ArcInfo 8

Converting Document.aml Metadata to Metadata in XML Format

By Robert Nicholas, ESRI Technical Support

Metadata is information that describes data the way a card in a library's card catalog describes a book. When a data source has been properly documented, metadata can help you determine how current data is, what map scale is appropriate for presenting the data, and how accurate the data is.

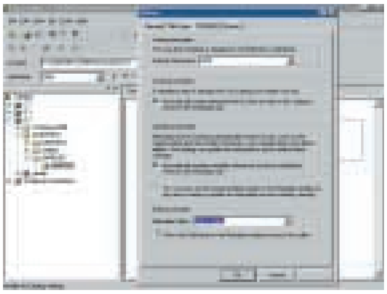
To create metadata in ArcInfo 8, all you have to do is click the data source in the ArcCatalog tree and then click the Metadata tab in the main ArcCatalog window. Metadata becomes a part of the data source itself. It is automatically moved, copied, and deleted along with the data source. Use ArcCatalog to create metadata for shapefiles, coverages, geodatabases, images, INFO tables—all the data sources supported by ArcInfo 8. ArcCatalog creates metadata that complies with Federal Geographic Data Committee standards.

Most types of metadata, except for geodatabase metadata, are stored in Extensible Markup Language (XML) files. XML, a markup language, is a subset of Standard Generalized Markup Language (SGML) and is similar to the familiar Hypertext Markup Language (HTML) used for Web pages. Geodatabase metadata is stored as a BLOB of XML within the database.

For many years, users have been creating and managing metadata in ArcInfo using document.aml. Some of these users may want to convert metadata created using document.aml to the new XML-based format to use the metadata management tools available in ArcCatalog. These tools include a new FGDC-compliant metadata editor, single-click updating of metadata files, and the ability to display metadata in different ways using one of the three stylesheets that come with ArcInfo or a custom stylesheet. Text files or Microsoft Word documents can also be attached to the data source's metadata in the manner of an e-mail message.

Follow these steps to convert document.aml metadata to the new XML-based format. This example uses an ArcInfo coverage called CAPITALS that has a metadata file that was created using the document.aml.

1. Start ArcInfo and go to an Arc prompt. Navigate to the workspace where the CAPITALS coverage resides and type `DOCUMENT capitals FILE capitals.txt` at the Arc prompt. This produces an ASCII text file containing all the metadata from the coverage. Close ArcInfo.
2. Open the ASCII file created in the previous step in a text editor such as Microsoft Notepad. Delete the second line, which contains several dashes followed by the output file's path. Save the changes to the file and close the text editor.
3. The next step requires a preparsing program called cns. If you don't have cns, go to the USGS's Web site (geology.usgs.gov/tools/metadata) to download and install it. This program, contained in a self-extracting Zip file, prepares metadata files for parsing. After installing cns, add the folder containing cns.exe to the system's PATH variable. The text file created by using the DOCUMENT command closely resembles the FGDC text format but doesn't match it exactly. Use cns to correctly map the metadata for CAPITALS to FGDC standards. If metadata created with document.aml is not corrected by using cns, this metadata will probably not display correctly in ArcCatalog and may contain errors.
4. Choose Start>Programs>Command Prompt to open a Command Prompt window on a Windows system. Navigate to the directory where capitals.txt resides. Type cns -o capitals_cns.txt capitals.txt at the prompt. This will create a file called capitals_cns.txt that can be imported into ArcCatalog.
5. Close the Command Prompt window. Start ArcCatalog. Select the CAPITALS coverage in the ArcCatalog tree and click on the Metadata tab in the main ArcCatalog window. Click on the Import Metadata button on the Metadata toolbar. In the Import Metadata dialog box, click the Format drop-down arrow and then click FGDC CSDGM (TXT). Click on the Browse button and navigate to capitals_cns.txt and click Open. After clicking on the OK button, the metadata will be imported to XML format and will be displayed using the current stylesheet.



Metadata is easy to create in ArcCatalog. ArcInfo metadata tools include a new FGDC-compliant editor.

Delete the two temporary text files, capitals.txt and capitals_cns.txt.

Atlas GIS 4

Selecting Features by Location

By Linda Schmidt, ESRI Technical Support

The Query menu tool can be used to determine how many features are located within a selected item such as the number of counties within a state. The subset created by this query can be saved as a file or copied to a new or existing layer. The following example, showing how to determine the number of counties in the State of California, will demonstrate the process.

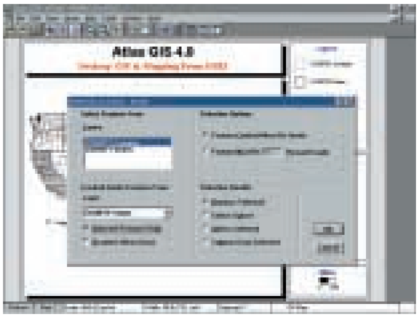
1. From the sample data that comes with Atlas GIS 4, open the County.agf file. This geofile contains both county and state boundaries.
2. Make sure that COUNTY:States is the only layer in the Default Layer Set by clicking on the Layers tool viewing the Default Layer Set dialog box. If other layers are included, choose Map>Layers & Themes, click on the Visibility Tab, and turn off the other layers.
3. Graphically select California by clicking on it.
4. Return to Map>Layers & Themes and turn on the COUNTY:Counties layer.
5. Choose Query>Select By Location> Inside from the menu.

Continued on page 41

Selecting Features by Location

Continued from page 40

6. In the Select by Location–Inside dialog box, choose COUNTY:Counties in the Select Features From Layers section.
7. In the same dialog box, select COUNTY:States in the Located Inside Features From Layer section and check the box for Selected Features Only.
8. Click OK. The status bar at the bottom of the interface should show 59 selected items indicating that 59 counties were found inside California.



The Query menu choice Select by Location–Inside returns the number of features inside a selected item.

The subset created by this query can be saved as a geofile by choosing File>Save from the menu or can be saved to a separate layer. To save to a separate layer, choose Edit>Copy to Layer. In the Copy to Layer dialog box, make sure the Copy Features from Layer choice is set to COUNTY: Counties and the Selected Features Only box is checked. In the Copy Features To section, click on the New Layer button. If a table is linked to the geofile, a new table can be created at this time. Click OK to finish copying just the counties of California to a new layer. AU

User Level
New/Casual Advanced

desktop notes

ArcView GIS

Open ArcView GIS to a Specific Project

By Mike Jensen, ESRI Technical Support

There may be instances when you want ArcView GIS to open to a specific project file upon start-up. Creating a shortcut in Windows to the ArcView GIS executable file and changing the shortcut's properties to start in a specific project file does not always work properly. The best way to accomplish this is to modify the Appl.Initialize script to point to a specific project file and create a new system default.apr file. The steps to do this are shown below.

Modify Appl.Initialize

1. Start ArcView GIS. In the default start-up dialog that asks if you want to open a project with a view, open a blank project, or open an existing project and uncheck the box next to Show This Dialog. Open a new blank project.
2. From the File menu select Extensions. Make sure none of the extensions are selected. Click OK.
3. Open a New Script window. From the Script menu select Load System Script.
4. Select the Appl.Initialize script. Click OK. This will open the Appl.Initialize script in a window with the default name Script1. Select all text in this script and copy it onto the Windows clipboard by pressing CNTRL + C. Open a text editor such as Microsoft Notepad. Paste the original Appl.Initialize script file into a new text file using CNTRL + V. Save this text file in a separate directory so that the original Appl.Initialize script can be easily restored if you desire.
5. Return to ArcView GIS. From the Script menu select Properties. Rename Script1 to Appl.Initialize. Click OK.
6. Locate and delete the following lines, located near the top, from the script.

```
startfile = "$HOME/avstart.txt".AsFileName
if (File.Exists(startfile).Not) then
   startupExt = Extension.Open("$AVHOME/
tools/avstart.___".AsFileName)
if (startupExt <> nil) then
   startupExt.AddDefaultOwner
   av.FindDialog("StartUp").Open
end
end
```

7. Add the following code to the end of the Appl.Initialize script.

```
theProject = av.GetProject
theFName = "C:\Temp\proj1.apr".AsFileName '--
Change path
if (nil <> theFName) then
if (nil <> theProject) then
theProject.Close
end
theProject = nil
av.ClearGlobals
av.DelayedRun("Project.OpenPart2", theFName, 0)
end
```

8. In the third line of code (`theFName = "C:\Temp\proj1.apr".AsFileName '-- Change path`) change the file name path to point to the project (.apr) file you want to open at start-up.
9. Compile the script by clicking on Checkmark button. **Do not run the script.** Close the script window.

Create a New Default Project

1. Leave ArcView GIS running and open Windows Explorer or another file management program. Create a folder on the C: drive called Avtemp.
2. Minimize Explorer, restore ArcView GIS, and open a new ArcView GIS script. Rename this script z_default using Script>Properties.
3. Paste the following line into the z_default script.

```
av.GetProject.MakeSysDefault("C:\Avtemp\
default.apr".asfilename,false)
```

4. Compile and run the new script z_default. This will create a new default.apr file in the Avtemp directory. Exit ArcView GIS. If prompted to save the new project, click No.
5. Go to Windows Explorer or equivalent and navigate to Esri/AV_Gis30/Arcview/Etc. Rename the default.apr file to default.old.
6. Go to the Avtemp folder and select the new default.apr file. Copy this file to the Esri/AV_Gis30/Arcview/Etc folder.
7. Restart ArcView GIS.

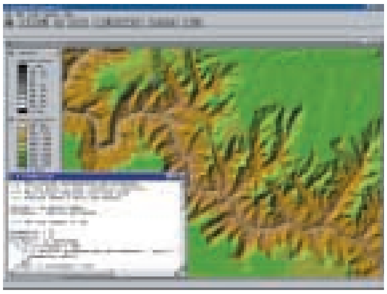
ArcView GIS will now open to the desired project, but you will still be able to open existing projects or create new ones. AU desktop notes

ArcView Spatial Analyst

Incorporating the Earth’s Curvature in Visibility Analysis

By Thomas Whitenack, ESRI Technical Support

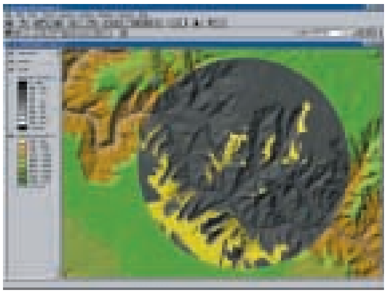
The Calculate Viewshed function, available from the Surfaces menu in the ArcView Spatial Analyst extension, doesn’t take into account the curvature of the earth or the refraction of visible light. In ArcInfo, these factors are included in the VISIBILITY function if the grid has a valid projection (prj) file. The script below incorporates the same formula used in ArcInfo to make this correction to surfaces calculated using ArcView Spatial Analyst.



This script uses the formula from ArcInfo software’s VISIBILITY function to factor in the earth’s curvature.

Download a copy of this script from *ArcUser Online.*

```
‘--- This script incorporates the curvature
‘--- of the earth to your visibility analysis.
‘--- This script requires that your grid
‘--- surface theme’s units are meters.

theview = av.getactivedoc
thethemes = theview.getthemes

‘--- Get the themes to use

pthemelist = {}
gthemelist = {}
for each t in thethemes
  if (t.is(ftheme)) then
    if (t.getftab.getshapeclass.getclassname = “point”) then
      pthemelist.add(t)
    end
  elseif (t.is(gtheme)) then
    gthemelist.add(t)
  end
end
if ((pthemelist.count =0) or (gthemelist.count = 0)) then
  msgbox.info
    (“At least 1 point theme and 1 Grid theme must be present”,””)
  return nil
end
thegtheme = msgbox.listasstring
    (gthemelist,”Select the Elevation grid”,”K-factor visibility”)
if (thegtheme = nil) then return nil end
thegrid = thegtheme.getgrid
thepttheme = msgbox.listasstring
    (pthemelist,”Select the Point theme”,”K-factor visibility”)
if (thepttheme = nil) then return nil end
theftab = thepttheme.getftab
thesfield = theftab.findfield(“shape”)
theselection = theftab.getselection

‘--- You can only do this visibility one point at a time

if (theselection.count <> 1 ) then
  msgbox.info
    (“There needs to be one and only one point selected”,””)
  return nil
end
for each rec in theselection
  p1 = theftab.returnvalue(thesfield,rec)
end

‘--- Set up the analysis environment

maxdist = msgbox.input
    (“Enter the maximum distance to consider”,””,”5000”)
if (maxdist = nil) then return nil end
if (maxdist.isnumber.not) then
  msgbox.info(“input must be numeric”,””)
  return nil
end
maxdist = maxdist.asnumber
therect = rect.makexy
    (p1.getx+maxdist,p1.gety+maxdist,p1.getx-maxdist,p1.gety-maxdist)
therect.scale(5)
grid.setanalysisextent(#GRID_ENVTYPE_VALUE, therect)
```



The script shades areas that will not be visible from the chosen viewpoint.

```
grid.setanalysiscellsize(#GRID_ENVTYPE_VALUE, thegrid.getcellsize)

'--- Create the 'd1' distance surface

pgrid = theftab.asgrid(prj.makenull,nil,nil)
d = pgrid.EucDistance(nil,nil,Maxdist)
d2 = d^2.asgrid

'--- Write out the formula constants as grids

Ediameter = 12740000.asgrid

'--- Calculate formula to create a result surface

thenewgrid = thegrid-(0.87.asgrid*(d2/Ediameter))

'--- Perform the visibility function on the corrected surface

thevisgrid = thenewgrid.visibility(theftab, prj.makenull, false)

'--- Create a theme and add it to the view

newtheme = gtheme.make(thevisgrid)
theview.addtheme(newtheme)

'--- End of Script
```

User Level

New/Casual Advanced

desktop notes

MapObjects 2

Tricks for Geometric Objects

By Jason Hine, ESRI Developer Support

The objects used in MapObjects are divided into five broad groups—Data Access, Address Matching, Map Display, Project, and Geometric objects. With Geometric objects, you can interact with map extents and work with points returned from the Map control by mouse events. Geometric objects can be used for returning positions and shapes drawn on the Map control, extracting selected shapes from a MapLayer, and manipulating the map extent of the Map control.

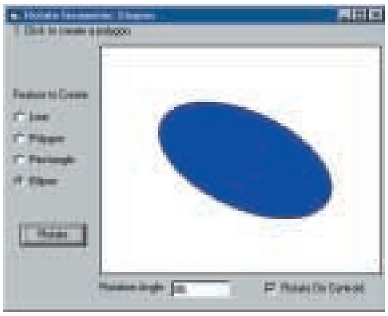
The functionality of Geometric objects has expanded with the release of MapObjects 2. Geometric objects now have the ability to perform Buffer, Intersection, Union, and Inverse Intersection operations. Techniques for using this new functionality to convert an ellipse to a polygon, convert a rectangle to a polygon, rotate a Geometric object, and detect self-intersecting polygons are described below.

Point, Line, and Polygon Geometric objects have the same geometric definition as the corresponding features in supported vector data. The Points object has the same geometric definition as a multipoint shape in a shapefile or Spatial Database Engine (ArcSDE) layer. However, none of the vector data sources directly store an Ellipse or Rectangle object, though geometry of these shapes can be approximated with polygon shapes. The first two techniques describe how to work with Ellipse and Rectangle objects.

Convert an Ellipse to a Polygon

An Ellipse object is a calculated curve that does not have any vertices. It is stored in memory as a mathematical formula to reduce memory usage and yield the smoothest possible curve. There may be times when you may need to convert an ellipse to a polygon. If you want to select the point features in a MapLayer that fall within the boundary of an ellipse, you can't use the SearchShape method. An ellipse is not a valid ShapeType. It cannot be passed to the SearchShape method because the SearchShape method's algorithm depends on the shape's vertices and the ellipse object lacks vertices.

To use an ellipse for point feature selection, convert it to a polygon using this simple trick: buffer the ellipse with a distance of zero (0). The Buffer method will return a polygon that approximates the calculated curve of the ellipse. Using the optional Extent argument of the Buffer method will produce the most accurate approximation.



An Ellipse or Rectangle object must be converted to a polygon before it can be rotated.

Rectangle to Polygon Conversion

To convert a rectangle to a polygon you need to write code that will create four new point objects, set their *x* and *y* values to match the corners of the rectangle, and then add the points to a new empty polygon object. The following sample code shows how to do this.

```
Dim rect As MapObjects2.Rectangle
Dim pt As New MapObjects2.Point
Dim pts As New MapObjects2.Points
```

Continued on page 44

Tricks for Geometric Objects

Continued from page 43

```
Dim poly As New
MapObjects2.Polygon

Set rect = Map1.Extent
pt.X = rect.Left
pt.Y = rect.Top
pts.Add pt
Set pt = Nothing
pt.X = rect.Right
pt.Y = rect.Top
pts.Add pt
Set pt = Nothing
pt.X = rect.Right
pt.Y = rect.Bottom
pts.Add pt
Set pt = Nothing
pt.X = rect.Left
pt.Y = rect.Bottom
pts.Add pt
Set pt = Nothing
poly.Parts.Add pts
Map1.FlashShape poly, 3
```

Rotate a Geometric Object

MapObjects considers the sides and axes of ellipse and rectangle objects to be parallel to the *x* and *y* axes of the Map control's coordinate system. Before rotating a Geometric object, it must be converted to a polygon. Use the method described in the previous section; then use standard trigonometry to rotate each vertex in the object around a given point by a given angle. See this concept applied in the Visual Basic sample, Rotate Geometric Shapes, that is available from the ESRI Web site. This sample may be downloaded from the MapObjects section of the ArcScripts Web site (**www.esri.com/arcscripts**).

Detecting Self-Intersecting Polygons

Determining whether a polygon intersects itself or not can be important. While a shapefile will store self-intersecting polygons, an SDE layer will not. Also, the new geometric operations available with MapObjects 2—Buffer, Intersection, Union, and Inverse Intersection—will not work with self-intersecting polygons. This characteristic can be used to identify self-intersecting polygons. Attempt to intersect the suspect polygon with another valid shape such as the polygon's extent. Using the Intersect method with a self-intersecting polygon will cause Visual Basic to generate an error message. The following code fragment demonstrates this concept.

```
Dim poly As MapObjects2.Polygon
Dim rect As MapObjects2.Rectangle
Dim iPoly As MapObjects2.PolygonP

Set poly = Map1.TrackPolygon
Set rect = poly.Extent
Set iPoly = poly.Intersect(rect)

If iPoly Is Nothing Then
  MsgBox “Poly is not valid.”
End If
```

Disclaimer

The user assumes all responsibility for use of the sample routines as well as implementation of them to achieve the intended results. The user is responsible for fully testing each portion of the routine prior to relying on it. This information is offered as a sample only, and ESRI assumes no obligation for its operation, use, or any resultant effect in spite of this offer. This information and these sample routines are provided on an “as is” basis, without warranty of any kind.

Please remember to back up your data prior to using this information.