Task: Join CCD school information with the school attendance boundary dataset

Reference Powerpoint:

W:\Ashwini\Minnesota\_shape\_table\Demo to Students\ Matching school attendance boundaries with schools.pptx (This has brief description, flowcharts of the process comprising of 3 steps in all, and detailed help on Advanced Field Calculation using scripts).

Reference MXDs: W:\Ashwini\Minnesota\_shape\_table\Demo to Students\ ProcessDocumentation.mxd (contains my processing of MN data 2008-2009 according to method 1)

W:\Ashwini\Minnesota\_shape\_table\SABSConformToBlocks\ ProcessSABSasBlocks.mxd (contains my processing of school attendance boundary dataset to make it conform to the census blocks)

Reference word document: W:\Ashwini\Minnesota\_shape\_table\Demo to Students\Task and process.docx (This document)

Reference calculation scripts:

W:\Ashwini\Minnesota\_shape\_table\Demo to Students\field\_calculator\_scripts\\*.cal (contains all the scripts used by me during this work)

Datasets:

School attendance boundary: sdatt08.shp for MN

Census school districts for your state: tl\_2009\_unsd\_MN

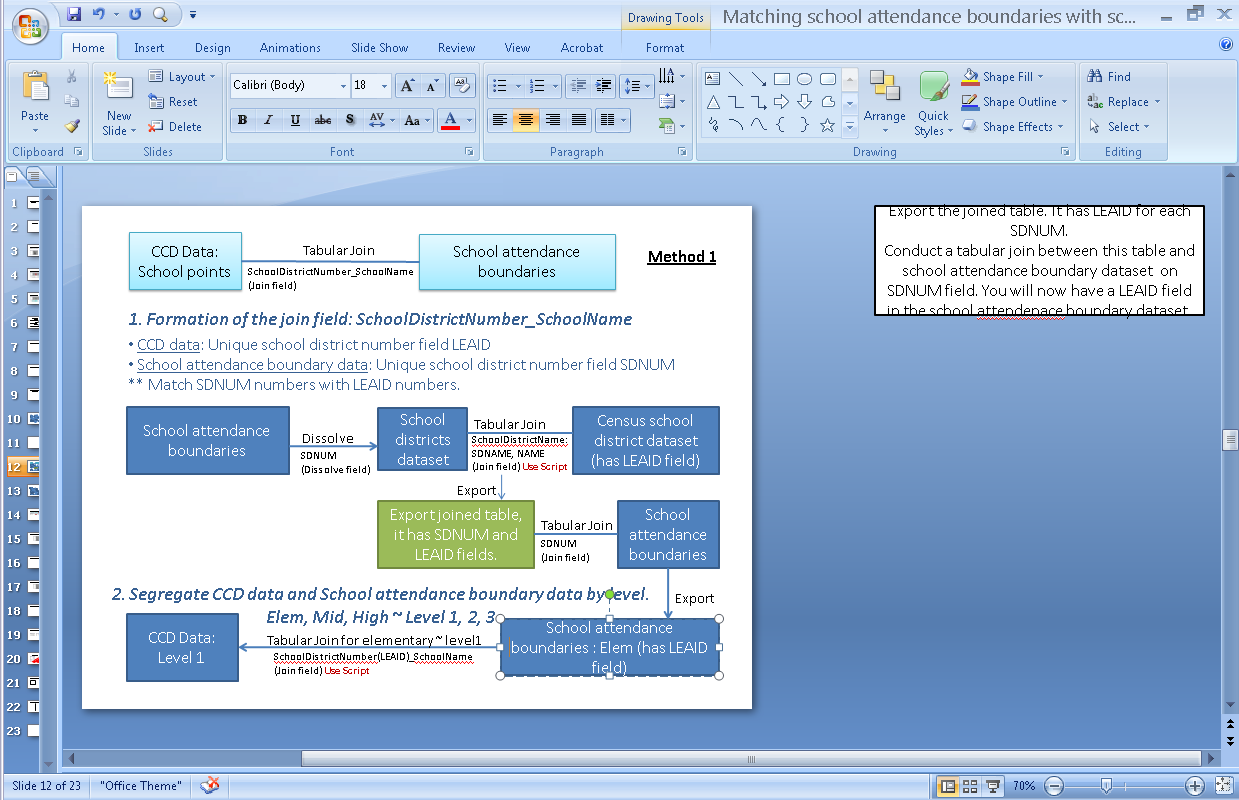
CCD data for your state: CCD200708MN (filtered from the original CCD data by state FIPS code of MN 27)

Census blocks for your state

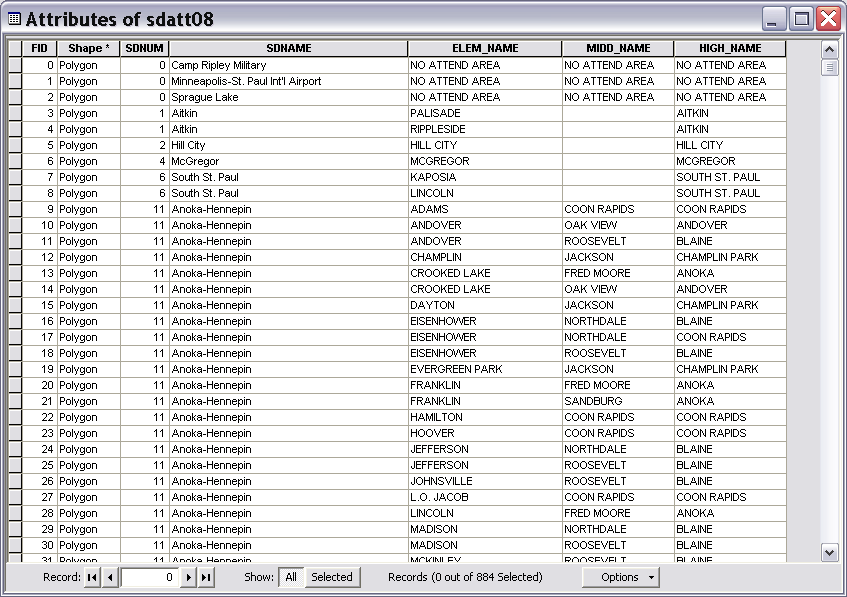
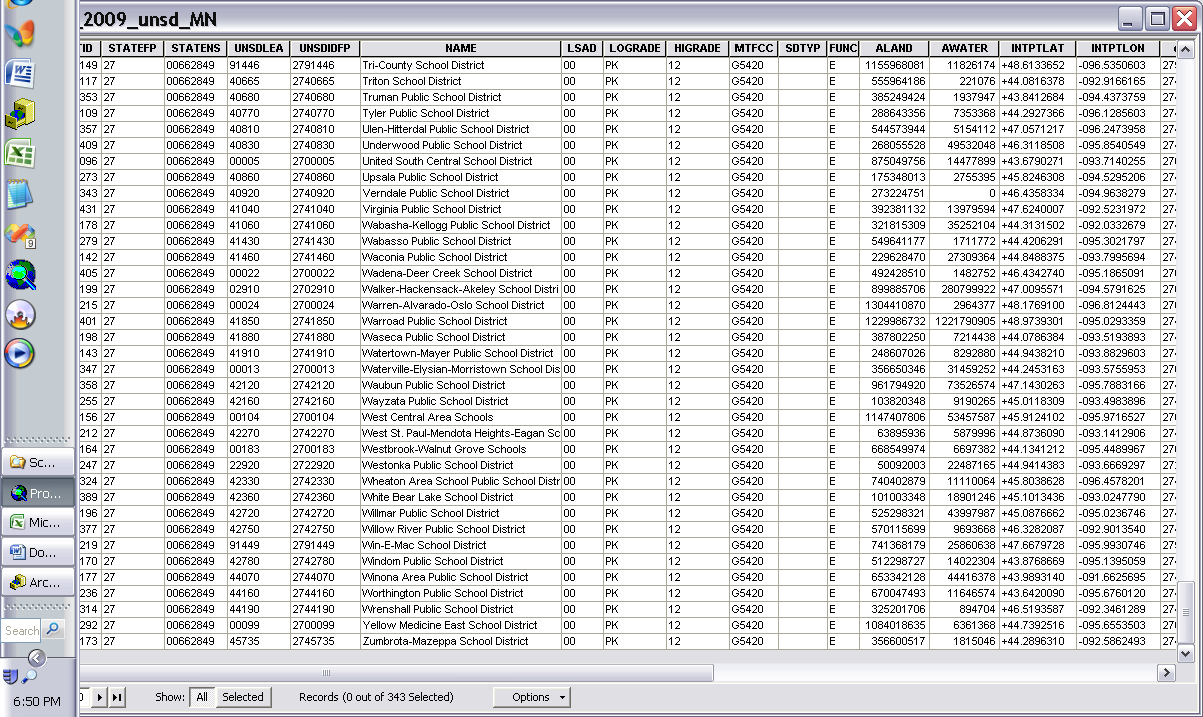
Prior to commencing work according to these instructions, please look at the power point presentation and modify school attendance boundary dataset according to the flowchart slide describing part 0 to make it conform to the census blocks.

Method 1

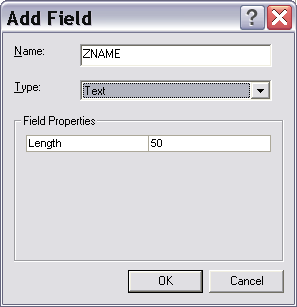
Process

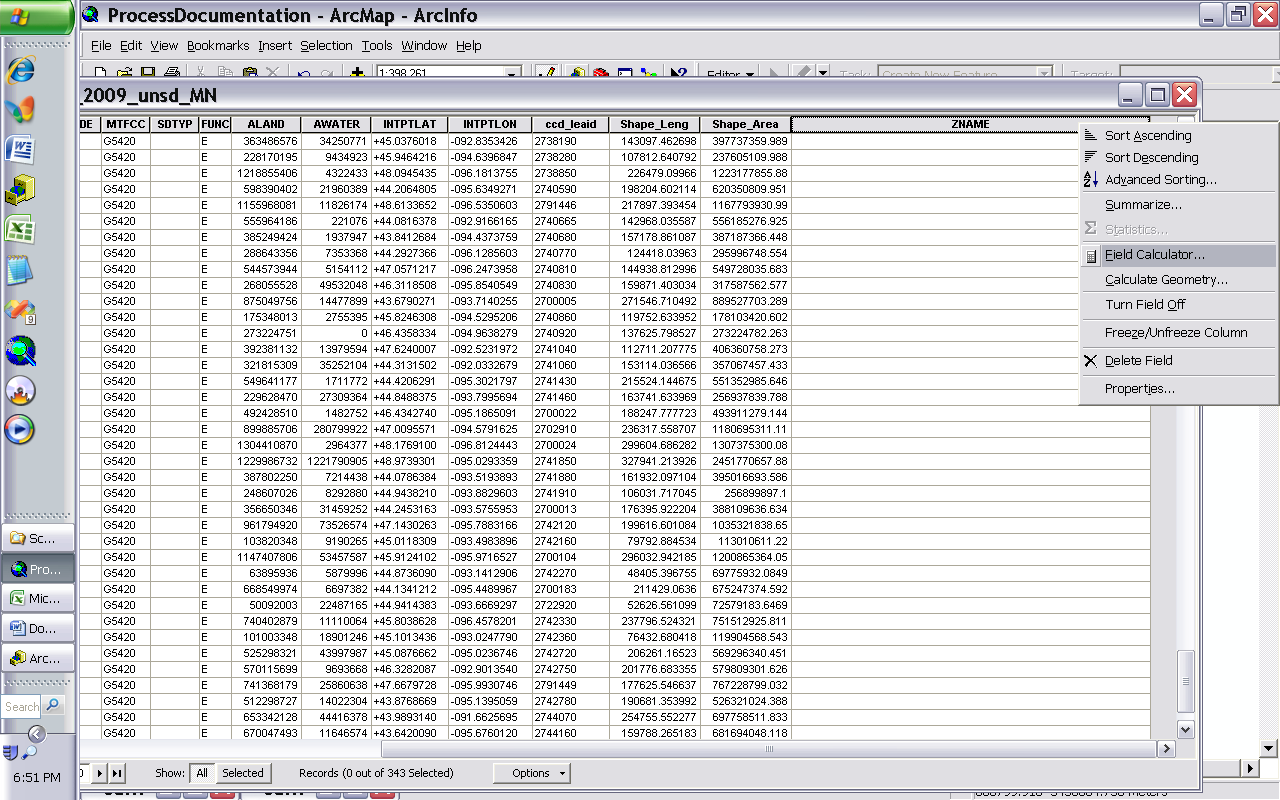


1. Dissolve the school attendance boundary dataset sdatt08 on school districts based on either sdnum and sdname fields.

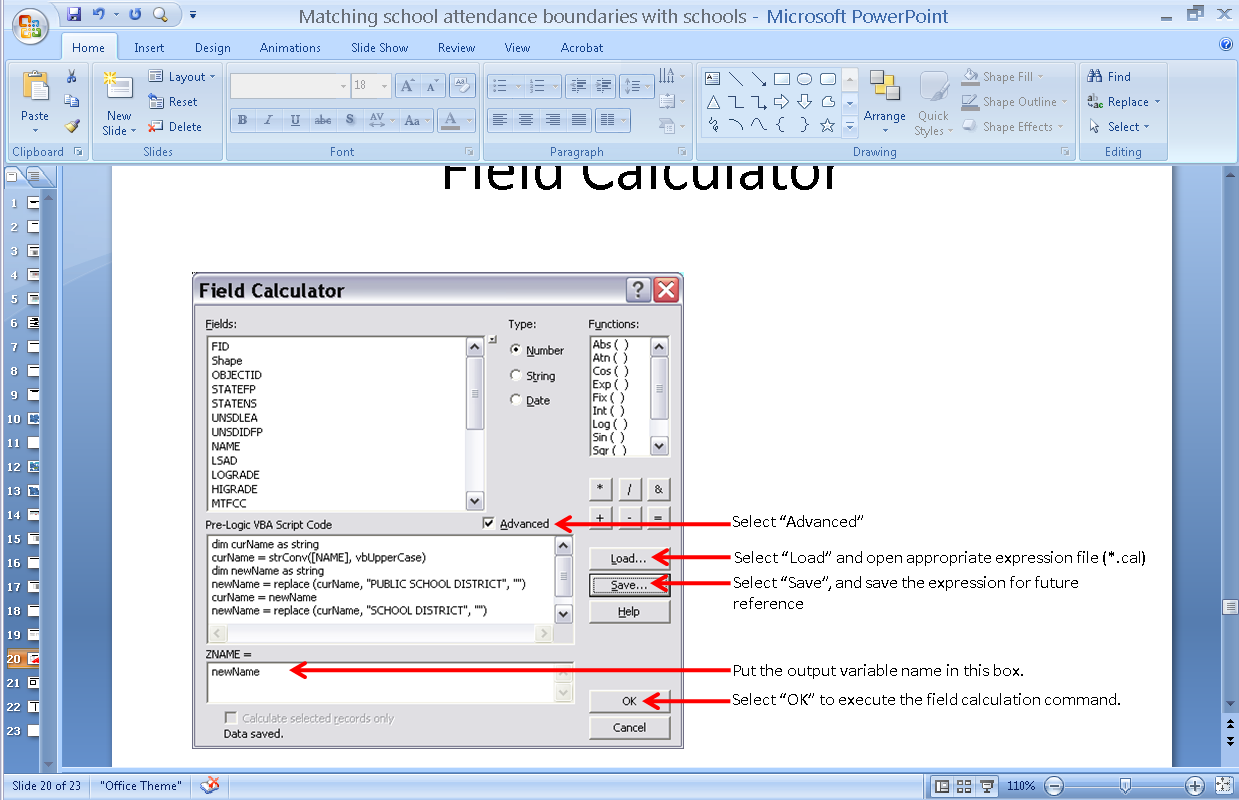
 

1. Now get the census school district dataset, and examine names of school districts in the census data and compare those with the school district names present in the dissolved layer obtained in step 1. Identify differences, especially common words from the district name field that may be removed from one or both datasets to facilitate exact match on the name field.
2. In this step we will modify district names to facilitate a name match. Start editor. Open the attribute table of the census school district dataset. Add a new text field in this dataset to store modified district names. I have named the newly added field with Z as a prefix to the original school district name field. For instance if the original field is named NAME then I added a new field with title ZNAME. Right click on the new field, and invoke Field Calculator command.



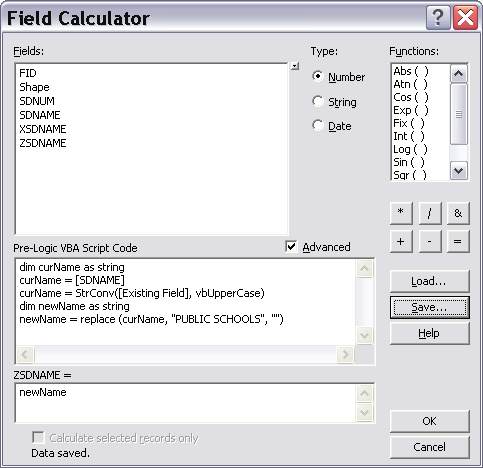


In the field calculator select Advanced, then load a field calculation script “ExpressionReplacePublicSchoolDistrictAndSchoolDistrictFromCensusSchoolDistricts.cal” and make appropriate changes to the find replace strings in it, depending on your dataset. Save your final script in a suitable location in your folder, and execute the script.

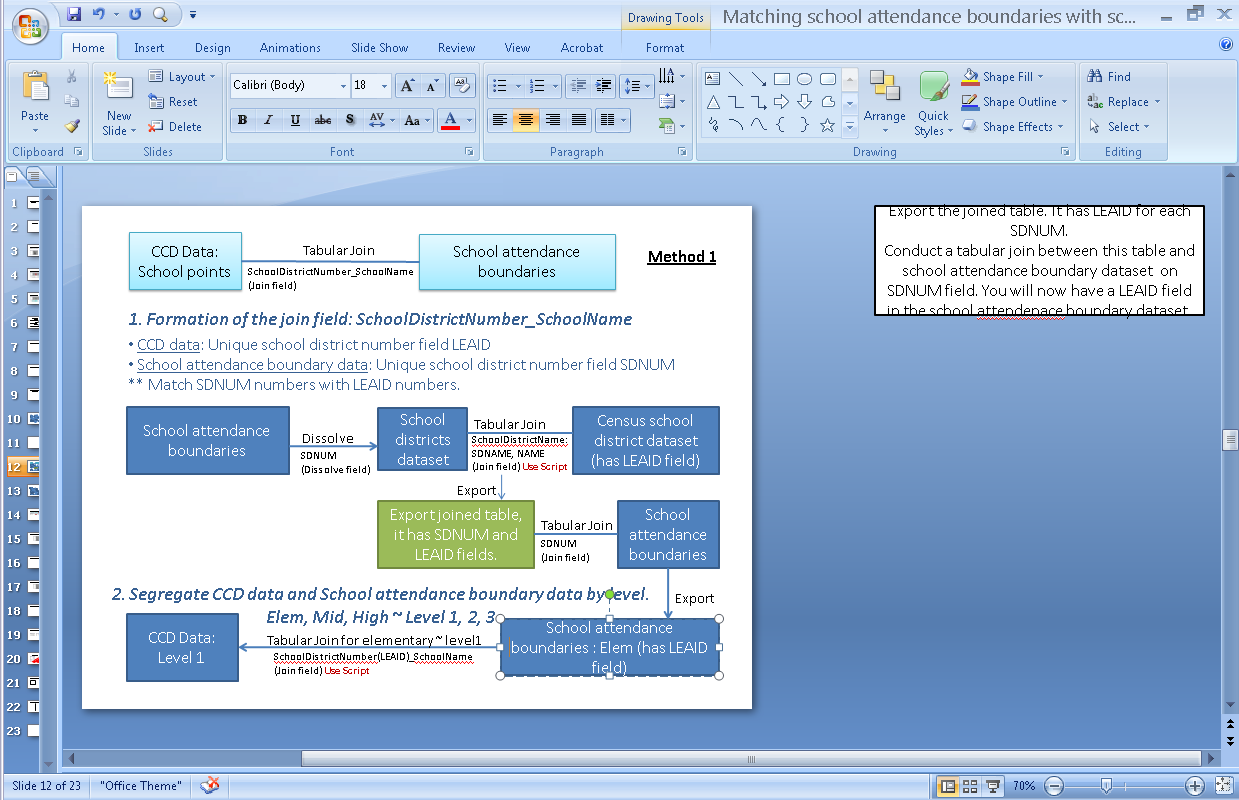


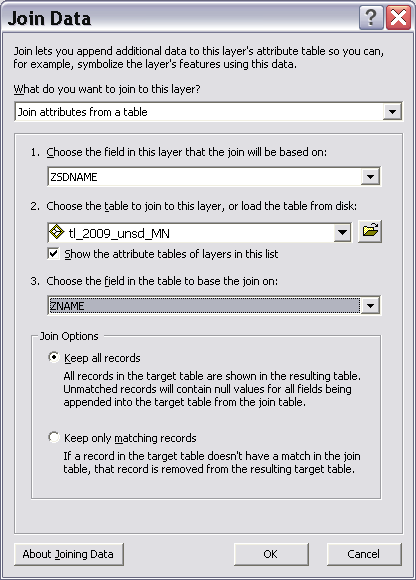
Examine values in the newly calculated field, and if you wish to rerun the script with modifications, again invoke the Field Calculator command, load script and modify it. To see the entire script, copy it in Notepad and make changes. Then paste it back into script area. One of the scripts and syntax are explained in detail in the power point presentation.

1. If necessary also edit school district names in the dataset obtained from dissolve operation performed in step 2.



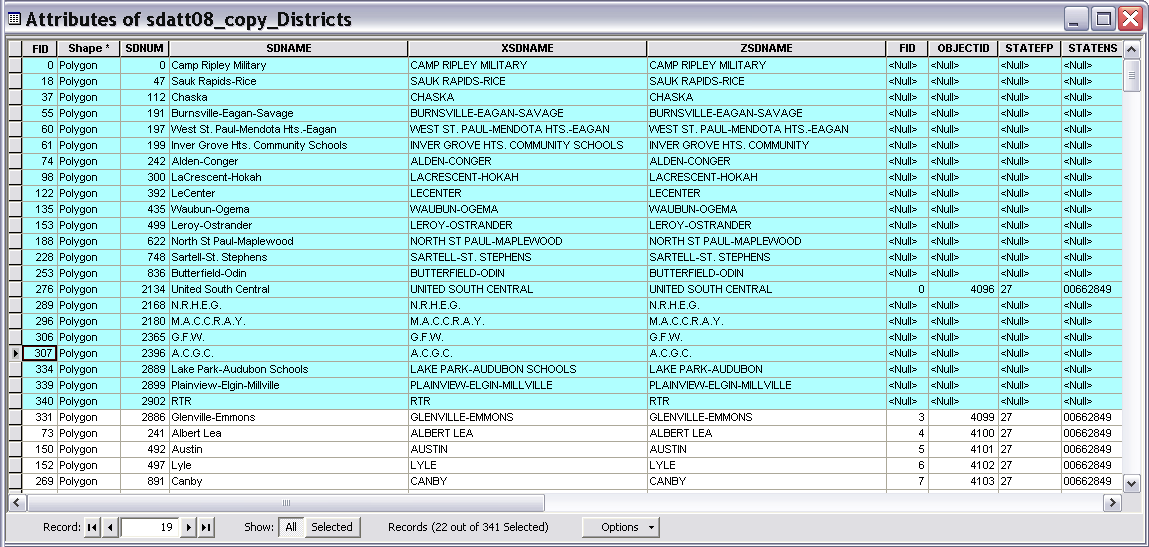
1. After both the datasets are edited up to your satisfaction, then proceed with carrying out an attribute join on them.

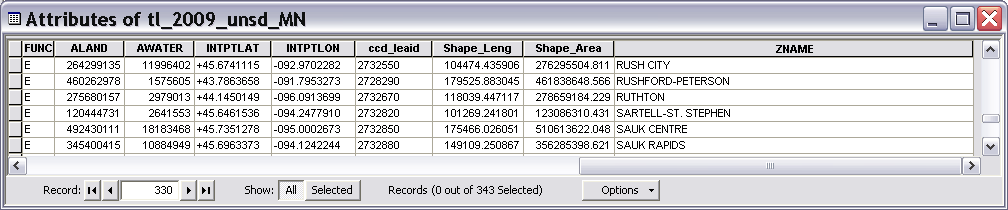




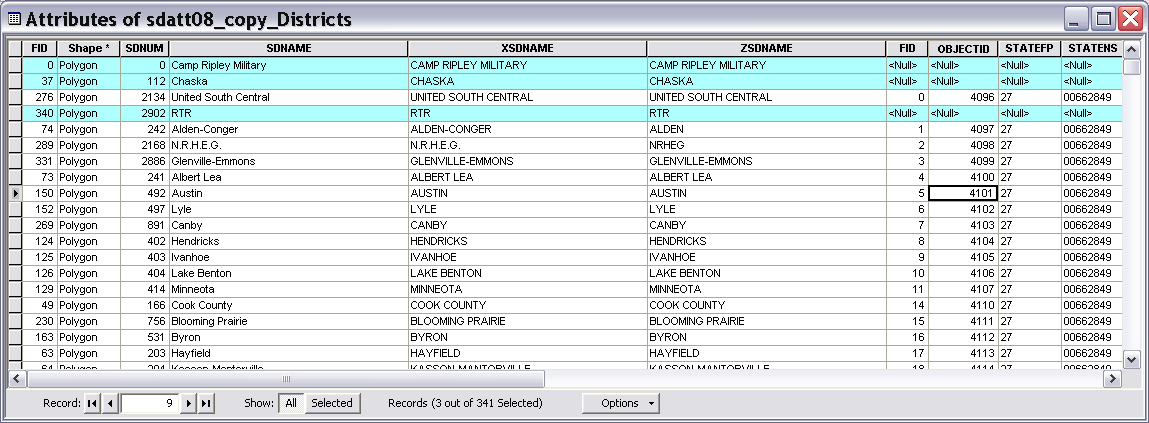


1. View the joined dataset. Most rows may have joined, some might be missed. Identify missed records by sorting on one of the joined layer’s fields and look at those rows that have values equal to null in joined fields. Also open the joining layer’s attribute table and sort it on the district name field. Then try and figure out records with missed join.

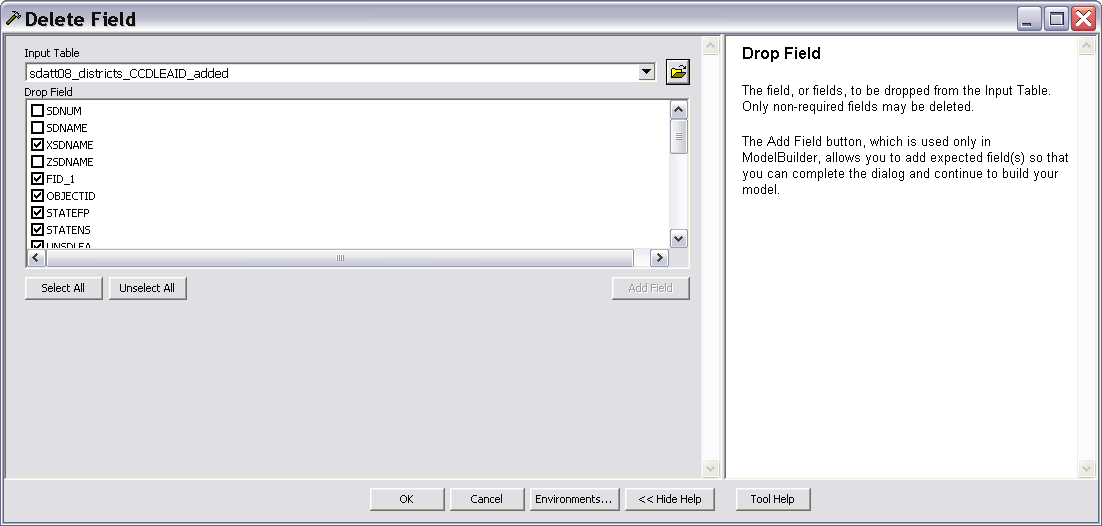
****

****

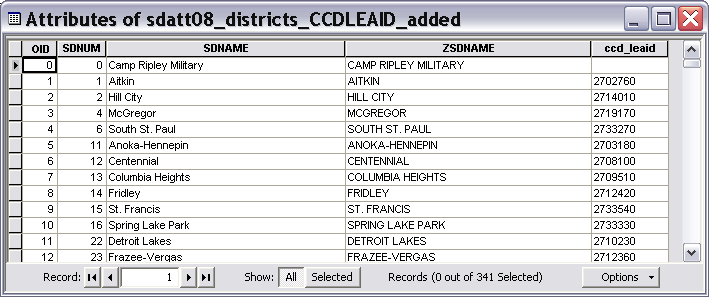
Start editor, and manually edit some of the obvious missed joins in the joined layer’s attribute table, and correct the join. For instance in one case one of the school district names is written as “SAUK RAPIDS” and its match is written as “SAUK RAPIDS RICE”, so I manually edited the words from the joining table to SAUK RAPIDS RICE, which corrected the join for that record. IF you manually edit some names like this, please keep a record of the FID / objected numbers for those. After I manually edited a few records, all except 3 records matched.



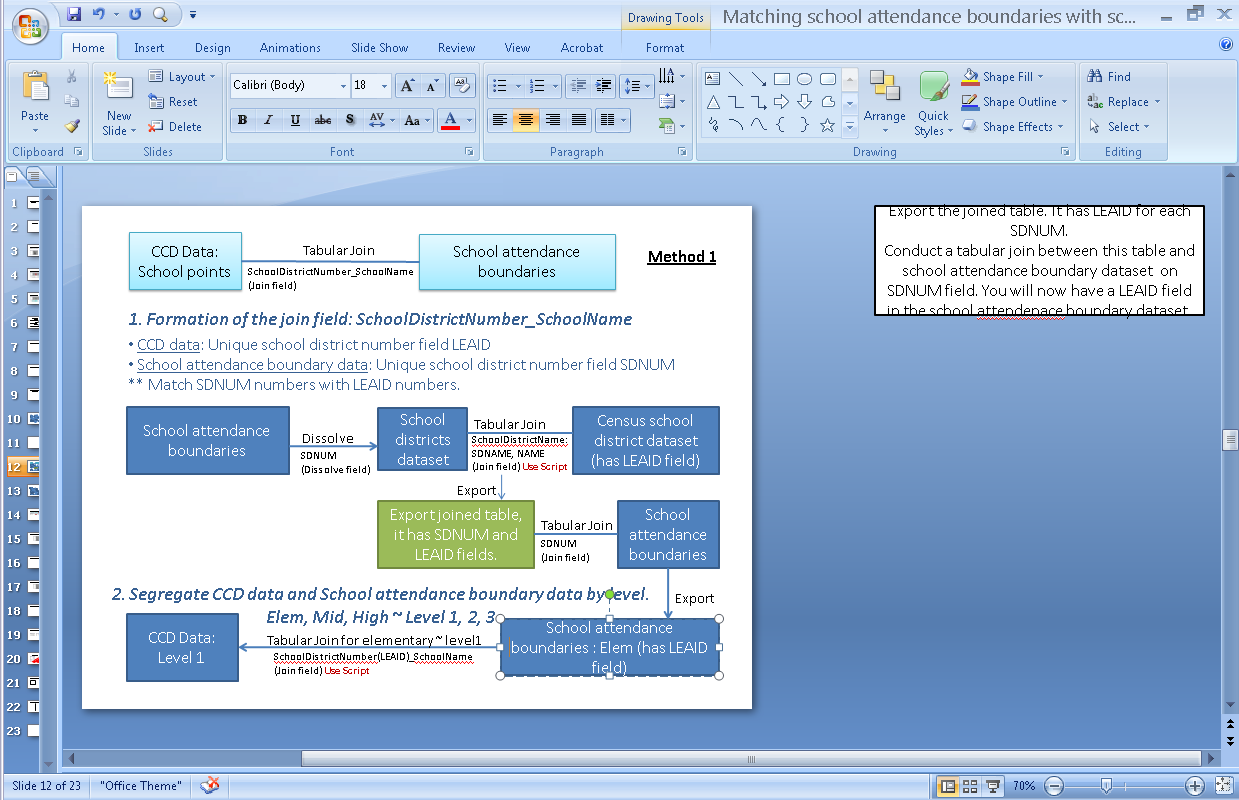
1. Export all the records to a table a dbf file. From that delete all the fields except for the ccd\_leaid field, school district name and number field from school attendance boundary dataset and any other relevant fields.

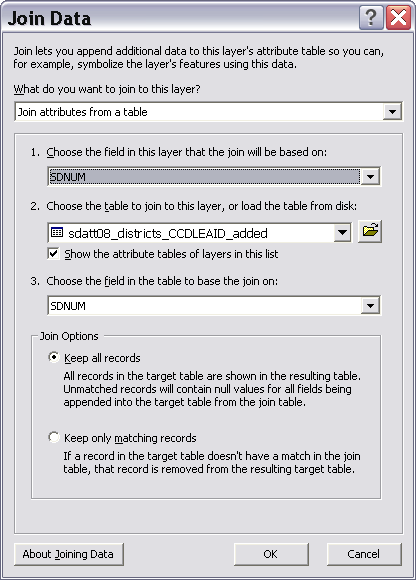


The following shows fields from the output table after I cleaned up redundant fields from it.

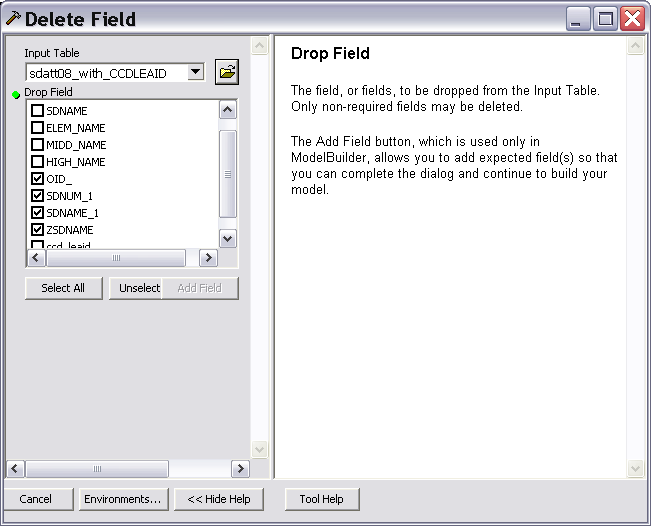


1. Now join the exported table to the school attendance boundary dataset to add (ccd\_leaid) LEAID field.

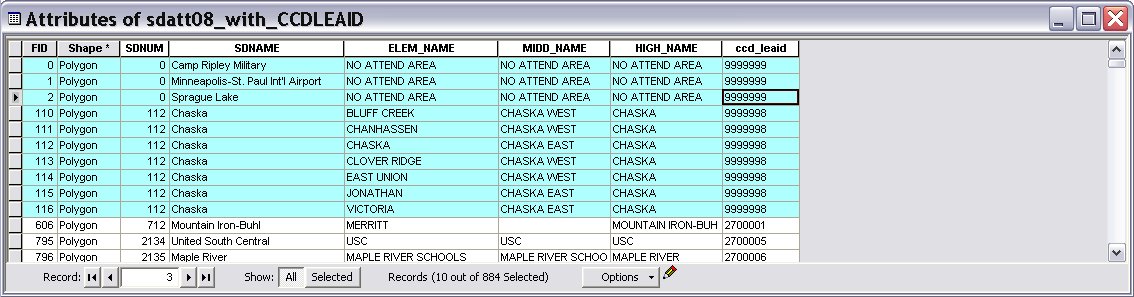




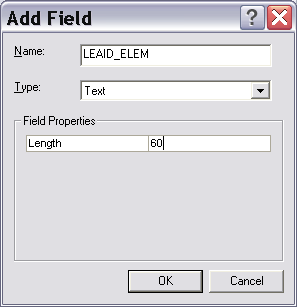
After the join, export the school attendance boundary dataset, to make the LEAID field a permanent part of it, and drop unnecessary joined fields from the exported shape file.



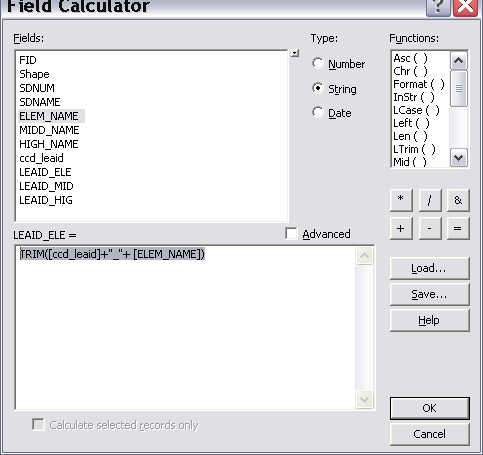
To avoid occurrence of nulls in the CCD\_LEAID field I added a value 9999999 in place of nulls, using the editor toolbar.

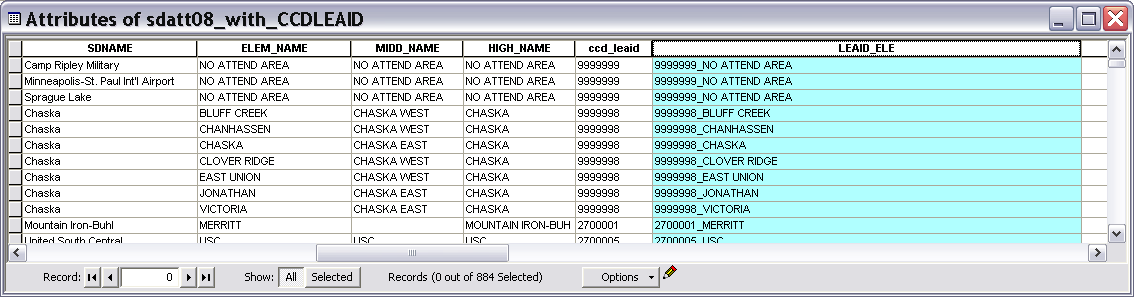


1. Now in the exported school attendance boundary dataset, add 3 fields LEAID\_ELE, LEAID\_MID, LEAID\_HIG to create fields on which elementary, middle and high school boundaries can be dissolved.

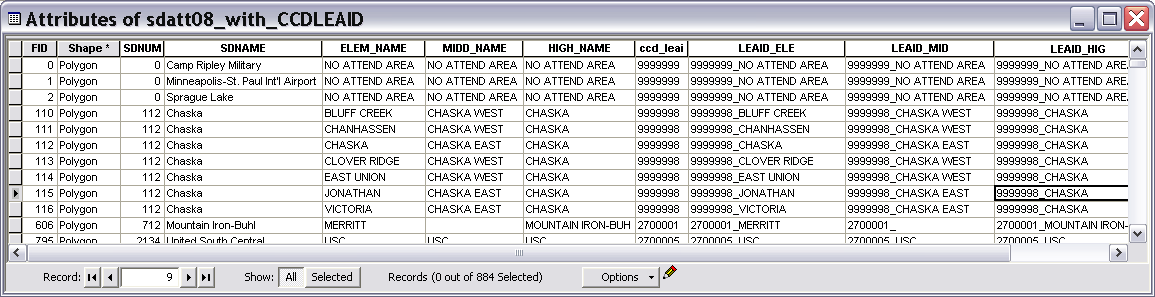


Use field calculator to calculate LEAID\_ELE field with formula TRIM([ccd\_leaid]+"\_"+ [ELEM\_NAME])

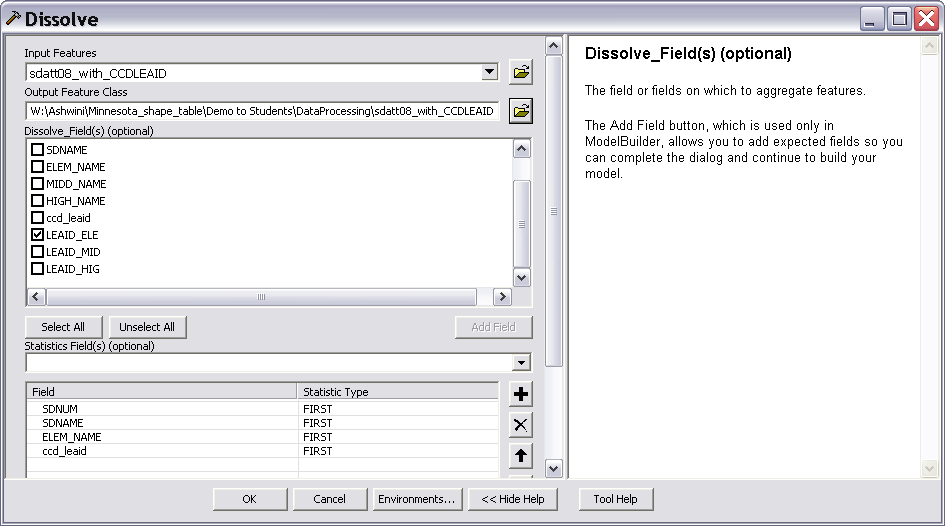


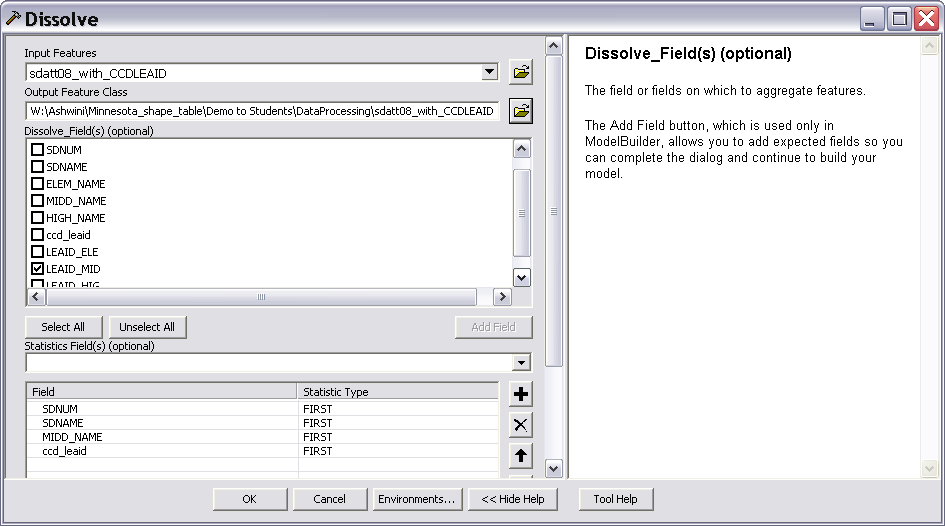


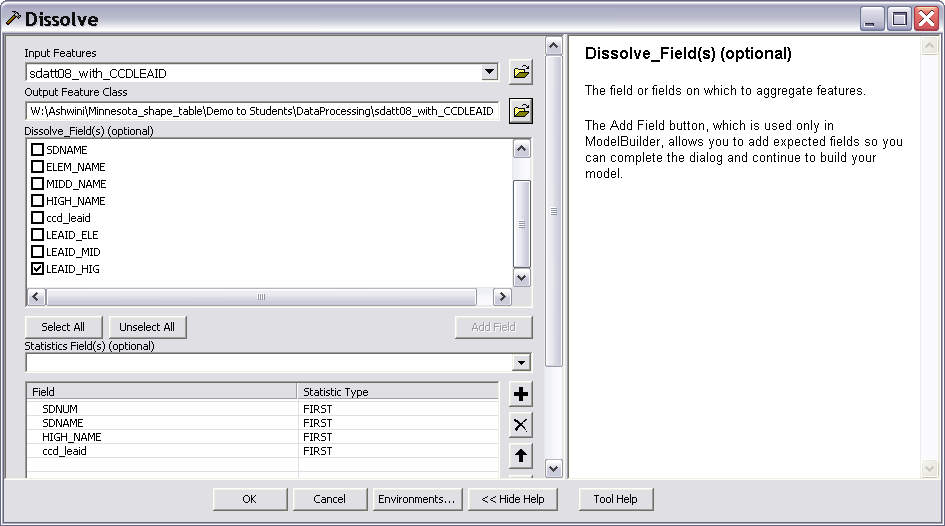
The attribute table of the school attendance boundary dataset should look like the following image.



1. Dissolve the school attendance boundary dataset on elem, middle, high levels to form 3 datasets of school attendance boundaries by level.

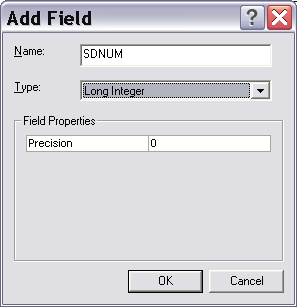


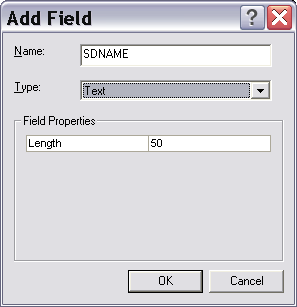


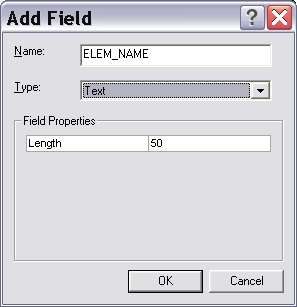


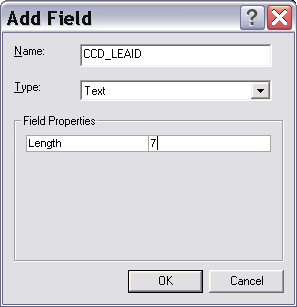
To restore the field names to original names present in the dataset prior to the dissolve operation, add new fields, name them appropriately and fill those with values by copying them from the corresponding field. For instance, note that the following image shows the new name of the “SDNAM” field is now “FIRST\_SDNA”. To correct that I added a field SDNAME, and calculated values in it by formula SDNAME = [FIRST\_SDNA] , and then deleted the field FIRST\_SDNA.











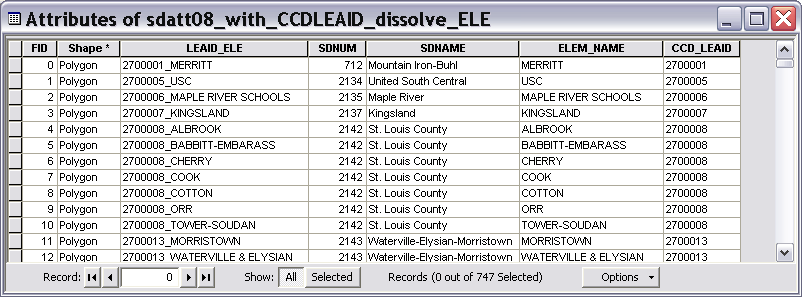
Field calculator formulae:

SDNUM = FIRST\_SDNU

SDNAME = FIRST\_SDNA

ELEM\_NAME = FIRST\_ELEM

CCD\_LEAID = FIRST\_CCD\_



Do the same for high and middle schools.

1. Now open the CCD dataset. If you have not created a shapefile from CCD data, make one now by using ADD X, Y data command. Select Long field for X axis, and LAT field for Y axis. Export the newly added point file and name it appropriately. If you have not isolated CCD records for your state, using the select by attribute query, select records for your state from the FIPS field, and export those to form a new CCD shapefile for your state. Then create 5 new datasets (Elementary, Middle, High, Charter, Magnet schools) from the CCD data based on the following criteria. Use Select by Attribute query for selection and export selected features to create new shape files.

Magnet: MAGNET07 = 1

Charter: CHARTER07 = 1

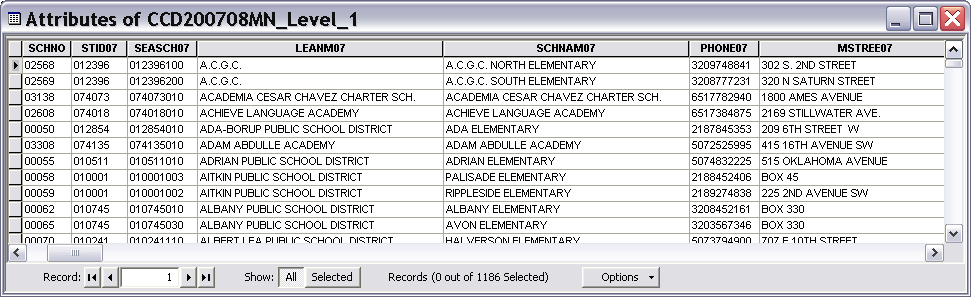
Elementary: LEVEL07 = 1

Middle: LEVEL07 = 2

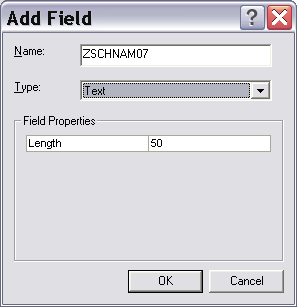
High: LEVEL07 = 3

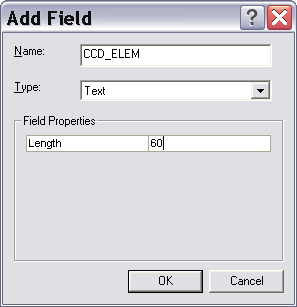
\*\* Note that numbers “07” in the field names indicate year of the CCD data, which may vary for your particular case.

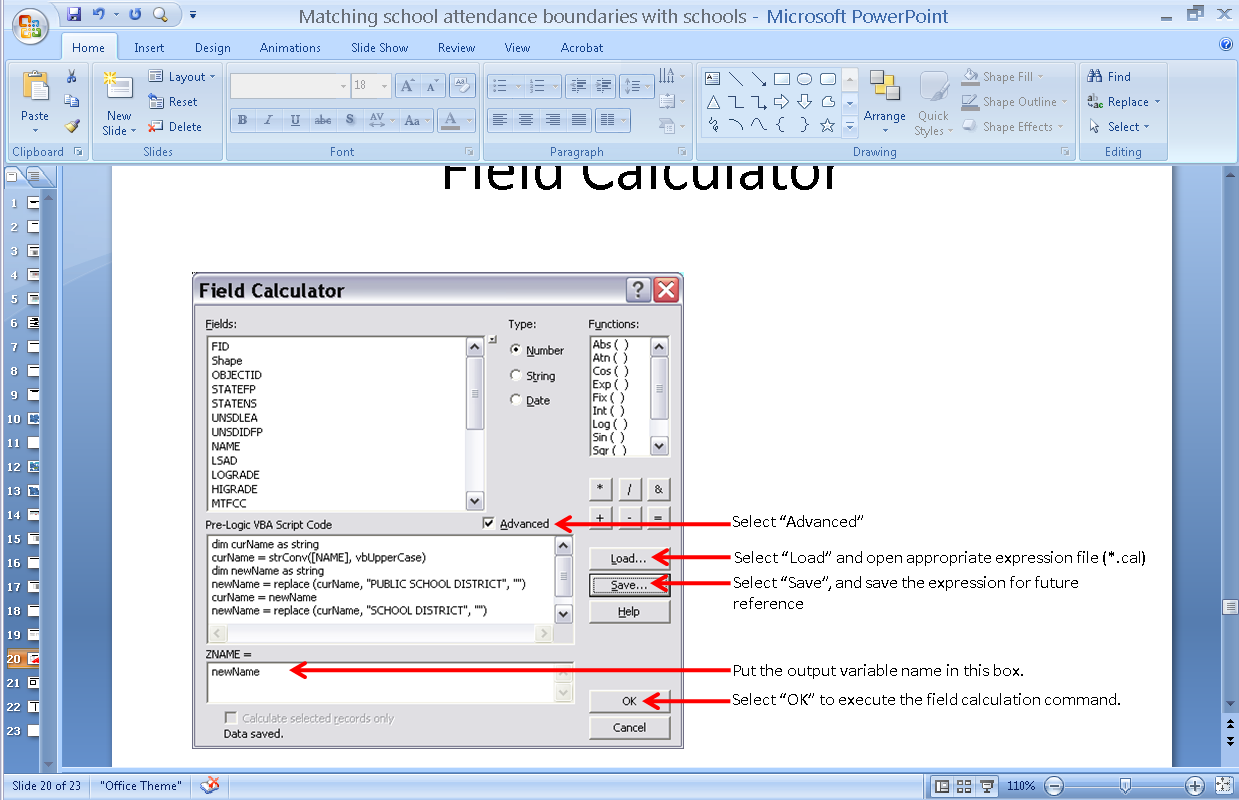
1. Now examine the CCD data level 1 dataset. Level 1 stands for elementary.



With the editor, add 2 fields in the CCD data, one to hold elementary names edited from the field SCHNAM07, and the other to hold joined elementary school names in format LEAID\_SCHOOLNAME. We will name new fields as ZSCHNAM07 and CCD\_ELEM.

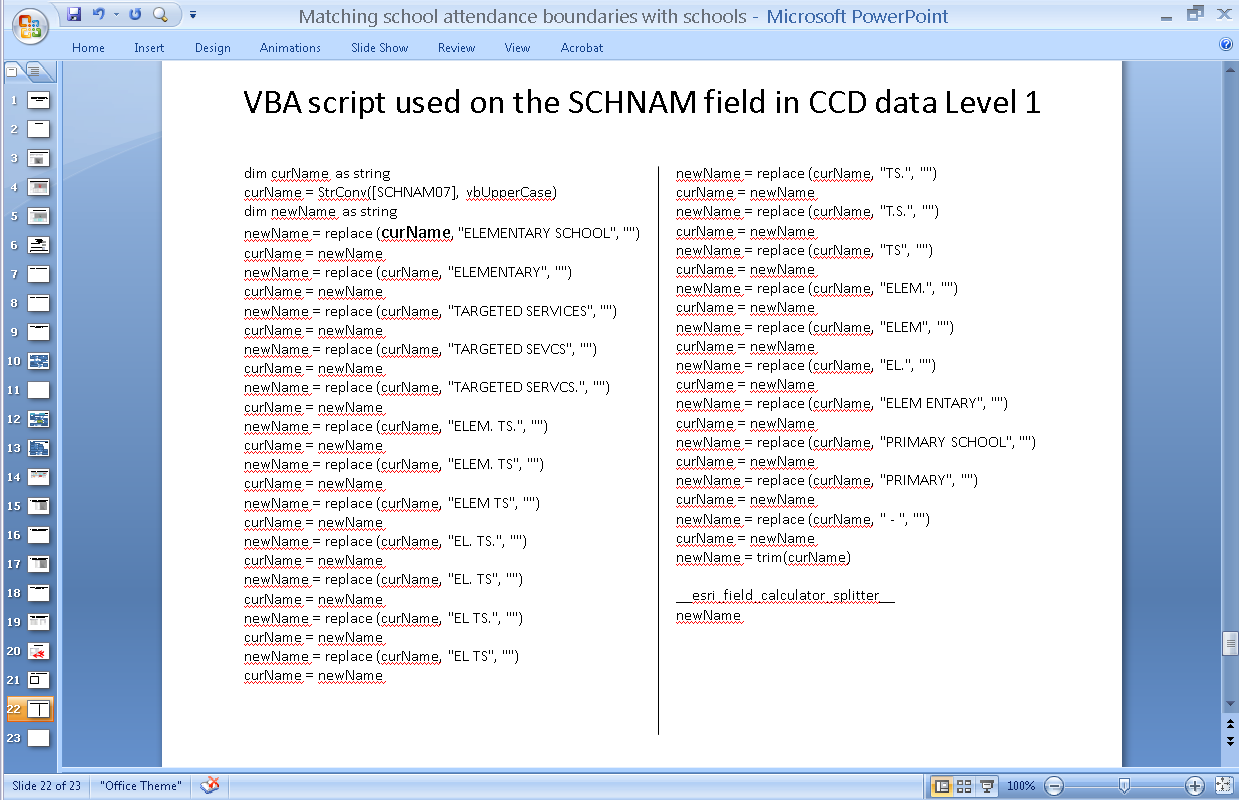




I used a script to modify to calculate values in the ZSCHNAM07 field. The purpose of this calculation is to match the school names in CCD data level 1 as closely as possible to the school names in elementary school attendance boundary data. Therefore examine these values, and identify the words that you would like to remove. 

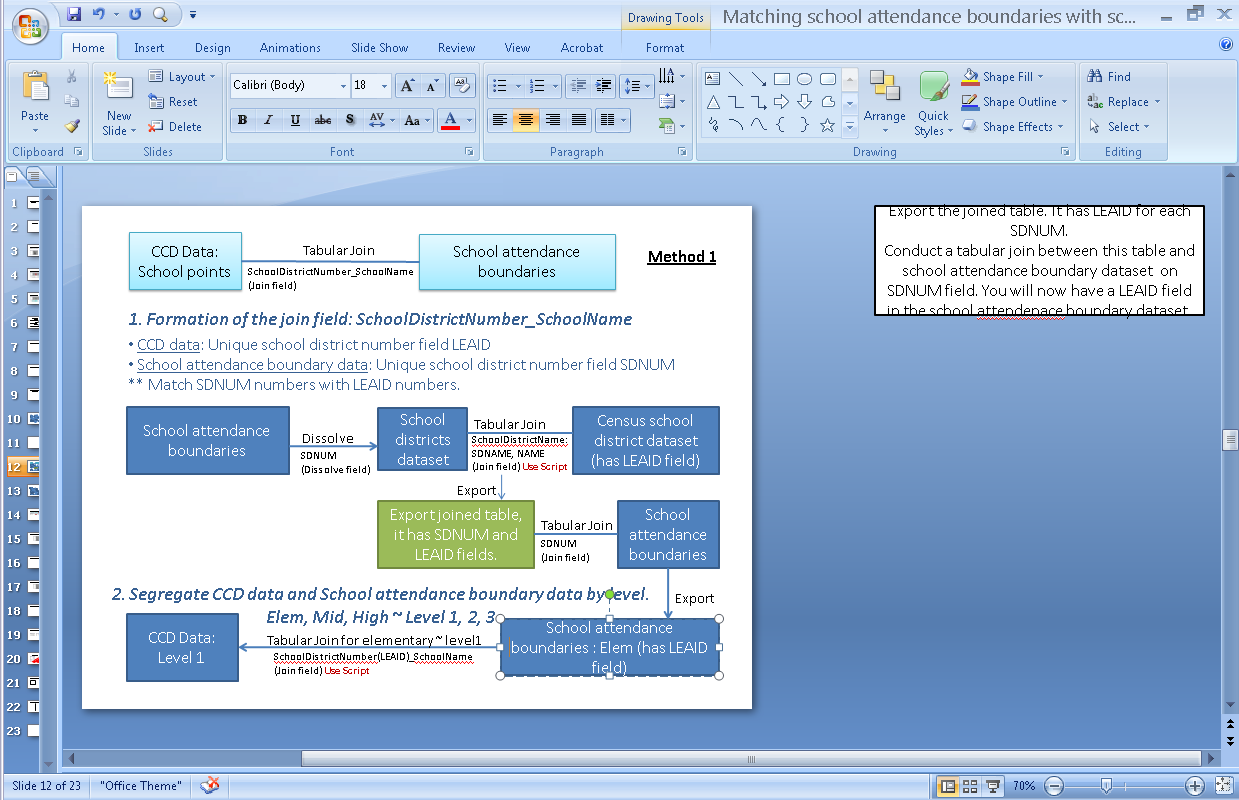
In the field calculator select Advanced, then load a field calculation script “ExpressionCCD\_ZSCHNAMCalculationELEM.cal” and make appropriate changes to the find replace strings in it, depending on your dataset. Save your final script in a suitable location in your folder, and execute the script.

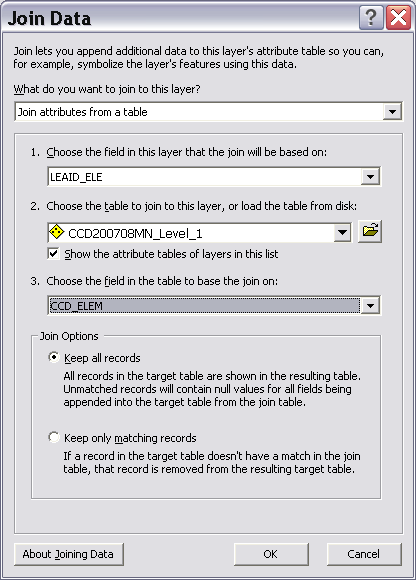
The script looks like the following.

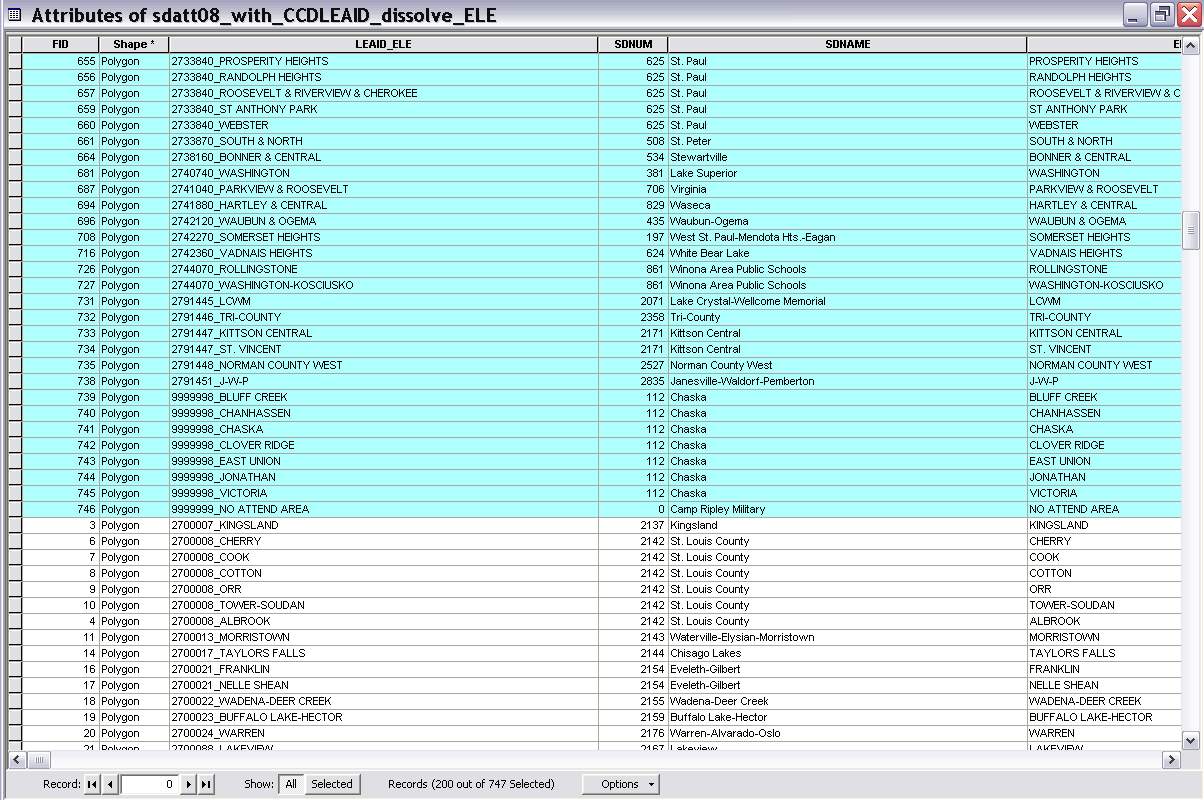


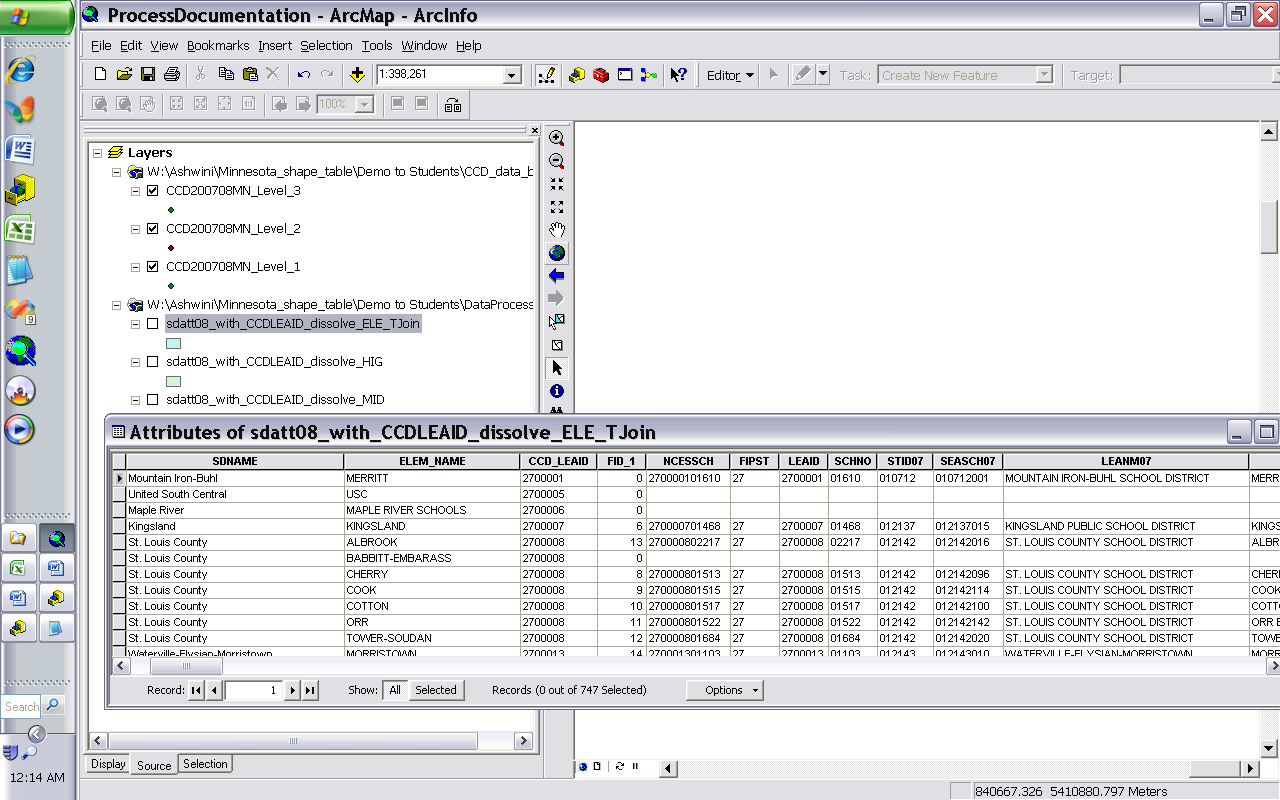
Detailed description of the script syntax is provided in the power point slides related to scripts.

1. After the school names are edited up to your satisfaction, calculate the newly added CCD\_ELEM field with formula trim([LEAID]+"\_"+ [ZSCHNAM07])
2. Now you can execute a tabular join on the CCD data and school attendance boundary data with join field as CCD\_ELEM and LEAID\_ELEM .

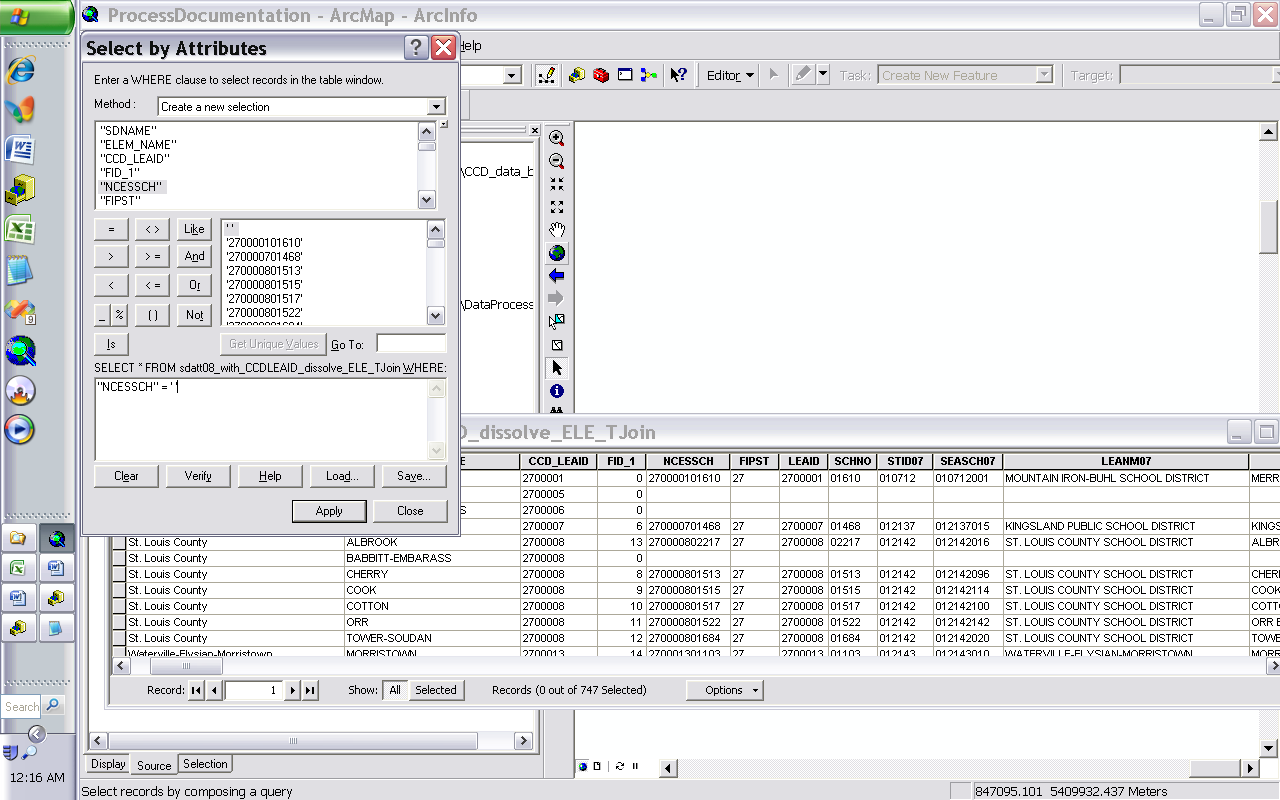






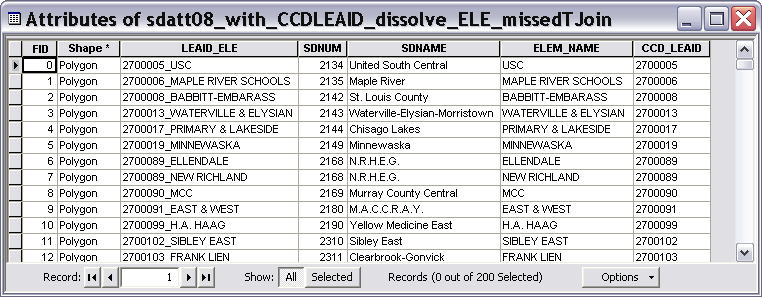
15. Export the joined shapefile to make joined fields permanent in the dataset. 

16. Examine the records from newly exported file and identify records where there is no join. You can sort on any field that contains null values, to identify records with no join.

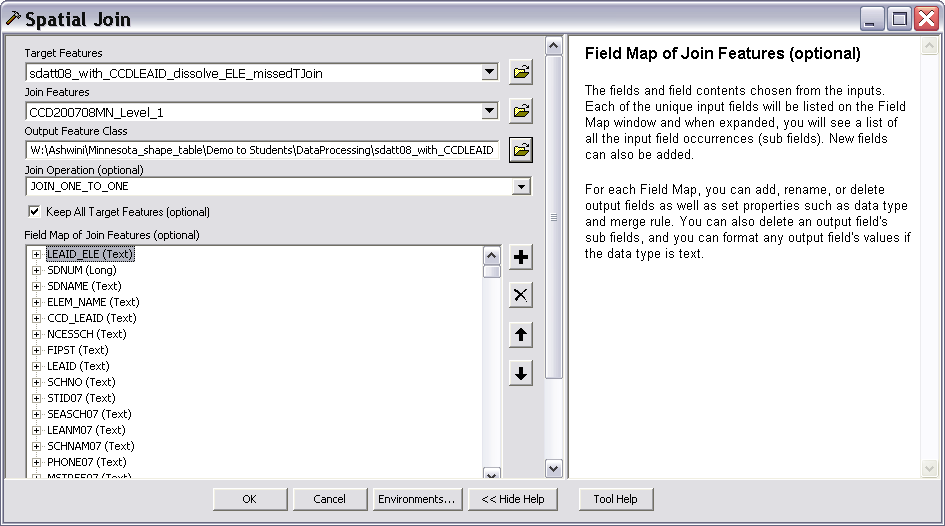


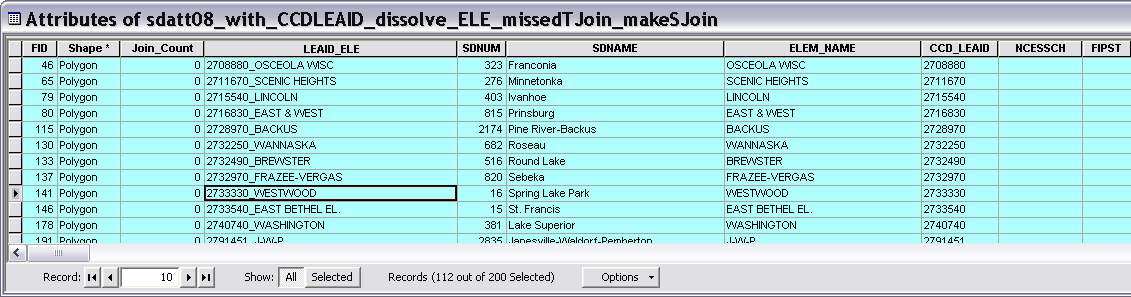
17. We will export the records that didn’t join and use spatial join on those. Create a new shape file with only the records that didn’t join.



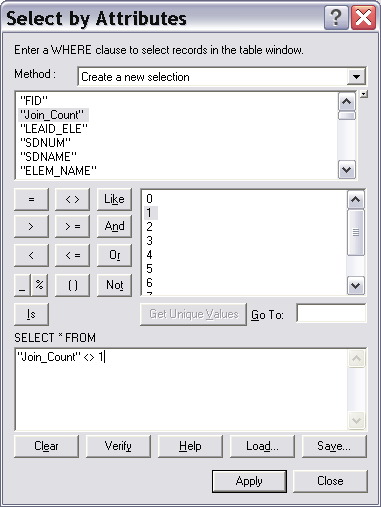


18. Carry on spatial join with the school attendance boundaries and newly created shape file.

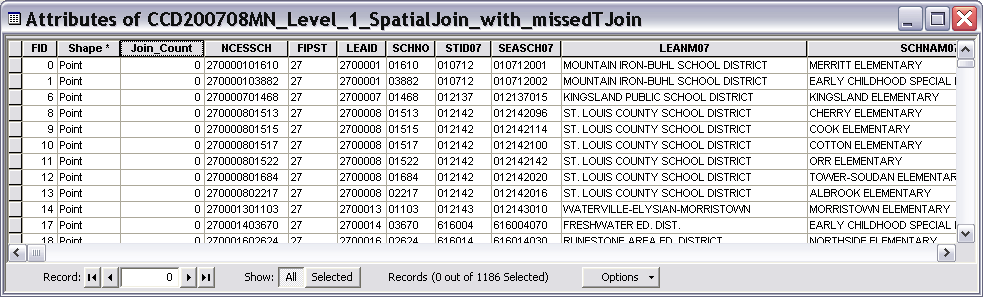


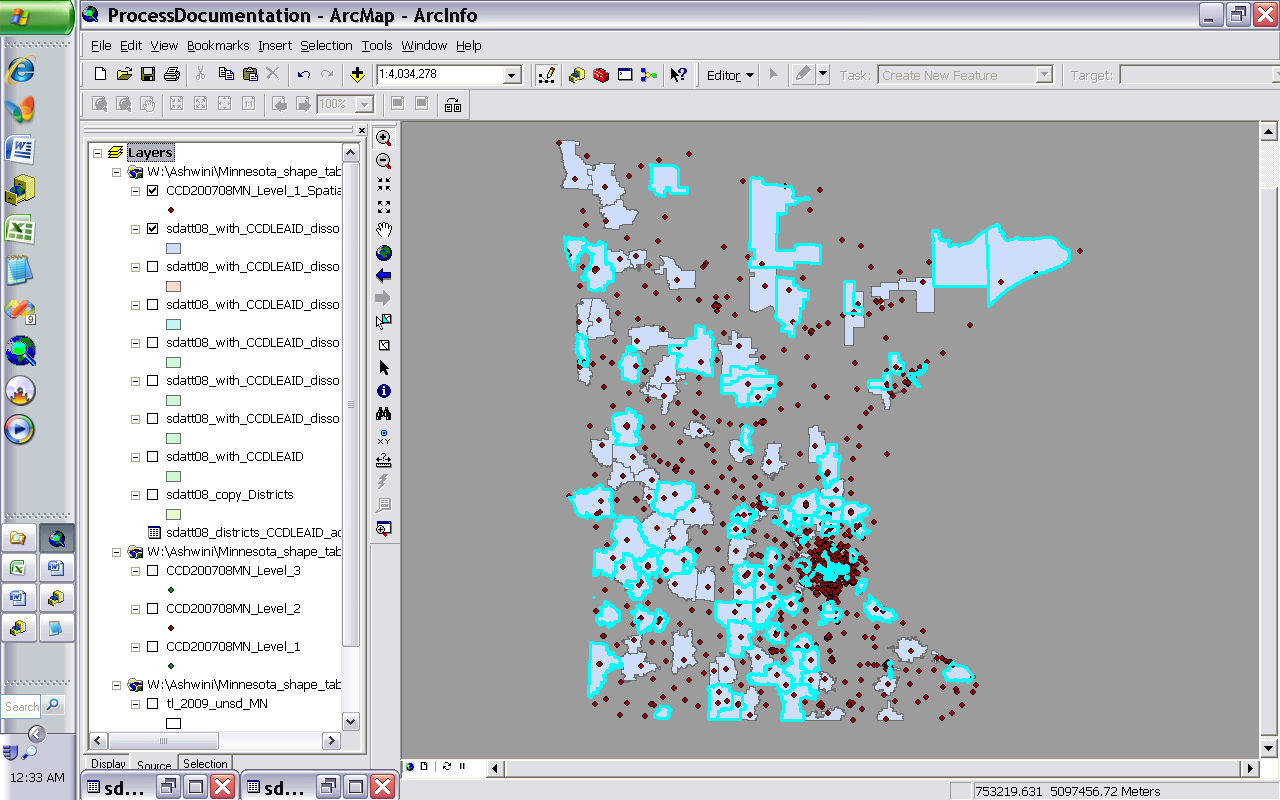


19. In the spatial join output file, examine how many records joined more than once or had a 0 join, by using the join count field values.



20. Also create a reverse spatial join with the same 2 datasets used in step 18, but interchange the target dataset and joining dataset. For instance this time join CCD missed records dataset with the school attendance boundary records. Then overlay the joined point dataset on top of the joined polygon dataset. Select those records in the polygon dataset that have join count not equal to 1, and examine visually overlaid points on those.





By identifying selected records individually and considering the school names, and school district names, it may be possible to understand why the spatial join didn’t work in these cases. Each case needs to be reviewed individually from this step.

**HowTo:  Use VBA functions in the field calculator**

|  |  |
| --- | --- |
| *Article ID:* | 31807 |
| *Software:* | **ArcGIS - ArcEditor** 8.3, 9.0, 9.1, 9.2, 9.3, 9.3.1 **ArcGIS - ArcInfo** 8.3, 9.0, 9.1, 9.2, 9.3, 9.3.1 **ArcGIS - ArcView** 8.3, 9.0, 9.1, 9.2, 9.3, 9.3.1 |
| *Platforms:* | N/A |

**Summary**

The field calculator dialog box allows for calculating the values of a field by specifying a calculation expression. You can type the expression directly into the box and also add fields, functions and operators into the box by clicking on them in the dialog box. You can include any Visual Basic Number, String or Date functions in the calculation expression to define. Commonly used functions are listed in the dialog box for the current choice of data Type. Simple field calculator expressions are entered into Field Calculator directly. More complex expressions, such as multi-line scripts, looping and branching, are entered in the Pre-Logic VBA Script code box by clicking the Advanced check box on the field calculator dialog box.   
  
For more information on using the Field Calculator, consult the ArcGIS help documentation. To access this documentation:   
  
- Open the field Calculator dialog box and click the Help button.   
- Open the ArcGIS Desktop Help and go to the Index Tab > Calculating > Fields in Attribute Tables > Making Field Calculations.   
  
There is also a help topic for working with date fields in the field calculator:   
-Open the ArcGIS DT Help and go to the Index Tab > Calculating > Fields in Attribute Tables > Working with Date Fields.   
  
The scope of this article is to demonstrate some common VBA functions that can be used within the Field Calculator.

**Procedure**

This document looks at some common VBA functions: LEFT, RIGHT, MID, INSTR, LEN, REPLACE, CHR, & OPERATOR and IF...THEN...ELSE statements. VBA has a very large array of components that can be used in the Field Calculator. These example are only a few possibilities.   
  
Note: In the VBA function syntax below, optional parameters are shown in square brackets ([ ]). In the examples, field names are shows in square brackets.

* **Left Function**   
    
  Returns a Variant (String) containing a specified number of characters from the left side of a string.   
    
  **Syntax**   
    
  Left(string, length)   
    
  **Left Function Example**   
    
  MyStr = Left([MyField], 1)
* **Right Function**   
    
  Returns a Variant (String) containing a specified number of characters from the right side of a string.   
    
  **Syntax**   
    
  Right(string, length)   
    
  **Right Function Example**   
    
  MyStr = Right([MyField], 1)
* **Mid Function**   
    
  Returns a Variant (String) containing a specified number of characters from a string.   
    
  **Syntax**   
    
  Mid(string, start[, length])   
    
  **Mid Function Example**   
    
  MyString = "Mid Function Demo" ' Create text string.   
  FirstWord = Mid(MyString, 1, 3) ' Returns "Mid".   
  LastWord = Mid(MyString, 14, 4) ' Returns "Demo".   
  MidWords = Mid(MyString, 5) ' Returns "Function Demo".
* **Len Function**   
    
  Returns a Variant (Long) containing the number of characters in a string.   
    
  **Syntax**   
    
  Len(string)   
    
  **Len Function Example**   
    
  MyString = [MyField] ' Initialize variable.   
  MyLen = Len(MyString)
* **InStr Function**   
    
  Returns a Variant (Long) specifying the position of the first occurrence of one string within another.   
    
  **Syntax**   
    
  InStr([start, ]string1, string2[, compare])   
    
  **InStr Function Example**   
    
  MyPosition = InStr([address], " ")
* **Replace Function**   
    
  Returns a string in which a specified substring has been replaced with another substring a specified number of times.   
    
  **Syntax**   
    
  Replace(expression, find, replace[, start[, count[, compare]]])   
    
  **Replace Function Example:**   
    
  NewString = Replace([comments], "#", "!")
* **Chr Function**   
    
  Returns a String containing the character associated with the specified character code. Note: a link to an external ASCII character chart is included in the Related Information section at the bottom of this page.   
    
  **Syntax**   
    
  Chr(charcode)   
    
  **Chr Function Example**   
    
  ' Replace a carriage return character with an exclamation   
  NewString = Replace([comments], chr(13), "!")
* **& Operator**   
    
  Used to force string concatenation of two expressions.   
    
  **Syntax**   
    
  result = expression1 & expression2   
    
  **& Operator Example**   
    
  MyStr = [MyField1] & " " & [MyField2]
* **If...Then...Else Statement**   
    
  Conditionally executes a group of statements, depending on the value of an expression.   
    
  **Syntax**   
    
  If condition Then [statements] [Else elsestatements]   
    
  Or, you can use the block form syntax:   
    
  If condition Then   
  [statements]   
  [ElseIf condition-n Then   
  [elseifstatements] ...   
    
  [Else   
  [elsestatements]]   
  End If   
    
  **If...Then...Else Statement Example**   
    
  Number = 53 ' Initialize variable.   
  If Number < 10 Then   
  Digits = 1   
  ElseIf Number < 100 Then   
  ' Condition evaluates to True so the next statement is executed.   
  Digits = 2   
  Else   
  Digits = 3   
  End If   
    
    
    
  Below are examples of field calculator expressions that use the above VBA functions, plus many more.
* Concatenate fields in a table. [**-show me-**](javascript:openWin('22606',%20500,%20500);)

|  |
| --- |
| **Summary** |
| Instructions provided describe the necessary steps to concatenate values from two or more fields in a table using ArcMap. |
| **Procedure** |
| * 1. Start ArcMap   2. Add a dataset or stand alone table to ArcMap.   3. Right-click the layer or table in the Table of Contents.   4. Select the 'Open Attribute Table' or 'Open' option, depending on the data source.   5. Select 'Options > Add Field.'   http://support.esri.com/common/style/graphics/icons/note.gif It is possible to use an existing field, but the output of this process will overwrite existing data in that field.   * 1. Define the new field in the 'Add Field' dialog box and click OK.   2. Right-click the field name of the new field.   3. Select 'Calculate Values.'   4. Click 'Yes' to the following prompt:   [O-Image] Calculate values   5. Type the following expression, where [Field Name] is the name of each individual field to be concatenated:   6. [Field Name] & " " & [Field Name] & " " & [Field Name]   http://support.esri.com/common/style/graphics/icons/note.gif This expression adds a space between the values that are being concatenated. For further information on the use of the concatenation operator, see the Microsoft Visual Basic Editor Help files. In ArcMap, select Tools > Macros > Visual Basic Editor. In the Visual Basic Editor, click Help > Microsoft Visual Basic Help. Click the Index tab, type "concatenation operators", and click 'display.' Select '& operator' and click 'display.'   * 1. Click OK. |
|  |

* Convert a string to proper case. [**-show me-**](javascript:openWin('31541',%20500,%20500);)

|  |
| --- |
| **Summary** |
| Instructions provided describes how to use the ArcMap Field Calculator to convert a string that is upper case, lower case or mixed case to proper case. For example, a string that is in the following formats:   "hello world"  "HELLO WORLD"  "hELLO wORLD"   the VBA function "StrConv" converts the string to: "Hello World".   For information on how to do the equivalent steps in a label expression, please see the link in the Related Information section below. |
| **Procedure** |
| * 1. Add a new text field to store the new string values.  [-show me-](http://support.esri.com/index.cfm?fa=knowledgebase.techarticles.articleShow&d=31734)   2. Open the ArcMap field Calculator for the field created in the step above by right-clicking the new field name heading in the Table View and click Calculate Values.   3. In the field calculator dialog box, type the following code in the text box under "Your Field Name = "   StrConv([Existing Field], vbProperCase)   Change [Existing Field] to match the name of the existing field that contains the string values that are not currently in proper   4. case.   5. Click OK in the field calculator dialog box. |
| * 1. [Convert a string to proper case in a label expression](http://support.esri.com/index.cfm?fa=knowledgebase.techArticles.articleShow&d=31542) Instructions provided describe how to use a label expression to convert a string that is upper case, lower case or mixed case to proper case. For example, if a string that is in the following formats: "hello world" "HELLO WORLD" "hELLO wOR... |

* Round numbers of attribute table to the nearest nth value. [**-show me-**](javascript:openWin('31778',%20500,%20500);)

|  |
| --- |
| **Summary** |
| Instructions provided describe how to round values in an attribute table to the nearest nth value using Field Calculator. |
| **Procedure** |
| * 1. Open attribute table.   2. Right-click field heading and select Calculate Values.   3. Place a check in the Advanced checkbox on the field calculator.   4. Paste the following code into the Pre-Logic VBA Script Code box.   5. Dim dblRnd as Double   dblRnd = math.round([Field]/N)\*N   * 1. Replace [field] with field to be rounded and N with value to round the values too.   Nearest Half N = 0.5  Nearest Tenth N = 0.1  Nearest Hundredth N = 0.01   2. In the final dialog box type "dblRnd" without the quotes. |
| * 1. [Use VBA functions in the field calculator](http://support.esri.com/index.cfm?fa=knowledgebase.techArticles.articleShow&d=31807) The field calculator dialog box allows for calculating the values of a field by specifying a calculation expression. You can type the expression directly into the box and also add fields, functions and operators into the box by clicking on them in... |

* Create a sequential unique ID field. [**-show me-**](javascript:openWin('27427',%20500,%20500);)

|  |
| --- |
| **Summary** |
| In ArcView 3.x, typing 'rec+1' in the field calculator creates a sequential unique ID field for a layer. Instructions provided describe how to do the equivalent in the ArcMap field calculator. |
| **Procedure** |
| * 1. Add the number field.  [-show me-](http://support.esri.com/index.cfm?fa=knowledgebase.techarticles.articleShow&d=31844)   2. Click Start Editing on the Editor toolbar.   3. Right-click the new number field. Click Calculate Values.   4. Click the Advanced check box.   5. Copy the following code into the Pre Logic VBA Script code:   6. Static rec As Long   7. Dim pStart As Long   8. Dim pInterval As Long   10. ' adjust start value if you want it to   11. ' start at a value other than 1.   12. ' For example, 1000.   13. ' ====================================   15. pStart = 1   17. ' adjust interval value if you want it to   18. ' increment at a value other than 1.   19. ' For example, 2.   20. ' =======================================   22. pInterval = 1   24. If (rec = 0) Then   25. rec = pStart   26. Else   27. rec = rec + pInterval   End If   * 1. Type the word 'rec' in the text box under [Your Field Name] =.   2. Click OK. |
|  |

* Convert Decimal Degree values to Decimal Minute Seconds. [**-show me-**](javascript:openWin('31243',%20500,%20500);)

|  |
| --- |
| **Summary** |
| Instructions provided describe how to use the Field Calculator to convert Decimal Degrees stored in a Numeric field to Degrees Minutes Seconds stored in a Text field. The output by default will be in the following format:   DDÃ‚ÂºMM'SS.SS"N |
| **Procedure** |
| Follow the steps below:   * 1. Add the table to ArcMap.   2. Right-click on the Table in the Table of Contents and select Open.   3. Verify 'Edit' mode is not enabled. Click the Options button and select Add Field.   4. Enter DMSLat in the Name field and select Text from the Type drop-down list. If DMSLat is already used as a field name, select a name that is not being used.   5. Change the length to 20.   6. Right-click on the DMSLat field and select Calculate Values.   7. Click Yes, if presented with a message box.   8. Check the Advanced check box.   9. Paste the following code into the expression box:   10. Dim DDField   11. Dim zM1 As Double   12. Dim Suffix As String   13. Dim DMS As String   14. Dim zY As Double   15. Dim zD As Double, zM As Double, zS As Double   16. Dim Dchr As String, Mchr As String, Schr As String   17. Dim Degree As String, Minute As String, Second As String   18. '=================================================   19. 'Adjust the variables below   20. DDField = [DDLat] 'Change to field with decimal degree values   21. Dchr = Chr(186) 'Character after degrees   22. Mchr = Chr(39) 'Character after minutes   23. Schr = Chr(34) 'Character after seconds   24. '=====================================================   25. zY = DDField   26. If zY >= 0 Then   27. Suffix = "N" 'N if Latitude, E if Longitude   28. Else   29. Suffix = "S" 'S if Latitude, W if Longitude   30. End If   31. zY = Abs(zY)   32. zD = Int(zY)   33. Degree = CStr(zD)   34. zM = (zY - zD) \* 60   35. zM1 = Int(zM)   36. Minute = CStr(zM1)   37. zS = FormatNumber(((zM - zM1) \* 60), 2)   38. zS1 = Int(zS)   39. Second = CStr(zS)   DMS = Degree & Dchr & Minute & Mchr & Second & Schr & Suffix   * 1. Change the value within the brackets next to 'DDField =' to the field in the table that contains the latitude decimal degree values. To change the characters after degrees minutes and seconds edit the value after 'Dchr =', 'Mchr =', and 'Schr =' respectively.   2. Paste the following code into the 'DMSLat =' box at the bottom of the dialog box.   DMS   * 1. Click OK to run the Field Calculator.   2. Repeat steps 3 through 12 for the longitude values, but change the values within the code where Suffix = "N" to Suffix = "E" and where Suffix = "S" to Suffix = "W". Also, change the value next to 'DDField =' to the field in the table that contains the longitude decimal degree values. |
| * 1. [Convert Degrees Minutes Seconds values to Decimal Degree values using the Field Calculator](http://support.esri.com/index.cfm?fa=knowledgebase.techArticles.articleShow&d=22455) Instructions provided describe how to use the Field Calculator to convert Degrees Minutes Seconds stored in a string field to Decimal Degrees stored in a number field. The values must be stored in a field in a table as Degrees Minutes Seconds with... |

* Convert Degrees Minutes Seconds values to Decimal Degree values. [**-show me-**](javascript:openWin('22455',%20500,%20500);)

|  |
| --- |
| **Summary** |
| Instructions provided describe how to use the Field Calculator to convert Degrees Minutes Seconds stored in a string field to Decimal Degrees stored in a number field. The values must be stored in a field in a table as Degrees Minutes Seconds with no symbols. For example:   25 35 22.3   In the example, 25 is degrees, 35 is minutes and 22.3 is seconds. |
| **Procedure** |
| Follow the steps below.  http://support.esri.com/common/style/graphics/icons/note.gif Initially perform the steps below on positive numbers. After the conversion is complete, multiple any fields that need to be negative by -1     * 1. Add the table to ArcMap.   2. Right-click on the Table in the Table of Contents and select Open.   3. Verify 'Edit' mode is not enabled. Click the Options button and select Add Field.   4. Enter Lat2 in the Name field and select Double from the Type drop-down list. If Lat2 is already used as a field name, select a name that is not used.   5. Verify the Scale and Precision is set to 0 and click OK.   6. Right-click on the Lat2 field and select Calculate Values.   7. Click Yes if presented with a message box.   8. Check the Advanced check box.   9. Paste the following code into the expression box:   10. Dim Degrees as Double   11. Dim Minutes as Double   12. Dim Seconds as Double   13. Dim DMS as Variant   14. Dim DD as Double   16. DMS = Split([Latitude])   17. Degrees = CDbl(DMS(0))   18. Minutes = CDbl(DMS(1))   19. Seconds = CDbl(DMS(2))   DD = (Seconds/3600) + (Minutes/60)+ Degrees   * 1. Find the line that begins 'DMS. . .' The text within the brackets [ ] is the name of the field holding the latitude values. Replace the word Latitude in the code with the name of the field that stores the latitude values in the table.   2. Paste the following code into the 'Lat2 =' box at the bottom of the dialog box.   CDbl(DD)   * 1. Click OK.   2. Repeat steps 3 through 12 for the longitude values. |
|  |

* Convert Decimal Minutes to Decimal Degrees. [**-show me-**](javascript:openWin('28520',%20500,%20500);)

|  |
| --- |
| **Summary** |
| Instructions provided describe how to convert decimal minutes data to decimal degrees (DDEG) data. This script works for string decimal minutes values in the format of: <Deg> <space> <Decimal\_minutes>.   For example:  -10 30.57  100 49.3  0 25.2789  45 0  -2 0.25 |
| **Procedure** |
| * 1. Create a new field with type double to store the DDEG data.  [-show me-](http://support.esri.com/index.cfm?fa=knowledgebase.techarticles.articleShow&d=31845)   2. Select Editor > Start Editing from the Editor toolbar.   3. Right-click on the new field and select Calculate Values.   4. Click the Advanced check box.   5. Copy the following code into the Pre-Logic VBA Script code:   6. dec\_min=[dec\_min]   7. if IsNull(dec\_min) or dec\_min = "" or IsEmpty(dec\_min) then   8. dec\_deg = 0   9. else   10. space\_pos=InStr(dec\_min," ")   11. deg=Left(dec\_min,space\_pos-1)   12. min=Right(dec\_min,Len(dec\_min)-space\_pos)   13. min=min/60   14. if deg < 0 then   15. dec\_deg=(Abs(deg)+min) \* -1   16. else   17. dec\_deg=deg+min   18. end if   end if   * 1. On line 1 of the code sample, change [dec\_min] to be the name of the field that stores the decimal minutes values.   2. Type the word 'dec\_deg' in the text box under [Your field Name] =.   3. Click OK. |
|  |

* Replace one value for another within a field. [**-show me-**](javascript:openWin('20119',%20500,%20500);)

|  |
| --- |
| **Summary** |
| Instructions provided demonstrate how to use the Replace function in the Field Calculator, to replace one value for another within a field in a table. |
| **Procedure** |
| * 1. Load the table into ArcMap.   2. Select Start Editing from the Editor menu.   3. Open the table.   4. Right-click the desired field.   5. Select Field Calculator.   6. Type the following in the expression box:   replace()   * 1. Place the cursor between the parenthesis of the line just typed.   2. Select the name of the field from the Fields list.   3. Type a comma after the end bracket surrounding the field name.   4. Type the value to be replaced and the replacement value; enclose each value in quotation marks and separate with a comma. For example:   replace([FLD\_name],"Wisconsin","WI")   * 1. Click OK. |
|  |

* Truncate a field. [**-show me-**](javascript:openWin('20856',%20500,%20500);)

|  |
| --- |
| **Summary** |
| The ArcMap Field Calculator allows truncating an existing value. Several Visual Basic string manipulation functions are available for truncating values.   The example provided in this article uses the Visual Basic Left function to return a specified number of characters from the left side of a string, and truncates after that number. |
| **Procedure** |
| * 1. Right-click the layer in the ArcMap table of contents.   2. Select Open Attribute Table.   3. Select Add Field from the Options dropdown list.   4. Create a new text field.   5. Start editing.   6. Right-click the new field name in the attribute table window.   7. Click Field Calculator.   8. Insert the following code in the expression box:   Left( [field\_name],#)  where [field\_name] is the field to truncate and # is the number of characters you want to preserve. |
|  |

* Populate a date field with unique values. [**-show me-**](javascript:openWin('23194',%20500,%20500);)

|  |
| --- |
| **Summary** |
| Instructions provided describe how to generate unique date values after adding a new date field to an attribute table. |
| **Procedure** |
| * 1. Start ArcMap.   2. Right-click on a layer and Select 'Open Attribute Table.'   3. Select 'Options > Add Field'. Name the field.   4. Select 'Date' from the 'type' menu. Click OK.   5. Right-click on the date field.   6. Select 'Calculate Values'.   7. Select the 'Date' radio button in the Field Calculator dialog box.   8. Copy the code below into the text box.   DateValue("mm/dd/yyyy")  or  DateValue("February 11, 2002")   * 1. Click OK. |
|  |

* Calculate a date field to be the current date. [**-show me-**](javascript:openWin('23213',%20500,%20500);)

|  |
| --- |
| **Summary** |
| Instructions provided describe how to calculate a date field to the current date using ArcMap. |
| **Procedure** |
| * 1. Open the attribute table that contains the date field.   2. Right-click the date field and select Calculate Values.   3. If you are comfortable calculating values outside of an edit session click Yes.   4. From within the Field Calculator dialog box set the Type radio button to 'Date'.   5. Set the field calculator equal to:   NOW()   * 1. Click OK on the field calculator to perform the calculation. |
|  |

**Related Information**

* [ASCII Character Look Up Table](http://www.lookuptables.com/)
* [Label expression by way of VBScript](http://support.esri.com/index.cfm?fa=knowledgebase.techarticles.articleShow&d=20154)
* [Use advanced label expressions in ArcMap](http://support.esri.com/index.cfm?fa=knowledgebase.techArticles.articleShow&d=17149)