EVALIDatorReports: Reporting Beyond the FIADB

Patrick D. Miles[1]

Abstract: *Tools for analyzing data collected by the U.S. Forest Service's Forest Inventory and Analysis (FIA) program are available in Microsoft Access© format. Databases have been created for every state, except Hawaii, and are available for downloading. EVALIDatorReports is a Visual Basic Application that is stored within each Microsoft Access© database containing FIA data for each state. EVALIDatorReports can generate reports for over 40 different types of estimates including forest area, number of trees, growing-stock volume, growth, removals, and mortality. The code for this Visual Basic Application can be viewed, modified, and extended. This approach provides advantages for users interested in creating customized data summaries utilizing external data and/or programs. The system does not require Internet access after the databases are downloaded to a personal computer. This paper contains one simple and three complex examples illustrating how items can be added to the list of classification variables or to the list of estimates within the EVALIDatorReports program.*

KEYWORDS: Forest inventory, database, reporting tool, forest statistics

Background

The needs of most customers of the U.S. Forest Service's Forest Inventory and Analysis (FIA) program can be met by Web applications such as the Forest Inventory Mapmaker program (Miles 2002) and more recently Forest Inventory Data Online (FIDO) (Wilson and Ibes 2008). The reporting capabilities of these tools are continually being improved but they will never be able to meet all the needs of power users. Power users – biometricians, statisticians, and analysts - will always require access to the underlying data.

FIA began providing FIA downloadable datasets in 1996. Initially, datasets for the eastern states were available in Eastwide database format (Hansen et al. 1992) while datasets for the western states were available in Westwide database format (Woudenberg and Farrenkopf 1995). In 2001 a standardized national format for all datasets became available and is known as the FIADB (Forest Inventory and Analysis Program 2008).

---

[1]Research Forester, U.S. Department of Agriculture, Forest Service, Northern Research Station, 1992 Folwell Ave., St. Paul, MN 55108; fax: 651-649-5140, phone: 651-649-5146, e-mail: pmiles@fs.fed.us

In: McWilliams, Will; Moisen, Gretchen; Czaplewski, Ray, comps. 2009. 2008 Forest Inventory and Analysis (FIA) Symposium; October 21-23, 2008: Park City, UT. Proc. RMRS-P-56CD. Fort Collins, CO: U.S. Department of Agriculture, Forest Service, Rocky Mountain Research Station. 1 CD.

Comma separated values (CSV) has been the standard format for dataset delivery. Users wrote scripts to read data from these CSV files into their programs or databases. Mistakes were frequent, such as loading a real number as an integer thereby inadvertently truncating or rounding the real number.

Beginning in February of 2008, FIA datasets became available in Microsoft Access© 2003 format in addition to CSV format. Microsoft Access© is available on most Forest Service computers and is used widely in the academic and research communities. Many databases and software programs can read and correctly interpret data from Microsoft Access© databases, thereby eliminating the formatting problem so often encountered with CSV files.

Microsoft Access© databases containing FIA data and reporting tools have been created for every state except Hawaii and are available for download from http://www.fia.fs.fed.us/tools-data/datamart.html. Each of these state databases also includes a form (i.e., a database tool) that allows users to import CSV data for additional states and create a multi-state database. Unfortunately, due to the two-gigabyte size limit for Microsoft Access© files, multi-state databases are typically limited to two or three states.

Data delivery in Microsoft Access© format creates an opportunity to ensure that results provided by the Forest Inventory Mapmaker and FIDO programs can be duplicated. To that end, a number of Structured Query Language (SQL) scripts were written and stored in the Microsoft Access© databases. These SQL scripts are useful to Microsoft Access© database users and help to verify that both the program used to create the CSV files and the import specifications used to load the data from the CSV files into the Microsoft Access© databases were working correctly.

While the delivery of FIA data in Microsoft Access© format was initially intended solely as a method of data transfer, it soon became apparent that there are additional opportunities to meet customer needs that could not be met by existing Web applications. The FIA program can provide personal open-ended databases with a reporting tool based on peer reviewed estimation protocols (Bechtold and Patterson 2005).

Advantages to having FIA data in personal database

There are several advantages to having FIA data in Microsoft Access©:

1) Archiving – data does not change
2) Availability – data is always available
3) Transparency – data and programs in one place
4) Scalability and flexibility – opportunity to add data and programs to the database

Archiving – A common complaint from users is that the numbers coming out of the Forest Inventory Mapmaker or FIDO have changed since the last time they ran a report. Mapmaker and FIDO use data stored in FIA's corporate Oracle database. The corporate database is not static. Corrections to field data or changes to compilation procedures

result in different answers from the reporting tools. A researcher with a personal copy of the database could choose when or if to update the information.

Availability – Network outages, denial of service attacks, funding cuts, and other unforeseen circumstances can limit access to FIA Web applications. Researchers can download and retain a static copy of the database to use in their research and publications.

Transparency – The code for the reporting tool (EVALIDatorReports) is stored inside the database and can be viewed and modified by the user. The code, data, and reporting tool are all located in a single Microsoft Access© file that can easily be copied and transferred.

Scalability – Users can add to the list of classification variables and the list of estimates. Three examples of extending the reporting capabilities of the database and reporting tools are provided here.

The EVALIDatorReports Reporting Tool

EVALIDatorReports has inputs and outputs that are essentially equivalent to those of the Forest Inventory Mapmaker and FIDO.

The user inputs the following information:

1) Geographic area of interest
2) Optional filters (for restricting the query to a specific ownership, species, etc.)
3) Attribute of interest (timberland area, number of trees, volume, etc.)
4) Classification variables to be used for page, column, and row headings

EVALIDatorReports outputs:

1) Table of population estimates
2) Corresponding table of sampling errors

EVALIDatorReports differs from the Forest Inventory Mapmaker and FIDO in that the user can add to the list of classification variables and to the list of attributes of interest.

Simple EVALIDatorReports Examples

To start EVALIDatorReports, open the Microsoft Access© database and click on the "Forms" object and then double-click on the "EVALIDatorReports" name (Fig. 1).

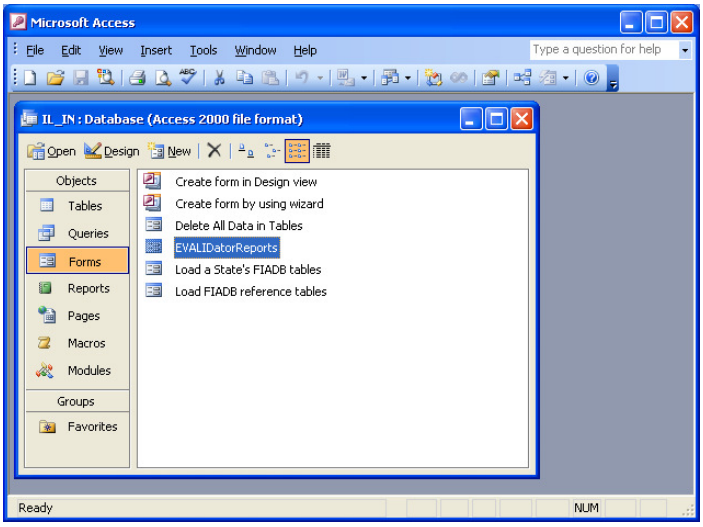

Figure 1: Starting EVALIDatorReports.

The user must first specify the variable to be estimated (Fig.2). In this example the estimate selected is “3 Area of timberland (acres)”. Once the user specifies this variable, a list of inventories available in the database for this estimate appears in the EvalID listbox. In this example, inventories of Illinois and Indiana for the year 2006 were selected from the available list. The retrieval can be further limited to national forest lands by placing the filtering clause “and cond.owncd=11” in the filtering textbox. The user must also specify the three classification variables for page, row, and column. Here a single page report is specified by picking “None” for the page classification variable. This single page report will have “Basal area live tree” classes for the row headings and “Stand-size” classes for the column headings.

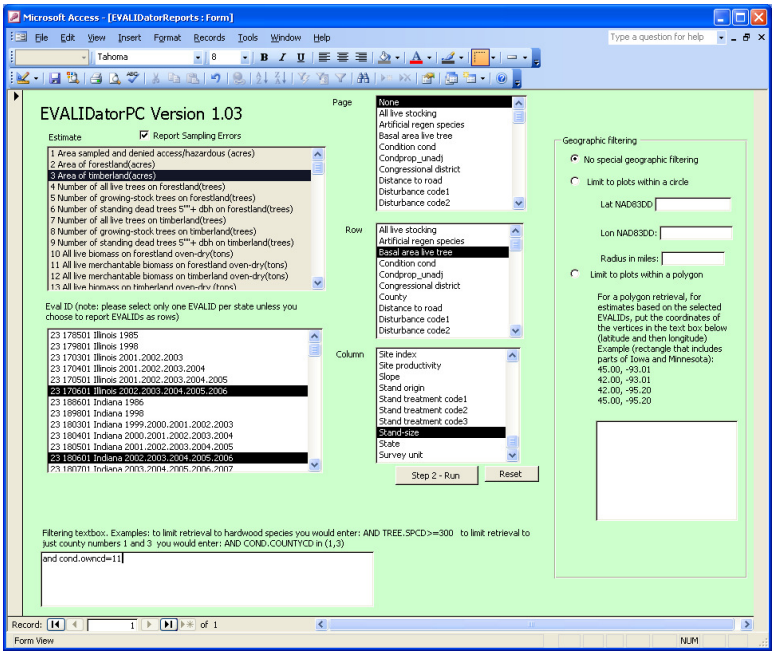

Figure 2: EVALIDatorReports input screen.

The resulting tabular reports (Tables 1 and 2) show that in 2006 there were 449,928 acres of national forest timberland in Indiana and Illinois plus or minus 3.85 percent (one standard deviation).

Table 1. National Forest timberland acres by basal area and stand-size, Illinois and Indiana, 2006.

	Stand-size			
Basal area	**Total**	**Large diameter**	**Medium diameter**	**Small diameter**
Total	449,928	368,894	64,532	16,502
0-40 sqft/ac	12,156	156	5,604	6,395
41-80 sqft/ac	74,137	44,740	19,290	10,107
81-120	211,918	181,456	30,462	-

sqft/ac				
120+ sqft/ac	151,718	142,542	9,176	-

Table 2: Sampling error percent for estimates in Table 1.

	Stand-size			
Basal area	Total	Large diameter	Medium diameter	Small diameter
Total	3.85	5.17	19.84	36.82
0-40 sqft/ac	42.78	101.38	67.74	56.24
41-80 sqft/ac	17.63	24.22	33.84	49.66
81-120 sqft/ac	9.62	10.52	32.95	-
120+ sqft/ac	10.78	11.2	49.43	-

Example 1: Adding a Runtime Classification Variable: Classification variables are stored in the table REF_PRC. To open the REF_PRC table in Microsoft Access© (Fig. 3) click on the "Tables" object and then double-click on the "REF_PRC" name.

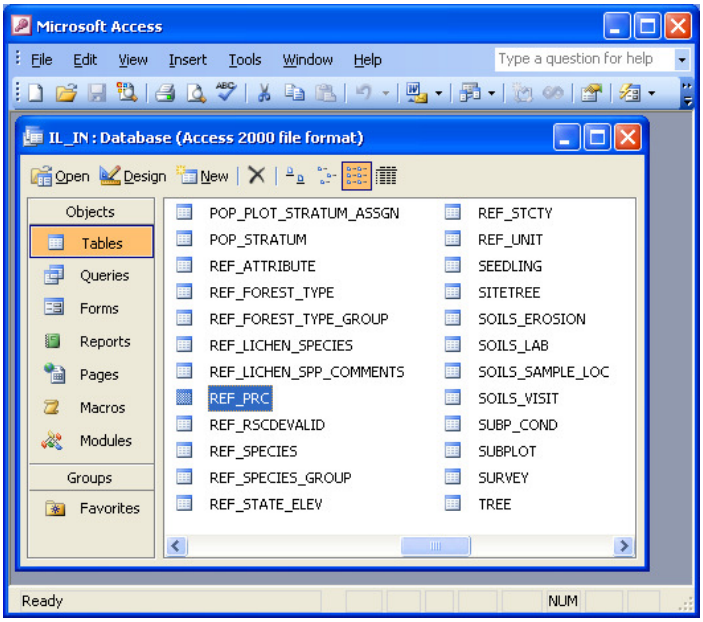

Figure 3: Opening the REF_PRC table in Microsoft Access©.

The values in the Page, Row, and Column lists in the EVALIDatorReports program are drawn from the column CLASSNM in the REF_PRC table (Table 3). In the simple example "Basal area live tree" and "Stand-size" were selected as the classification variables. In the Microsoft Access© table REF_PRC these are just 2 of the 54 classification variables available to the EVALIDatorReports program. A new item can be added to the classification variable list in EVALIDatorReports by adding a new record to the REF_PRC table.

Table 3: A subset of records in REF_PRC table.

CLASS NBR	COND TREE SEED	CLASSNM	FUNCTIONNM	PAGE CLASS	ROW CLASS	COL CLASS
3	COND	Stand-size	stdszcdLabel(cond.stdszcd)	Y	Y	Y
11	COND	Basal area live tree	baliveLabel(cond.balive)	Y	Y	Y

Basal area is commonly used for describing stands in the east. In the west, forest managers often prefer to use stand-density index (SDI). SDI (Long and Daniel 1990) is not currently available in EVALIDatorReports. But it can be added in two simple steps:

- Step 1 - Add the record in Table 4 to the REF_PRC table.
- Step 2 - Add function sdiLabel (appendix A) to the "Functions module" in the Microsoft Access© database (Fig. 4). To open the Functions module click on the "Modules" object and then double-click on the name "Functions module". Then copy the sdiLabel function code from Appendix A into the module. This sdiLabel function generates the classification variable at runtime. For each CONDITION record, all of the trees are selected and an SDI value is calculated for the condition. The resulting SDI is then assigned an SDI class.

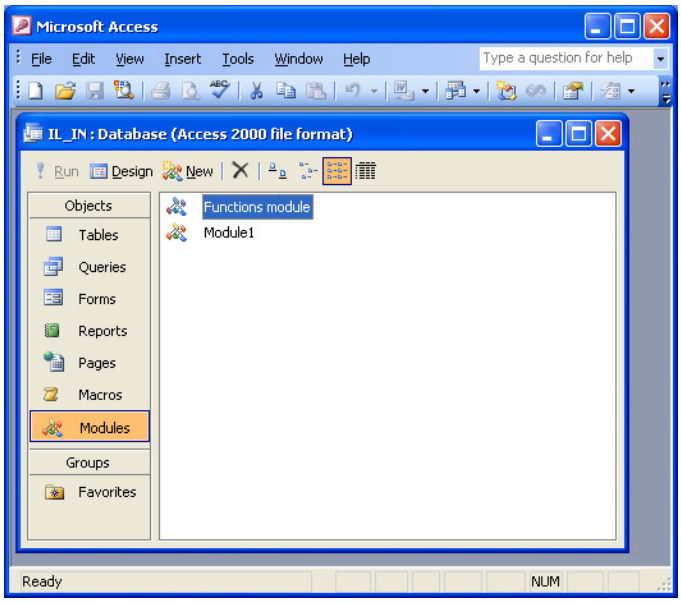

Figure 4: Opening the Functions module in Microsoft Access©.

Table 4: SDI classification variable record to be added to REF_PRC table

CLASS NBR	COND TREE SEED	CLASSNM	FUNCTIONNM	PAGE CLASS	ROW CLASS	COL CLASS
55	COND	SDI	sdiLabel(cond.plt_cn, cond.condid, cond.condprop_unadj)	Y	Y	Y

Now when the simple example is rerun using SDI as the row classification variable instead of "Basal area all live", the output tables provide a slightly different perspective to the land manager (Tables 5 and 6).

Table 5. National forest timberland acres by basal area and stand-size, Illinois and Indiana, 2006.

	Stand-size			
SDI	**Total**	**Large diameter**	**Medium diameter**	**Small diameter**
Total	449,928	368,894	64,532	16,502
less than 100 SDI	15,300	5,771	5,604	3,925
100 to 199 SDI	203,310	153,153	40,487	9,671
200 to 299 SDI	196,777	181,205	12,665	2,907
300 to 399 SDI	28,891	23,115	5,776	-
400 to 499 SDI	5,032	5,032	-	-
500 to 599 SDI	618	618	-	-

Table 6: Sampling error percent for estimates in Table 5.

	Stand-size

SDI	Total	Large diameter	Medium diameter	Small diameter
SDI	3.85	5.17	19.84	36.82
less than 100 SDI	38.78	65.78	67.74	65.83
100 to 199 SDI	9.65	11.6	26.72	51.55
200 to 299 SDI	9.18	9.86	41.26	88.47
300 to 399 SDI	29.2	32.97	66.97	-
400 to 499 SDI	55.86	55.86	-	-
500 to 599 SDI	101.38	101.38	-	-

Example 2: Adding a User-Provided Classification Variable to the FIADB:The state of Minnesota often reports information by Department of Natural Resources (DNR) administrative boundaries. Within the Microsoft Access© database for Minnesota (MN.mdb) a new column can be permanently added to the PLOT table. In this case the new column is called AREA_NAME. By overlaying the plot coordinates (LAT and LON) on an administrative boundary shapefile (Fig. 5) provided by the Minnesota DNR, each plot can be assigned to an administrative unit. The resulting administrative boundary codes are then assigned to the AREA_NAME column of the PLOT table.

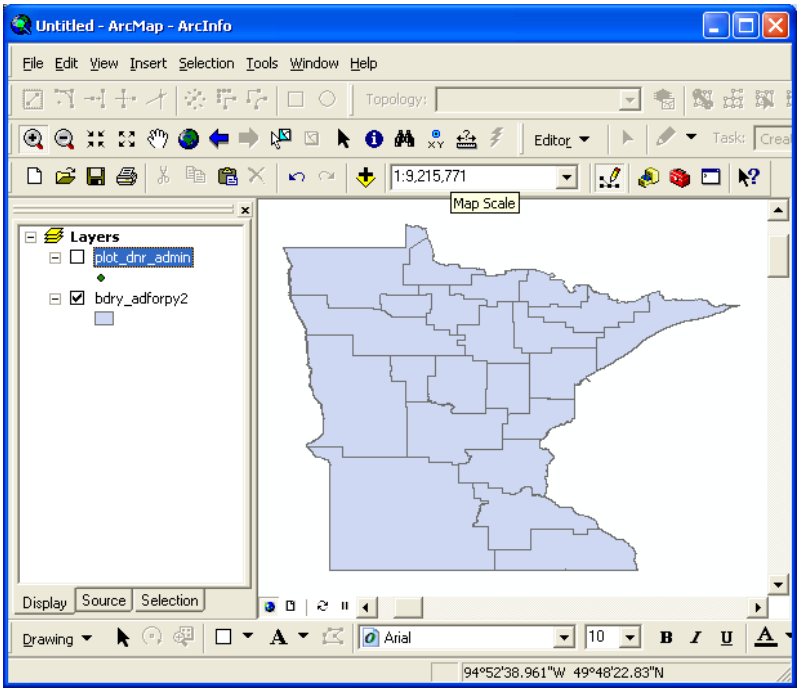

Figure 5: Minnesota DNR Administrative boundaries as viewed in ArcGIS.

Two additional steps are required before the EVALIDatorReports program can report information by administrative boundary. First a record must be added to the REF_PRC table. In this example the record in Table 7 is added to the REF_PRC table.

Table 7: AREA_NAME classification variable record to be added to REF_PRC table.

CLASS NBR	COND TREE SEED	CLASSNM	FUNCTIONNM	PAGE CLASS	ROW CLASS	COL CLASS
56	COND	AREA_NAME	area_nameLabel(plot.area_name)	Y	Y	Y

Then the function area_nameLabel (Appendix b) must be added to the "Functions module". Note that in this case the function simply assigns the area name class. A portion of the resulting output from this retrieval is displayed in Table 8.

Table 8: Area of timberland by DNR administrative area and stand-size, Minnesota, 2006.

	Stand-size	Large diameter	Medium diameter	Small diameter	Nonstocked
Total	15,112,725	4,115,614	5,564,086	5,193,133	239,892
Aitkin Area	757,616	214,109	303,031	229,298	11,178
Backus Area	865,762	284,596	333,551	237,369	10,246
Baudette Area	548,271	66,092	220,576	252,656	8,947
…	….	…	…	…	…

Example 3: Adding a Runtime Estimate: There is a difference between adding a classification variable and adding an estimate. In both cases a function must be added to the "Functions module" However, when adding a classification variable a record must be inserted into the REF_PRC table, while when adding an estimate a record must be inserted into the REF_ATTRIBUTE_ACCESS table.

The REF_ATTRIBUTE_ACCESS table has four columns. A single record of this table is presented in TABLE 9. The first column in the table is ATTRIBUTE_NBR. Currently there are 46 rows in the REF_ATTRIBUTE_ACCESS table so the rows are numbered 1 through 46. The second column is ATTRIBUTE_DESCR. The ATTRIBUTE_DESCR for the row depicted in Table 9 is "All live biomass on forestland oven-dry(tons)". The third column, VBA_SUMFROMWHERE, contains a segment of SQL code used by the EVALIDatorReports program to query the database. The fourth column, PEA_SURROGATE, is blank for all 46 rows that have been predefined. When adding a new row to the REF_ATTRIBUTE_ACCESS table the PEA_SURROGATE column should be filled with the ATTRIBUTE_NBR for the row that most closely reflects what is being estimated. The program will then use the surrogate's PEA records to identify the inventories that can be queried for the new estimate.

Table 9: A single record from the REF_ATTRIBUTE_ACCESS table.

ATTRIBUTE_NBR	ATTRIBUTE_DESCR	VBA_SUMFROMWHERE	PEA_SURROGATE
10	All live biomass on forestland oven-dry(tons)	SUM(tree.TPA_UNADJ*tree.drybiot* IIf(IsNull(tree.dia),PPP.adj_factor_subp, IIf(tree.dia<5,PPP.adj_factor_micr, IIf(IsNull(MACRO_BREAKPOINT_DIA), PPP.adj_factor_subp, IIf(dia<MACRO_BREAKPOINT_DIA, PPP.adj_factor_subp,adj_factor_macr)))))/2000 AS ESTIMATED_VALUE FROM TREE INNER JOIN (COND INNER JOIN ((POP_PLOT_STRATUM_ASSGN	

```
 INNER JOIN
 POP_STRATUM as PPP ON
POP_PLOT_STRATUM_ASSGN.STRATUM_CN = PPP.CN)
 INNER JOIN
 PLOT ON POP_PLOT_STRATUM_ASSGN.PLT_CN =
PLOT.CN) ON COND.PLT_CN = PLOT.CN)
ON TREE.PLT_CN = PLOT.CN
 WHERE ((TREE.STATUSCD)=1) AND
((COND.COND_STATUS_CD)=1) and
TREE.CONDID=COND.CONDID AND
 tree.TPA_UNADJ is not null
and tree.drybiot is not null and
```

The EVALIDatorReports program only allows the user to query those state inventories for a particular estimate where the FIA program says it is appropriate. It is not appropriate, for example, to estimate growth on forest land for inventories conducted prior to 1998 because trees were not always measured on unproductive and reserved forest land. For inventories conducted prior to 1998, it may only be appropriate to estimate growth on timberland. The POP_EVAL_ATTRIBUTE (PEA) table identifies which estimates can be computed for each state inventory. When new estimates are added either new rows must be added to the PEA table or an ATTRIBUTE_NBR for an estimate that has entries in the PEA table must be entered in the PEA_SURROGATE column of the REF_ATTRIBUTE_ACCESS table.

In this example, a user wants to compute all live aboveground tree biomass using Jenkins biomass equations rather than using the stored biomass values that were developed from FIA regional biomass equations. To use the stored biomass values, the researcher would select “10 All live biomass on forest land oven-dry(tons)” from the estimate list in EVALIDatorReports. The program would then use the tree biomass number stored in the DRYBIOT column of the TREE table.

To replace DRYBIOT with a value calculated using the Jenkins biomass equations the user would add a new row to the REF_ATTRIBUTE_ACCESS table (Table 10). The value of ATTRIBUTE_NBR would be 47. The value in ATTRIBUTE_DESCR would be “Jenkins All live biomass on forestland oven-dry(tons)”. The value in VBA_SUMFROMWHERE would be identical to the value from Table 9 with the exception that “tree.drybiot**”** would be replaced with the function call

"JenkinsBiomass(tree.spcd, tree.dia, tree.diahtcd, 0)". The value of PEA_SURROGATE would be 10.

Table 10: Jenkins biomass estimate in the REF_ATTRIBUTE_ACCESS table.

ATTRIBUTE_NBR	ATTRIBUTE_DESCR	VBA_SUMFROMWHERE	PEA_SURROGATE
47	Jenkins All live biomass on forestland oven-dry(tons)	SUM(tree.TPA_UNADJ*JenkinsBiomass(tree.spcd, tree.dia, tree.diahtcd, 0)* IIf(IsNull(tree.dia),PPP.adj_factor_subp, IIf(tree.dia<5,PPP.adj_factor_micr, IIf(IsNull(MACRO_BREAKPOINT_DIA), PPP.adj_factor_subp, IIf(dia<MACRO_BREAKPOINT_DIA, PPP.adj_factor_subp,adj_factor_macr)))))/2000 AS ESTIMATED_VALUE FROM TREE INNER JOIN (COND INNER JOIN ((POP_PLOT_STRATUM_ASSGN INNER JOIN POP_STRATUM as PPP ON POP_PLOT_STRATUM_ASSGN.STRATUM_CN = PPP.CN) INNER JOIN PLOT ON POP_PLOT_STRATUM_ASSGN.PLT_CN = PLOT.CN) ON COND.PLT_CN = PLOT.CN) ON TREE.PLT_CN = PLOT.CN WHERE ((TREE.STATUSCD)=1) AND ((COND.COND_STATUS_CD)=1) and	10

TREE.CONDID=COND.CONDID AND

tree.TPA_UNADJ is not null

and tree.drybiot is not null and

The researcher could then run EVALIDatorReports to generate biomass estimates based on either Jenkins (Table 11) or based on the regional biomass numbers (Table 12).

Table 11: Jenkins all-live biomass (oven-dry tons) on timberland Illinois and Indiana, 2006.

	Species group major		
State	**Total**	**Softwoods**	**Hardwoods**
Total	591,296,419	13,460,746	577,835,673
IL	292,274,945	5,790,358	286,484,587
IN	299,021,473	7,670,388	291,351,086

Table 12: Regional all-live biomass (oven-dry tons) on timberland Illinois and Indiana, 2006.

	Species group major		
State	**Total**	**Softwoods**	**Hardwoods**
Total	486,143,719	11,234,641	474,909,078
IL	234,606,405	4,812,103	229,794,301
IN	251,537,314	6,422,538	245,114,776

Future Developments

New permanent meaningful partnering opportunities may arise from providing FIA data and tools in a completely transparent and open-ended system. Microsoft Access© databases containing FIADB data and the EVALIDatorReports program were downloaded 543 times from the FIA website in the first 5 months. Interest has already been expressed by this small user community in developing a mechanism for sharing code for new classification and attribute variables. Usually future development is driven

by customer demand. In the case of the EVALIDatorReports program, future development may be driven by customer supply.

References

Bechtold, W.A.; Patterson, P.L. 2005. The enhanced forest inventory and analysis program: national sampling design and estimation procedures. Gen. Tech. Rep. SRS-80. Asheville, NC: U.S. Department of Agriculture, Forest Service, Southern Research Station. 85 p.

Forest Inventory and Analysis Program. 2008. The forest inventory and analysis database: database description and users manual version 3.0 for Phase 2, revision 1. U.S. Department of Agriculture, Forest Service, Washington Office. http://www.fia.fs.fed.us/library/database-documentation/FIADB_user_manual_v3-0.pdf.

Hansen, M.H.; Frieswyk, T.; Glover, J.F.; Kelly, J.F. 1992. The Eastwide forest inventory data base: users manual. Gen. Tech. Rep. NC-151. St. Paul, MN: U.S. Department of Agriculture, Forest Service, North Central Forest Experiment Station. 48 p.

Jenkins, J.C.; Chojnacky, D.C.; Heath, L.S.; Birdsey, R.A. 2003. National scale biomass estimators for United States tree species. Forest Science. 49(1): 12-35.

Long, J.N.; Daniel, T.W. 1990. Assessment of growing stock in uneven-aged stands.Western Journal of Applied Forestry. 5:93-96.

Miles, P.D. 2002. Forest inventory mapmaker web-application version 3.0. St. Paul, MN: U.S. Department of Agriculture, Forest Service, North Central Research Station. www.ncrs2.fs.fed.us/4801/fiadb/index.htm [Internet only]

Wilson, B.T;. Ibes, W.S. 2008. Forest inventory and data online (FIDO) web application, version 1.0.0r4. St. Paul, MN: U.S. Department of Agriculture, Forest Service, Northern Research Station. http://fiatools.fs.fed.us.

Woudenberg, S.W.; Farrenkopf, T.O. 1995. The Westwide forest inventory data base: user's manual. Gen. Tech. Rep. INT-GTR-317. Ogden, UT: U.S. Department of Agriculture, Forest Service, Intermountain Research Station. 67 p.

Appendix A – sdiLabel Function

```
Public Function sdiLabel(cn, condid, condprop_unadj)
Dim sSQL As String
Dim sdi As Double
   sSQL = "SELECT sum(nz(tpa_unadj,0)/ " & Str(condprop_unadj) &
"*(nz(dia,0)/10)^1.6)"
   sSQL = sSQL + " FROM tree where plt_cn=""" & cn & """ and condid=" &
Str(condid)

   Dim cnn As ADODB.Connection
   Set cnn = CurrentProject.Connection
   Dim rs As New ADODB.Recordset
   rs.Open sSQL, CurrentProject.Connection
      Do Until rs.EOF
         If IsNull(rs.Fields(0)) Then
            sdi = 0
         Else
            sdi = rs.Fields(0)
         End If
         rs.MoveNext
      Loop
   rs.Close
   cnn.Close
   Debug.Print sdi
   If (sdi < 100) Then
   sdiLabel = "0001 less than 100 SDI"
   ElseIf (sdi < 200) Then
   sdiLabel = "0002 100 to 199 SDI"
   ElseIf (sdi < 300) Then
   sdiLabel = "0003 200 to 299 SDI"
   ElseIf (sdi < 400) Then
   sdiLabel = "0004 300 to 399 SDI"
   ElseIf (sdi < 500) Then
   sdiLabel = "0005 400 to 499 SDI"
   ElseIf (sdi < 600) Then
   sdiLabel = "0006 500 to 599 SDI"
   ElseIf (sdi < 700) Then
   sdiLabel = "0007 600 to 699 SDI"
   ElseIf (sdi < 800) Then
   sdiLabel = "0008 700 to 799 SDI"
   ElseIf (sdi < 900) Then
   sdiLabel = "0009 800 to 899 SDI"
   ElseIf (sdi < 1000) Then
   sdiLabel = "0010 900 to 999 SDI"
   ElseIf (sdi < 1100) Then
   sdiclass = "0011 1000 to 1099 SDI"
```

```
    Else
    sdiLabel = "0012 1100+ SDI"
    End If
End Function
```

Appendix B – Area_nameLabel Function

```
Public Function area_nameLabel(area_name)
  Select Case area_name
Case "Aitkin Area"
area_nameLabel = "0001 Aitkin Area"
Case "Backus Area"
area_nameLabel = "0002 Backus Area"
Case "Baudette Area"
area_nameLabel = "0003 Baudette Area"
Case "Bemidji Area"
area_nameLabel = "0004 Bemidji Area"
Case "Blackduck Area"
area_nameLabel = "0005 Blackduck Area"
Case "Cambridge Area"
area_nameLabel = "0006 Cambridge Area"
Case "Cloquet Area"
area_nameLabel = "0007 Cloquet Area"
Case "Deer River Area"
area_nameLabel = "0008 Deer River Area"
Case "Detroit Lakes Area"
area_nameLabel = "0009 Detroit Lakes Area"
Case "Hibbing Area"
area_nameLabel = "0010 Hibbing Area"
Case "Lake City Area"
area_nameLabel = "0011 Lake City Area"
Case "Little Falls Area"
area_nameLabel = "0012 Little Falls Area"
Case "Littlefork Area"
area_nameLabel = "0013 Littlefork Area"
Case "New Ulm Area"
area_nameLabel = "0014 New Ulm Area"
Case "Orr Area"
area_nameLabel = "0015 Orr Area"
Case "Park Rapids Area"
area_nameLabel = "0016 Park Rapids Area"
Case "Red Lake Reservation"
area_nameLabel = "0017 Red Lake Reservation"
Case "Rochester Area"
area_nameLabel = "0018 Rochester Area"
Case "Sandstone Area"
area_nameLabel = "0019 Sandstone Area"
Case "Tower Area"
area_nameLabel = "0020 Tower Area"
Case "Two Harbors Area"
area_nameLabel = "0021 Two Harbors Area"
Case "Warroad Area"
```

```
area_nameLabel = "0022 Warroad Area"
Case Else
 area_nameLabel = "0023 Other "
End Select
End Function
```

Appendix C – JenkinsBiomass function

```
 Public Function JenkinsBiomass(p_spcd, p_dbh, p_diahtcd, p_wdldstem)
    Dim v_sppgrp     As String
    Dim v_b0, v_b1, v_bm, v_dbh, v_live_aboveground        As Double
    Dim pied, drcp, quga,drcq, stm            As Double

   v_dbh = 2.54 * p_dbh                  'Convert diameter from inches to cm
'-------------------------------------------------------------------------------
'Adjust diameter for woodland species that are measured at diameter root collar not dbh
    If (p_diahtcd = 2) Then
      pied = 0
      drcp = 0
      quga = 0
      drcq = 0
      If (p_wdldstem <= 1) Then
        stm = 1
      Else
        stm = 0
      End If
      If (p_spcd < 300) Then
        pied = 1
        drcp = v_dbh
      Else
        quga = 1
        drcq = v_dbh
      End If
      v_dbh = -6.818 + (1.0222 * v_dbh) + (1.8879 * stm) + (1.8971 * pied) - (0.0399 *
drcp) + (3.11 * quga) - (0.0689 * drcq)
      If (v_dbh < 2.54) Then
        v_dbh = 2.54
      End If
    End If
'-------------------------------------------------------------------------------
'Assign FIA species codes (spcd) to Jenkins species group codes (v_sppgrp)
Select Case p_spcd
    Case 350, 351, 352, 353, 354, 355, 740, 741, 742, 743, 744, 745, 746, 747, 748, _
                749, 752, 753, 754, 920, 921, 922, 923, 924, 925, 926, 927, 928, 929
      v_sppgrp = "aa"
    Case 40, 41, 42, 43, 57, 60, 67, 68, 70, 71, 72, 73, 81, 211, 212, 220, 221, 222, 240,
241, 242
      v_sppgrp = "cl"
    Case 200, 201, 202
      v_sppgrp = "df"
    Case 310, 311, 312, 313, 315, 316, 317, 319, 320, 370, 371, 372, 373, 374, 375, 376,
377, 378, 379
      v_sppgrp = "mb"
```

```
   Case 323, 330, 331, 332, 333, 334, 336, 337, 341, 345, 356, 357, 358, 360, 361, 362,
367, 381, 391, _
            420, 421, 422, 423, 424, 430, 431, 450, 451, 452, 460, 461, 462, 463, 471, 481,
490, 491, 492, _
            500, 501, 502, 503, 504, 505, 506, 507, 508, 509, 510, 511, 512, 513, 514, 520,
521, 522, 540, _
            541, 542, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 555, 561, 571, 580,
581, 582, 583, _
            590, 591, 600, 601, 602, 603, 604, 605, 606, 611, 621, 631, 641, 650, 651, 652,
653, 654, 655, _
            657, 658, 660, 661, 662, 663, 664, 680, 681, 682, 683, 684, 690, 691, 692, 693,
694, 701, 711, _
            712, 720, 721, 722, 729, 730, 731, 732, 760, 761, 762, 763, 764, 765, 766, 769,
770, 771, 772, _
            773, 774, 852, 853, 854, 855, 856, 857, 858, 859, 860, 863, 864, 865, 866, 873,
874, 876, 877, _
            882, 883, 884, 885, 886, 887, 888, 890, 891, 895, 896, 897, 901, 906, 907, 908,
909, 910, 912, _
            913, 914, 915, 919, 931, 934, 935, 936, 937, 940, 950, 951, 952, 953, 970, 971,
972, 973, 974, _
            975, 976, 977, 981, 982, 986, 987, 988, 989, 991, 992, 993, 994, 995, 996, 997,
998, 999, 5091, 5092, 5093, 7211
     v_sppgrp = "mh"
   Case 314, 318, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413,
531, 800, 801, 802, _
        804, 805, 806, 807, 808, 809, 811, 812, 813, 815, 816, 817, 818, 819, 820, 821,
822, 823, 824, 825, _
        826, 827, 828, 830, 831, 832, 833, 834, 835, 836, 837, 838, 839, 840, 841, 842,
844, 845, 846, 847, 850
     v_sppgrp = "mo"
   Case 0, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115,
116, 117, 118, _
   119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135,
136, 137, 138, 139, 140, 142, 143, 144, 298, 299
     v_sppgrp = "pi"

   Case 90, 91, 92, 93, 94, 95, 96, 97, 98
     v_sppgrp = "sp"

   Case 10, 11, 12, 14, 15, 16, 17, 18, 19, 20, 21, 22, 230, 231, 232, 250, 251, 252, 260,
261, 262, 263, 264
     v_sppgrp = "tf"

   Case 50, 51, 52, 53, 54, 55, 56, 58, 59, 61, 62, 63, 64, 65, 66, 69, 300, 303, 304, 321,
322, 475, 755, _
            756, 757, 758, 768, 803, 810, 814, 829, 843, 902, 990
```

```
      v_sppgrp = "wo"
    Case Else
      v_sppgrp = "xx"
    End Select
'--------------------------------------------------------------------------------
'Assign coefficients based on Jenkins species group
    If (v_sppgrp = "aa") Then
      v_b0 = -2.2094
      v_b1 = 2.3867
    ElseIf (v_sppgrp = "mb") Then
      v_b0 = -1.9123
      v_b1 = 2.3651
    ElseIf (v_sppgrp = "mh") Then
      v_b0 = -2.48
      v_b1 = 2.4835
    ElseIf (v_sppgrp = "mo") Then
      v_b0 = -2.0127
      v_b1 = 2.4342
    ElseIf (v_sppgrp = "cl") Then
      v_b0 = -2.0336
      v_b1 = 2.2592
    ElseIf (v_sppgrp = "df") Then
      v_b0 = -2.2304
      v_b1 = 2.4435
    ElseIf (v_sppgrp = "tf") Then
      v_b0 = -2.5384
      v_b1 = 2.4814
    ElseIf (v_sppgrp = "pi") Then
      v_b0 = -2.5356
      v_b1 = 2.4349
    ElseIf (v_sppgrp = "sp") Then
      v_b0 = -2.0773
      v_b1 = 2.3323
    ElseIf (v_sppgrp = "wo") Then
      v_b0 = -0.7152
      v_b1 = 1.7029
    End If
    v_bm = Exp(v_b0 + v_b1 * Log(v_dbh))
'-----------------------------------------------------------------------------------------
'Calculate biomass
    v_live_aboveground = v_bm
    JenkinsBiomass = v_live_aboveground * 2.046 ' Convert kg to pounds

End Function
```