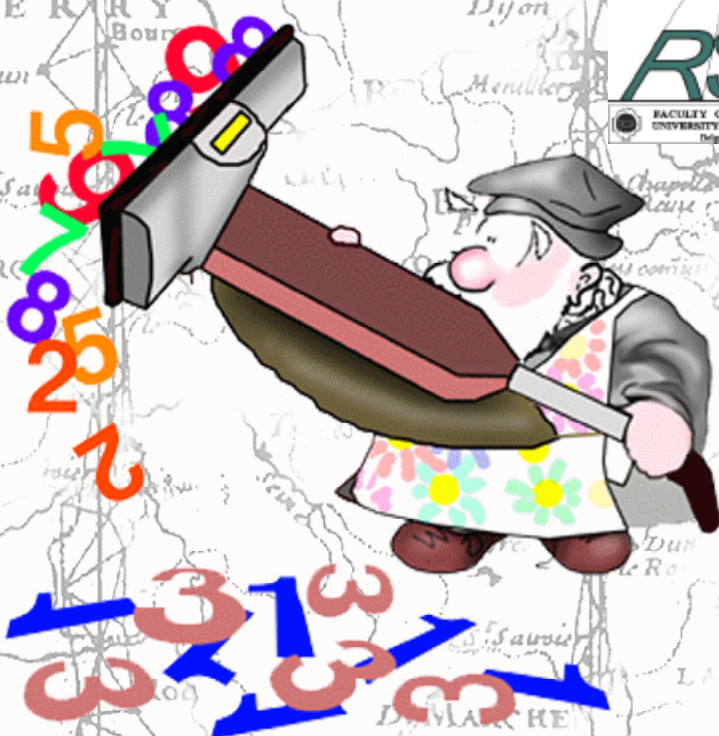


Getting Started



Izrada i korišćenje upita

Remote Sensing Center



sa

TNTmips®

TNTedit™

TNTview®

Pre nego što pocnete

Ova brošura Vas upoznaje sa upitima baza podataka TNT proizvoda i uci Vas kako da napravite upite da bi koristili informacije o atributima zakacenim za Vaše vektorske, CAD i TIN objekte. Na prvi pogled ovaj proces može izgledati komplikovan, ali ovaj set korak-po-korak vežbi, vodi Vas kroz zadate strukture i sintakse upita, napredujuci od jednostavnih primera do upita sa višestrukim uslovnim izjavama i procesnim petljama.

Potrebno predznanje Pretpostavlja se da ste uradili vežbe iz brošure *Getting Started:Displaying Geospatial Data* i *Getting Started:Navigating*. Te vežbe vas upoznaju sa osnovnom koncepcijom i veštinama koje ovde nisu ponovo pokazane. Za bilo koju proveru koja je potrebna, konsultujte ove brošure i TNT mips-ov referentni prirucnik.

Ogledni podaci Vežbe predstavljene u ovoj brošuri koriste ogledne podatke koji se distribuiraju zajedno sa TNT CD proizvodima, koje možete download-ovati sa Microlmage's web sajta. Ogledne fajlove možete pronaci u zbirci podataka QUERY. Obezbedite kopiju oglednih podataka na hard disku tako da sacuvane promene možete kasnije koristiti.

Dodatna dokumentacija Ova brošura predstavlja samo upoznavanje sa korišćenjem baza podataka upita. Konsultujte oblast "Display Vector, CAD and TIN by Query" u poglavlju Display u TNT referentnom prirucniku radi više informacija o Querz Editor-u, i Appendix 2: Database Queries za potpuno upucivanje o komandama i funkcijama upita.

TNT mips® i TNT lite® TNT mips postoji u dve verzije: profesionalna i besplatna TNT lite verzija. Ovoj brošuri odgovaraju obe verzija. Ako niste kupili profesionalnu verziju (koja zahteva hardverski kljuc), TNT mips radi u TNT lite modu, koji ogranicava velicinu objekta i ne dozvoljava izvoz podataka.

Upiti o bazama podataka se takode mogu koristiti da prikažu geoprostorne objekte u TNTview-u i da selektuju elemente za editovanje u TNT edit-u. Sve vežbe u ovoj brošuri mogu da se urade u TNT lite verziji uz pomoc obezbedenih uzoraka geopodataka.

Randall B. Smith, Ph.D., 17 September 2001

Ukoliko nemate kopiju ove brošure u boji, možda ce biti teško da utvrdite znacajne tacke na nekim ilustracijama. Možete štampati ili citati ovu brošuru u boji sa web sajta Microlimages-a. Web site je takode i Vaš izvor najnovijih *Getting started* brošura na druge teme. Možete download-ovati vodac za instalaciju, ogledne podatke i najnoviju verziju TNTlite-a.

<http://www.microimages.com>

Dobrodošli u izradu i korišćenje upita

TNTmips omogućava veliku fleksibilnost prilikom korišćenja atributa baze kao što su vektori, CAD, i TIN objekti za kontrolu prikaza i štampe objekata ili za selektovanje objekata radi korišćenja u različitim procesima. Bazni upiti omogućavaju najkompletnije i raznovrsne nacine za korišćenje ovih atributa.

Bazni upit je set instrukcija koji definišu određene kriterijume koji se koriste za izdvajanje zapisa iz baze podataka. Specifični prostorni elementi (kao što su linije ili poligoni) za koje su ti zapisi "prikaceni" su tada automatski selektovani za tekuci proces. Upit odgovara specifičnim tipovima elemenata i tada je moguće istovremeno korišćenje odvojenih upita za različite tipove elemenata u objektu. U Spatial Data Display, opcija Style By Script dozvoljava korisniku da oznaci prikazani parametar za selektovane elemente na osnovu atributa. Informacije o atributima koje razmatrate u upitima mogu biti kvalitativni (kao ime klase) ili kvantitativni (kao vrednost žetve).

Upiti moraju da koriste standardnu "gramatiku i sintaksu" da bi TNTmips mogao to da razume. Programski jezik za formiranje upita koji koristi TNTmips je Spatial Manipulation Language (SML).

Korisnik slaže upite u prozoru Query Editor. Ovaj editor poseduje menije koji olakšavaju konstrukciju validnih upita tako što omogućava izbor polja iz tabele baze podataka i ubacuje simbole i funkcije iz list prozora sa pravilnim sintaksama. Query Editor takode obezbeđuje kontrolu sintakse da bi se olakšalo nalaženje grešaka pre primene upita.

Vežbe u ovoj brošuri koriste upite za selektovanje ili oblikovanje elemenata za prikaz u vektorskim objektima. Treba imati na umu da upiti mogu biti korišćeni u svakom procesu koji selektuje komponente elemenata u vektorskom, CAD, ili TIN formatu. Takode je moguće koristiti upite u pojedinim rasterskim procesima za selektovanje vrednosti celija kod procesiranja.



KORACI

- ✍ obezbedite kopiju iz zbirke primeraka u QUERY kolekciji podataka na Vašem Hard disku
- ✍ pokrenite TNT mips
- ✍ pokrenite process Spatial Data Display i selektujte New 2D Group sa palete sa alatkama

Vežbe na stranama 4-9 upoznaju nas sa strukturom jednostavnog upita, operacijama poredenja i nekim korisnim alatima za građenje i proveru upita. Stranice 10-17 obraduju strukturu upita uključujući složene izjave, upotrebu promenljivih, komentare i korišćenje upita radi provere prikacenih upisa baze podataka. Stilizovanje skriptom nalazi se na 18-19 strani, zajedno sa upotrebom uslovne "if-then-else" konstrukcije. Primeri upita bazirani na prostornim i topološkim atributima vektorskih objekata, nalaze se na stranama 20-23. Strane 24-25 objašnjavaju kako se kreira izracunavanje polja u tabelama baza podataka, dok su na 26-27 strani dati primeri upita korisnih za editovanje vektorskih objekata.

Selektovanje upitom jednog polja

KORACI

- ✂ pritisnite na Add vector dugme i selektujte Add Vector Layer iz padajućeg menija
- ✂ selektujte CBOILS_LITE objekat iz CB_SOILQ Project File u QUERY kolekciji podataka
- ✂ u Vector Object Display Controls prozoru, proveriti da li je Select opcija na Lines panelu postavljena na All
- ✂ postaviti Select opciju na Labels panelu na None
- ✂ proveriti da li je Style opcija na Polygons panelu postavljena na By Attribute



Najjednostavniji vid upita je selektovanje specifičnih tipova prostornih elemenata (poligoni, linije i tačke u vektorskoj slici) na osnovu vrednosti pojedinih atributa iz baze. U ovoj vežbi je potrebno uneti jednostavne upite koji izdvajaju poligone tla na mapi, u vektorskom objektu prikaza.

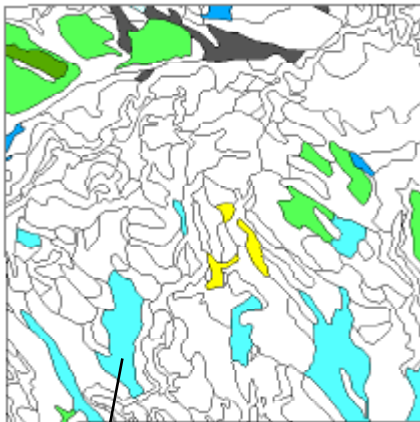
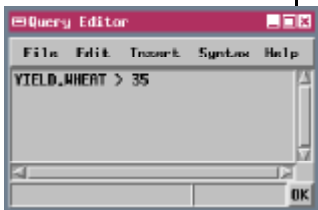
Svaki tip zemljišta ima pridruženu vrednost maksimalne kolicine roda za više žetvi. Upit selektuje poligone za koje je potencijalna žetva roda pšenice veća od 35 bušela po jutru. Upit je u sledećoj formi:

Attribute	Comparison Operator	Value
-----------	---------------------	-------

U upitu je potrebno definisati bazu koja sadrži tabele sa informacijama, odnosno specificovati određeno polje. Ova "pozicija atributa" informacija mora biti uneta u formi TABLE.FIELD. Vrednost u ovom primeru je jednostavna numerička vrednost, dok je komparativni operator "veći od" ($>$).



- ✂ postavite Select opciju na Polygons panelu na By Query i kliknite na dugme Specify
- ✂ u prozoru Query Editor unesite sledeći tekst YIELD.WHEAT > 35
- ✂ kliknite na OK u Query Editor prozoru

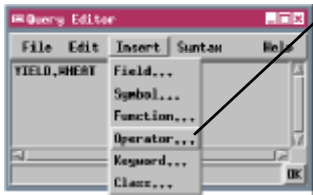


- ✂ pritisnite na OK u Vector Object Display Controls prozoru da bi podešavanja bila prihvacena

Pošto je selekcija podešena na All, iscrtane su sve spoljnje linije poligona tla. Poligoni selektovani upitom su popunjeni bojama i uzorci boja su bazirani na tipu tla, a prethodno su podešeni za prikaz By Attribute. Kriterijum za selekciju različitih tipova tla je prinos pšenice. Neselektovani poligoni ostaju nezapunjeni.

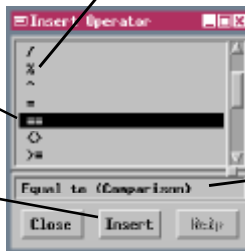
Korišćenje Insert Operator opcije

Poligoni zemlje selektovani prethodnim upitom pripadaju različitim klasama tipova tla. Moguće je preraditi kriterijum selekcije tako da upit selektuje samo one poligone sa potencijalnim rodnom pšecnice od 38 bušela po jutru. Da bi se odredio ovaj kriterijum selekcije, koristiti znak "=" (dupli znak ==) u upitu. Meni opcija Insert/ Operator u Query Editor-u otvara Insert operator prozor, koji nudi operatore sa izborne liste i umeće ih u upit na trenutnu poziciju kursora.



Izabrati operator iz Insert menija da bi otvorili Insert Operator prozor...

...koji daje listu svih dostupnih operatora upita



Selektujte operator "jednako".

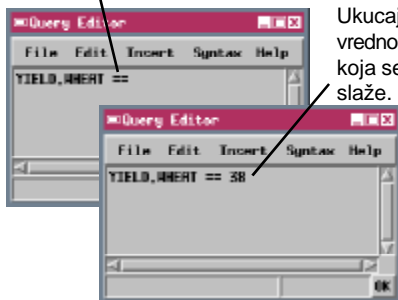
Pritisnite [Insert]...

...da unesete operator u upit.

KORACI

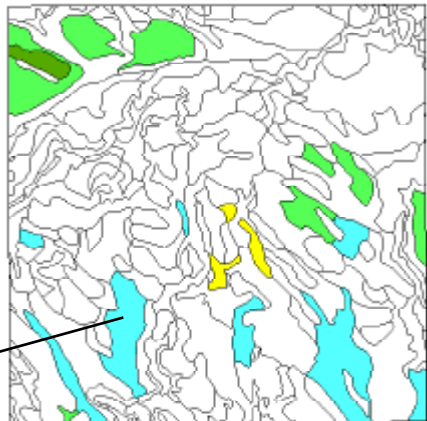
- ✗ kliknite na Vector koloni ikona da bi otvorili Vector Object Display prozor
- ✗ pritisnite Select:[Specify] u Polygon Options panelu
- ✗ oznacite ">35" u Query Editor prozoru i pritisnite Delete
- ✗ otvorite Insert meni i selektujte Operator
- ✗ u Insert Operator prozoru, oznacite "==" operator; kliknite na [Insert], a zatim [Close];
- ✗ u Query Editor prozoru uneti 38 desno od operatora, a potom OK
- ✗ pritisnite OK u Vector Object Display Controls prozoru da bi sve vrednosti bile prihvacene i prikazane

Tekstualno polje opisuje funkciju selektovanog operatora.



Ukucajte vrednost koja se slaže.

Za manji broj klasa zemljišta važe strožiji kriterijumi u preispitanom upitu.



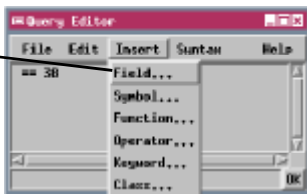
Korišćenje Insert Field opcije

KORACI

- ❌ otvorite Vector Object Display Controls prozor i Query Editor prozor
- ❌ izaberite YIELD.WHEAT u postojećem upitu, a potom <Delete>
- ❌ otvorite Insert Menu i izaberite Field
- ❌ u Insert Field prozoru, kliknite na YIELD u Table list-u
- ❌ pritisnite OATS u Field listi, a zatim [Insert]
- ❌ kliknite [Close] u Insert Field prozoru
- ❌ promenite vrednost na desnoj strani upita na 43 i zatim kliknite na OK
- ❌ pritisnite OK u Vector Object Display Controls prozoru

Moguće je koristiti Insert/Field meni opcije u Query Editor-u koja služi kao pomoć prilikom izrade ili izmene upita. Ova opcija otvara Insert Field prozor u kome se odabira Table ili Field i automatski ubacuje informacija o atributima lokacije u upit u pravilnom obliku.

Izaberite Field iz menija Insert da bi otvorili prozor Insert Field...



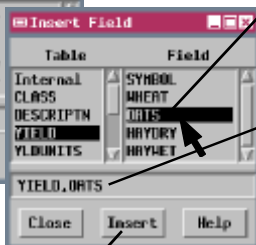
...koji daje spisak svih dostupnih tabela baze podataka za selektovani tip elementa.



Izabiranjem Table se isto prikazuje lista njenih polja.

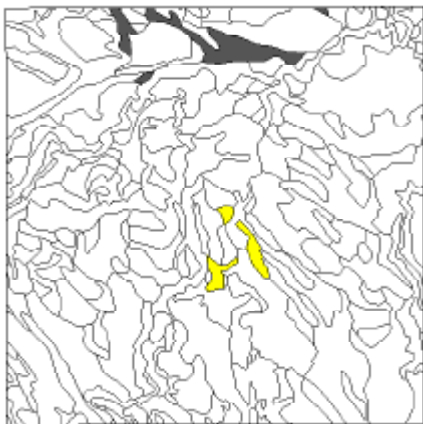


Izborom polja...

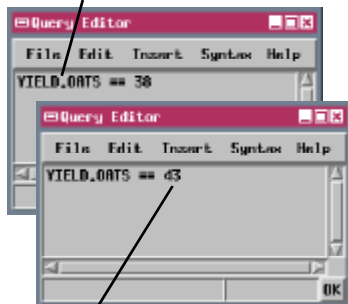


...kreira se TABLE.FIELD unos.

Ovim upitom je omogućena selekcija nekoliko istih klasa tla koje su već selektovane prethodnim upitom (strana 4).



pritisnite na Insert dugme...
... da unesete TABLE.FIELD unos u Vaš upit.



Promenite vrednost na desnoj strani izjave na 43.

Upiti String polja

Jezik upita korišćen u TNTmips-u je osetljiv na velicinu slova. Ukoliko tabela CLASS sadrži polje Class, unos TABLE.FIELD mora biti CLASS.Class; ukoliko se unese kao CLASS.CLASS upitni proces neće pronaci polje i prijaviće grešku. Korišćenje Insert Field procedure će pomoci u izbegavanju ovakvog tipa problema.

Polja iz baze koja su korišćena u upitu do sada su sadržala numericke vrednosti. Tabela YIELD za CBSOILS_LITE takode sadrži polja koja se zovu SYMBOL sa tipom zemljišta oznacenog u string formatu. Termin simbol je skracenica za "karakter string", što znaci da to polje nije oznaceno numericki i da može da sadrži tekst i ostale nenumericke karaktere. String polja mogu sadržati i brojeve (npr. CLASS1), ali se oni citaju kao karakater a ne kao broj. String vrednosti u upitu moraju biti zatvorene znacima navoda, a takode su osetljiva na velicinu slova (mala/velika slova).

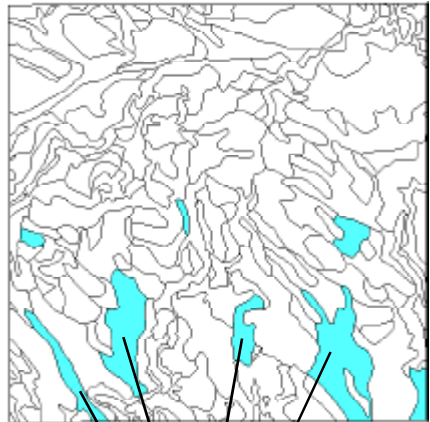
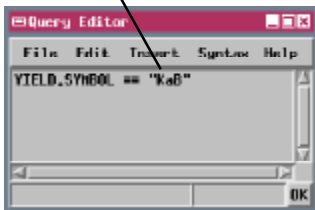
KORACI

- ✗ otvorite Vector Object Display Controls prozor i Query Editor prozor
- ✗ selektujte YIELD.OATS u postojećem upitu i pritisnite <Delete>
- ✗ upotrebite Insert Field proceduru i uvežite YIELD.SYMBOL na levu stranu upita
- ✗ promenite vrednost na desnoj strani upita u "KaB" (uključujući navodnike), zatim kliknite [OK]
- ✗ pritisnite na [OK] u prozoru Vector Object Display Controls



Polje SYMBOL sadrži string vrednosti.

Postavite string vrednost pod navodnike.



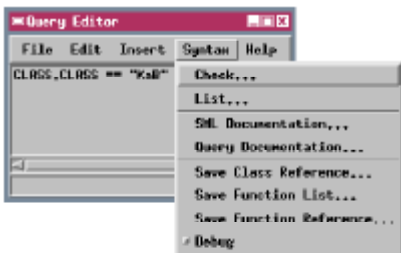
Selektovani poligoni tla koji pripadaju klasi KaB.

Provera sintakse formiranog upita

KORACI

- ✗ otvorite prozore Vector Object Display Controls i Query Editor
- ✗ rucno promenite levu stranu postojećeg upita u CLASS.CLASS (sva velika slova)

Pravila koja se ticu korišćenja velikih slova i navodnika za string vrednosti su primeri recenice TNTmips jezika. Sintaksa upita je proverena automatski kada se pritisne [OK] za izvršenje upita i kada se izabere Close iz File menija da bi se zatvorio Query Editor. (Ukoliko upit sadrži sintaksicku grešku, kada pokušate da zatvorite Query Editor, prozor ce ostati otvoren i prikazace se poruka o grešci).

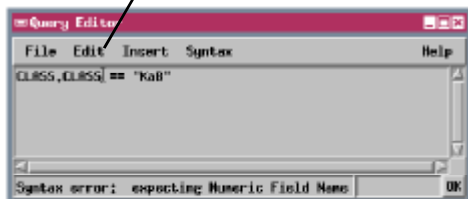


- ✗ izaberite Check iz menija Syntax
- ✗ oznaciti poruku o grešci i kliknuti OK u Message prozoru
- ✗ Promenite levu stranu upita u CLASS.Class
- ✗ izaberite Check iz menija Syntax
- ✗ opaziti "Syntax OK" poruku

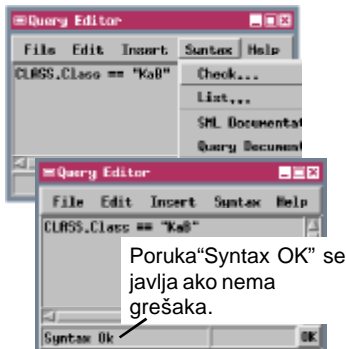
Moguca je provera sintakse upita pre njegovog izvršenja biranjem Check sa Syntax menija u Query Editor-u. Tako je moguće naci propuštene zagrade, pogrešno napisane upite ili ostale simbole ili reference ka nepostojecim poljima u bazi. Proces pocinje proverom na pocetku upita. Ukoliko nema sintaksickih gresaka, na dnu ce se pojaviti poruka Query Editor-

a "Syntax OK". Ako je otkrivena sintaksicka greška kursor ce biti postavljen na kraju prve komponente upita, koju racunar nije mogao da interpretira i poruka o grešci ce biti u dnu Query Editor-a. U ovom primeru provera sintakse je otkrila da ne postoji polje u bazi sa imenom CLASS u tabeli CLASS, tako da je kursor postavljen na kraju CLASS.CLASS upisa. Nakon ispravke sintaksicke greške, moguće je koristiti Syntax opciju za dalju proveru u preostalom tekstu.

U ovom primeru kursor je postavljen na kraju nepravilnog TABLE.FIELD unosa



U složenijim upitima, sintaksicka greška može biti rezultat ako nešto nedostaje posle tacke gde je kursor postavljen, kao što su zatvorene zagrade ili 'end' izjava pocetka / kraja petlje.



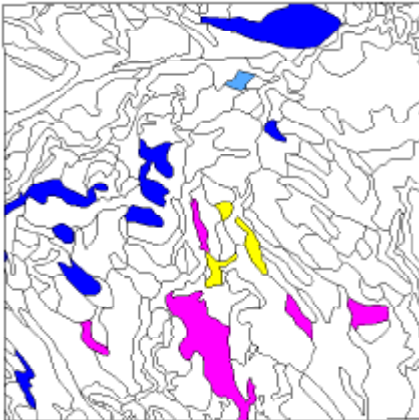
Poruka "Syntax OK" se javlja ako nema grešaka.

Izracunavanje u upitima

Vrednosti na desnoj strani upita mogu takode biti obezbedene poljem iz baze ili izracunavanjem polja iz baze. Kalkulacija u upitu moze da koristi standardne aritmetičke operacije: sabiranje (+), oduzimanje (-), množenje (*) i deljenje (/). Moguce je uneti simbol operacije iz Insert operatora. Opcija Insert/Function omogućava pristup trigonometrijskim i ostalim matematičkim funkcijama koje takode mogu biti upotrebljene u upitima.

Primer upita korišćen u ovoj vežbi izdvaja poligone zemljišta za koje je potencijalni rod ovsa tacno 5 bušela po jutru veci od roda pšenice.

Do sada ste verovatno primetili da su poslednji upit korišćen za odredeni objekat i tip elementa automatski postavljeni sa objektima i otvoreni sledeci put kada se izabere ista By Query opcija. Ukoliko se želi da se postavi nekoliko upita za isti objekat za buducu upotrebu, moguće je koristiti Save i Save As opcije u Query Editor meniju. Ove opcije omogućavaju snimanje upita koji su trenutno prikazani u Query Editor prozoru kao tekst fajl sa .QRY ekstenzijom ili kao tekst objekat u Project File-u. Da bi se opozvao snimljeni upit, koristiti opciju Open u File meniju.



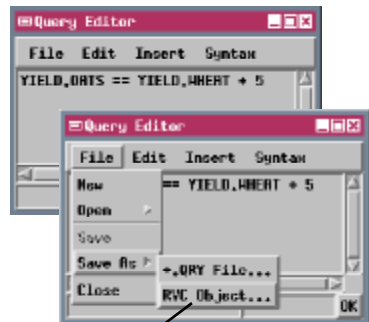
KORACI

- ✗ izabrati New iz File menija
- ✗ koristiti Insert proceduru i/ili rucno kreirati sledeci upit :

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

- ✗ izabrati Save As iz File menija, zatim *.QRY iz menija
- ✗ izabrati File Selection prozor da bi se dalo ime novom fajlu koji sadrži upit
- ✗ pritisnuti [OK] u Query Editor prozoru
- ✗ kliknuti [OK] u Vector Object Display Controls prozoru

Obicno ce upit raditi samo sa specificnim objektom zbog odnosa sa jedinstvenim poljem baza podataka. Ako imate niz objekata sa identicnim formatima baza podataka, ili se upit odnosi samo na polja u standardnim tabelam a koje je napravio TNTmips, onda možete koristiti isti upit za bilo koji objekat.



Izaberite RVC Object da sacuvate upit kao tekstualni objekat u Project File-u.

Spajanje upita

KORACI

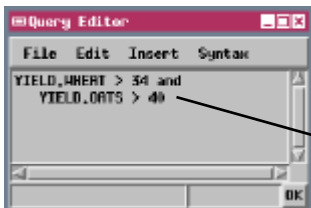
- otvoriti prozore Vector Object Display Controls i Query Editor
- izaberite New iz menija File
- koristeći Insert proceduru i/ili rucno kreirati sledeci upit:

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

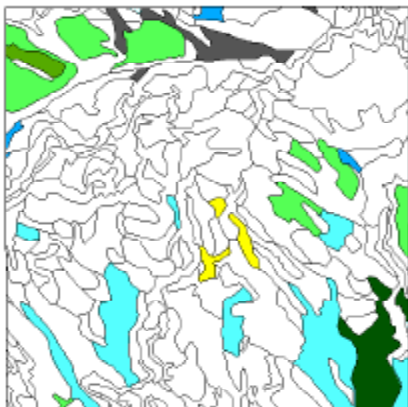
- kliknuti OK u prozoru Query Editor i u Vector Object Display Controls prozoru
- Ponoviti gornje korake ali, u upitu, "or" ubaciti umesto "and"

Svaki od upita korišćenih u prethodnim lekcijama upotrebljavaju jednu selekciju poredjenja za izbor poligona za prikaz. U mnogim slucajevima bice potrebno izdvajanje elemenata koristeći kombinaciju više kriterijuma. Serija selekcija poredjenja u upitu mora biti povezana jedna sa drugom jednim ili više logickih operatora kao što su "and", "or" i "not". Ovi operatori moraju biti uneti malim slovima. Moguce je logicke operatore uneti u upit korišćenjem Insert Operator procedure.

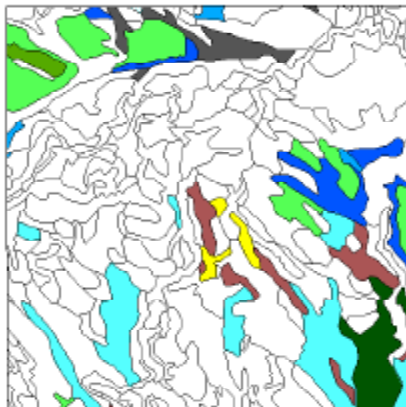
Kada su dva poredjenja povezana logickim "and" operatorom, oba poredjenja moraju biti istinita u cilju izrade citavog istinitog upita i selektovanja elementa. Kada su dva poredjenja povezana logickim "or" operatorom upit je istinit ako je jedan ili drugi upit istinit. Elementi koji ispunjavaju bilo koji kriterijum su selektovani.



Dugacak upit može da se nastavi u sledecoj liniji, kako bi pojasnili da su obe linije deo jedne izjave možete uvuci drugu liniju.



Poligoni za koje je potencijalni prinos pšenice veci od 34 bušela po jutru, i potencijalni prinos ovsa veci od 40 bušela po jutru.



Poligoni za koje je bio potencijalni prinos pšenice veci od 34 bušela po jutru, ili potencijalni prinos ovsa veci od 40 bušela po jutru.

Korišćenje “not equal to” operatora

Vecina tipova zemljišta u Crow Butte podrcjuju imaju veci potencijal roda ovsa nego pšenice, ali pšenica obicno donosi veci prihod kada se žetva prod. Pretpostavimo da je cena žetve po bušelu 3.25\$ za ovas i 4.00\$ za pšenicu. Ovaj primer upita je korišćen da bi izdvojio tip zemljišta u kome je potencijal roda po jutru ovsa veci ili jednak rodu pšenice (potencijalni prinos u bušelima po jutru pomnoženo sa cenom po bušelu).

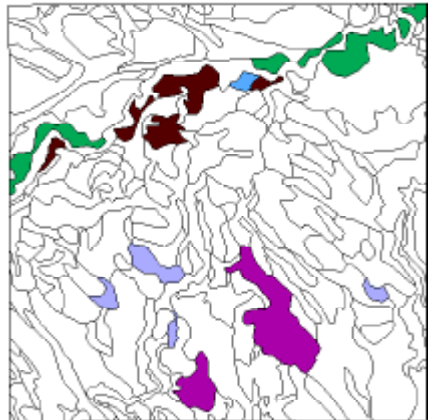
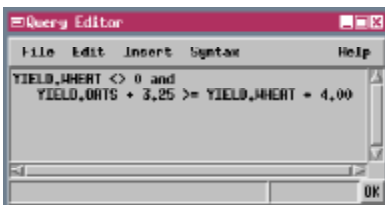
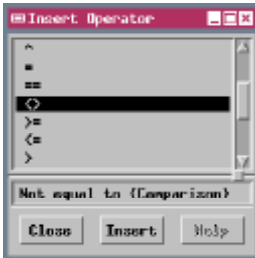
Ovaj upit je komplikovan zbog cinjenice da je potencijalni rod za zemljište koje ne može biti kultivisano oznaceno 0 i takav tip zemljišta bi zadovoljio poredenje u drugoj liniji upita. Prva linija upita isključuje tip zemljišta sa nultom žetvom i ilustruje korišćenje “not equal to” operatora (<> ili !=). Ovaj iskaz selektuje poligone za koje potencijalni rod pšenice nije jednak 0. Samo takvi poligoni se koriste za poredenje cena u drugoj liniji.

KORACI

- ⌘ otvorite prozore Vector Object Display Controls i Query Editor
- ⌘ izaberite New iz menija File
- ⌘ koristeći Insert proceduru i/ili rucno kreirati sledeci upit :

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

- ⌘ kliknite [OK] u Query Editor prozoru
- ⌘ kliknite [OK] u Vector Object Display Controls prozoru



Poligoni za koje bi prinos ovsa doneo vecu cenu po jutru od pšenice, pod pretpostavkom da se omoguci potencijalni prinos i stanje cena po bušelu.

Korišćenje komentara i promenljivih

KORACI

- ✎ otvorite prozore Vector Object Display Controls i Query Editor
- ✎ izaberite New iz menija File
- ✎ koristeći Insert proceduru i/ili ručno kreirajte sledeći upit:

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

# select polygons based on crop price
```

- ✎ kliknite [OK] u Query Editor prozoru
- ✎ pritisnite na [OK] u Vector Object Display Controls prozoru

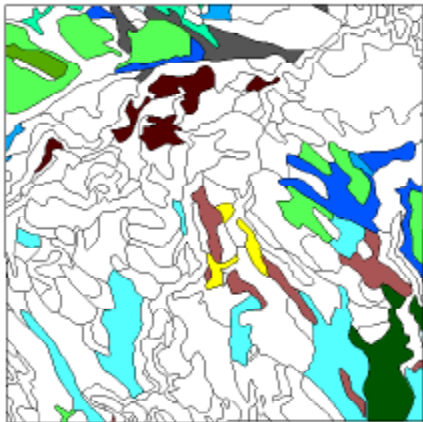
Moguće je poboljšati citljivost i kasniju iskorišćenost upita korišćenjem komentara. Komentar počinje “#” simbolom i može biti u posebnoj liniji ili na kraju upita. Komentari mogu biti u okviru upita da bi objasnili pojedine iskaze, ili to mogu biti uvodni komentari koji objašnjavaju nameru korišćenja upita i kom objektu su namenjeni.

TNTmips upitni proces takođe omogućava imenovanje i označavanje vrednosti promenljivih korišćenih u upitu. Ovaj primer upita selektuje poligone zemljišta koji prekoracuju traženu potencijalnu cenu roda po jutru bilo za ovas ili pšenicu. Prva linija upita je oznaka iskaza koji definiše numeričku promenljivu nazvanu “dolari”

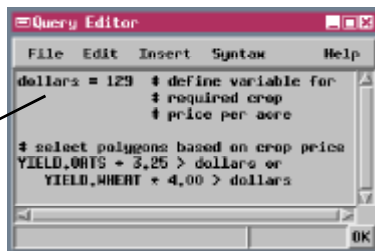
koja treba da sadrži očekivanu cenu i daje joj vrednost 129. Simbol “=” je korišćen da oznaci vrednost promenljive.

Promenljive su korisne kada se ista vrednost koristi više puta u upitu. Ukoliko je potrebno pokrenuti ovaj upit ponovo sa drugom cenom, potrebno je promeniti samo vrednost oznacenu kod promenljive “dolari”.

Ukoliko se ovaj upit piše bez promenljive, potrebno je promeniti dve numeričke vrednosti u oznacenom iskazu.



Imena promenljivih moraju uvek biti napisana malim slovima, i moraju da počinju slovom. Ime promenljive ne može da bude isto kao komanda upita ili tabela baze podataka ili kao ime polja.



Upotreba String promenljive

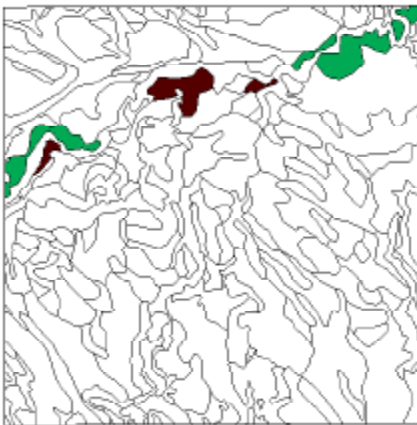
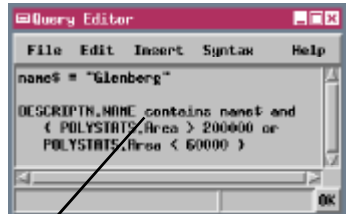
Moguće je takođe definisati promenljivu koja sadrži string vrednost. Naziv koji se dodeljuje string promenljivoj mora se završavati sa \$. Upit u ovoj vežbi definiše string promenljivu name\$, kojom je oznacena vrednost "Glenberg". Upit selektuje niz poligona koji pripadaju seriji Glenberg tla koji uključuju dva tipa tla u Crow Butte području. Umesto korišćenja dve klase simbola za selektovanje poligona ovaj upit ima prednost zbog činjenice da polje NAME u DESCRIPTN tabeli omogućava opis tla koja počinju imenom "Glenberg" za obe klase. Upit koristi "sadržajni" ("contains") operator za selekciju elemenata pomoću specifičnih string karaktera koji odgovaraju svim ili delovima string polja. U ovom slučaju karakter stringa koji treba da odgovara ("Glenberg") je postavljen u name\$ promenljivu. Poligoni koji odgovaraju ovoj selekciji su prikazani na osnovu svog rasprostranjenja (u kvadratnim metrima), a podaci o tome su uskladišteni u polju Area koje se nalazi u standardnoj POLYSTATS tabeli (POLYSTATS tabela je prisutna samo u slučaju kada su izračunati standardni atributi za vektorski objekat).

KORACI

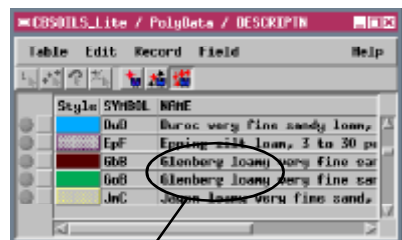
- ✦ otvorite prozore Vector Object Display Controls i Query Editor
- ✦ izaberite New iz menija File
- ✦ koristeći Insert proceduru i/ili ručno kreirati sledeći upit :

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

- ✦ pritisnite na [OK] u Query Editor prozoru
- ✦ kliknite [OK] u Vector Object Display Controls prozoru



Operator "contains" selektuje poligone za koje se karakteri string-a u name\$ promenljivoj poklapaju sa bilo kojim delom DESCRIPTN.NAME string polja.



Tekstualni string "Glenberg" je uključen u DESCRIPTN.NAME polje za oba tipa zemljišta koja pripadaju Glenberg serijama.

Korišćenje logickog “not” operatora

KORACI

- ☞ otvorite prozore Vector Object Display Controls i Query Editor
- ☞ izaberite New iz menija File
- ☞ koristeći Insert proceduru i/ili rucno kreirajte sledeci upit :

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

- ☞ kliknite na [OK] u Query Editor prozoru
- ☞ pritisnite [OK] u Vector Object Display Controls prozoru

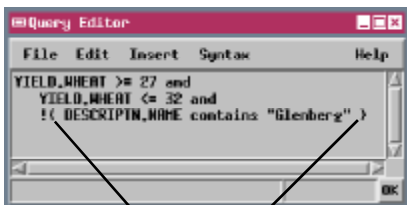
Brojna zemljišta u Crow Butte imaju potencijalni pšenicni rod koji se može porediti sa dva Glenberg zemljišta (sa 27 i 32 bušela po jutru). Upit u ovoj vežbi izdvaja sve tipove zemljišta u okviru ovog raspona roda osim vrednosti zemljišta “Glenberg”.

Prve dve linije upita selektuju poligone u kojima je potencijalni pšenicni rod između oznacenih vrednosti. Treća linija upita pocinje logickim “not” operatorom (!), koji okreće rezultat promenljive,

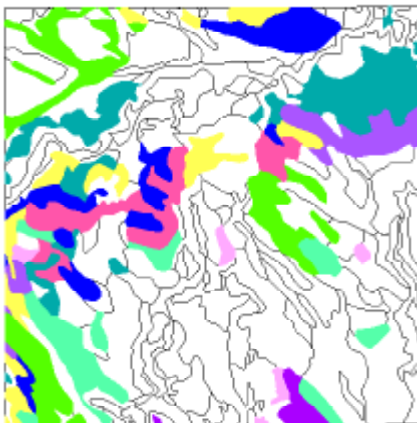
operatora ili izraza koji se nalaze u nastavku linije posle operatora. U ovom slucaju izraz u zagradi ce

selektovati poligone koji pripadaju “Glenberg” zemljištima. “Not” operator (!) okreće ovaj rezultat i saglasno prethodnim zahtevima selektuje sve poligone koji odgovaraju prinosu roda od 27 - 32 bušela, izuzev poligona Glenberg tla.

“Not” operator je posebno koristan kada postoji veliki broj vrednosti koji želimo selektovati i manje, lakše izdvojene grupe vrednosti koje ne želimo da izdvojimo.



Operator ‘not’ preokrece element koji ga sledi (ukljucujuci promenljive ili druge operatore). Ako želite da ‘not’ operator primenite za ceo izraz (kao u ovom primeru), izraz mora biti unutar zagrada.



Proveravanje prikacenih zapisa

Upit formiran u ovoj vežbi se razlikuje od svih do sada korišćenih u prethodnim vežbama. Umesto formiranja upita na bazi vrednosti određenih atributa, ovaj upit je zasnovan na broju zapisa u određenoj tabeli baze nekog poligona. Upit koristi SetNum() grupu funkcija, koji vraća broj zapisa nekog skupu. U ovom slučaju skup je obezbeđen bilo kojim izrazom koji ima formu Table[*] i obuhvata sve zapise prikacene za neki element u imenovanoj tabeli.

Ovaj upit izdvaja sve poligone različitih klasa zemljišta, koji nemaju prikacen ni jedan zapis u tabeli YIELD. Suprotan upit [SetNum(YIELD[*])>1] ce izdvojiti sve vektorske elemente koji imaju privezani za sebe više od jednog zapisa. Moguce je koristiti upite sa ovom formom da se izdvoje poligoni za koje nije povezan ni jedan atribut ili koji imaju zakacene dodatne zapise. U ovom primeru, svi poligoni koji nemaju prikacen zapis u YIELD tabeli pripadaju klasi WATER. Oni predstavljaju mala jezera i ribnjake i tako nemaju potencijalni rod.

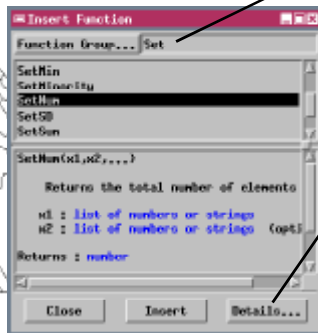
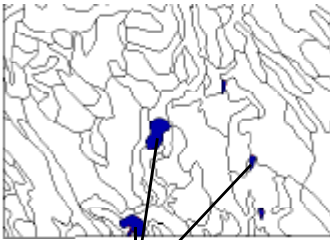
KORACI

- ✗ otvorite prozore Vector Object Display Controls i Query Editor
- ✗ izaberite New iz menija File
- ✗ upotrebite Insert proceduru i/ili rucno unesite sledeci upit:

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

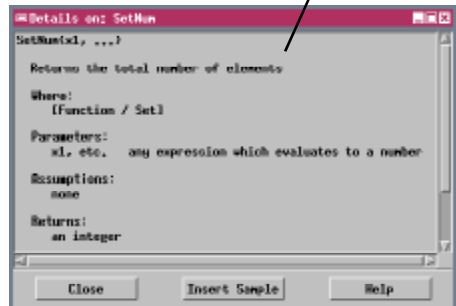
- ✗ pritisnite [OK] u Query Editor prozoru
- ✗ kliknite na [OK] u prozoru Vector Object Display Controls

Upotrebite Function Group opciju u Insert Function prozoru da selektujete koju grupu funkcija želite da izlistate.



Pritisnite [Details] na Insert Function prozoru da vidite dalje informacije o trenutno selektovanoj funkciji i primer kako se koristi.

Svi poligoni bez zakacenih zapisa u tabeli YIELD baze podataka pripadaju klasi WATER. Takođe možete selektovati nezakacene elemente desnim klikom na ikonu za odgovarajucu tabelu u prozorima Group Controls ili Layout Controls i biranjem Select All Unattached Elements iz padajuceg menija.



Selekcija pomocu više prikacenih zapisa

KORACI

- ☞ otvorite prozore Vector Object Display Controls i Query Editor
- ☞ izaberite New iz menija File
- ☞ koristite Insert proceduru i/ili rucni unos sledeceg upita :

```
"WB" in
```

- ☞ kliknite na [OK] u Query Editor prozoru
- ☞ pritisnite [OK] u Vector Object Display Controls prozoru

Poligoni klasa tla koji imaju atmosferilijama izloženi bedrok kao deo svog tipicnog profila.

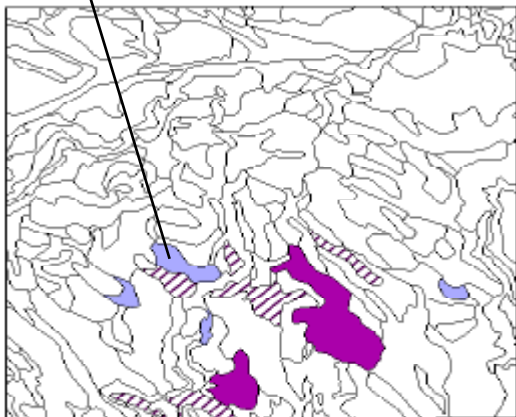


Tabela LAYER za poligone klasa tla u CBSOILSQ objektu sadrži informacije o razlicitim slojevima u tipicnom profilu tla za svaku klasu. Tu se nalazi izdvojeni zapis za svaki sloj u profilu, a tako je sa svakim poligonom povezano više zapisa. Izdvajanje elemenata na bazi selekcije više atributa izmedu razlicitih zapisa zahteva posebnu upitnu sintaksu.

U ovoj vežbi, na primer, želimo da izdvojimo tipove zemljišta koji sadrže stenovito tlo, a izloženi su uticaju vremenskih prilika u svakom delu profila tla. Ovaj atribut je oznacen stringom "WB" u polju tekstura. Ukoliko pokušate korišćenje uobicajenih upita `LAYER.texture == "WB"`, videcete da ni jedan poligon nije selektovan, cak iako pojedini delovi tla imaju stenovito zemljište u nižim delovima profila. Ovaj tip upita kontroliše samo *prvi* zapis u tabeli za svaki poligon, što je u ovom slucaju zapis sloja broj 1, koji sadrži attribute samo za najviše delove tla. Sledeci zapisi za dublje delove tla se zanemaruju.

Za upit teksturnog polja *svih* prikacenih zapisa, moramo koristiti izraz `LAYER[*].texture`, koji vraca spisak sadržaja teksturnih polja za svaki zapis povezan sa aktuelnim poligonom. Zatim je potrebno odrediti da li bilo koji od clanova skupa odgovara željenom atributu "WB". Najlakši nacin da se ovo uradi je korišćenje reci "in" kao logickog operatora. Upit je

istinit ukoliko promenljiva koja prethodi operatoru tacno odgovara bilo kom elementu u skupu prikazanom izrazom koji sledi operator. Ova konstrukcija može biti korišćena kako sa string, tako i sa numerickim poljima.

Pronalaženje poligona “ostrva”

Vektorski poligon koji se u potpunosti nalazi u okviru većeg poligona se naziva **island poligon** (“ostrvo poligon”). S obzirom da island poligon ima često različite atribute od poligona koji ga okružuje, procesi izmene topologije ili atributa dodeljenih vektorskom objektu moraju tretirati međusobni odnos objekta sa island poligonom.

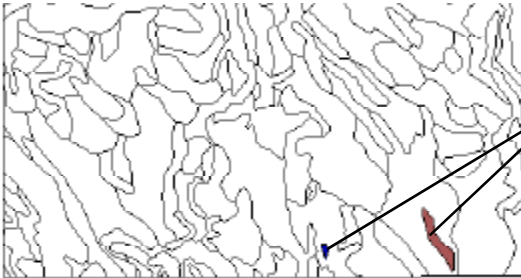
Interna tabela poligona obuhvata nekoliko polja koja sadrže informacije vezane sa island poligon. Možete napraviti upite kojim ćete selektovati island poligon ili poligone koji sadrže ostrvo. Polje `Internal.Inside` sadrži, ukoliko postoje, broj elemenata okruženog poligona. Svi okruženi poligoni će imati ne-nula vrednosti u ovom polju. Prvi upit stoga selektuje sve island poligone. `NumIslands` polje prikazuje broj ostrva koja se nalaze u svakom poligonu. Drugi upit u ovoj vežbi selektuje poligone koji imaju `NumIsland` vrednost veću od 0, omogućavajući tako selekciju svih poligona koji sadrže ostrva.

KORACI

- ✍ otvorite prozore `Vector Object Display Controls` i `Query Editor`
- ✍ izaberite `New` iz menija `File`
- ✍ koristeći `Insert` proceduru i/ili ručno kreirajte sledeći upit:


```
Internal.Inside > 0
```
- ✍ pritisnite `[OK]` u `Query Editor` prozoru
- ✍ kliknite na `[OK]` u `Vector Object Display Controls` prozoru
- ✍ ponoviti gornji potupak koristeći sledeći upit :

```
Internal.NumIslands >
```



Island poligoni selektovani prvim upitom. Svako ostrvo pripada različitoj klasi tla u odnosu na okolni poligon.

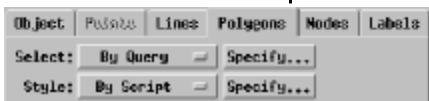


Dva poligona u `CBISOILS_LITE` vektorskom objektu uključuju island poligone, pa su zato selektovani drugim upitom.

Stilizovanje skriptom

KORACI

- ✦ otvorite Vector Object Display Controls prozor
- ✦ u Lines tabbed panelu, uključite Draw Lines Before Polygons dugme

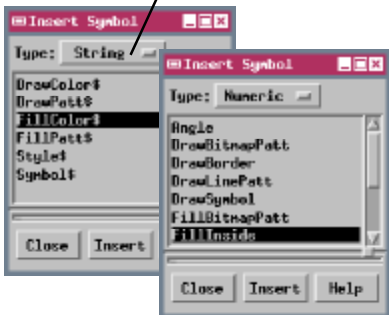


- ✦ na Polygons tabbed panelu, podesite opciju Style u By Script, i kliknite na susedno dugme (Specify)
- ✦ upotrebite Insert Symbol proceduru (Numericku i String opciju) da kreirate sledeci skript stil:

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

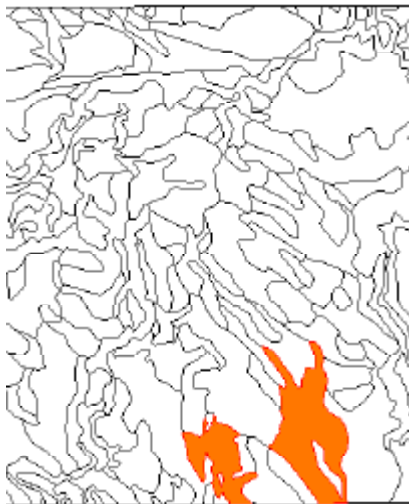
- ✦ kliknite na [OK] u Query Editor prozoru
- ✦ pritisnite na [OK] u Vector Object Display Controls prozoru

Koristite opciono dugme na Insert Symbol prozoru da pristupite razlicitim listama simbola: konstante ili razliciti tipovi promenljivih (uključujući između ostalih numericke, string i vektorske tipove).



Opcija Style by Script omogućava da se odrede karakteristike prikaza za svaki selektovani element na osnovu njihovih atributa. Kao uvod u ovu opciju, ova vežba zadržava prethodno selektovane upite, ali koristi skup skriptova za odredbu novih parametara prikaza svih selektovanih poligona. (Normalno bi koristili opciju All Same da postignete ovo).

Kada postavljate stilove uz pomoc skripta, Insert meni u Query Editor prozoru omogućava pristup dopunskim promenljivima koje su upotrebljene u skupu karakteristika prikaza. FillInside i DrawBorder su numericke promenljive koje oznacavaju vrednosti od 1 do popune selektovanog poligona i crtaju granicu oko njih. FillColor\$ i DrawColor\$ su string promenljive koje služe da oznace boju poligona i njegovih granica. Vrednosti koje su im dodeljene (pod navodnicima) mogu biti ime boje (crvena, zelena, plava, crna, bela, žuta, narandžasta, braon, cijan, ljubicasta ili siva) ili skup RGB vrednosti (svaka od 0 do 100%).



Složeni stil skriptovi

U ovoj vežbi svi poligoni klasa tla su selektovani za prikaz, a stil skriptovi su korišćeni da definišu dva različita skupa parametara za prikaz poligona celokupnog poligonalnog područja.

Kada se žele odrediti alternativne akcije u upitu ili stil skriptu, neophodno je koristiti "if-then-else" komande, koje eksplicitno definišu logiku zahteva. Iskaz u ovom skriptu se može prevesti kao "ako poligon ima područje veće od 200 000 kvadratnih metara, **onda** ga popuni žutom, **inace** ga popuni bitmap šrafurom (BitmapPatt4)". Kada više srodnih iskaza prati "then" ili "or" komanda, grupa iskaza mora biti zatvorena unutar begin/end komande. Izostavljanje begin/end komande nakon "else" u ovom upitu neće izazvati sintaksnu grešku. Međutim, u tom slučaju ce samo prvi iskaz biti primenjen kao alternativa "then" akciji; preostali iskazi ce biti interpretirani i primenjeni globalno na **sve** poligone (kao prve dve linije upita), prelazni parametri stila su definisani ranije.

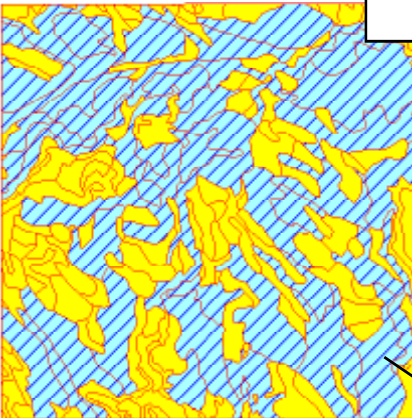
KORACI

- ✗ otvorite Vector Object Display Controls prozor
- ✗ u Polygons tabbed panelu, podesite Select opciju na All, ostavite Style opciju podešenu na By Script i kliknite na susjedno Specify dugme
- ✗ izaberite New iz menija File
- ✗ koristeci Insert proceduru i/ili rucno kreirajte sledeci upit :
- ✗ kliknite na [OK] u Query Editor prozoru
- ✗ pritisnite [OK] u Vector Object Display Controls prozoru

```

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

```



Da bi koristili bitmap-u popunjavanja šrafurom u skriptu, uzorak dodeljen FillPatt\$ promenljivoj mora da se nalazi u User Set-u definisanih uzoraka za objekat. Da bi nacrtali uzorak, promenljive FillInside i FillBitmapPatt obe moraju da budu podešene na 1. Videti brošuru *Getting Started: Creating and Using Styles* radi informacija o kreiranju i popunjavanju šrafurom i formiranju drugih stilova.

Poligoni površine od 200,000 kvadratnih metara ili veće se popunjavaju sa prugastim uzorkom definisanim u BitmapPatt2. Manji poligoni se popunjavaju žutom bojom.

Nalaženje poligona umetanjem tacke

KORACI

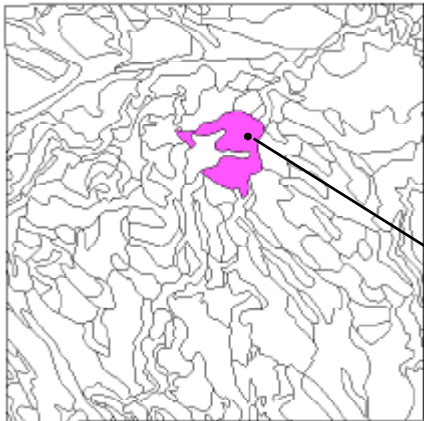
- ✍ otvorite Vector Object Display Controls prozor u Polygons tabbed panelu, podesite Style opciju na By Attribute
- ✍ podesite Select opciju na By Query i kliknite na susjedno dugme (Specify)
- ✍ izaberite New iz menija File u Query Editor prozoru
- ✍ koristeći Insert proceduru i/ili ručno kreirajte upit koji sledi:
- ✍ pritisnite [OK] u Query Editor prozoru
- ✍ kliknite na [OK] u Vector Object Display Controls prozoru

Upitni proces uključuje i brojne prostorne funkcije koje mogu biti upotrebene za selektovanje elemenata. FindClosestPoly() funkcija (u vektorskoj funkcijskoj grupi) vraća broj elementa poligona koji okružuje tacku sa određenim x i y koordinatama. Željeni parametar funkcije uključuje u upit vektorski objekat, njegove x i y koordinatne vrednosti i broj georeferenciranog podobjekta koji se može koristiti prilikom obrade vrednosti koordinata. Broj georeferenciranog podobjekta je obezbeden funkcijom GetLastUsedGeorefObject() (u georeferentnoj funkcijskoj grupi), koji se u iskazu dodeljuje promenljivoj "georef". Definisana pomenljiva "Vect" (u Vector listi u prozoru Insert symbol) se koristi da oznaci tekuci vektorski objekat. Prve dve linije upita definišu promenljive koje sadrže x i y vrednosti koordinate tacke.

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

FindClosestPoly() funkcija je korišćena prilikom određivanja iskaza i ona cuva brojeve elemenata u o k v i r u poligona, u

vidu numerickih promenljivih (c_poly u ovom primeru). Krajnji iskaz upita poredi brojeve elemenata svakog poligona u objektu (Internal.ElemNum) sa brojem koji je smesten u c_poly da bi se pronašao poligon koji ce biti prikazan.



Lokacija tacke određena koordinatama xvar (dužina) i yvar (širina). Vrednosti korišćene u upitu moraju biti u istom koordinatnom sistemu kao i određeni georeferencirani podobjekat. Širina / Dužina koordinata mora biti izražena u decimalnim stepenima.

Upit za susedne poligone: Logicki

Selekcija upita takode može koristiti topografske informacije povezane sa vektorskim objektima. Svaka linija u vektorskom objektu ima pocetni i krajnji nod, koji definišu levu i desnu stranu linije. Svaki poligon je napravljen od specifičnih linijskih elemenata i interna tabela linija uključuje polja sa brojevima elemenata poligona koji leže sa leve, odnosno desne strane linije. GetVectorPolyAdjancedPolyList() funkcija (u Vector listi u prozoru Insert Function) koristi ove informacije za odredbu koji je poligon susedni tekucem poligonu. Ova funkcija može biti korišćena u upitu za selektovanje poligona koji su susedni poligonima specifične klase.

Kao primer, treba razmotriti upit za CBSOILS_LITE vektor objekta koji selektuje poligone koji pripadaju klasi zemljišta "SrD", a koji su susedni sa klasom poligona "Sa". Da bi se razmatralo okruženje, poligoni moraju da dele glavnu granicnu liniju, a ne samo glavni nod. Opšti način korišćenja ovakvog upita je sledeci:

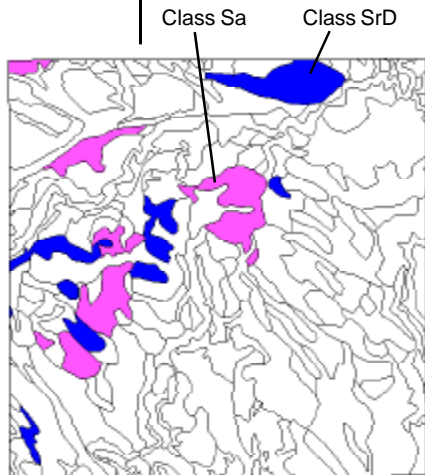
- 1) Definirati klasu koja se selektuje.
- 2) Ukoliko poligon pripada klasi koja je selektovana, treba uraditi sledeće korake (testirati susedne poligone), a u protivnom ih zaobici.
- 3) Preci na listu poligona koji su susedni samo sa osmatranim poligonom.
- 4) Proveriti pridruženu klasu svakog susednog poligona. Ukoliko bilo koja od njih odgovara definisanoj susednoj klasi, selektovati taj poligon. Ukoliko ne odgovara, treba ga odbaciti.

Sintaksa za ovaj upit je prikazana i objašnjena na sledećoj strani.

KORACI

- ✗ otvorite prozore Vector Object Display Controls i Query Editor
- ✗ izaberite New iz menija File
- ✗ koristeci Insert proceduru i/ili rucno kreirajte upit prikazan na sledećoj strani:
- ✗ kliknite na [OK] u Query Editor prozoru
- ✗ pritisnite [OK] u Vector Object Display Controls prozoru

Vektorski CBSOILS_LITE objekat sa selektovanim svim poligonima klasa SrD i Sa (radi poredenja sa ilustracijom na sledećoj strani).



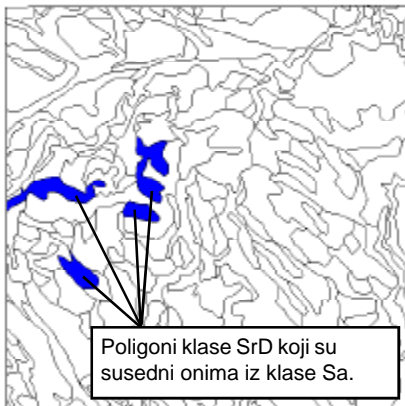
Upit za susedne poligone: Sintaksicki

```

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

```

1. Dodeljuje broj tekućeg susednog poligona (odreden na osnovu svog indeksa u redu) promenljivoj "poynum"-u.
2. Pocinje procesna petlja koja proverava klasu susednih poligona.
3. Definiše jednodimenzionalni niz "polylist" da bi zadržao listu elemenata brojeva poligona koji su susedni sa tekucim poligonom. Inicira red velicine od 10 elemenata (velicina se automatski menja putem funkcije u sledecoj izjavi).
4. Definiše promenljivu "numpolys" koja je dodeljena vrednosti jednako broju susednih poligona koji su u okruženju tekuceg poligona. Ova vrednost se vraca funkcijom GetVectorPolyAdjacentPolyList, koja takode nalazi elemente brojeva susednih poligona i cuva ih u "polylist" nizu. Prethodno zadata promenljiva "Vect" se koristi da ukaže na tekuci vektorski objekat.
5. Pocinje procesna petlja koja ispituje klase svih poligona u redu. Petlja se pokrece jednom za svaki element u redu, pocevši od prve pozicije (indeks reda 1) i nastavlja do krajnje pozicije (koja je odredena tekucom vrednošću promenljive "numpolys"). U svakoj petlji promenljivoj "i" se dodeljuje vrednost indeksa tekuceg reda koja ce se koristiti u sledecoj izjavi.
6. Dodeljuje broj tekućeg susednog poligona (odreden na osnovu svog indeksa u redu) promenljivoj "poynum".
7. Upoređuje klasu tekuceg susednog poligona sa navedenom susednom klasom. Specifikacija baze podataka je u formi Object.database[record#.table.field]. (Znak \$ na kraju specifikacije baze podataka ukazuje da je zadato polje tipa string). Ukoliko se klase poklapaju, onda...
8. Izjava "vratiti se na 1" eksplicitno govori da je upit istinit za poligon koji zadovoljava gore pomenute uslove, tako da ce poligon biti selektovan za prikaz.
9. Kraj reda procesne petlje.
10. Ako svi susedni poligoni padnu na gornjem testu klasa, izjava "vratiti se na 0" govori da je upit netacan.
11. Kraj petlje okruženja poligona.
12. Govori da je upit netacan za poligon koji ne udovoljava pocetnoj uslovnoj selekciji klasa iz izjave broj 1.





Popis granica u TIGER podacima

Vektorski objekti koji su uveženi iz US Census Bureau TIGER / Line dokumenata, sacinjeni su od linijskih segmenata koje predstavljaju prirodne ili tehnogene fizicke karakteristike, kao i od državnih granica. Granice popisa pojaseva (i ekvivalentne Block Numbering Areas ili BNA) i njihovi sastavni popisni blokovi, se obicno poklapaju sa ostalim karakteristikama i nisu eksplicitno identifikovane od strane Census feature Class Code (CFCC) kao osnovne karakteristike karte.

Popis blokova granicnih linija može biti selektovan za prikaz ili izdvajanje korišćenjem upita koji markira linije za koje brojevi blokova sa desne i leve strane nisu isti. Blokovi koji su podeljeni zadržavaju isti broj, ali su oznaceni razlicitim slovima u levom i desnom dodatku polja razlicitim sufiksom; drugi set iskaza u primeru upita selektuje ove granice. Na kraju, blokovi u susedstvu BNA (Block Numbering Areas) mogu imati iste brojeve, pa krajnji iskaz izdvaja linije razlicitog BNA.

KORACI

- ☞ izbrišite sloj koji je korišćen u prethodnoj vežbi 
- ☞ kliknite na ikonu Add Vector i sa padajućeg menija selektujte Add Vector Layer 
- ☞ selektujte ALMEDA objekat iz TIGER Project File-a
- ☞ u prozoru Vector Object Display Controls, na panelu Lines podesite Style Option na All Same i opciju Select na By Query
- ☞ otvorite Query Editor prozor i izaberite New iz File menija
- ☞ koristeći Insert proceduru i/ili rucno kreirajte upit koji je prikazan niže
- ☞ kliknite na [OK] u Query Editor prozoru
- ☞ pritisnite [OK] u Vector

Object Display Controls prozoru

```
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 )
```



Racunanje u polju iz višestrukih zapisa

KORACI

- ☞ smanjite Display prozor iz prethodne vežbe
- ☞ izaberite Edit / Attribute bazu podataka iz glavnog TNTmips menija
- ☞ pronadite i selektujte objekat CBSOILS_LITE iz Project File-a CBSOILSQ
- ☞ uključite dugme Polygon u prozoru Select
- ☞ u prozoru Database Editor, kliknite desnim dugmetom miša na Class table kvadrat, i selektujte Edit Definition u prozoru definition za Class table pritisnite na ikonu Add Field
- ☞ u polju liste, oznacite ime po default-u za novo ime i ukucajte ClassArea
- ☞ na panelu Field info, selektujte Computed iz Field Type menija i kliknite na Edit Expression
- ☞ unesite upit koji je pokazan dole, u prozor Query Editor



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

- ☞ unesite upit koji je pokazan dole, u prozor Query Editor
- ☞ unesite 12 u Width tekst boks i 2 u Places tekst boks
- ☞ kliknite na [OK] u prozoru definition
- ☞ kliknite dva puta na boks Class table da bi ga otvorili

Skriptovi takode mogu biti korišćeni prilikom definisanja vrednosti za izracunavanje u polju tabele baze. U mnogo slucajeva ovi skriptovi zahtevaju kreiranje jednostavne aritmeticke relacije više polja u istom zapisu. Zadatak u ovoj vežbi je malo složeniji: treba izvršiti izracunavanje u polja poligona tabele Class za CBSOILS_LITE koja prikazuju ukupnu površinu podrucja za svaki tip zemljišta.

Površine poligona su smeštene u POLYSTATS tabeli za pojedinačne poligone, ali mi kreiramo izracunavanje vrednosti zapisa polja iz tabeli Class, koja ima po jedan zapis za svaki tip zemljišta. Skript koji je ovde pokazan je dizajniran da sumira podrucja poligona za svaku klasu zemljišta i da vrati rezultat kao vrednost polja.

Skript definiše numericke vrednosti promenljive "sum" koji se koristi da sumira podrucja u POLYSTATS.Area polju. Ova promenljiva mora da vrati vrednost na 0.0 za svaku klasu. Promenljivoj "num" je dodeljena vrednost (za tekucu klasu tla) koja je jednaka broju prikacenih zapisa u POLYSTATS tabeli. Ova promenljiva se koristi da postavi broj ponavljanja petlje koja sumira površinu podrucja.

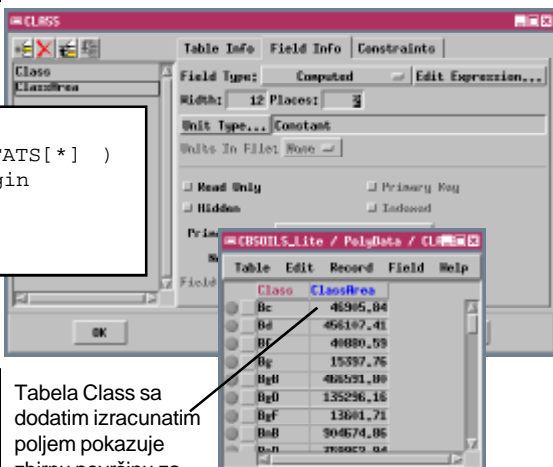


Tabela Class sa dodatim izracunatim poljem pokazuje zbirnu površinu za svaki tip tla.

- ☞ izaberite Table / Close da bi zatvorili Class table.

Polja sa String Izrazom

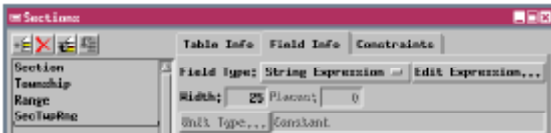
Polja sa string izrazom su poseban tip izracunatih tabela u bazi. Najjednostavnije korišćenje string izraza je kopiranje sadržaja string polja u drugu vezanu tabelu u okviru tekuće tabele. Izraz u tom slučaju je jednostavno odgovarajuća TABLE.FIELD referenca. Takođe je moguće koristiti string izraz za spajanje sadržaja nekoliko string polja u jedno polje. Na primer, tabela zvana NAME može imati odvojena polja za ime i prezime. Da bi se oni spojili potrebno je iskoristiti operator "+". Izraz

NAME.FIRST + " " + NAME.LAST

će napraviti unos u formi "John Doe". Izraz mora sadržati bilo koji karakter za odvajanje (space, zarez) kako je i prikazano u primeru. Moguće je spojiti string izraze za polja da bi se dobio tekst za dodatne informacije Data Tips.

Izraz korišćen u ovoj vezbi, koristi sprintf() funkciju koja dozvoljava da se formatira složeni string izraz mnogo lakše. Prvi argument funkcije je kontrolni string (u zagradi), za kojim slede reference string polja. Svaki "%s" unos u kontrolnom stringu označava referencu jednog string polja. Kontrolni string takođe može uključiti tekst, prazan prostor i interpunkciju.


```
sprintf( "Sec %s Twp %s Rng %s",
Sections.Section, Sections.Township,
```



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

Formatiran tekst kreiran pomocu string izraza.

KORACI

- ✗ izaberite File / Open Database u prozoru Database Editor
- ✗ pronadite i selektujte objekat CBSOILS_LITE iz Project File-a CBSOILSQ
- ✗ uključite dugme Polygon u prozoru Select
- ✗ u prozoru Database Editor, kliknite desnim dugmetom miša na boks za Sections tabelu i selektujte Edit Definition
- ✗ u prozoru definition za Sections table, kliknite na Range field u listi i kliknite na Add Field ikonu 
- ✗ u listi polja, označite ime po defaultu za novo polje i ukucajte SecTwpRng
- ✗ selektujte String Expression iz menija Field Type i kliknite na Edit Expression
- ✗ unesite upit koji je pokazan dole u prozoru Query Editor

- ✗ kliknite OK u prozoru Query Editor
- ✗ unesite 25 u tekst boks Width
- ✗ kliknite OK na definition prozoru
- ✗ kliknite dva puta na boks za Selection tabelu da bi je otvorili
- ✗ izaberite File / Close iz oba prozora Database Editor-a kada završite ovu vezbu.

Provera digitalizovanih objekata upitima

Kliknite na Select / Deselect ikonu i izaberite Select by Query da otvorite standardni prozor Query Editor gde možete uneti upit za taj tip elementa.

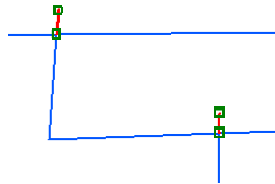


Izabrani upiti mogu takode biti od koristi kada se kreira ili prikazuje vektorski objekat koristeći TNTmips Spatial Data Editor. Složeni vektorski objekti mogu sadržati digitalizovane greške kao što su prebacene linije, nezatvoreni poligoni i iscepkani poligoni. Mnoge greške ovog tipa nisu vidljive, osim ako se ne uvelicaju, što može samo dodatno otežati rad i oduzeti vreme. Korišćenjem upita je moguće ubrzati i olakšati nalaženje ovih grešaka kao što je prikazano u primeru. Spatial Data Editor prozor obezbeđuje ikone koje omogućavaju kreiranje i primenu upita za određeni tip elementa.

PREBACENE LINIJE

Prebacene linije su kratki segmenti linija koje se nepravilno produžuju iza preseka linija. Ukoliko ste prolazili kroz proces Standard Attributes za vektorski objekat, možete koristiti selektovani upit koji je baziran na dužini linija, kako bi selektovali sve veoma male linije radi ispitivanja i moguceg uklanjanja.

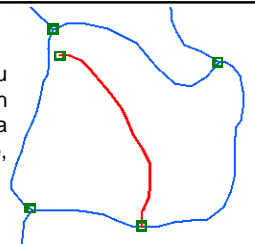
LINESTATS.Length < [vaša vrednost dužine]



NEZATVORENI POLIGONI

Ako vektorski objekat sadrži mrežu poligona, praznina između dve linije koje bi trebalo da se seku, može da ostavi jedan poligon na mestu na kome bi trebalo da postoje dva odvojena poligona. Upitom je moguće pronaći linije koje nisu zatvorene, jer imaju isti poligon sa obe strane.

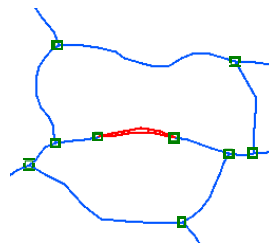
Internal.LeftPoly == Internal.RightPoly



ISCEPKANI POLIGONI

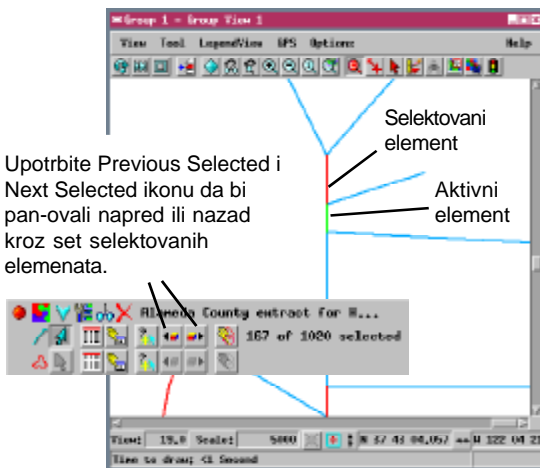
Dvostruko crtanje granica može stvoriti nepostojeće iscepkane poligone duž granice dva dodirna poligona. Iscepkani poligoni obično imaju mnogo manju površinu nego glavni poligon, i obično su izduženi (sa veoma sažetim odnosom). Koristite kombinovan upit u polju Area and CompactRatio, u tabeli POLYSTATS, kako bi selektovali iscepkane poligone.

**POLYSTATS.Area < [vaša vrednost površi] ili
POLYSTATS.CompactRatio > 3.00**






Pan-ovanje Upitom

Izvršeni upit iz Element Selection prozora u Spatial Data Editor-u ili u Spatial Data Display često selektuje više od jednog elementa. Jedan od ovih selektovanih elemenata je označen kao "aktivni" element; aktivni i selektovani elementi su označeni razlicitom bojom. Operacija uredjenja može biti primenjena i na aktivni i na selektovani element. Moguće je koristiti Previous Selected i Next Selected ikonu u Element Selection prozoru da bi se išlo napred ili nazad kroz selektovani skup elemenata, pri čemu se za svaki od njih aktivira kada na njega dodje selekcija. Prikaz se automatski repositionira (ako je neophodno) za prikaz trenutno aktivnog elementa. Ova "pan by query" karakteristika omogućava da se zadrži zum na elementu koji se proučava (i možda edituje) dok se lagano prelazi kroz selektovani skup.




Upotrebite Previous Selected i Next Selected ikonu da bi pan-ovali napred ili nazad kroz set selektovanih elemenata.

KORACI

- ☞ vratite prozore View i Group Controls
- ☞ otvorite prozor Vector Object Display Controls
- ☞ u Panelu Lines podesite opciju Select na All i kliknite OK
- ☞ otvorite meni Options u prozoru Spatial Data Display View i proverite da li je Show Scale / Position opcija uključena
- ☞ u tekst boks Scale unesite "5000", na dnu prozora View, i pritisnite <Enter>
- ☞ kliknite na ikonu Show Details na lejeru Iconrow 
- ☞ pritisnite Select ikonu za liniju 
- ☞ kliknite na ikonu Select / Deselect za linije i izaberite Select By Query sa padajućeg menija 
- ☞ u Query Editor unesite sledeci upit:

```
LINESTATS.Length<50
```

- ☞ kliknite na Apply na Select by Query prozoru
- ☞ pritisnite ikonu Next Selected 

Vežbe u ovoj brošuri Getting Started su Vas upoznale sa osnovama strukture i sintakse upita baza podataka koje se koriste u TNT mips-u, TNT edit-u i TNT view-u. Jezik upita je podset Spatial Manipulation Language (SML) koji se koristi u TNT mips-u i deli istu sintaksu. U prilog dokumentacije o upitima, koja je navedena na strani 2, možda ćete želeći da konsultujete SML dokumentaciju u TNT mips-ovom referentnom priručniku za dodatne namere vezane za programiranje, kako bi proširili svoja znanja o mogućnostima upita.

Napredni Software za Geoprostornu Analizu

Microlimages, Inc. izdaju kompletan asortiman profesionalnog softvera za naprednu vizuelizaciju geoprostornih podataka, analizu i izdavaštvo. Kontaktirajte nas ili posetite naš Web Site za detaljnije informacije o proizvodu.

TNTmips TNTmips je profesionalni sistem za potpuno integrirani GIS, analizu slike, CAD, TIN, desktop kartografiju i menadžment geoprostornih baza podataka.

TNTedit TNTedit obezbeđuje interaktivne alate za kreiranje, georeferenciranje i editovanje vektorskih, CAD, TIN, slikovnih i srodnih projektnih materijala baza podataka, u širokom spektru formata.

TNTview TNTview ima iste mocne performanse prikazivanja kao TNTmips i odlican je za one kojima ne treba tehnicko procesiranje i pripremanje materijala u TNTmips-u.

TNTAtlas TNTAtlas Vam dozvoljava da izdajete i distribuirate Vaše prostorne projektne materijale na CD-ROM-u po niskim cenama. TNTAtlas CD-ovi mogu da se koriste na bilo kojoj popularnoj platformi.

TNTserver TNTserver Vam dozvoljava da izdajete TNTAtlas na Internetu ili Vašoj internoj mreži. Surfujte kroz Geodata Atlas pomocu Web browser-a i TNTclient Java applet-a.

TNTlite TNTlite je besplatna verzija TNTmips-a za studente i profesionalce koji rade manje projekte. Možete skinuti TNTlite sa Microlimages Web Site-a ili ga možete naruciti na CD-ROM-u.

Index

Aritmeticke operacije	9	Selekcija elemenata (nastavak)	
Cuvanje upita	9	Umetanje tacke poligona	20
Insert Field prozor	16	Višestruko zakaceni zapisi	16
Insert Operator prozor	5	Selektovanje v. selekcija elem.	
Izracunavanje polja	24,25	SetNum funkcija	15
Komentari	12	Sintaksa, provera	8
Logicki operatori (i, ili, ne)	10,14	Složeni upiti	10,19
Namena izjava	12	Stilizovanje skriptom	3,18,19
Otvaranje upita	9	String polje	7
Pan-ovanje upita	27	String polje – izjava	25
Promenljive	12,13	String promenljiva	13
Selekcija elemenata		Table[*] izjava	15
Iscepkani poligoni	26	TIGER podaci	23
Neprebacene linije	26	Upiti baza podataka, definicija	3
Nezakaceni zapisi	15	Znaci za poredenje	4
Ostrva poligona	17	Jednako, ==	5
Pomocu upita	3	Nejednako, <>	11
Prebacene linije	26	Sadrži	13
Susedni poligoni	21,22	Vece od, >	4



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