

Guam Coconut Rhinoceros Beetle Eradication Project



Technical Note: Using QGIS to Detect Georeferencing Errors in an Online MySQL Database

Prepared by
Aubrey Moore
University of Guam Cooperative Extension Service

March 30, 2012

Abstract

The Guam Coconut Rhinoceros Beetle Eradication Project uses an online MySQL database to keep track of trapping data and observations. Point locations for traps and observations are stored in fields for latitude and longitude stored as decimal degrees. This note describes use of QGIS, an open source geographical information system, to find points which do not lie within Guam's shoreline. A QGIS plugin named eVis is used to connect directly to the MySQL database to download and visualize georeferenced trap and observation data. A QGIS project and associated data which demonstrate the methods can be downloaded from <http://guaminsects.net/anr/content/technical-note-using-qgis-detect-georeferencing-errors-online-mysql-database>.

1 Introduction

The project's online MySQL database uses georeferenced tables (**trap** and **crb_obs**) to keep track of trapping data and observations. Each table contains fields containing latitude and longitude as decimal degrees. Location of traps and observations are entered into the database in several ways:

- direct upload from GPS devices
- GPS locations recorded on a datasheet
- locations approximated using Google Earth

The last 2 methods require transcription, which is error prone. Even direct upload from GPS devices may produce errors if the device has not been set up correctly, for example, if the unit is set to report degrees,

minutes, and seconds instead of decimal degrees. Another source of error is accuracy of the Guam island outline in the shape files we are using. GPS waypoints recorded on the beach may be outside the island outline polygon.

This note outlines a method for identifying errors in latitude and longitude recorded in project database **trap** and **crb_obs** tables using QGIS (Quantum GIS) which is a free, user-friendly Open Source Geographic Information System (GIS) which may be downloaded from <http://qgis.org/>. The methods below use two QGIS plugins:

eVis connects to the database and downloads georeferenced data specified by SQL query statements

Spatial Query selects points which are inside or outside a polygon, etc.

2 Methods

1. QGIS 1.7.3 was opened and a new project was started.
2. An outline of Guam, **newguam.shp**, was opened as a vector layer.
3. Locations of traps and CRB observations were imported from the project's online MySQL database as vector layers in the QGIS project using the eVis plugin. eVis can read database collection parameters and SQL query statements saved in an xml file, in this case a file I wrote called **preferredSQL.xml** (Listing 1).
4. When the trap location layer is displayed on top of the island polygon, some errors are evident (Fig. 1).
5. The area of the newguam polygon was buffered by 0.0005 degrees and the buffered polygon was saved as a new vector layer, **newguam_buf_0_0005**. This slightly increases the area covered by the Guam map so that points recorded on the beach are included (Fig. 2).
6. The QGIS Spatial Query plugin is used to select points lying outside the buffered island polygon (Fig. 3). The locations and IDs for these selected points can be exported in several formats, including kml, csv, and shp, using the menu selection **Layer | Save Selection as vector file ...**. Exporting selected points to a **kml** file allows viewing them in Google Earth which facilitates making corrections.

3 Results

On March 29, 2012, the above methods found 5 of 2086 location errors in the **trap** table and 380 of 3164 records errors in the **crb_obs** table.

By exporting attributes of selected points from the **crb_obs** table to a comma separated values (csv) file, it was plain to see that most of these observation records were not georeferenced when added to the database:

- 199 points had latitude set to 0.0 and longitude set to 0.0
- 103 points had latitude set to 1.0 and longitude set to 1.0
- 78 points had points located in the ocean surrounding Guam, probably the result of transcription and data entry errors

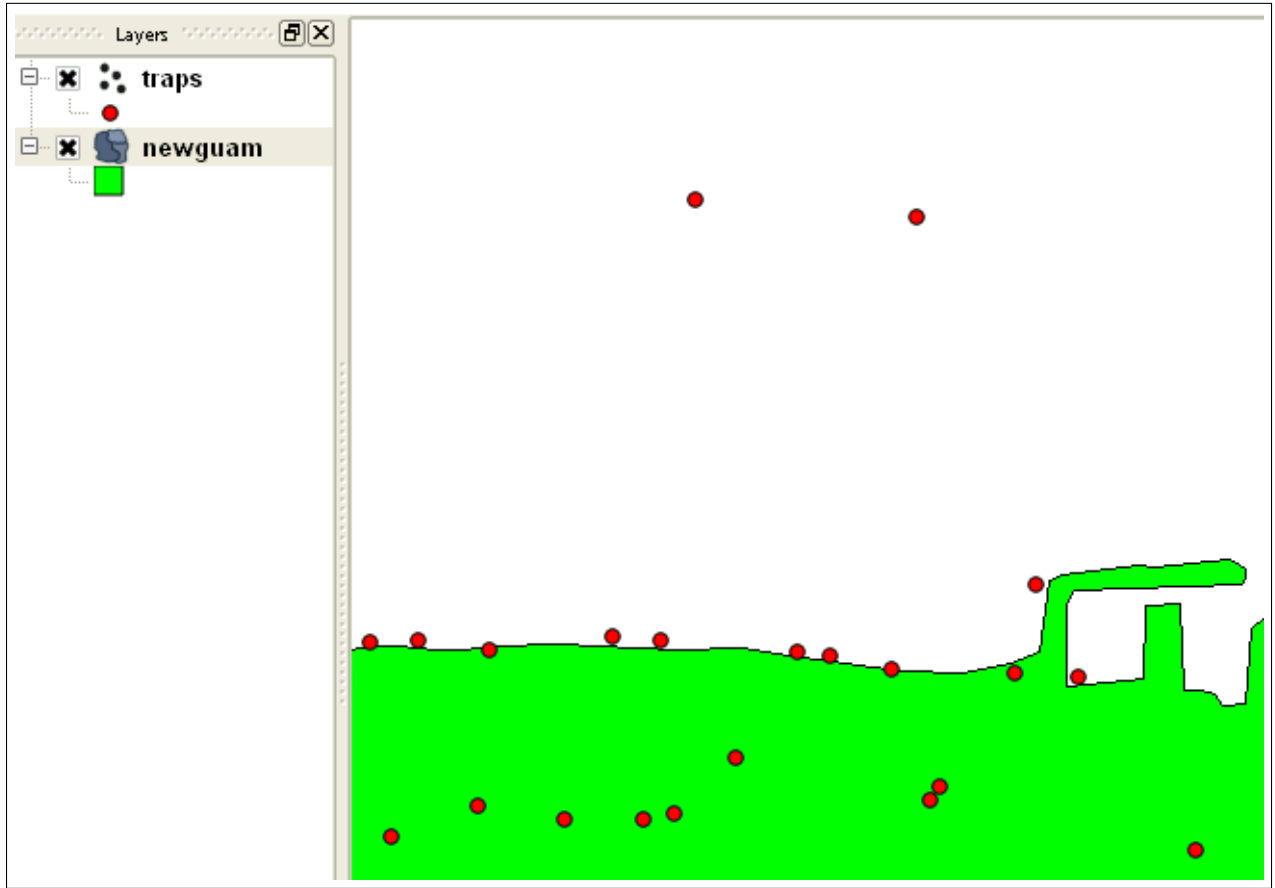


Figure 1: Trap locations in the vicinity of the Agana boat basin. Note that several trap locations are just outside the island polygon. There are 2 locations, however, which are obvious errors.

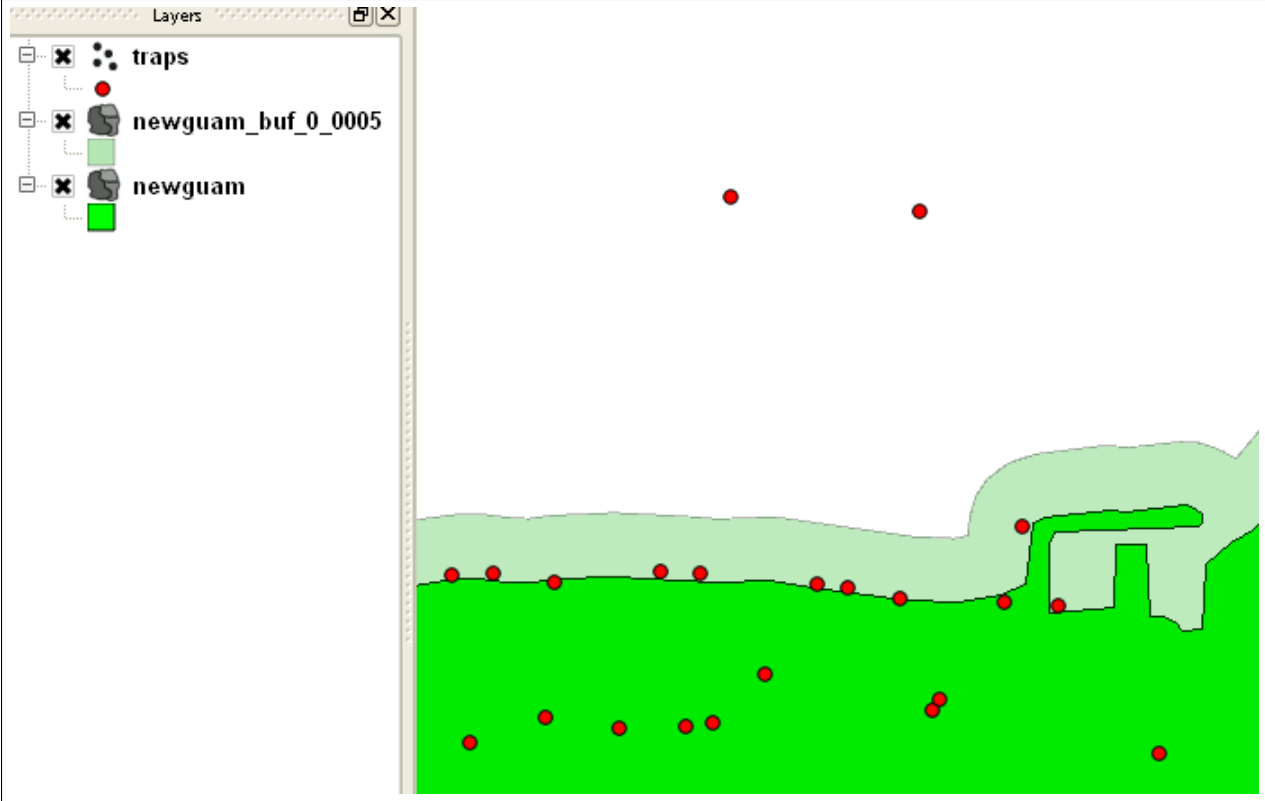


Figure 2: Trap locations in the vicinity of the Agana boat basin. The island polygon has been buffered by 0.0005 degrees to include trap locations on the beach.

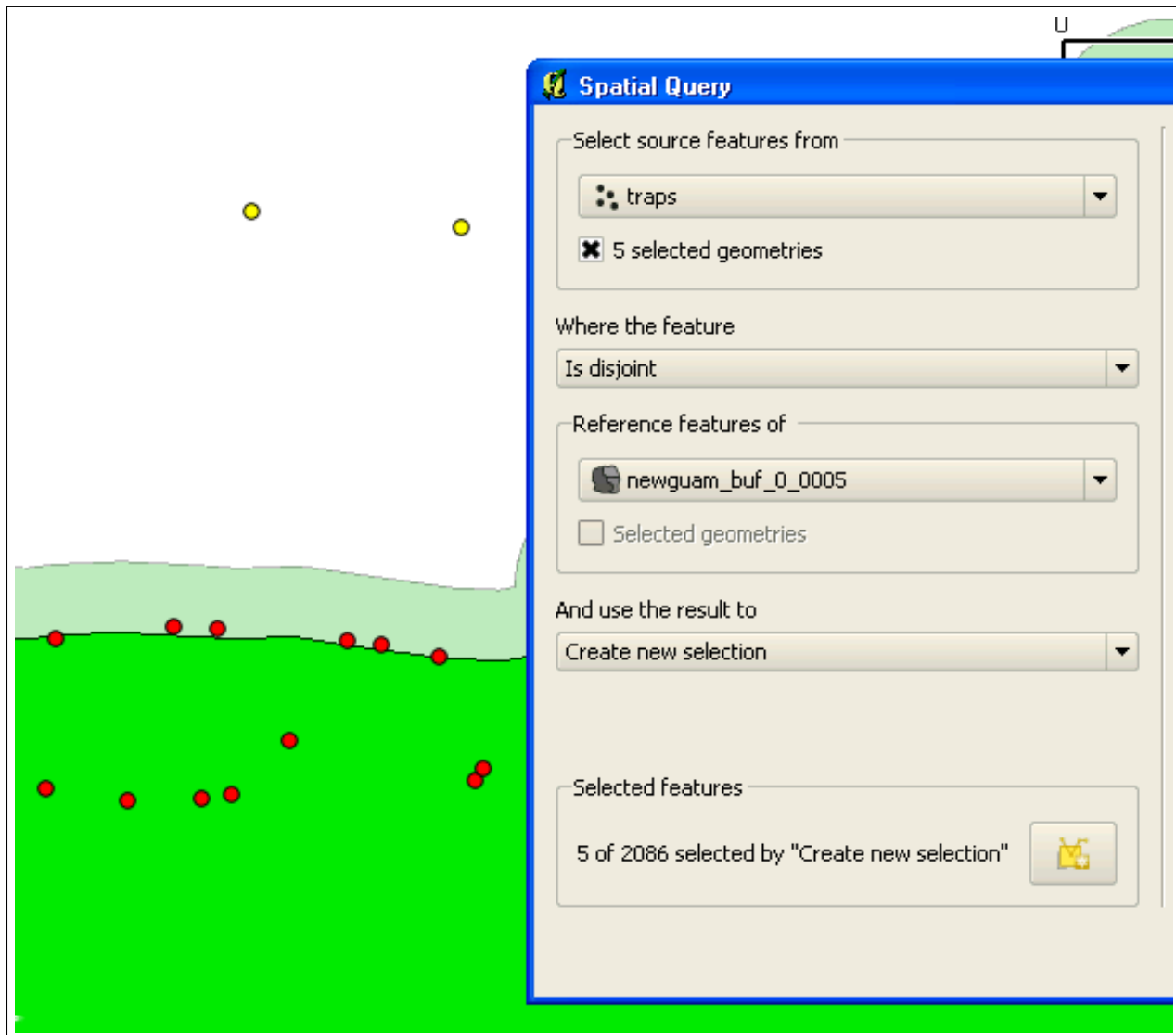


Figure 3: A spatial query selects trap locations which lie outside the buffered island polygon. Selected points are colored yellow.

4 Listings

Listing 1: Preferred SQL

```
<?xml version="1.0"?>
<doc>
  <query>
    <shortdescription>Import all trap locations</shortdescription>
    <description>This command will import all trap locations from the project
      database.
    </description>
    <databasetype>ODBC</databasetype>
    <databasehost>mysql.guaminsects.net</databasehost>
    <databaseport>3306</databaseport>
    <databasename>oryctes</databasename>
    <databaseusername>readonlyguest</databaseusername>
    <databasepassword>mangilao</databasepassword>
    <sqlstatement>SELECT trap_id, latitude, longitude FROM trap</sqlstatement>
    <autoconnect>true</autoconnect>
  </query>
  <query>
    <shortdescription>Import all obs locations</shortdescription>
    <description>This command will import all obs locations from the project
      database.
    </description>
    <databasetype>ODBC</databasetype>
    <databasehost>mysql.guaminsects.net</databasehost>
    <databaseport>3306</databaseport>
    <databasename>oryctes</databasename>
    <databaseusername>readonlyguest</databaseusername>
    <databasepassword>mangilao</databasepassword>
    <sqlstatement>SELECT crb_obs_id, decimal_latitude AS latitude,
      decimal_longitude AS longitude FROM crb_obs</sqlstatement>
    <autoconnect>true</autoconnect>
  </query>
</doc>
```