

# Populating the CPT: the Southeast Division's Experience

*[Brit Minor](#), Southern Resource Office*

In 2004, the TNC Southern Resource Office entered ecoregional assessment data for TNC's Southeast Division into the Conservation Planning Tool (CPT). Included were nine terrestrial ecoregions and four freshwater basins. The task was completed in six months; the author spent 100% of his time leading this effort.

These data existed in several different formats: published reports, Microsoft Excel spreadsheets, text files, Microsoft Word documents, and ESRI ArcView shape files. Information from all needed to be incorporated into the CPT, a Microsoft Access database developed by TNC to help standardize ecoregional assessment data and to "roll-up" assessments across the organization.

In the Southeast Division (now part of the Southern U.S. Region), team membership has varied and several project leaders no longer are available for consultation, but we have benefited by having had one data manager, John Prince, for seven of the division's nine assessments. Since some of the region's assessments are now five or six years old, John's memory and archive retrieval skills proved invaluable.

## **Data Preparation**

As a first step, all data for each of the nine ecoregions were converted to formats that could be loaded into MS Access. At this stage, we chose to develop one database per ecoregion. This facilitated subsequent data manipulations for the CPT but allowed table structures to stay close to the ecoregional teams' original, albeit differing, concepts of assessment data. Maintaining data in databases rather than in scattered documents is expected to be of use during subsequent iterations.

Data and report files were copied from their various archives to a consistent folder and file structure, all in one place, with separate folders for each ecoregion and a folder for "All Regions." Readme files document the archive data sources [when properly maintained; see [Lessons Learned](#)]. E-mails that described the archived data were saved alongside the data files or were incorporated into the readme files (see [Tracking the Effort](#)). An example of the file structure is in [Appendix 1](#).

Data were reformatted as needed to allow their import into Access. For example, Excel spreadsheets that had been prepared for publication had their formatting removed; Word tables were exported to Excel or text; and text files were regularized into tab-delimited files with a first row containing field names. Simple Perl scripts were used for some of the more complex search-and-replace text operations.

After being imported into the appropriate ecoregional database, the data tables were further documented within the database in a special table (named "aaaTables" to come first in the database contents). This table contains name, source file, type of data, and a description of manipulations performed for each data table. As best as possible, every data appendix and most report tables were imported and documented for each ecoregion. A sample "aaaTables" is in [Appendix 2](#).

GIS shape file data required their own set of manipulations. In ArcView, all shape files were checked for duplicate polygons. Portfolio site polygons were dissolved into conservation areas, one per site name, prior to further manipulations. Field names that were not unique from others in their first 8 characters, an import requirement for dBase files, were either changed or the fields deleted. After the .dbf components of the shape files were imported into the appropriate ecoregional database, they too were documented in the database.

For ecoregion boundaries, conservation areas, and target occurrences, we were careful to use the same shape files as had been used in the reports. For other regional data that were requested by the CPT, such as

managed areas, land use/coverage, stream length, and population, we used shape files that were consistent across the Southeast Division ecoregions. These files may not have been available to the original assessment teams.

Target taxonomy changed during 1999-2004, the time period during which the nine ecoregional assessments were completed or at least brought to usable draft form (UEGCP; Piedmont is not yet to this state). This was especially true for community targets. To resolve differences, we contracted with NatureServe to update all target names to their current nomenclature. In addition, NatureServe provided us with many of the target-related data items that were requested by the CPT, such as common name, global and other ranks, habitat type, distribution, and association descriptions. The NatureServe contract period was one month, which turned out to be adequate.

Threats and strategies were addressed inconsistently or not at all in the various assessment reports, so we opted to include more recent and consistent results from the Sequencing Conservation Actions Tool (SCAT), in addition to the threats data collected during the ecoregional assessment process. Chris Szell's work on the SCAT database gave the great benefits of resolving conservation area boundary issues between ecoregions and of assigning unique IDs to all Southeast Division conservation areas.

## ***Populating the CPT***

We knew at the outset of this project that the CPT would undergo revisions. Therefore, it seemed prudent not to introduce our own CPT changes. Instead, data from the individual ecoregional databases (described above) were combined into intermediate databases, one for each major section of the CPT: Ecoregions (incorporating Planning Unit, Geographic Units, and Participants), Targets, ConservationAreas, Threats, and Methodology. These intermediate databases link to tables in the individual ecoregional databases and also link to tables in the CPT. Queries were created in the intermediate databases in order to format the data consistently and to rearrange the data to look more like what the CPT expected. Final sets of queries obtained the appropriate ID values from the CPT, then transferred data from the intermediate databases into the linked CPT tables. See [Appendix 3](#) for examples of these queries; also see the examples in *Managing Ecoregional Assessment Data using the Conservation Planning Tool (CPT) Database: Lessons learned*. (Farone et al 2003. draft)

At all stages of this process, validation queries were used to make sure that records and values were neither added nor lost and also to identify inconsistencies among the various assessments. The final validation included reviewing the data in the CPT's entry form. Populating CPT tables became an iterative process. When necessary, CPT records were deleted, data and queries in the intermediate databases were revised, and records were transferred anew to the CPT. To ease this process, queries were named carefully so that the intent and order of running the queries was apparent. Also, all queries for a particular series of operations were collected into Access macros, which ensured that the queries were run in the correct order (see [Appendix 4](#)).

Data for all nine ecoregions and four basins were entered into one CPT database. There are advantages and disadvantages to this approach. Having all data in one place makes it easy to produce descriptive tables, but it also makes it easy to produce misleading analyses of incompatible data. We used data sources that were consistent across ecoregions where feasible, as described above, but it would not be correct to compare results for percentage of goals met or total area protected, for example, across ecoregions of our Southern U.S. Region CPT database.

## ***Ecoregion and CPT ID Fields***

Ecoregional assessment teams came up with their own identifiers for conservation areas and other elements. The CPT generates random IDs for its elements and provides foreign ID fields to relate CPT records back to the original ecoregional data records. We considered the IDs from the original reports to be the important IDs, the ones likely to be familiar to our users. The random IDs used within the CPT are exactly what the database needs internally for its indexes and hash tables, but have no meaning to humans.

Also, the random IDs changed when CPT records were deleted and restored during the CPT population process. Therefore, no effort was made to propagate the random IDs back out to individual ecoregional databases or shape files.

## ***CPT Fields***

Our goal was to populate as many CPT fields as possible, sometimes using data sources that weren't available or considered by the assessment teams. For example, ecoregion descriptions were taken directly from the assessment reports and were quite verbose; population data came from the 2000 census; and stream length was calculated from RFI shape files. Land use/land cover percentages came from the 1992 NLCD. Managed area data came variously from the reports, from the Southern Resource Office, and from state sources that have been updated since the assessments were published. For political units, we opted to include both state and county records and, for Florida, water management districts. Data source decisions were documented primarily in e-mails, all of which were archived in one MS Outlook "personal folders" file, *cpt.pst*, which is stored alongside the CPT.

CPT data fields within the Targets table posed many problems, requiring some detective work to populate successfully. Target taxonomy, as mentioned earlier, was updated by NatureServe personnel. They also helped us to derive some of the CPT field values from the Biotics database. For example, "Species General Taxonomic Group" can be derived from the element ELCODE, as can the habitat type for some groups. For associations, we included the description in the CPT as "Other information." In many cases, we went back to archived files to obtain distribution values. Translation tables in the ecoregional or intermediate databases (see [Appendix 5](#) for an example) helped relate assessment field values to CPT field values [see [Note](#) below]. Rules to identify target selection rationales were written to duplicate those used by expert teams, which often were not documented in the assessment reports. Goal selection comments and essays on ranking criteria simply had to be left blank: we did not have the resources to go clear back to the expert team reports to see if such data existed in the archives.

Target occurrence data is sensitive. We opted not to include element occurrence latitude and longitude. We did obtain some viability and distribution values from EORANK and other fields in the element occurrence shape files.

Data fields within the ConservationAreas table also required careful study, although populating them proved less problematic than fields of the Targets table. Conservation area names had to be unique within ecoregions. For this reason, multi-polygon portfolio sites in the shape files were dissolved into one prior to import, as mentioned above. Also, a few aquatic site names, derived from the Mott-funded Freshwater Conservation Area study, duplicated terrestrial site names. In these cases, " (aquatic)" was appended to the aquatic site name. No site descriptions were provided in most of the assessments, so descriptions were created that summarized the site type, site ID values, and any comments found in input data sources. The most glaring omission in the Southeast Division CPT conservation area table regarded values pertaining to the field "Rationale for Status." For all cases, the value we chose to use was "Expert opinion," because that was the only method reported in the ecoregional assessments. Unfortunately, this Rationale field is of great importance to CLS users, and "Expert opinion" is considered an inadequate response (Jennie Gunther, personal communication).

Participant data were collected from acknowledgement sections and merged with TNC and Heritage staff directories. In a few cases, for key team members, internet searches were conducted to find current addresses.

Planning methodology essays were copied verbatim from assessment reports.

## ***CPT Extensions***

The CPT was extended so that sequencing (SCAT) data could be added for each conservation area. The extensions included a new “Stresses (SCAT)” tab on the conservation areas form, a new table, CASresses, a lookup table, xlkpSCATStressTypes, and a set of fields added to the xUsrConservationAreas table. Only stresses and relative stress scores have been added so far, but we plan to add the other SCAT components, irreplaceability and sequencing rank, in the near future.

Since we opted to include county-level political unit areas, we had to add county names to the xlkpPolAdminUnitName table. We also added Florida water management districts to this table, since those are important multi-jurisdictional units in Florida. The value “State Forests” was added to the xlkpOwnershipSubCategoryOne table.

As part of the documentation process, we added the field ERDataSource to xUsrEcoregionalPlans. The field value is the filename, including path, for the shape file that was used for calculating ecoregion area. This filename could have been assigned as the foreign ID in the EcoregionalPlans table, but we opted to put the ECOCODE (CSRV, EGCP, etc) in that field. Similar extensions were planned to document sources for other fields, but these weren’t added to the CPT due to lack of time. The “Methods” intermediate database (C\_plan9\_methods.mdb) collects all of the ecoregional documentation tables (the “aaaTables”) together into one table. A query transforms this table to yield a table that has one record per ecoregion, with fields that document data sources for all data types. This transformed table will be the source for the extended documentation fields that are expected in the next version of the CPT.

Another useful extension was a mechanism for adding footnotes to individual data values. For example, by right-clicking on the “stream length” field for the Tropical Florida ecoregion, the user could explain the seemingly low value of 387 km: in southern Florida, which is pretty much one extremely wide, slow-moving river, total inundated acreage would be more meaningful than stream length. Unfortunately, time ran out on this project before the footnoting tool could be adequately debugged. The footnoting tool is described further in [Appendix 6](#).

**Note:** while writing this, the author has realized that original, untranslated values could and probably should have been included in user-added fields for all cases where values were standardized for the CPT. This was not done partly due to lack of time and partly due to oversight.

## ***Tracking the Effort***

We devised a spreadsheet to track our progress. The spreadsheet had a column for each ecoregion and rows for each of the many steps of the major tasks: assessment report availability, data source identification, GIS data acquisition, Access database import, manipulations for CPT, CPT import, final quality assurance. The spreadsheet was updated weekly and distributed to team members at our weekly meetings.

All e-mails regarding the CPT were stored in one Outlook “personal folders” file and archived with the database. The file contained folders for each ecoregion as well as folders for such topics as targets and threats. E-mails that documented data sources also were saved as separate text files alongside those data source files.

Problems and unanswered questions would arise occasionally that had to be put aside so that the overall effort could continue. These were documented in e-mails and also, when appropriate, by database queries that were given names starting with “PROBLEM:”.

## ***Lessons Learned***

Populating a new database with archived data calls for at least four sets of skills: information management, database development, quality assurance, and documentation. It’s rare for one person to possess all these skills all the time.

Sources of data must be documented. It's all too likely that they are undocumented in the archives and only identifiable by date or someone's memory. If this is the case, then time must be taken to annotate the archive if the data are to be useful in a national effort like the CPT. This idea is not new; for example, see [The Ecological Information Challenge](#) (Terry Cook, 2000).

Database import and manipulation methods must be documented. Take advantage of the "Description" field for MS Access tables, queries, and macros. At a minimum, the original developer should be able to comprehend the database six months later!

During database development, it's critical to build in error checking and validation queries, but at the end, there must be a final quality assurance step. In software engineering terms, this is Verification and Validation, or V&V. You must verify that all data sources can be traced, and you must validate that the content of the data sources is reflected in the final database.

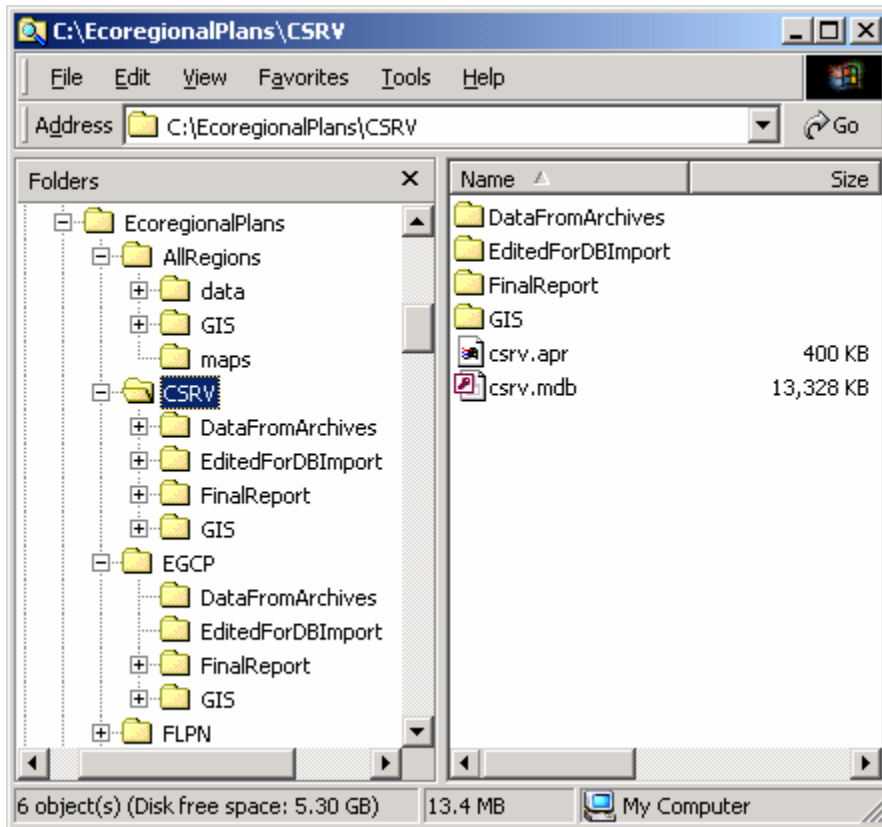
Undocumented data sources and manipulations will slip in despite best efforts. It is important to step back from the details occasionally and look at the whole process.

Set up a foreign ID structure early on in order to improve data source documentation. Check data values early on in order to improve data quality. Don't cover your tracks by deleting intermediate steps before properly documenting them. On the other hand, don't leave behind undocumented, dead trails, either. Tell people what you are doing and make sure that people listen and respond when necessary.

Above all, beware of assumptions.

## Appendices

### 1. File Structure



#### All\_Regions

Overall ArcView project

Separate Access databases for ER, CA, Targets, Threats, Methods.

Cross-boundary data: landcover, target nomenclature, staff directories.

Regional GIS data: counties, ecoregions, managed areas, rivers, census.

#### <Ecoregion>

<ecoregion>.mdb – Access database for ecoregion.

<ecoregion>.apr -- ArcView project for ecoregion.

**DataFromArchives** folder – copy unedited data from non-report sources here, with documentation.

**EditedForDBImport** folder—collect here edited data from report or archive sources, with documentation.

**FinalReport** folder – should contain entire published report, including any extra files such as from a published CD.

**GIS** folder – include all shape files, in multiple projections as needed.

### 2. Access Table Documentation: excerpt from “aaaTables”

ID	Category	Table	Description	Source	Comments
1		(all)		\\EcoregionalPlans\EGCP	Non-spatial data tables are from a set of diskettes stored in John Prince's office. Spatial data were imported from arcdata\egcp on SRO's GIS server; the specific shape files were designated by John Prince.
2	CA	AppA	Appendix A: EGCP Portfolio Site Summary	\\EcoregionalPlans\EGCP\FinalReport\1999 Final Plan\53app_a.xls	Comments were spread over 7 cells in the Excel spreadsheet. After import, the 7 comment fields were concatenated into one memo field by running the macro named "AppA_MakeFullComments".
3		AppH	Appendix H: Additional Species of Conservation Concern	\\EcoregionalPlans\EGCP\FinalReport\1999 Final Plan\53app_h.xls	Access attempted to import STRAT and P1 as numeric fields. After import, the data types for STRAT and P1 were changed to text, and the Excel column data for STRAT was recopied into the Access table to fix import errors. Also, P1 value "2?" was restored. NOTE: the published appendix hid several columns dealing with goals and occurrences. These columns were imported from the spreadsheet, but probably should not be used for any analyses or compilations.
4	CA	AppL	Appendix L: Conservation Areas Evaluation	\\EcoregionalPlans\EGCP\FinalReport\1999 Final Plan\53app_l.doc	The word document was saved as a text file, EditedforDBImport\53app_l_mod.txt. This text file was modified by hand so that all name and county info was on one line per record. Multiple consecutive tabs were translated to a single tab with this one-line perl command, executed from the DOS command line: perl -p -i.bak -e "s/(\t*)/t/g" 53app_l_mod.txt. Commas in acreage values were removed with this perl command: perl -p -i.bak -e "s/(\d+),(\d+)/\$1\$2/g" 53app_l_mod.txt. The modified text file was imported into Access several times and refined further until no import errors were reported.
5	ER	GIS_egcp	Ecoregion GIS	\\SCSE5200\ArcData\egcp\egcp.dbf	per e-mail from John Prince; copied to \\EcoregionalPlans\EGCP\GIS\projected_tncstd

### 3. Sample Queries Used to Load the CPT

The following pair of queries were used to load target records into the CPT. The first query does most of the work. It obtains the ERPID from the CPT's EcoregionalPlans table, then does several manipulations on target rank values. The second query inserts the results into the CPT. These two queries could have been combined into one, but doing it as two allowed the first query to be thoroughly debugged without changing the CPT

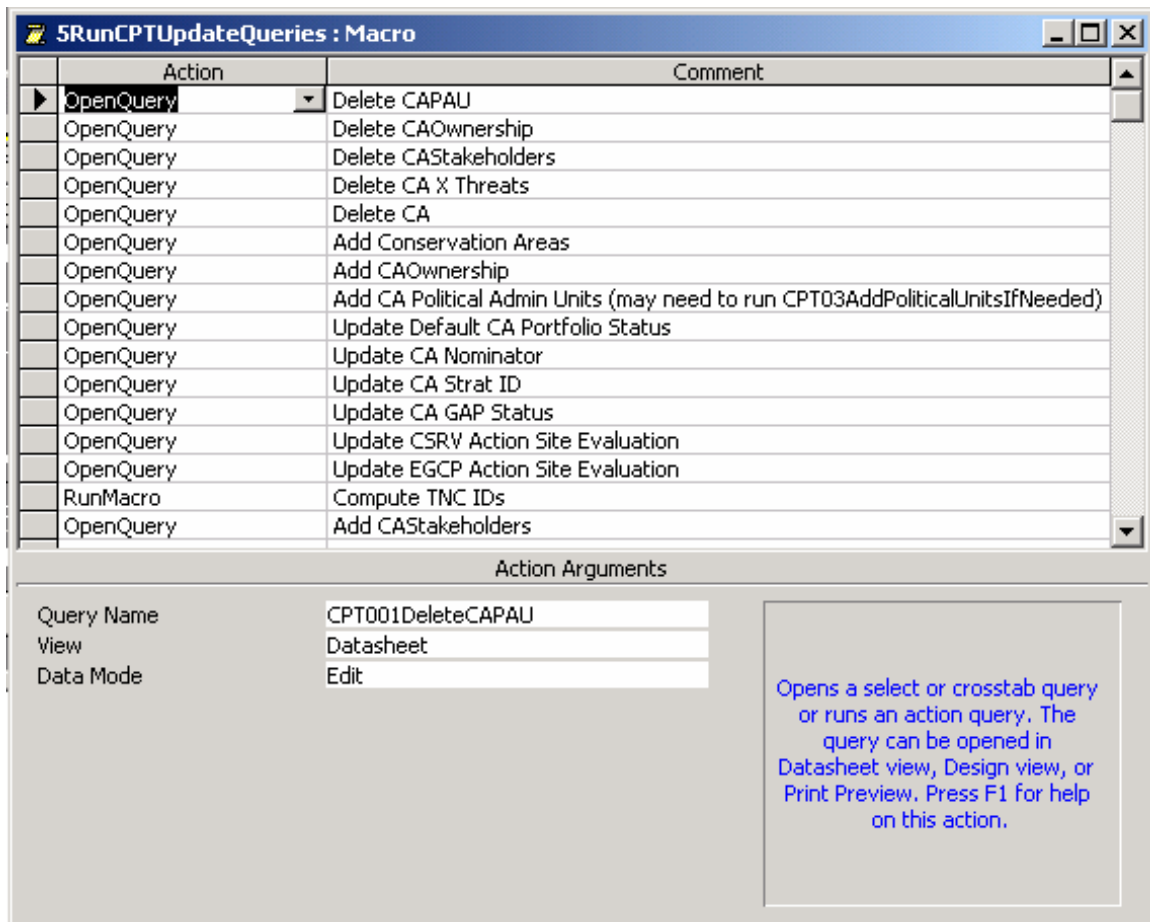
```
SELECT DISTINCT [STD_GNAME] AS SciName,
Iif(Len([CommonName]),[CommonName],[STD_GNAME]) AS CommName, STD_ELCODE AS
ElementCode, Iif(Len([STD_GRANK]),[STD_GRANK],Iif(len([GRANK]),[GRANK],Null)) AS
GlobalRank,
Iif(Len([STD_GRANKRounded]),[STD_GRANKRounded],Iif(len([RoundedGRANK]),[RoundedGRANK],Null)) AS GlobalRankRounded, Iif(Len([USESARank]),[USESARank],Null) AS USESARank,
Iif(Len([IUCN_CD]),[IUCN_CD],Null) AS IUCNRank, Iif(Len([CITES_CD]),[CITES_CD],Null) AS
CITESRank, iif(len(t.EcolType)>0,t.EcolType,'Unknown') AS EcolType, t.Taxon, t.NumGoal, t.MetYN,
t.Captured, t.DataSource, t.SrcID, c.ERID, c.ERPID, t.ERNAME
FROM AllTargetGoalsMatched AS t, CPTecoregionalPlans AS c
WHERE (((CStr(1000+t.ERID))=c.erid));
```

```
INSERT INTO CPTTargets ( TargetScientificName, TargetCommonName, TargetGlobalElementCode,
TargetGlobalRank, TargetGlobalRankRounded, TargetUSESARank, TargetIUCNRank,
TargetCITESRank, TargetLevelBiologicalOrganization, TargetTaxonomicGroup, TargetConservationGoal,
TargetGoalMet, TargetAmountCaptured, TargetForeignID, TargetForeignIDNumber, ERID, ERPID )
```

```
SELECT [SciName], [CommName], [ElementCode], [GlobalRank], [GlobalRankRounded],
[USESARank], [IUCNRank], [CITESRank], [EcolType], [Taxon], [numgoal], [metYN], [captured],
[DataSource], [SrcID], [ERID], [ERPID]
FROM CPT01FindTargets;
```

#### 4. Sample Macro

The intermediate database that's devoted to conservation areas contains the following Access macro. The macro runs all the tasks needed to load conservation area records and related records into the CPT. First, it runs several queries to delete any existing records. It then runs some queries to insert records, using tables that themselves were created by running other macros. These records are updated in the next several macro steps, and the TNC ID is computed for each conservation area by running some VBA code (shown in Appendix 7).



#### 5. Sample Translation Table

The following image shows a simple table used for translating protection status values. These values were stored in several different ways within the assessments, as shown in the ProtStat column. Standardized GAP status values (GapStat) for the CPT were obtained by running a query similar to the following:  
 SELECT gapstat, sedivid, hectares FROM allCAOwnership as ca, TranslateProtectionStatus as t WHERE ca.protstat=t.protstat

	ProtStat	GapStat
▶		Unk
	?	Unk
	0	Unk
	1	L1
	2	L2
	3	L3
	4	L4
	I	L1
	III	L3
	L1	L1
	L1/L2	L2
	L2	L2
	L3	L3
	L4	L4
	*	

Record: 1 of

## 6. Footnoting Tool

The footnoting tool, still in development, allows annotations to be associated with particular data values on a form. It adds one table, tblFootnotes, three macros for interactions with the form, and a VBA module to do the work. To enable the tool, the user adds an item to a global menu, creates a custom right-click menu for the database (preferred method), or adds a button to the form. The current version of the tool works well enough, but has some lingering issues with displaying values on subforms, deleting values and with changing to a different ecoregion without first closing the database. For more information, please [contact the author](#).

## 7. VBA code to compute TNC ID

The following code is designed for CPT v1.5. Add the code to a new or existing Access module. To run the code, create an Access macro that calls ComputeTNCIDs, then run the macro. Note the warning about the AssignID routine!

Option Compare Database

Option Explicit

Public Function ComputeTNCIDs()

' Update all CA records with TNC ID. The ID is computed from centroid

' lat/long using the AssignID routine, which was imported from the CPT.

' IMPORTANT: if the AssignID routine changes in CPT, then it must be

' re-imported here.

On Error GoTo LineError

Dim rstCA As ADODB.Recordset

Set rstCA = New ADODB.Recordset

rstCA.Open "SELECT \* FROM CPTConservationAreas", CurrentProject.Connection, adOpenKeyset,  
adLockOptimistic

rstCA.MoveFirst

```

Dim TNCID As String
Dim CALat As Double
Dim CALong As Double

While Not rstCA.EOF
    If (rstCA("ConservationAreaCentroidLat") And rstCA("ConservationAreaCentroidLong")) Then
        rstCA.Update "CATNCID", AssignID(rstCA("ConservationAreaCentroidLat"),
rstCA("ConservationAreaCentroidLong"), "frmSites")
    End If
    ' Debug.Print rstCA("ConservationAreaCentroidLat"),rstCA("ConservationAreaCentroidLong")
    rstCA.MoveNext

Wend

LineExit:
    On Error Resume Next
    rstCA.Close
    Set rstCA = Nothing
    Exit Function

LineError:
    MsgBox "An unexpected error has occurred. "
        & "For help, report this error message to SRO CPT support: "
        & "CPTmodUtilities:ComputeTNCIDs " & Err.Number & " " & Err.Description
    GoTo LineExit

End Function

Public Function AssignID(dblCenLat As Double, dblCenLong As Double, strForm As String) As String
'Assigns the ID for a Conservation Area (Ecoregional Planning) or a Conservation Site (Site Conservation
'Planning}, which is based on a formula derived from the centroid latitude and longitude.

On Error GoTo LineError

Dim T1 As Integer
Dim t2 As Integer
Dim t3 As Long
Dim t4 As String
Dim t5 As Integer
Dim t6 As Long

Select Case strForm
    Case "frmEcoregionEntry", "frmSites"
        T1 = 10
End Select

If dblCenLong < 0 Then
    t2 = 1
Else
    t2 = 2
End If

t3 = Abs(dblCenLong * 100000)

If t3 < 10000000 Then

```

```
t4 = "0"  
End If  
  
If dblCenLat < 0 Then  
    t5 = 1  
Else  
    t5 = 2  
End If  
  
t6 = Abs(dblCenLat * 100000)  
  
AssignID = CStr(T1) & CStr(t2) & CStr(t3) & t4 & CStr(t5) & CStr(t6)  
  
LineExit:  
    Exit Function  
  
LineError:  
    MsgBox "An unexpected error has occurred. "  
        & "For help, report this error message to SRO CPT support: "  
        & "CPTmodUtilities:AssignID " & Err.Number & " " & Err.Description  
    GoTo LineExit  
  
End Function
```