

# San Diego County *Roads\_All* Utility

---

*For ArcGIS 10.0*

Author: Jeff Hamblin

Original Date: May 12, 2010

Revision Date: August 4, 2011

Contact: [jphamblin@gmail.com](mailto:jphamblin@gmail.com)

The San Diego Geographic Information Source (SanGIS) – [www.sangis.org](http://www.sangis.org) – makes available for download the "Roads\_All.shp" shapefile which contains all the roadways in the county. It is a valuable resource for creating a roads layer on maps of the region. However, there are some issues within the data (such as leading zeros in some numeric street names) that can be obstacles to quick use; and its great detail can clutter maps at small scales.

The purpose of this project is to provide a set of models that transforms the "Roads\_All.shp" shapefile into a geodatabase feature class named "Roads\_All\_Extra" with additional fields to aid in subtyping, symbolizing, labeling and geocoding addresses. An accompanying "Roads\_Extra.lyr" layer file references the new feature class and provides pre-configured symbology, labeling, and scale-dependent display.

Whenever the Roads\_All.shp file is updated (quarterly, I believe), the enhanced roads feature class can be re-generated easily without requiring any changes to the maps that reference it.

Several of the tools used require at minimum ArcEditor licensing. So this process cannot be performed with an ArcView license (though the data can be used by ArcView, of course).

This utility is not meant as a complete solution. Instead, it is meant to lay the groundwork for further refinement based on a project's requirements, and to provide examples on how to achieve them. No work has been put into detailed labeling or symbology. The default labeling engine is used, and fonts are mostly at their defaults as well. This is by design so that users can do their own polishing depending on the map scale and their needs. The display is most noticeably ugly at smaller scales out beyond 1:100,000.

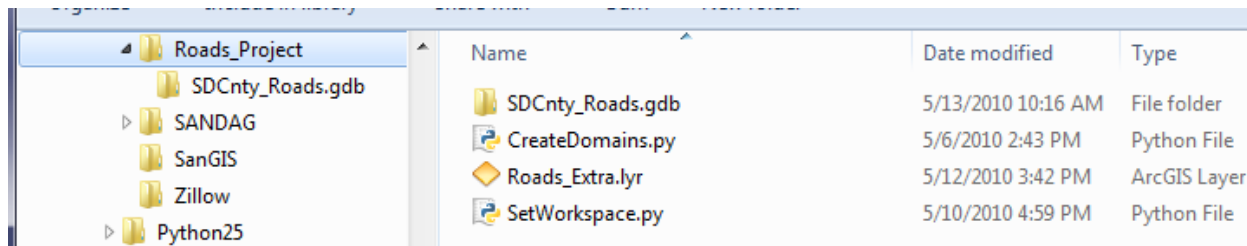
For ArcGIS 9.3 users, please download the manual and data for that version. Some of the models with calculations fail when attempting to run the models produced in v10.0 on a v9.3 installation.

The changes required to update this utility to function with v10.0 mainly involved switching from VB script to Python for field calculations.

If you have problems or suggestions, please feel free to contact me via the email address above.

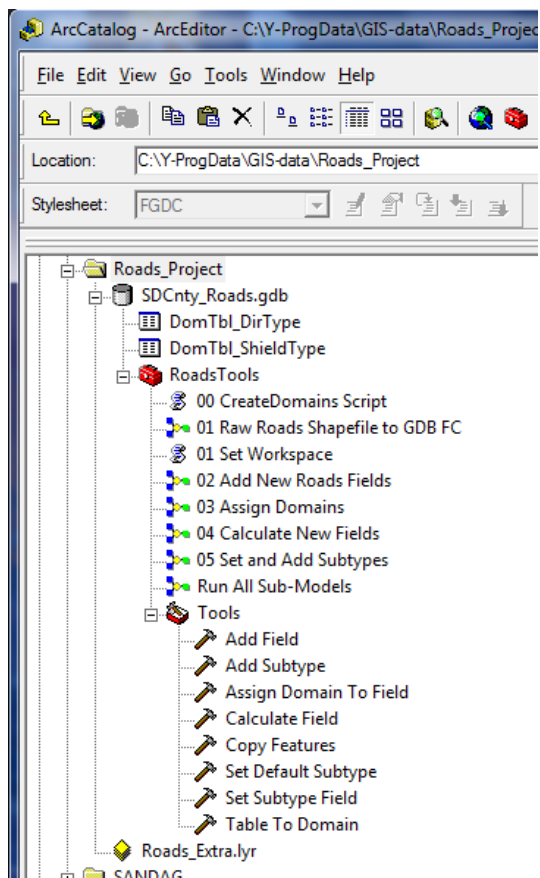
## Quick Start

Extract the downloaded zip file (from [gis.qtools.com](http://gis.qtools.com)). In Windows Explorer you should see a single geodatabase folder and three other files:



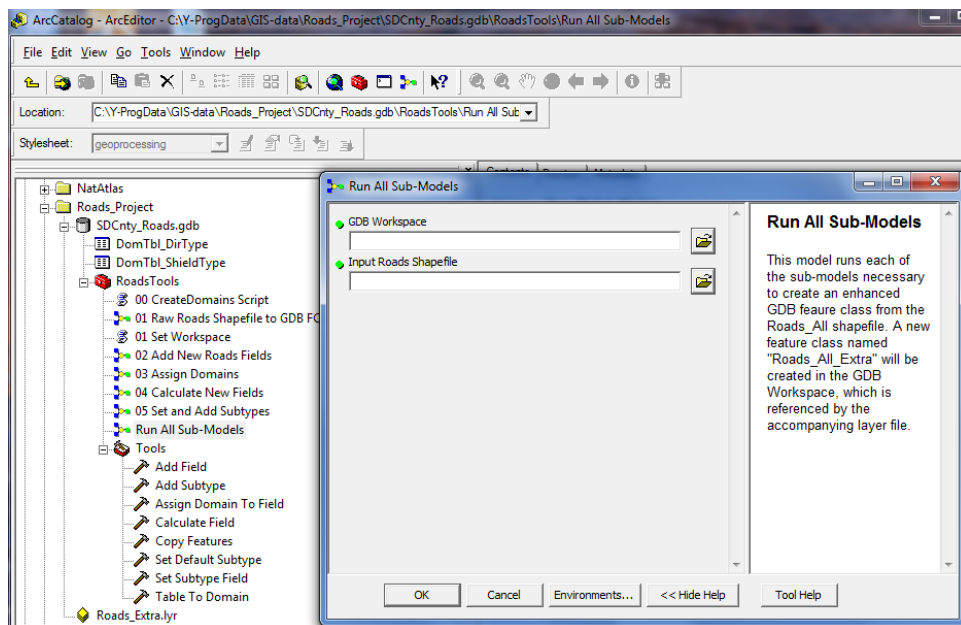
If you decide to move the GDB, be sure to move the two python script files and the “Roads\_Extra.lyr” layer file along with it.

The view of the same project folder in ArcCatalog:

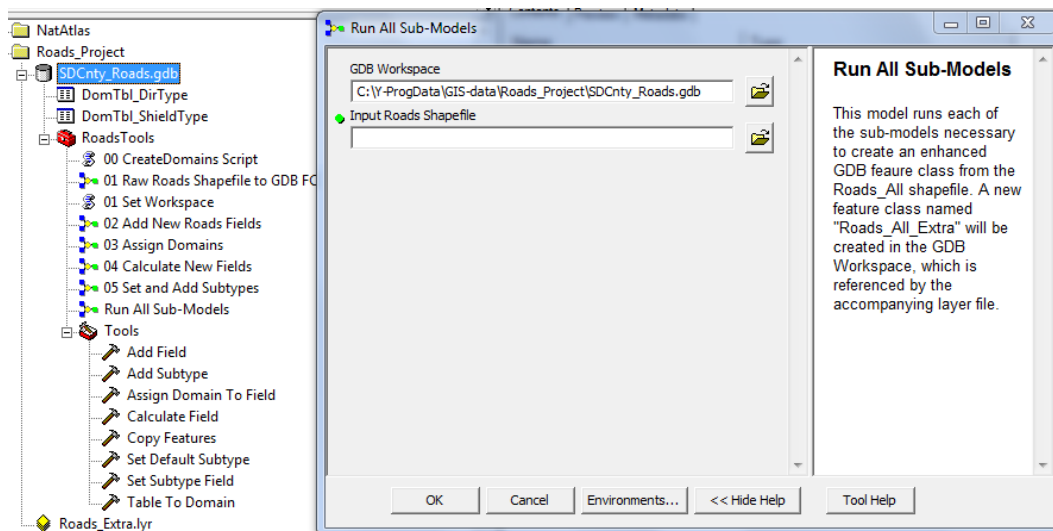


To create the new roads feature class, in ArcCatalog, run the model named “Run All Sub-Models”

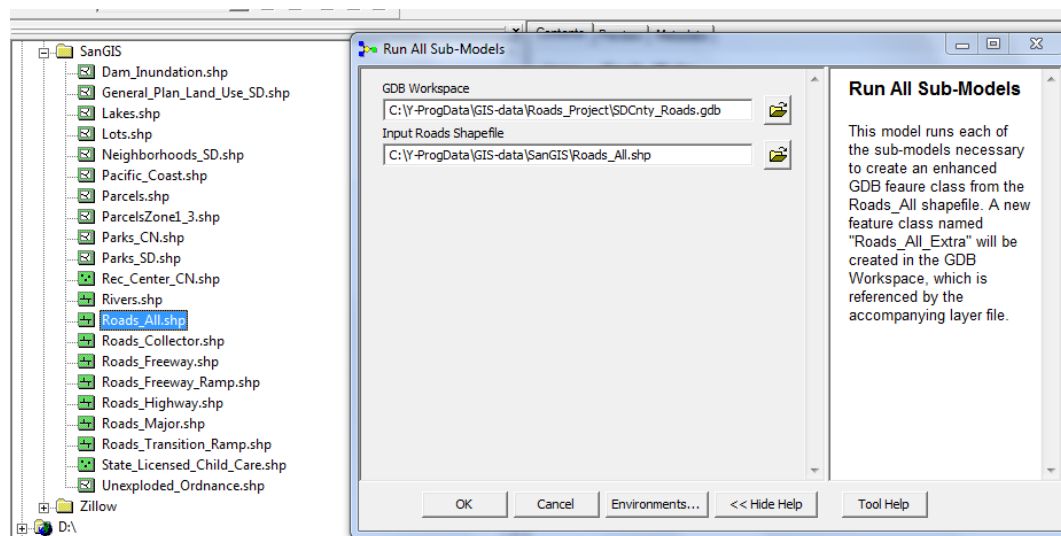
(Note: If the “Run All Sub-Models” model has a red “X” through it, you will need to edit the model in Model Builder, re-link each sub-model, and set the “Output Roads FC” parameter to “%GDB Workspace%\Roads\_All\_Extra” in the first sub-model. Save the model and run it.):



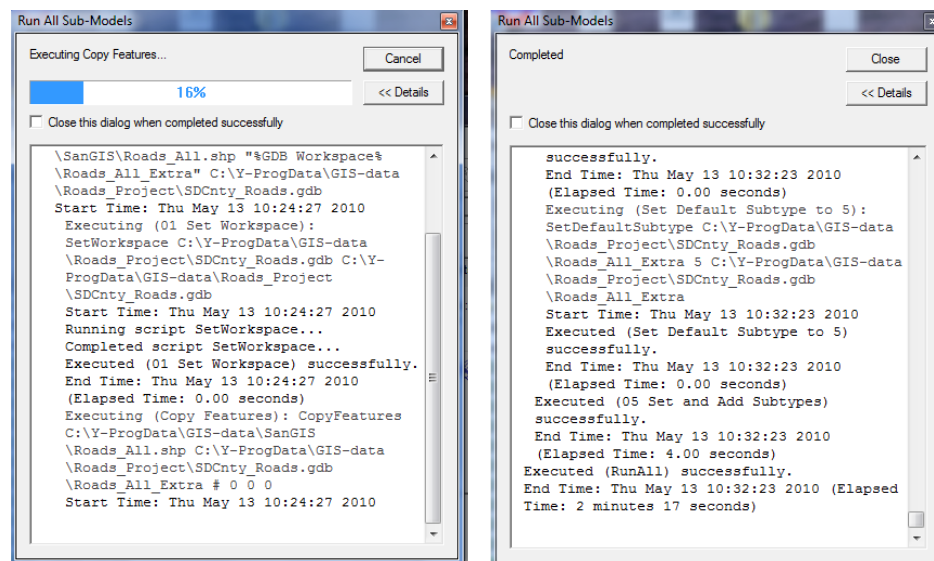
Enter the “GDB Workspace” into the model. Either drag and drop the “SDCnty\_Roads.gdb” from ArcCatalog, or browse to the same using the dialog button:



Enter the path to the “Input Roads Shapefile” into the model using the same process. This must be the “Roads\_All.shp” available from SanGIS:



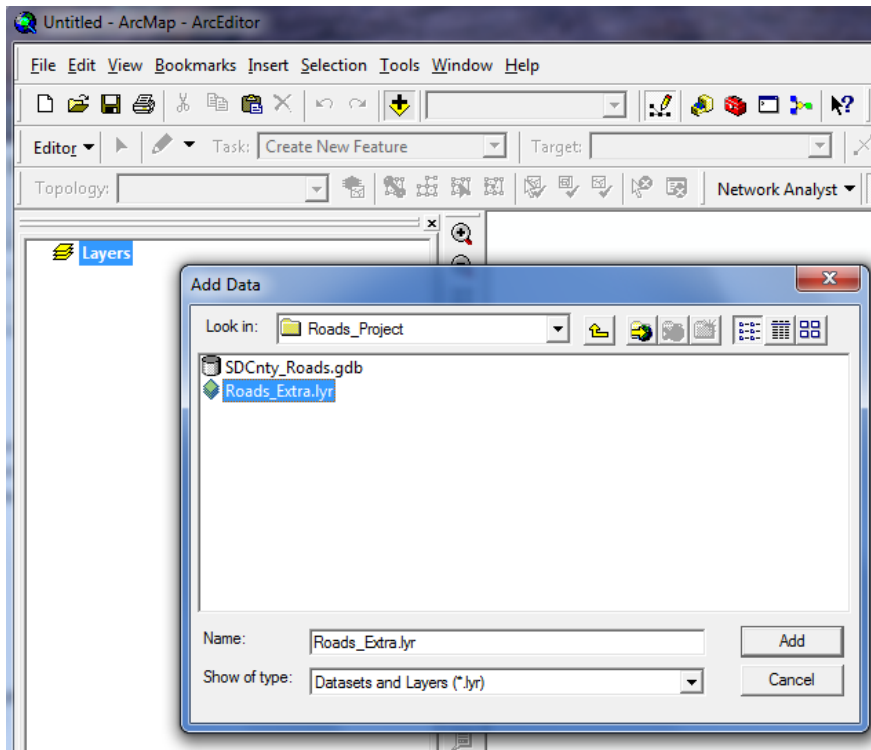
Click the “OK” button to start the process of building the new roads feature class. The usual model progress dialog will appear.



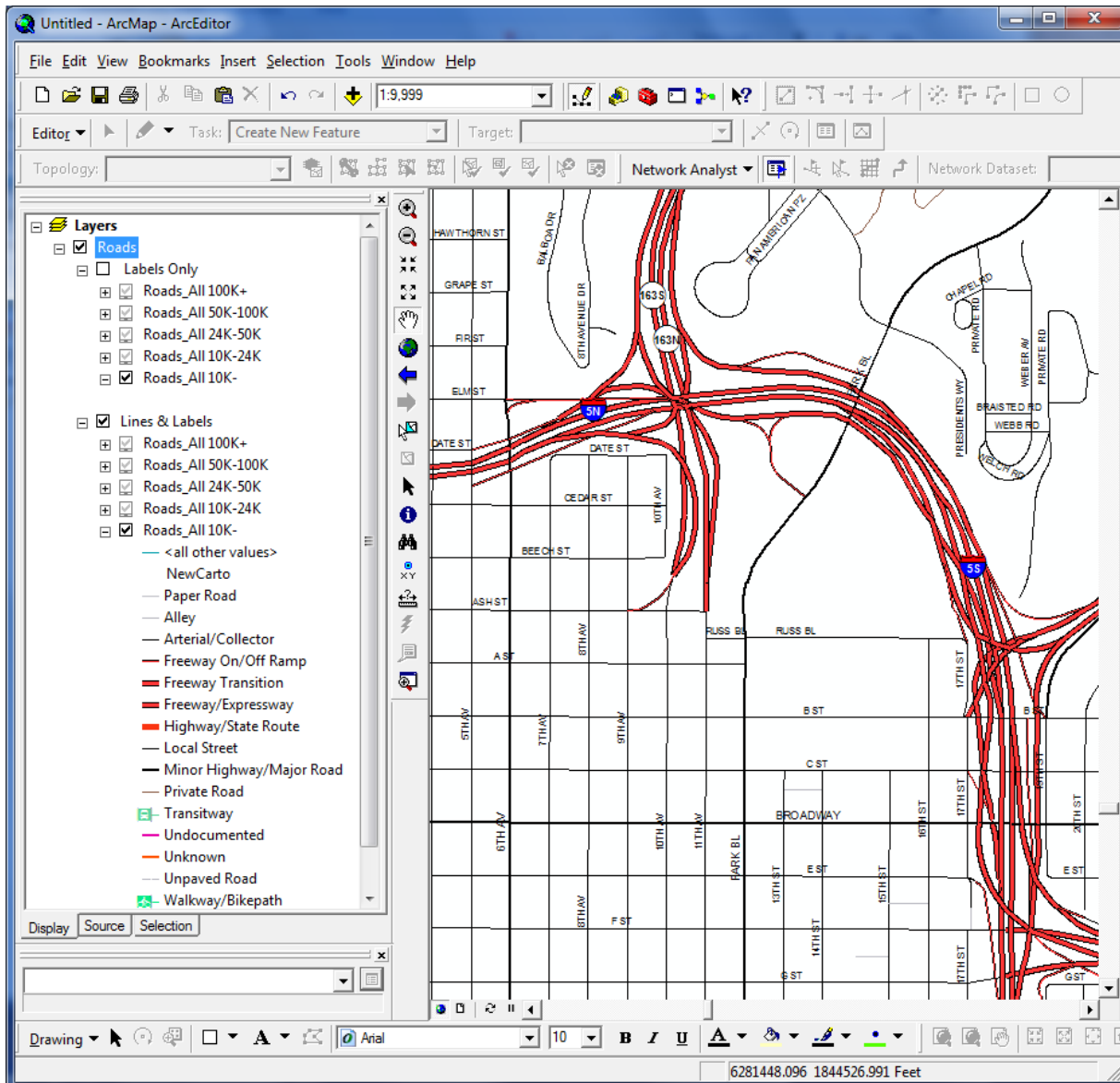
After a couple of minutes (depending on system), the process will complete and the new roads feature class will be in the GDB. (If re-running the model an existing “Roads\_All\_Extra” will be overwritten)



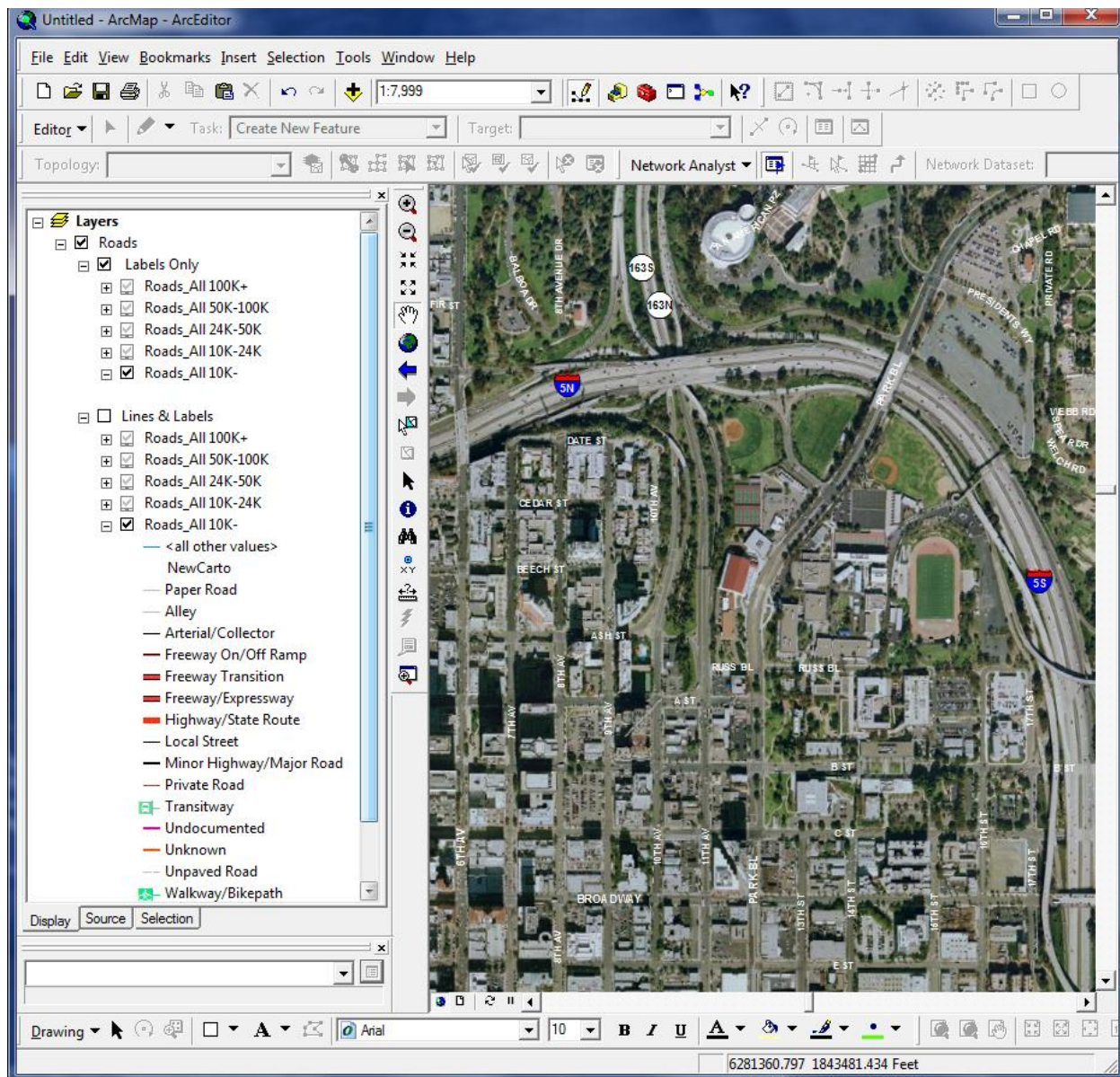
Open a new or existing map in ArcMap and add the "Roads\_Extra.lyr" to the map.



Zoom in to the downtown San Diego area and this is what you should see:



To use the “Labels Only” group layer: turn off the “Lines & Labels” group layer, add an imagery layer to the map, and turn on the “Labels Only” layer.



## Detailed Information

Key items provided with this project are:

### SDCnty\_Roads.gdb

This file geodatabase is pre-configured with 2 domains:

1. **ShieldTypes:** Applied to a new ShieldType field that determines which highway shield to use for labeling.
  - 0: None
  - 1: Interstate
  - 2: State Route
  - 3: Highway
2. **DirTypes:** Applied to a new DirType field that reflects the direction of a highway. Used mainly to drop-out one side of a highway at smaller scales and label the direction on shields at large scales.
  - 0: Unknown
  - 1: North
  - 2: East
  - 3: South
  - 4: West

Note: There is an included script tool "00 CreateDomains Script" and two non-spatial tables "DomTbl\_DirType" and "DomTbl\_ShieldType". Running the script will create the above domains on a fresh GDB if you decide to output the new roads feature class to a different GDB. It **MUST** be run **BEFORE** running the models that create the roads feature class. To do this, you must first copy the two non-spatial tables and the "RoadsTools" toolbox to the new GDB, and run the tools from that location. Also copy the "Roads\_Extra.lyr" to the folder containing the GDB.

### Models & Tools

A set of models and tools are provided. A single model named "Run All Sub-Models" performs the entire operation. When it is complete, the output is a new (or updated) feature class named "Roads\_All\_Extra", which has the following highlights:



## New Fields:

- DispName:** Text (25)  
 This field is meant to replace the original RD20NAME field. It has any leading zeroes stripped so that "04TH" becomes "4TH". This improves labeling and also eliminates errors when geocoding addresses.
- NumName:** Text (10)  
 This field is extracted from RD20NAME data. It is the numeric name for highways used to place in highway shield symbols.
- ShieldType:** Short Int  
 This field is calculated based on RD20NAME data. It assigns a value corresponding to the ShieldTypes domain for Interstate, State Route, or Highway so that the corresponding shield can be used for labeling.
- DirType:** Short Int  
 This field is calculated from RD20NAME data. It is the direction of the road, using the corresponding DirTypes domain. This is used to label highway shields with the correct direction at large scales where both sides of the highway are shown, and to drop-out one side of the highway at small scales to remove duplicate labels and doubled lines.
- NewCarto:** Short Int  
 This field is used to subtype the feature class. It is based on the SEGCLASS field. Line symbology is based on these subtypes.

Description	NewCarto	SEGCLASS
Freeway/Expressway	1	1
Highway/State Route	2	2
Minor Highway/Major Road	3	3
Arterial/Collector	4	4
Local Street	5	5
Unpaved Road	6	6
Private Road	7	7,Z
Freeway Transition	8	8
Freeway On/Off Ramp	9	9
Alley	10	A
Speed Hump	5	H
Military Base Street	5	M
Paper Street	14	P
Undocumented	11	K,Y
Walkway/Bikepath	12	W
Transitway	13	
Unknown	0	

The provided "Roads\_Extra.lyr" has the following highlights:

Two group layers:

- **Labels Only**  
This group is designed for use against an imagery background when line symbology is not necessary. It shows shields and text labels. Text labels are set to white for better display over imagery.
- **Lines & Labels**  
This group is best used over typical light-colored, symbolized backgrounds.

Each group contains five layers that provide decreasing level of detail as scale decreases, using definition queries and label classes with their own queries, expressions and scales.

All layers reference the single new roads feature class in the geodatabase.

## Appendix 1 – Field Calculations

### DispName

Copy existing RD20NAME field, removing any leading zeros .

Expression (PYTHON\_9.3):

```
getDispName(!RD20NAME!)
```

Code:

```
def getDispName(RD20NAME):
    if not RD20NAME:
        return ""
    else:
        if RD20NAME[0:1] == "0":
            return RD20NAME[1: len(RD20NAME)]
        else:
            return RD20NAME
```

### ShieldType

Calculate the value based on text found in existing RD20NAME field.

Expression (PYTHON\_9.3):

```
getShieldType(!RD20NAME!, !SEGCLASS!)
```

Code:

```
def getShieldType(RD20NAME, SEGCLASS):
    if not RD20NAME or not (SEGCLASS == "1" or SEGCLASS == "2"):
        return 0
    else:
        if RD20NAME[:2] == "I-":
            return 1
        elif RD20NAME[:3] == "SR-":
            return 2
        elif RD20NAME[:7] == "HIGHWAY":
            return 3
        elif RD20NAME[:11] == "OLD HIGHWAY":
            return 3
        elif RD20NAME[:12] == "OLDE HIGHWAY":
            return 3
        else:
            return 0
```

## NumName

Extract the numeric route number from existing RD20NAME field.

Expression (PYTHON\_9.3):

```
getNumName(!RD20NAME!)
```

Code:

```
def getNumName(RD20NAME):
    if (not RD20NAME):
        return ""
    elif (len(RD20NAME) < 2):
        return ""
    else:
        num = getHwyNumber("I-", RD20NAME)
        if num:
            return num
        num = getHwyNumber("SR-", RD20NAME)
        if num:
            return num
        num = getHwyNumber("HIGHWAY ", RD20NAME)
        if num:
            return num
        num = getHwyNumber("OLD HIGHWAY ", RD20NAME)
        if num:
            return num
        num = getHwyNumber("OLDE HIGHWAY ", RD20NAME)
        if num:
            return num
        return ""

def getHwyNumber(prefix, RD20NAME):
    prefixLen = len(prefix)
    if RD20NAME.find(prefix, 0, prefixLen) == 0:
        iPos = RD20NAME.find(" ", prefixLen)
        if iPos > -1:
            return RD20NAME[prefixLen: iPos].strip()
        else:
            return RD20NAME[prefixLen: len(RD20NAME)].strip()
    else:
        return ""
```

**DirType**

Calculate a numeric direction value based on text in the existing RD20NAME field.

Expression (PYTHON\_9.3):

```
getDirType(!RD20NAME!, !ShieldType!)
```

Code:

```
def getDirType(RD20NAME, ShieldType):  
    if not RD20NAME or ShieldType < 1:  
        return 0  
    else:  
        if RD20NAME.find(" NB") > 0:  
            return 1  
        elif RD20NAME.find(" EB") > 0:  
            return 2  
        elif RD20NAME.find(" SB") > 0:  
            return 3  
        elif RD20NAME.find(" WB") > 0:  
            return 4  
        else:  
            return 0
```

## NewCarto

Calculate a numeric value based on the value in the existing SEGCLASS field.

Expression (PYTHON\_9.3):

```
getNewCarto(!SEGCLASS!)
```

Code:

```
def getNewCarto(SEGCLASS):
    sSegVal = SEGCLASS.strip()
    try:
        return int(sSegVal)
    except ValueError:
        if not sSegVal:
            return 0
        else:
            if sSegVal == "A":
                return 10
            elif sSegVal == "H":
                return 5
            elif sSegVal == "M":
                return 5
            elif sSegVal == "P":
                return 14
            elif sSegVal == "K":
                return 11
            elif sSegVal == "Y":
                return 11
            elif sSegVal == "W":
                return 12
            elif sSegVal == "Z":
                return 7
            else:
                return 0
```

## Appendix 2 – Properties for Layers in Layer File

### Roads\_All 100K+

- Out: none, In: 100,000
- Def Q: ("NewCarto" = 1 OR "NewCarto" = 2) AND ("DirType" <> 2 AND "DirType" <> 3)

Label Classes: (don't label default)

#### Interstate

Def Q: "ShieldType" = 1  
 Field: NumName  
 Symbol: Interstate Shield, Arial, 7, Bold  
 Place: Horiz, Remove dupes  
 Scale: Same as feat

#### StateRoute

Def Q: "ShieldType" = 2  
 Field: NumName  
 Symbol: State Route Shield, Arial, 7, Bold  
 Place: Horiz, Remove dupes  
 Scale: Same as feat

#### HWY

Def Q: "ShieldType" = 3  
 Field: NumName  
 Symbol: US Route Shield, Arial, 7, Bold  
 Place: Horiz, Remove dupes  
 Scale: Same as feat

### Roads\_All 50K-100K

- Out: 99,999, In: 50,000
- Def Q: ("NewCarto" > 0 AND "NewCarto" < 4) AND ("DirType" <> 2 AND "DirType" <> 3)

Label Classes:

#### Default

Def Q: "ShieldType" = 0  
 Field Expr: [RD20PRED] & " " & [DispName] & " " & [RD20SFX]  
 Symbol: Arial, 7  
 Place: Parallel, Above:Page:0, At Best, Remove Dupes  
 Scale: Same as feat

#### Interstate

Def Q: "ShieldType" = 1  
 Field: NumName  
 Symbol: Interstate Shield, Arial, 7, Bold  
 Place: Horiz, Remove dupes  
 Scale: Same as feat

#### StateRoute

Def Q: "ShieldType" = 2  
 Field: NumName  
 Symbol: State Route Shield, Arial, 7, Bold

Place: Horiz, Remove dupes

Scale: Same as feat

#### HWY

Def Q: "ShieldType" = 3

Field: NumName

Symbol: US Route Shield, Arial, 7, Bold

Place: Horiz, Remove dupes

Scale: Same as feat

### Roads\_All 24K-50K

- Out: 49,999, In: 24,000

- Def Q: ("NewCarto" > 0 AND "NewCarto" < 5) AND ("DirType" <> 2 AND "DirType" <> 3)

Label Classes:

#### Default

Def Q: "ShieldType" = 0 AND "NewCarto" > 3

Field Expr: [RD20PRED] & " " & [DispName] & " " & [RD20SFX]

Symbol: Arial, 6

Place: Parallel, Above:Page:0, At Best, Remove Dupes

Scale: Same as feat

#### MajorRoad

Def Q: "ShieldType" = 0 AND "NewCarto" = 3

Field Expr: [RD20PRED] & " " & [DispName] & " " & [RD20SFX]

Symbol: Arial, 7

Place: Parallel, Above:Page:0, At Best, Remove Dupes

Scale: Same as feat

#### Interstate

Def Q: "ShieldType" = 1

Field: NumName

Symbol: Interstate Shield, Arial, 7, Bold

Place: Horiz, Remove dupes

Scale: Same as feat

#### StateRoute

Def Q: "ShieldType" = 2

Field: NumName

Symbol: State Route Shield, Arial, 7, Bold

Place: Horiz, Remove dupes

Scale: Same as feat

#### HWY

Def Q: "ShieldType" = 3

Field: NumName

Symbol: US Route Shield, Arial, 7, Bold

Place: Horiz, Remove dupes

Scale: Same as feat



**Roads\_All 10K-24K**

- Out: 23,999, In: 10,000

- Def Q: "NewCarto" > 0 AND "NewCarto" < 10

Label Classes:

Default

Def Q: "ShieldType" = 0 AND "NewCarto" > 4 AND "NewCarto" < 8

Field Expr: [RD20PRED] & " " & [DispName] & " " & [RD20SFX]

Symbol: Arial, 6

Place: Parallel, Above:Page:0, At Best, Remove Dupes

Scale: Out: 17,999, In: 10,000

Collector

Def Q: "ShieldType" = 0 AND "NewCarto" = 4

Field Expr: [RD20PRED] & " " & [DispName] & " " & [RD20SFX]

Symbol: Arial, 7

Place: Parallel, Above:Page:0, At Best, Remove Dupes

Scale: Same as feat

RampsTrns (do not label)

Def Q: "ShieldType" = 0 AND "NewCarto" > 7

Field Expr: [RD20PRED] & " " & [DispName] & " " & [RD20SFX]

Symbol: Arial, 7

Place: Parallel, Above:Page:0, At Best, Remove Dupes

Scale: Same as feat

MajorRoad

Def Q: "ShieldType" = 0 AND "NewCarto" = 3

Field Expr: [RD20PRED] & " " & [DispName] & " " & [RD20SFX]

Symbol: Arial, 7

Place: Parallel, Above:Page:0, At Best, Remove Dupes

Scale: Same as feat

Interstate

Def Q: "ShieldType" = 1 AND ("DirType" <> 2 AND "DirType" <> 3)

Field: NumName

Symbol: Interstate Shield, Arial, 7, Bold

Place: Horiz, Remove dupes

Scale: Same as feat

StateRoute

Def Q: "ShieldType" = 2 AND ("DirType" <> 2 AND "DirType" <> 3)

Field: NumName

Symbol: State Route Shield, Arial, 7, Bold

Place: Horiz, Remove dupes

Scale: Same as feat

HWY

Def Q: "ShieldType" = 3 AND ("DirType" <> 2 AND "DirType" <> 3)

Field: NumName

Symbol: US Route Shield, Arial, 7, Bold

Place: Horiz, Remove dupes

Scale: Same as feat

**Roads\_All 10K-**

- Out: 9,999, In: none

- Def Q: none

Label Classes:

Default

Def Q: "NewCarto" > 4 AND "NewCarto" < 8

Field Expr: [RD20PRED] & " " & [DispName] & " " & [RD20SFX]

Symbol: Arial, 6

Place: Parallel, Above:Page:0, At Best, Remove Dupes

Scale: Out: 17,999, In: 10,000

Alley (do not label)

Def Q: "NewCarto" = 10

Field Expr: [RD20PRED] & " " & [DispName] & " " & [RD20SFX]

Symbol: Arial, 7

Place: Parallel, Above:Page:0, At Best, Remove Dupes

Scale: Same as feat

Paper Road (do not label)

Def Q: "NewCarto" = 14

Field Expr: [RD20PRED] & " " & [DispName] & " " & [RD20SFX]

Symbol: Arial, 7

Place: Parallel, Above:Page:0, At Best, Remove Dupes

Scale: Same as feat

Collector

Def Q: "NewCarto" = 4

Field Expr: [RD20PRED] & " " & [DispName] & " " & [RD20SFX]

Symbol: Arial, 7

Place: Parallel, Above:Page:0, At Best, Remove Dupes

Scale: Same as feat

RampsTrns (do not label)

Def Q: "NewCarto" = 8 OR "NewCarto" = 9

Field Expr: [RD20PRED] & " " & [DispName] & " " & [RD20SFX]

Symbol: Arial, 7

Place: Parallel, Above:Page:0, At Best, Remove Dupes

Scale: Same as feat

MajorRoad

Def Q: "ShieldType" = 0 AND "NewCarto" = 3

Field Expr: [RD20PRED] & " " & [DispName] & " " & [RD20SFX]

Symbol: Arial, 7

Place: Parallel, Above:Page:0, At Best, Remove Dupes

Scale: Same as feat

Interstate

Def Q: "ShieldType" = 1 AND ("DirType" <> 2 AND "DirType" <> 3)

Field: NumName + DIR (See expression below)

Symbol: Interstate Shield, Arial, 7, Bold

Place: Horiz, Remove dupes

Scale: Same as feat

StateRoute

Def Q: "ShieldType" = 2 AND ("DirType" <> 2 AND "DirType" <> 3)

Field: NumName + DIR (See expression below)

Symbol: State Route Shield, Arial, 7, Bold

Place: Horiz, Remove dupes

Scale: Same as feat

HWY

Def Q: "ShieldType" = 3 AND ("DirType" <> 2 AND "DirType" <> 3)

Field: NumName + DIR (See expression below)

Symbol: US Route Shield, Arial, 7, Bold

Place: Horiz, Remove dupes

Scale: Same as feat

Expression for 10K- labels:

```
Function FindLabel ( [NumName], [DirType] )
  If ([DirType] = "North") Then
    FindLabel = [NumName] & "N"
  ElseIf ([DirType] = "East") Then
    FindLabel = [NumName] & "E"
  ElseIf ([DirType] = "South") Then
    FindLabel = [NumName] & "S"
  ElseIf ([DirType] = "West") Then
    FindLabel = [NumName] & "W"
  Else
    FindLabel = [NumName]
  End If
End Function
```