8. FIELD DATA QUALITY ASSURANCE/QUALITY CONTROL (QAQC I)

Desert tortoise monitoring QA/QC is accomplished in three phases, of which the field data collection teams are responsible for the first phase, or QA/QC I. The goal of QA/QC I is to provide the most accurate, valid and complete data to USFWS. Each and every crew member is responsible for QA/QC I. As a field monitor you are responsible for double-checking your data at each entry as well as reviewing it at the end of each transect for completeness and correctness before submitting it. At this point your team's QA/QC specialist receives your data and downloads it into a Collection database that contains all your fellow crew members' data. To keep an unaltered copy of this Collection database, it is imported into the Contractor database before edits are identified and made. The QA/QC specialist uses all tools (provided scripts and visual inspection) to search the data for inconsistencies and then corrects them. In the process of correcting the original field data, your QA/QC specialist may only need to review your paper data sheets. In other cases, the specialist will need your assistance in correcting errors in your data. The outline below details the individual objectives and standards each data collection team must meet. Field technicians and QA/QC specialists need to work together to meet the goal of these objectives: to deliver correct and complete data to benefit recovery of the desert tortoise.

Objective 1: Limit and correct data entry errors
Each field technician will 1) understand data required under every data field; 2) enter paper and electronic data in parallel; 3) use these data forms in tandem to check for completion and correctness at data entry, at transect completion, and when reviewing data from other teams; 4) understand each QA/QC check; and 5) understand checks that will be performed at later stages of QA/QC.

Objective 2: Execute QA/QC scripts
Each QA/QC specialist will be trained to 1) import field data, 2) understand conceptually how the scripts work, 3) understand each QAQC check, and 4) understand how and where errors are logged

Objective 3: Identify errors that are not addressed by QA/QC scripts
QA/QC specialists will be trained to 1) systematically work with each data table to visually identify errors.

Objective 4: Correct errors before data delivery
Each QA/QC specialist will be trained to 1) identify error records, 2) add error records for errors not identified by QA/QC scripts, 3) identify if error is correctable by contacting crew member or using paper data sheets, 4) correct errors, and 5) update the error table fields (old_value, new_value, resolution, resolver and the error_status field to explain action taken.

Objective 5: Database back up and delivery
Each QA/QC specialist will be trained to 1) back up database to external device daily, and 2) send the database to appropriate personnel by uploading to FTP sites.

Objective 6: RDA recovery from system failure
Each QA/QC specialist will be trained to recover RDA units in case of total system failure. This

includes, 1) performing hard reset, 2) setting date and time, 3) syncing RDA and installing Pendragon forms software, 4) pairing GPS, 5) loading field data collection forms, and 6) testing the RDA setup.

Metrics: Each QA/QC specialist will be tested on their understanding of basic QA/QC process operation and ability to restore a faulty RDA. They will be given several RDA units to download data from, they will run QA/QC scripts, correct identified errors, identify additional data errors and correct them, backup the database and send it to appropriate personnel. In addition, they will be required to restore a faulty RDA. They will be re-trained if they are not proficient with the QA/QC process and restoring RDA. Continued delivery of poor quality data will result in the recommendation to USFWS that they not participate in monitoring.

Objective 1: Limit and Correct Data Entry Errors

1. **Crews proof their RDA and paper data sheets for missing entries every day.**

   In so doing they should find and complete any blank data fields assuming an answer can be found on the paper data sheet. If they cannot complete a blank data field it should be reported to the QA/QC specialist when data from their RDA is downloaded.

2. **Crews compare their RDA and paper data sheets for discrepancies.**

   If possible, discrepancies between the two should be resolved immediately. Any discrepancies that you are unable to resolve should be reported to your QA/QC specialist when data are transferred. These errors should be recorded in the discrepancy spread sheet so they do not continue to cause the QA/QC specialists to expend further effort.

3. **Crews will be apprised of identified issues so they can correct the way they enter data for future deliveries.**

   All resolvable, correctable, or fixable (synonyms used for emphasis!) data entry issues should be resolved before submitting the data to MDEP. With each passing day issues such as these become more and more difficult to fix.

Objective 2: Executing QA/QC Scripts

1. **Import Collection (Pendragon) database into the Contractor (QA/QC) Database**
    a. Open the QA/QC scripts database by double clicking "Import_QAQC.mde" file.
    b. Select the "Import data" tab on top if not already selected. Click on "Import Raw Data" to import the Collection database into the Contractor database. After importing, certain fields are automatically populated in the Contractor database. For instance, "tran_num" is populated in sub- forms; latitude and longitude are calculated from GPS grabs, etc.
    c. All the errors encountered during the import process are logged into ("LDS_Import_Log.txt") file located in "C:\Program Files\Forms3". This file must be checked after each import to see if any errors were encountered during import process

    NOTE: Before running the QA/QC scripts for the first time make sure that the Contractor database (Import_QAQC.mde) exists in the same folder as the Collection database (C:\Program Files\Forms3). If not, copy the latest version of the Contractor database in this folder.

2. **Understand how to execute QA/QC checks**
    a. To run the QA/QC checks select the "Transect Database Scripts" or "Training Database Scripts" tab on top. Select the desired checks to be performed and then click the "Run selected QA/QC checks" button.
    b. If updates are made to tran_num, stratum, team_num fields in Transects table or date and G0_site fields in G0_Start table, or trial_number, team_number, training_line_color, transect, training_date, transect_bearing fields in Tran_Train table the copied fields in the subform can be updated automaticallyby clicking the "Re-Calculate Fields" button.

3. **Understand Each QA/QC Check**
    a. **Relationship Checks**
        i. **Missing Waypoints**
        This checks if a xxx format transect is missing waypoints 0, 1, 99, 100 and if xxx.x format transect is missing waypoint 99.
        Tables included in check: *Waypoints*
        Error logged: *"Transect xxx: missing Waypoint xx"*

        ii. **Orphan Records**
        This checks for orphan records created after deleting a parent record without first deleting the related child records.

Tables included in check: *Waypoints, OppCarcObs, OppLiveObs, TranCarcObs, TranLiveObs, g0_Obs, g0_OppLiveObs.*
Error Logged: *"Orphan Record Found"*

b. Duplicate Checks

i. Transect numbers

This checks for duplicate transect numbers in transect table.
Tables included in check: *Transect*
Error Logged: *"Duplicate Transect: Count=x"*

ii. Waypoints for transect

This checks if duplicate waypoints are present for a transect.
Tables included in check: *Waypoints*
Error Logged: *"Duplicate Waypoint x Transect: xxx"*

iii. Transects

This checks if a team has recorded a duplicate transect for the same trial number in training database.
Tables included in check: *Train_Tran*
Error Logged: *"Duplicate Training Transect x for team x and trial x"*

iv. Tortoise ID

This checks if a team has recorded a duplicate tortoise in one day
Tables included in check: *Train_Obs*
Error Logged: *"Duplicate TortoiseID for team x on date x"*

c. Spatial Condition Checks

i. UTM Zone

This checks for missing or incorrect UTM zone. An error is logged if sites other than "BD", "BD2" and "GB" have UTM zone 12.
Tables included in check: *Waypoints, OppCarcObs, OppLiveObs, TranCarcObs, TranLiveObs, G0_Obs, G0_OppLiveObs.*
Error Logged: *"Missing UTM Zone" or "Incorrect UTM Zone"*

ii. Easting and Northing

This checks if the spatial location of the record is within the strata boundary buffer. An error is logged if the co-ordinates are not within 3km buffer around the strata boundary.

Tables included in check: *Waypoints, TranCarcObs, TranLiveObs, G0_Obs.*
Error Logged: *"Incorrect Easting co-ordinates for stratum" or "Incorrect Northing co-ordinates for stratum"*

iii. **Missing Location Data**
This checks for records missing GPS grab and hand entered co-ordinates. An error is logged if GPS grab as well as manual entry co-ordinates are null.
Tables included in check: *Waypoints, OppCarcObs, OppLiveObs, TranCarcObs, TranLiveObs, G0_Obs, G0_OppLiveObs.*
Error Logged: *"Missing Location data"*

d. **Attribute Condition Checks**

i. **Lead and Follow**
This checks if there are inconsistencies between observer1 and observer2. An error is logged if (a) observer1 or observer2 is missing or (b) if observer1 and observer2 are the same.
Tables included in check: *Transects.*
Error Logged: *"Missing Observer" or "Same Observer1 and Observer2"*

ii. **Existing Tag information**
This checks if there are inconsistencies between existing tag and new tag. An error is logged if (a) existing tag is "yes" and tag attached is "yes" or (b) if existing tag is "unknown" and new tag attached is "yes" or (c)if existing tag is "no" and new tag attached is "no"
Tables included in check: *OppLiveObs, TranLiveObs, G0_OppLiveObs.*
Error Logged: *"Inconsistency between existing tag and new tag attached"*

iii. **Incorrect Time**
This checks for records with incorrect time. An error is logged if the time is before 5:00AM or after 6:00PM.
Tables included in check: *Transects, Waypoints, TranCarcObs, TranLiveObs, G0_Start and G0_Obs.*
Error Logged: *"Incorrect Time (valid: 5:00AM - 6:00PM)"*

iv. **MCL Values**
This checks for inconsistencies between mcl>=180, mcl_mm, mass_g, location and carc_condition fields.

1. mcl_mm and mcl_greater_180

An error is logged if (a)mcl_mm > 180 and mcl_greater_180 is "no", or (b) mcl_mm < 180 and mcl_greater_180 is "yes", or (c) mcl_mm > 0 and mcl_greater_180 is "unknown" or (d) if mcl_mm is 0

2. mcl_mm and mass_g
   An error is logged if (a) mcl_mm > 0 and mass_g <= 0 or (b) mcl_mm =< 0 and mass_g >0
3. location, mcl_mm and mass_g
   An error is logged if location is not burrow and mcl_mm or mass_g is null.
4. Carc_condition and mcl_mm
   An error is logged if (a) carc_condition is "Intact" and mcl_mm is "null" or (b) if carc_condition is "Disarticulated" and mcl_mm > 0

Tables included in check: *OppCarcObs, OppLiveObs, TranCarcObs, TranLiveObs, G0_OppLiveObs.*
Error Logged: *"Inconsistency between mcl_mm and mcl_greater_180, or mcl_mm =0 or mcl_mm not entered" or "Inconsistency between mcl_mm and mass_g" or "Inconsistency between location, mcl_mm and mass_g" or "Inconsistency between carc_condition and mcl_mm"*

v. **Visibility**
This checks inconsistency between visibility and behavior. An error is logged if invisible tortoise has behavior other than "unknown"
Tables included in check: *G0_Obs.*
Error Logged: *"Inconsistency: tortoise visibility and behavior"*

vi. **Observer name and position**
This checks if the observer name and observer position does not match lead and follow in training database.
Tables included in check: *Train_Tran, Train_Obs*
Error Logged: *"Observer name and position not consistent in Train_Tran and Train_Obs tables"*

vii. **Transect Segment Number**
This checks if the transect_seg_number is not matching the calculated value.
Tables included in check: *Train_Obs*
Error Logged: *"Transect_seg_number in incorrectly calculated."*

viii. **Time**

This checks the time values in training database and error is logged if (a) Training start time is not before training end time or (b) Observation time not between training start time and training end time

Tables included in check: *Train_Tran, Train_Obs*

Error Logged: *"Inconsistency between Training start and end time" or "Observation time not between start time and end time"*

ix. **Radial Distance**

This checks if the radial distance has more than one decimal

Tables included in check: *Train_Obs*

Error Logged: *"Radial Distance contains more than one decimal"*

x. **Bearing**

This checks if the local bearing not within 40 degrees of transect bearing.

Tables included in check: *Train_Obs*

Error Logged: *"Local Bearing not within 40 degrees of Transect Bearing"*

xi. **Perpendicular distance**

This checks the following conditions on perpendicular distance. An error is logged if (a) if the perpendicular distance greater than radial distance or (b) If the perpendicular distance is greater than 25m

Tables included in check: *Train_Obs*

Error Logged: *"Perpendicular distance greater than radial distance" or "Perpendicular distance greater than 25m"*

xii. **Burned**

This checks if burned field is not null for sites other than "Coyote Springs".

Tables included in check: *G0_Obs*

Error Logged: *"Burned field should be null for sites other than CS"*

4. Understand How and Where Errors are Logged

After executing the QA/QC checks all errors encountered are logged into the Errors table. For subsequent QA/QC checks, all uncorrected errors will be logged again into the Errors table unless the error status is marked as "exception allowed" to errors that cannot be resolved.

Objective 3: Identify errors that are not addressed by QA/QC scripts

The following lists illustrate checks for the most common errors that are not identified by provided scripts. This list is not exhaustive, but does illustrate one way to methodically examine data tables for missing or inconsistent values. Errors/inconsistencies are identified by sorting one or more fields to find the described issues. These are reported out by table and with identification of individual records using the TableRecord_ID field (The field identifying records is built off the appropriate name for each table.)

Checks common to all tables

1. Check that timestamps are valid. Sort by time (ascending and descending).
2. Check numeric fields for errors (e.g. decimals after numbers, zero preceding number). Sort by field.
3. Check if manual UTM zone values or manual coordinates are entered when successful Bluetooth GPS grab has also been indicated. Sort by Grab then manual UTM field. If both Bluetooth grab and manual grab are recorded then comments should specify which grab is correct.

Transects table

1. Check if do_time, tran_start_time, tran_end_time and ret_do_time are null (1:00AM). Sort by time.
2. Check if Transect_Standard field is null. Sort by field.
3. Check for the accuracy of Transect_Standard field. Visual check of transect waypoints converted to lines in GIS software. Display by Transect_Standard. Only typical square 12km transects should be Transect Standard = Y.
4. Check for invalid entries for “date”. Sort dates ascending then descending.
5. Check do_time, tran_start_time, tran_end_time and ret_do_time for accuracy. For example, do_time should be earlier than tran_start_time, etc.
6. Compare date and time fields with PDA date/time stamp.

Waypoints Table

1. Lead or Follow fields should be null only for waypoints 0, 99 and 100.
2. Check for waypoint numbers that are missing or invalid (skipped, duplicate, or with decimal places).

OppCarcObs Table

1. Check for the accuracy of detection numbers.
2. mcl_mm should be null for disarticulated carcass and mcl_mm should be > 0 for intact carcass.

OppLiveObs Table

1. Check for the accuracy of detection numbers.
2. If tortoise location is burrow, burrow_visibility and tortoise_in_burrow_visibility should not be null. For other locations the tortoise_visibility field should not be null.
3. The mcl_mm and mass fields should be null if the temperature is greater than 95°F.
4. If new tag attached is "Yes" then new tag number should not be null and if existing tag is yes then existing tag number should not be null.

TranCarcObs Table

1. Check for the accuracy of detection numbers.
2. If the carcass condition is Disarticulated then mcl_mm should be null.
3. If the carcass condition is Intact then mcl_mm should not be null.
4. If existing tag is "Yes" then existing tag number should not be null.

TranLiveObs Table

1. Check for the accuracy of detection numbers.
2. If tortoise location is burrow, burrow_visibility and tortoise_in_burrow_visibility should not be null. For other locations the tortoise_visibility field should not be null.
3. The mcl_mm and mass fields should be null if the temperature is greater than 95°F.
4. If the tortoise is in the "Open" then the mcl_mm and mass should not be null.
5. Check if the existing tag is "No" and new tag attached is "No".
6. Sort ascending and descending on both mcl_mm and mass and look for unrealistic discrepancies.
7. Check for New tag number format. Should be "FWxxxx". Should not be "FW-xxxx".

G0_Start Table

1. Check for start_time or end_time null values (1:00AM). Sort ascending and descending.

G0_OppLiveObs Table

1. Check for the accuracy of detection numbers.
2. If the tortoise location is burrow, check if the burrow_visibility and tortoise_in_burrow_visibility should not be null. For other locations the tortoise_visibility field is not null.
3. The mcl_mm and mass fields should be null if the temperature is greater than 95°F.
4. If new tag attached is "Yes" then new tag number should not be null and if existing tag is "Yes" then existing tag number should not be null.

G0_Obs Table

1. The burned field should be null for all sites except CS.
2. If the tortoise is not visible the behavior can only be unknown, digging or moving.
3. If the tortoise is in open then behavior should not be unknown.
4. If the tortoise location is burrow, check if the burrow_visibility and tortoise_in_burrow_visibility should not be null. For other locations the tortoise_visibility field should not be null.

Objective 4: Correcting Errors before Data Delivery

1. **Identify Error Records**
   After the QA/QC checks are run, all the errors encountered are logged into the Errors table in Contractor database. Following are the fields present in the Errors table. Fields (a)-(h) are automatically populated by the scripts and fields (i)-(m) are manually entered by QA/QC specialist after correcting error.

   a. ID – auto-number used to identify record
   b. date – the date when the QA/QC scripts were run
   c. table_name – table name where error was found.
   d. prime_key – primary key for the table with error record(for transect table – tran_prime_key, for waypoints – wp_key, for OppCarcObs – OppCarcObs_key, etc)
   e. tran_num – transect number for error record, -99 if not applicable
   f. stratum – stratum for the error record, null if not applicable
   g. team_num – team number for the error record, -99 if not applicable
   h. error_desc – short description on type of error found
   i. old_value – old incorrect value of the field
   j. new_value – new correct value of the field
   k. resolution – null, filled in while correcting errors
   l. resolver – null, filled in while correcting errors
   m. error_status – null, filled in while resolving errors

   The error records can be identified using the information in table_name, prime_key, tran_num, stratum and team_num fields in the Errors table.

2. **Add error records for errors not identified by QA/QC scripts**
   For errors that were identified during systematic visual inspection of tables, there will not be an automatic error record logged into the Errors table. A manual error record will need to be entered. For each manually entered record, the QAQC specialist will need to populate fields (b)-(h) above. Care should be taken to enter the information accurately as these fields are critical for identifying the correct record.

3. **Identify if Error is Correctable**
   After identifying the error record, a decision must be made if the error can be corrected or not. To determine if the error can be corrected, review the paper datasheets or contact crew members, if necessary.

4. **Correct Errors in the Contractor Database**
   If the error can be corrected, the error record identified should be corrected in the corresponding table in the Contractor database.

5. **Update the error table resolution fields and the record's error_status field to explain action taken**

After the error has been fixed or determined that it cannot be fixed, the resolution, resolver and status fields in the Errors must be manually filled in for the corresponding error record. The "resolution" column should clearly mention steps taken to correct the error or should explain why the error could not be corrected. The "old_value" columns should contain the original incorrect value of the field and "new_value" column should contain the correct value replaced.. For example if the time field is corrected, the resolution should be "changed time to match the datasheet" and the "old_value" field should contain 6:00PM and "new_value" should contain 6:00AM. If the error cannot be corrected, the "resolution" column should explain why. The resolver column should contain the name of the person/agency correcting the error. The error_status column should contain "resolved" if the error was corrected or "exception allowed" if the error cannot be corrected. Marking the status to "exception allowed" will ignore the error and it will not be logged again in subsequent QA/QC checks.

Objective 5: Database Backup and Delivery

1. **Backup Collection (Pendragon) database**
   The Collection database should be backed up daily on external storage device. The backup should be performed at the end of the day when the new data collected on all RDA's have been imported (hotsynced) into the Collection database.
   The Collection database file "FORMS32K.MDB" is located in "C:\Program Files\Forms3" folder on the machine used for hot-syncing RDA's. To back up the database copy this file ("FORMS32K.MDB") onto external storage device and then rename the file on external storage device to "FORMS32K_MM_DD_YY.MDB". Do not rename the source file found in "C:\Program Files\Forms3".

2. **Deliver Collection Database**
   The Collection database backed up on the external storage device ("FORMS32K_MM_DD_YY.MDB") should be uploaded to MDEP FTP site. After the database has been uploaded, an email should be sent to appropriate personnel about data delivery. Details on accessing and uploading data on MDEP FTP site will be provided by MDEP.

3. **Backup Contractor Database**
   The contractor database should be backed up after each QA/QC session. QA/QC session consists of importing new data from the Collection database, running QA/QC checks and correcting all the errors found in Errors table.
   The contractor database file "Import_QAQC.mde" is located in "C:\Program Files\Forms3" folder on the machine used for hot-syncing RDA's. To back up the database copy this file ("Import_QAQC.mde") onto external storage device and then rename the file on external storage device to "Import_QAQC_MM_DD_YY.MDE". Do not rename the source file found in "C:\Program Files\Forms3".

4. **Deliver Contractor Database**
   The Contractor database backed up on the external storage device ("Import_QAQC_MM_DD_YY.MDB") should be uploaded to MDEP FTP site. After the database has been uploaded, an email should be sent to appropriate personnel about data delivery. Details on accessing and uploading data on MDEP FTP site will be provided by MDEP.

5. **Deliver Paper Datasheets**
   Each week, QA/QC specialists send photocopies or scanned versions of that week's paper datasheets to MDEP. Originals are kept with the QA/QC specialist to add necessary annotations when data errors and inconsistencies are addressed. On the date of final data delivery, all datasheets, with annotations as needed, must be submitted along with electronic data. Note that it may be reasonable and convenient to send original datasheets (with any annotations) to MDEP after a few weeks have passed and the associated electronic data have been through a couple weeks of independent evaluation by FWS.

Before sending in original datasheets, and if the QA/QC specialist chooses to submit scanned datasheets rather than copied ones, all datasheet pages for a given transect are scanned together into one document. The file naming convention is:

transect number_stratum_year	Example: 719_PT_2007

G_0 datasheets should be treated the same way - all data sheets for a given day scanned into one document. The file naming convention in this case:

G_0 site_two-digit-monthdayyear	Example: CK_042807

If crews complete G_0 during multiple time periods in a given day (0530 - 1000, 1200 - 1330, 1530 - 1700), the naming convention should be

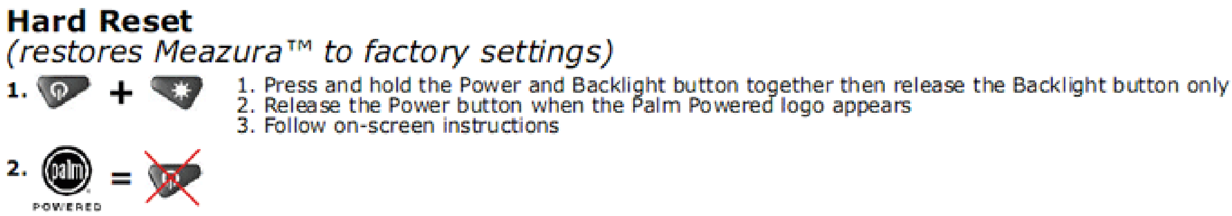
G_0 site_two-digit-monthdayyear_timegroup	Example: CK_042807_A (B for second period and C for third period if necessary etc.).

After the scanned sheets have been uploaded or the weekly packet of copied sheets have been sent, an email should be sent to appropriate personnel at MDEP and FWS announcing data delivery. Details on accessing and uploading data on MDEP FTP site will be provided by MDEP.

Objective 6: RDA Recovery from System Failure

This procedure should only be performed by QA/QC personnel if soft reset as well as hard reset with system restore does not work. This method clears all the data from RDA and PC and then does a fresh install of all software.

1. **Hard Reset**
   Perform a Hard reset on RDA.



2. **RDA Setup**
   Follow on screen instructions to complete the RDA setup. Make sure it displays correct date and time, if not then change it to current date and time.
3. **Delete Palm Desktop Account from PC**
   On PC, start Palm desktop application by clicking start->Programs-> Palm Desktop and then Palm desktop. On the right hand top corner you will see users dropdown, click on the dropdown and select “Edit Users”. Click on the current RDA user and click “Delete”. Select “Yes” for deletion confirmation. After this, close the Palm Desktop application.
4. **Setup RDA hotsync**
   On RDA, perform hotsync. It will ask for a User account on PC. Click “New” to create a new account and enter the name of the RDA.
5. **Install Pendragon database**
   After previous hotsync is successfully completed, on PC click “Install Forms 5.1 on Palm OS Device” found windows programs. (Start->programs->Pendragon Forms 5.1). Select the new user created in the previous step and click “Done”. Perform HotSync on RDA and you should see the Forms 5.1 on RDA.
6. **Install FormsGPS and Mathlib files.**
   On PC, navigate to “C:\Program Files\Forms3 folder and double click on “FormsGPS.prc” file. Select the new user created and click “Ok” and “Done” on the next dialogs. Similarly install the “Mathlib.prc” file from “Mathlib” folder. Perform Hotsync on RDA to install the files.
7. **Load LDS 2009 forms**
   On RDA, open “Forms 5.1” if “Transects” is not visible then make sure the RDA username exists in “Users” and “Default Group” in “Pendragon Forms Manager”. Perform Hotsync on RDA. You should now see all the forms on the RDA.
8. **Pair Bluetooth GPS with RDA**
   Instructions for pairing can be found in “Pendragon Objective4” document.
9. **Test RDA Setup**
   Test the RDA setup to confirm that all the forms and subforms are accessible and confirm that the perpendicular distance calculation and GPS grabs are working.