión string.

- clic [OK] en la ventana Query Editor
- ingrese 25 en la caja de texto Width
- clic [OK] en la ventana definition
- doble-clic en la caja para abrir la tabla Sections
- escoja File / Close de las dos ventanas Database Editor cuando haya completado este ejercicio

Consultas para Verificar Problemas de Digitalización

Clic el icono del botón Select / Deselect y escoja Select by Query para abrir la ventana Standard Query Editor de forma que pueda ingresar una consulta para ese tipo de elemento.

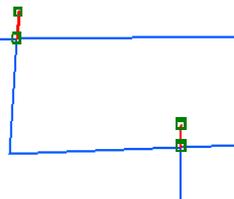


Las consultas de selección pueden también ser muy útiles cuando esta creando o editando objetos vector usando el Spatial Data Editor de TNTmips. Los objetos vector complejos pueden contener errores de digitalización tales como líneas excedidas (overshoots), polígonos sin cerrar y polígonos irrelevantes. Muchos de estos defectos no son visibles excepto a niveles de ampliación muy altos, lo que hace difícil y consumidora de tiempo la verificación manual. Usted puede acelerar la búsqueda de potenciales problemas topológicos, usando consultas de selección tal como los ejemplos de abajo. La ventana Spatial Data Editor provee iconos que le permiten crear y aplicar consultas de selección para un tipo particular de elemento.

OVERSHOOTS – LÍNEAS EXCEDIDAS

Los Overshoots son segmentos de líneas cortas que se extienden incorrectamente más allá de una intersección de líneas. Si usted ha ejecutado el proceso de Standard Attributes para el objeto vector, puede usar una consulta de selección basada en la longitud de la línea para seleccionar todas las líneas muy cortas para su examen y posible remoción:

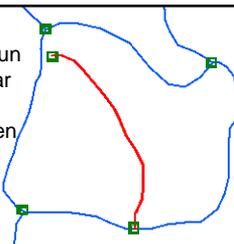
LINESTATS.Length < [your length value]



POLÍGONOS SIN CERRAR

En un objeto vector conteniendo una red de polígonos, un gap entre dos líneas que deberían intersectarse, podría dejar un solo polígono donde debería existir dos polígonos separados. Las líneas que fallan al cerrar el polígono pueden ser encontradas por consulta ya que ellas tienen el mismo polígono en los dos lados:

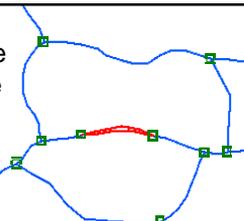
Internal.LeftPoly == Internal.RightPoly



POLÍGONOS IRRELEVANTES

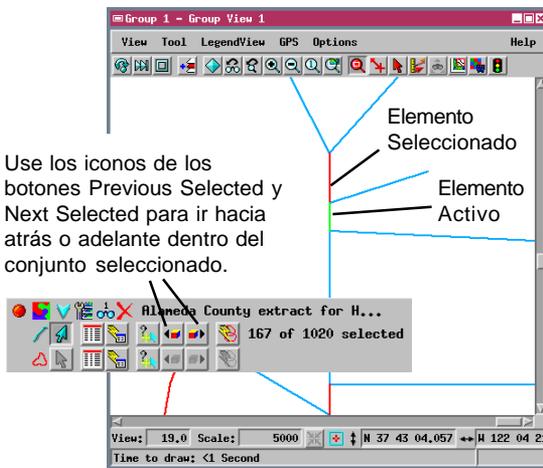
El trazado doble de los límites de polígonos puede crear extraños polígonos irrelevantes a lo largo de la frontera entre dos polígonos colindantes. Los polígonos irrelevantes (sliver polygons) normalmente tienen un área mucho menor que los polígonos principales, y son usualmente altamente elongados (con un alto coeficiente de Compactación). Utilice una consulta combinada en los campos de Area y CompactRatio en la tabla POLYSTATS para seleccionar polígonos irrelevantes

**POLYSTATS.Area < [your area value] or
POLYSTATS.CompactRatio > 3.00**



Paneo por Consulta

Una consulta ejecutada desde la ventana Element Selection en el Spatial Data Editor o en Spatial Data Display a menudo selecciona más de un elemento. Uno de estos elementos seleccionados es designado el elemento “activo”; los elementos seleccionados y el activo son resaltados en colores diferentes. Las operaciones de edición pueden aplicarse bien al elemento activo o a los seleccionados. Usted puede usar los botones de los iconos de Previous Selected y Next Selected en la ventana Element Selection para avanzar o retroceder un paso a través del conjunto de elementos seleccionados, haciendo en turno a cada uno, el elemento activo. La vista es automáticamente reposicionada (si es necesario) para desplegar el elemento activo actual. Esta característica de “paneo por consulta” permite mantener en la ampliación escogida para examinar (y talvez editar) cada elemento mientras fácilmente se moviliza a lo largo del conjunto seleccionado.



PASOS

- restablecer las ventanas View y Group Controls
- abrir la ventana Vector Object Display Controls
- fijar la opción Select a All en el panel Lines y clic [OK]
- abrir el menú Options en la ventana Spatial Data Display View y asegurarse que la opción Show Scale / Position esté encendida
- escriba “5000” en la caja de texto Scale al final de la ventana View, y presione <Enter>
- clic el icono del botón Show Details en la fila de iconos de Layer 
- clic el icono del botón Select para líneas 
- clic el icono del botón Select / Deselect para líneas, y escoja Select by Query del menú desplegable 
- ingrese la siguiente consulta en el Query Editor:

```
LINESTATS.Length < 50
```
- clic [Apply] en la ventana Select by Query 
- clic el icono del botón Next Selected

Los ejercicios en este folleto Tutorial han introducido los fundamentos de la estructura y sintaxis de las consultas de bases de datos para uso en TNTmips, TNTedit, y TNTview. El lenguaje de consulta es un subconjunto del Lenguaje de Manipulación Espacial - Spatial Manipulation Language (SML) - usado en TNTmips, y comparte la misma sintaxis. Adicionalmente a la documentación sobre consultas citada en la página 2, usted podría consultar la documentación de SML en el Manual de Referencia de TNTmips para obtener consejos adicionales de programación que expandan sus capacidades de consulta.

Software Avanzado para Análisis Geoespacial

MicroImages, Inc. produce una línea completa de software profesional para visualización, análisis y publicación de datos geoespaciales. Contáctenos o visite nuestra página en Internet para información detallada del producto.

TNTmips TNTmips es un sistema profesional con una completa integración GIS, análisis de imágenes, CAD, TIN, cartografía de escritorio y gestión de Bases de Datos geoespaciales.

TNTedit TNTedit provee de herramientas interactivas para crear, georeferenciar y editar materiales de proyectos tipo vector, imagen, CAD, TIN y Bases de Datos Relacionales en una gran variedad de formatos.

TNTview TNTview tiene las mismas características poderosas de despliegue de TNTmips y es perfecto para aquellos que no necesitan las características de procesamiento técnico y preparación de TNTmips.

TNTatlas TNTatlas permite publicar y distribuir materiales de proyectos en CD-ROM a bajo costo. Los CDs de TNTatlas pueden ser usados en cualquier plataforma popular de computadora.

TNTserver TNTserver permite publicar sus Atlas en TNT en Internet o en su Intranet. Navegue a través de atlas de geodatos con su navegador web y el applet Java TNTclient.

TNTlite TNTlite es una versión libre de TNTmips para estudiantes y profesionales con proyectos pequeños. Usted puede descargar TNTlite del sitio Internet de MicroImages, o puede ordenar TNTlite en CD-ROM con el conjunto actualizado de folletos *Tutoriales*.

Indice

Bienvenido a Construyendo y Usando Consultas.....	3
Seleccionar por Consulta a un Campo Único.....	4
Usando la Opción Insert Operator	5
Usando la Opción Insert Field	6
Consultando un Campo String.....	7
Verificando la Sintaxis de la Consulta.....	8
Usando Cálculos en las Consultas.....	9
Consultas Compuestas.....	10
Usando el Operador "not equal to".....	11
Usando Comentarios y Variables.....	12
Usando Variables String.....	13
Usando el Operador Lógico "not".....	14
Verificando Enlaces de Registros.....	15
Seleccionar usando Múltiples Registros Enlazados.....	16
Encontrar Polígonos Isla.....	17
Asignando Estilos por Guión.....	18
Guiónes para Estilo Compuesto.....	19
Encontrar el Polígono que Encierra un Punto.....	20
Consulta de Vecindad de Polígonos: Lógica.....	21
Consulta de Vecindad de Polígonos: Sintaxis	22
Fronteras Censales en Datos TIGER.....	23
Campos Calculados de Registros Múltiples	24
Campos de Expresiones String	25
Consultas para Verificar Problemas de Digitalización.....	26
Paneo por Consulta	27



MicroImages, Inc.

11th Floor - Sharp Tower
206 South 13th Street

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

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