

Title: How Do You Get GIS to the Casual User? A Small City's Approach to the Distribution of Data

Presenter: Tracy Schloss, GIS Coordinator, City of Lawrence, Kansas

Software: Arcview

Problem: How can I make my most common themes available to everyone, without teaching each user the names and locations of all the files necessary to create a basic map?

Solution: Create a theme dictionary, where the list of themes is predefined, and the location and legend are included.

'Name: Dict.SaveODB

'Description: Saves (writes) the active theme into the Theme Dictionary. You should have all properties (legend and name) defined before saving. They will be stored in the dictionary as defined.

```
theView = av.GetActiveDoc
theTheme = theView.GetActiveThemes.Get(0)
cloneTheme = theTheme.Clone
theThemeName = cloneTheme.GetName
theODBfilename = "c:/avcustom/ mythemes.odb"
theODBfile = theODBfilename.asfilename
theODB = ODB.Open(theODBfile)
_themeDictionary = theODB.Get(0)
tempDict = Dictionary.Make(10)
tempDict.Add(theThemeName, cloneTheme)
_themeDictionary.Merge(tempDict)
fullThemeLst = _themeDictionary.ReturnKeys
MsgBox.Info (theThemeName + " added to general features Theme Dictionary", "Update Dictionary")
if (theODB <> nil ) then
    theODB.Add(_themeDictionary)
    theODB.Commit
end
```

'Name: Dict.RestoreODB

'Description: Loads themes from a theme dictionary into your current view.

```
theView = av.GetActiveDoc
thefilename = "c:/avcustom/ mythemes.odb"
thefile = thefilename.asfilename
theODB = ODB.Open(thefile)
theDict = theODB.Get(0)
ThemeList = theDict.ReturnKeys
ThemeList.Sort(true)
myThemeChoiceLst = MsgBox.MultiListAsString(ThemeList, "General features available:", "ADD
THEMES")
for each i in myThemeChoiceLst
    theThemeName = theDict.Get(i)
    theView.AddTheme(theThemename)
    if (theThemeName.GetGraphics <> NIL) then
        for each g in theThemeName.GetGraphics
            theView.GetGraphics.Add(g)
    end
    theView.Invalidate
end
end
```

Problem: How do I use a script to load an existing .avl (legend) file?

Solution: Define the location of the legend file and load it using the leg.Load request.

```
'Name:Dict.AddParTheme
'
' Usage: Adds a parcel theme to a view
'
theView = av.GetActiveDoc
theViewName = theView.GetName

' These variables contain pathnames that may need to be modified
-----
legFile = "c:/avCustom/ parcel.avl".AsFileName
-----
parSecNm = MsgBox.Input("Key in Section Number in the format","T R Sec", "121936")
theSrcNm = (thePath+p"+parSecNm+f")
theSrc = SrcName.Make(theSrcNm++"poly")

if (nil = theSrc) then
    MsgBox.Warning("Invalid coverage or feature type", "")
    return nil
end
-----
' Loads property polygons
'
aTheme = Theme.Make(theSrc)
theView.AddTheme(aTheme)
aTheme.SetActive(true)
leg = aTheme.GetLegend
    if (nil = legFile) then
        Leg.SingleSymbol
        leg.GetSymbols.Get(0).SetColor(Color.GetYellow) ' if legend not found
    end
leg.Load(legFile, #LEGEND_LOADTYPE_ALL)
'
' Update the legend in the view TOC...
theThemeList = theview.GetActiveThemes
for each th in theThemeList
    th.InvalidateLegend
    th.SetVisible(true)' Draw the theme...
end
```

NOTE: These are abbreviated scripts, with some of the necessary lines removed for space considerations.
The complete scripts will be available from the MWGLAIG web site.