Tanzania Bird Atlas Project

Chelan Hoffman

December 2004

Abstract:

The union of GIS and the Internet have allowed for greater versatility in data dissemination. Digital maps allow for the user to personalize maps and create them for specific purposes. Interactive mapping sites allow maps to be dynamic as the user sees fit. This project was an informal partnership between the Tanzania Bird Atlas Project and the UNBC GIS department. A basic MapServer site was created using the data that was available from the Tanzania Bird Atlas Project. The process of working with the data has allowed for suggestions to be made that could lead towards a much more dynamic MapServer system. [1] Data used has been stored on a student drive temporarily until permanent server space is secured.

Introduction / background:

The Problem:
The objective of the Tanzania Bird Atlas Project (TBAP) is to better understand the breeding cycles and seasonality of Tanzania's birdlife. This is being accomplished by taking inventory and mapping observed distribution of birds and their breeding trends in Tanzania. The current state of the TBAP is a collection of over 600,000 datapoints. Static maps by species type have been made from these data and are available in PDF format online (Baker and Baker 2004). However, these PDFs do not allow for overlay of the differing species and other such dynamic analyses (i.e. query).

Objective:
To correct for the problem of only having static maps available, Dave Leman, on behalf of the TBAP, approached Scott Emmons of the University of Northern British Columbia (UNBC) GIS department to see if the data available could be modelled similar to OpenIGF. Instead, the MapServer application was used because it offers a similar, but much more simplistic, framework and the project could be done by a student of the Advanced GIS course. My objective was to take available data from the TBAP and create a basic MapServer system. This system will be presented to the TBAP in January 2005 to show the capability and versatility of such dynamic mapping techniques. As Pittman (2003) notes, MapServer is a good application for non-profit and community groups because it provides the opportunity for utlizing the abilities of a GIS for those groups who may not have the funding or technical knowledge to host a traditional GIS.

Study Area:
My study area was the country of Tanzania. Tanzania is located in Eastern Africa on the Indian Ocean borderd to the north by Kenya, the west by Rwanda, Burundi, the Democratic Republic of Congo and Zambia, and to the south by Malawi and Mozambique. Tanzania includes the islands of Mafia, Pemba and Zanzibar (CIA 2004).

Data (sources, formats and manipulation):

First, a base map of Tanzania was required. A satellite image was preferred, but the satellite images of Tanzania that I found were not complete. Instead, I reverted to the raster layer that Tyler Mitchell provided during the MapServer presentation. This raster layer only provides topographic information so I also required a detailed political boundary map that could be overlain to provide the country boundaries. This vector layer was also obtained from Tyler. Thirdly, I obtained a vector point layer of World Cities from Tyler. This layer provides the locations of cities.

This was enough base and reference data for what I was trying to achieve, but in searching the Internet I came across a website that tracked Livestock in Eastern Africa. The website is an interactive and dynamic mapping system as well, and has data layers for Tanzania including parks and protected areas, lakes, population density and river networks. This information would be useful in correlating species with their habitats, so I emailed the contact for this page, Dr. Jerry Stuth, to see if I could obtain access to the data layers. This data transfer is still in the process.

Bird data were obtained from Dave Leman on behalf of the TBAP. Data is collected and organized by the TBAP according to month to account for seasonality. This is accomplished by data being stored in a grid format. The grid is organized by lines of latitude and longitude. Each grid square is 2500km^2 (1/2 degree longitude by ½ degree latitude). The 2500km^2 is then further subdivided into a 2x2 matrix and each of the then 4 sub-grid squares is further subdivided into a 4x4 matrix to create 16 sub-sub-grid-squares to record monthly records within each square (example, fig. 1). The first 3 rows of the sub-sub-grid-square represent the months of the year and a point is used within the grid to demarcate which month the species was viewed in that location. The last row of the sub-sub-grid square is hatched out and no records are stored here.

Ashy Starling
Cosmopsarus unicolor

404 plots from
1,171 records

virgin YvB squares 43

one YvB month 38

two YvB months 44

J F M A
M J J A
S O N D

monthly records
within each
square

◇ presence
✕ breeding season
✱ month with eggs

Seasonality

	pres	bs	eggs
January	85	6	–
February	77	4	1
March	111	6	2
April	85	8	2
May	53	2	–
June	95	–	–
July	131	1	–
August	117	–	–
September	99	–	–
October	93	–	–
November	93	–	–
December	100	–	–

Tanzania Bird Atlas
Preliminary Map
October 2003
From database of
592,000 records

Mwanza, Arusha, Tabora, Dodoma, Dar es Salaam, Iringa, Mbeya, Songea
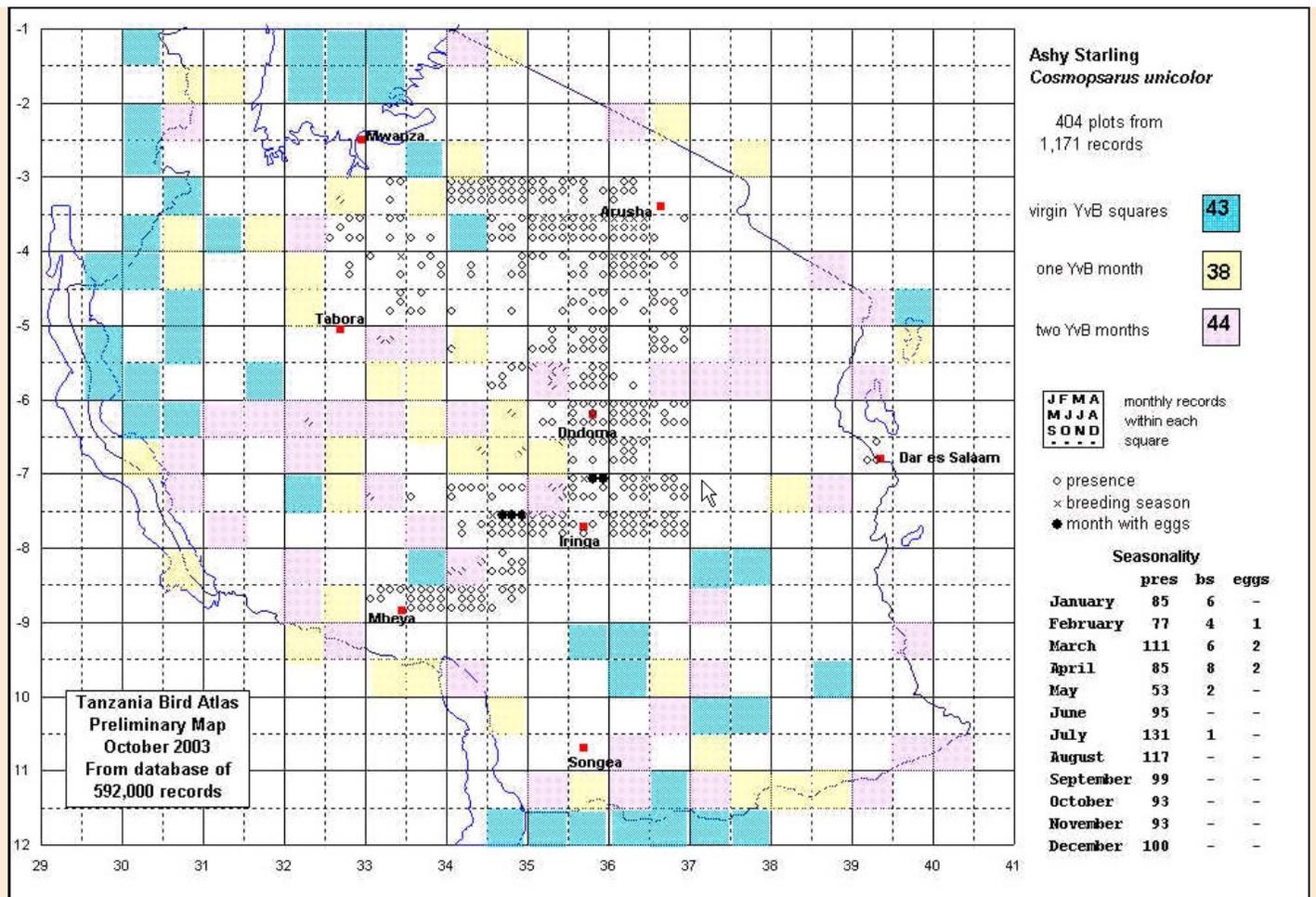
Fig 1: Sample of how TBAP organizes their data. **Note the Monthly Records within Each Square**

The results of this method of data storage and display is that many points are being overlain on each other and that the data are not accurate to the point of observation, but rather to 2500km^2. Using a GPS in data collection could greatly improve accuracy. Other errors in data collection as identified by the TBAP include original species identification errors, errors in recording data on the original field card and data entry errors. Furthermore, Baker and Baker (2003) also note that there is a bias in data collection towards where avid birders live with these areas being more closely and accurately monitored.

The bird data were received in .dbf format and I opened them in Microsoft Excel. Upon viewing the data, there were 10 different species contained in the file and these had to be broken into individual shapefiles based on species type. Each data entry gave x,y coordinates for that record. In ArcView I created an event theme from the entire file and then converted it to a shapefile. Upon viewing this layer relative to the World Countries layer, the birds were being displayed northwest of Tanzania in Central African Republic, Chad and Sudan. To correct for this I opened the bird layer in Microsoft Excel to check that the lat/long coordinates were correct for Tanzania. Once this was verified I checked the mapfile to ensure that I was using a correct projection for my layers and that the bird layer was utilizing the correct ellipsoid. Once the projection information was all verified to be correct, I had to revert to taking the bird layer in its entirety into ArcMap and use the 'Edit Feature' feature to select the birds and drag and drop them into Tanzania. Fortunately, the bird data offered a decent coverage of the entire country of Tanzania to allow for this action. Another factor that this error could be attributed to is the grid system that that TBAP used to collect and record data. This will have to be explored further.

ArcMap was used to break the collective shapefile into shapefiles specific to species-type. I selected the species by attribute [name] and then exported the data to a shapefile. These shapefiles are stored in hoffmanc on the ninkasi server.

Data Storage and Access in a MapServer Application:

The greatest part of the learning curve of this project was learning how the webserver (Apache) communicated with the MapServer, html-files and the mapfile. Mostly through trial and error and literature reviews was this process determined. The base mapfile and html-files that I used were those supplied by Tyler in the MapServer presentation. I linked to the species shapefiles from the mapfile, which is stored on the local machine. This ability to link to data is MapServer's most basic form of data access.

Role of the MapFile:
The mapfile defines the image to be displayed including its extent, projection, display size (in pixels), the output format and defines the html template that the generated image will be displayed in. The mapfile also defines the layers and their visual representation, directs the MapServer to where they are stored and allows for features to be displayed at maximum and minimum scales. The symbols and parameters used have to be defined using the MapFile and Symbol Set references supplied by the University of Minnesota MapFile Reference Site.

Role of the MapServer and WebServer:

MapServer then sends this generated image back to the webserver (Apache) which then, in turn, sends the information to the user. The ultimate power of the MapServer is the ability for the user to define the parameters of what will be displayed in the generated image. This is where the versatility and dynamics of a MapServer are realized. An element of this versatility and is the ability for a user to define a query. This function was exhaustively explored. Samples of it were found on the Internet (ex. MapServer Test Suite; Pittman 2003) and the query feature elements were isolated and emulated in my mapfile and html-file; however, this failed to produce queries from MapServer. I then copied and pasted the html source from MapServer Test Suite for the html template, the header and footer and inserted my own mapfile elements. This was the most progression made in determining the query function, but I was hindered by my limited knowledge in CGI variables and the nomenclature for defining them. As a result, the default mode of the system is browse mode which allows the user to use the map controls to turn layers on and off and navigate the image by panning and zooming in and out.

Results:

The html organizes the MapServer into a table format (fig. 2) and the MapServer offers the user the ability to display the bird species individually or in any combination that they choose only being limited by points that may share the same x,y coordinate. In this case the layer that is last defined in the mapfile will be the layer that is last displayed (the 'top' layer) in the generated image. The user also has the capability to zoom in and out at user defined parameters and to pan around the image. The legend is dynamic in the sense that features are displayed in the legend as they are selected to be displayed in the map. The scale is dynamic as it is defined by a CGI variable and re-calculated every time a new image is sent from the MapServer. The cities layer displays cities with populations greater than 500,000 at scales greater than 1:12,000,000 and cities with populations less than 500,000 at scales greater than 1:6,000,000. The birds are represented by colours that are associative to their physical appearance. Links are provided at the bottom of the page for users to access information relative to this project.
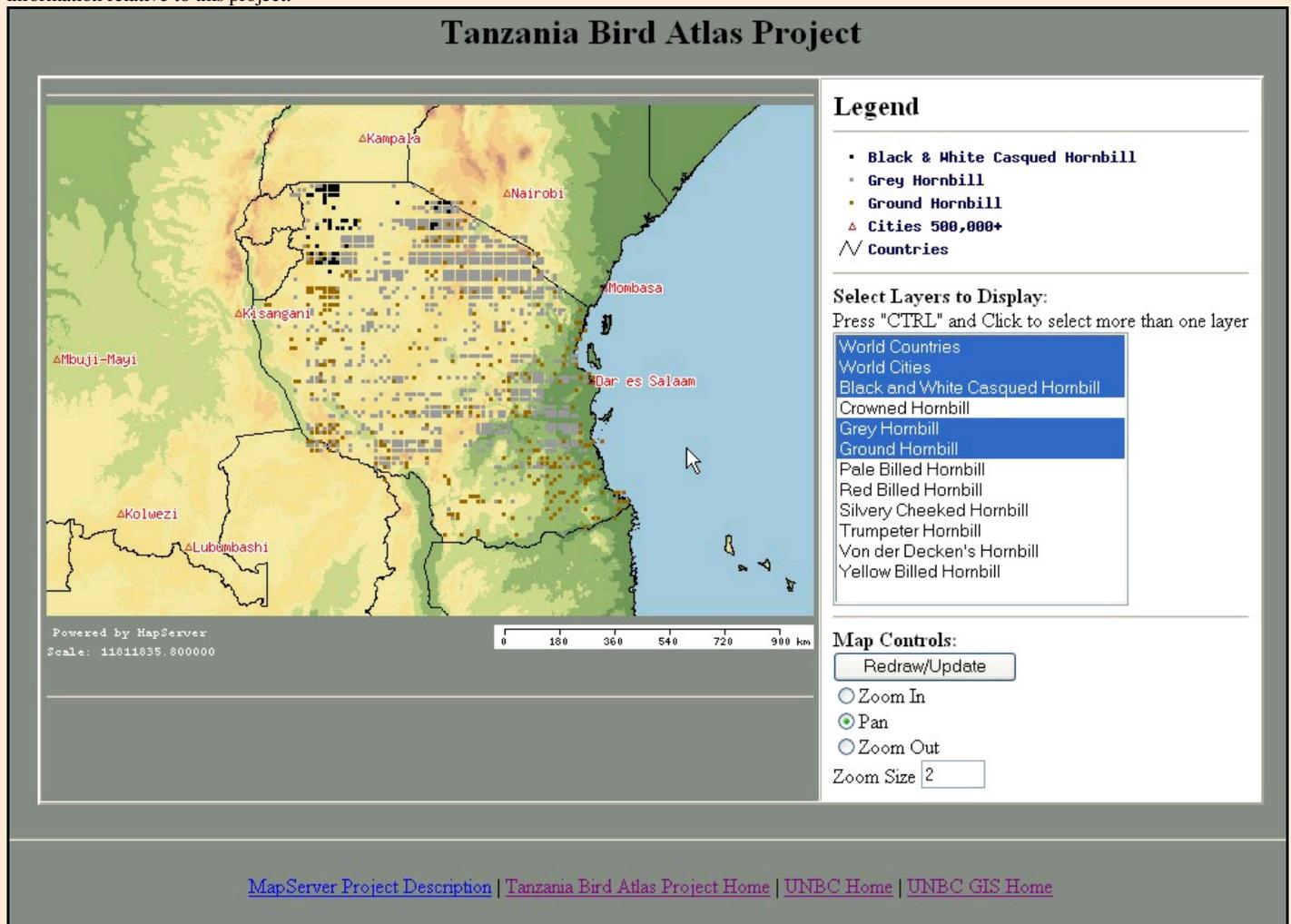

Fig 2: Appearance of the Tanzania MapServer.

Conclusion:

The MapServer system that I have established meets the very basic needs that were expressed by the TBAP. OpenIGF is the model that they were seeking to emulate; however, the resources were not available to do so. MapServer offers similar features and is a feasible option for non-profit groups in the early stages of development; however, there is still a lot of work to be done to portray the information that is being portrayed it the PDF static maps. In order to accurately display the seasonality of the breeding season the data would have to be broken into the seasons or months and then a raster layer could be

created to portray this information. Furthermore, to fully understand the workings of the MapServer a greater knowledge of CGI variables is required and the html nomenclature.

In summary, this MapServer system is a prime example of how data can be effectively displayed in a professional manner at a low cost (MapServer is a free development!). This project also exemplifies how partnerships can be formed between universities and non-profit groups and that they can be beneficial for both. The objective of this project was to build a basic MapServer system with available data for the Tanzania Bird Atlas Project and for me to develop a foundation of knowledge of the workings of a MapServer.

Future Developments:

- Currently the Tanzania Bird Atlas Project is comprised of numerous static maps available in PDF format on the

Tanzania Bird Atlas Webpage.
• The Tanzania Bird Atlas Project is hoping to obtain data from surrounding countries in attempt to understand where their birdlife is spending their non-breeding seasons. Also, as advanced mapping techniques are becoming available in OpenSource formats, the Tanzania Bird Atlas Project hopes to benefit from these and build the capacity within Tanzania to maintain databases and MapServer or OpenIGF applications. Furthermore, Internet mapping allows for greater and quicker dissemination of their data to the general public in a dynamic method where the user defines what they want to display.
• The Tanzania Bird Atlas Project hopes to improve their data collection techniques and accuracy of data by using GPS units in their data collection. Although GPS units do not offer 100% accuracy, they are a significant improvement over the system they are currently using.
• By using a dynamic mapping environment, the Tanzania Bird Atlas Project hopes to be able to associate birds more to their habitat by incorporating the other layers as discussed before (lakes, river networks and tree species).
• The query feature needs to be developed for this MapServer and to accomplish this more has to be known about CGI variables and html nomenclature.
• To better achieve the seasonality element of the Tanzania Bird Atlas Project, data could be broken up by season or by month and then created into a raster layer. This element has to be explored more.

References:

- Baker, N. and L. Baker. 2004. *Tanzania Bird Atlas: Distribution and Seasonality*. Available online at:

http://tanzaniabirdatlas.com Accesse on 29 November 2004.
• Central Intelligence Agency. 2004. *The World Factbook: Tanzania*. Available online at: http://www.cia.gov/cia/publications/factbook/geos/tz.html. Accessed on 29 November 2004.
• Nilsen, S. 2003. *Birds of Tanzania: Distribution and Seasonality*. Available online at: http://home.no.net/stenil1/TZbirdatlas/tzatlas.htm. Accessed on 29 November 2004.
• Pittman, S. 2003. *GIS and the Internet: Tools for Neighborhood Access to Information*. Available online at: http://www.npcr.org/reports/npcr1205/npcr1205.html. Accessed on 28 November 2004.
• Texas A&M University. 2004. *Livestock Early Warning System*. Available online at: http://cnrit.tamu.edu/maps/map_init.html. Accessed on 22 November 2004.
• University of Minnesota. 2004a. *Mapserver: MapFile Reference*. Available online at: http://mapserver.gis.umn.edu/doc42/mapfile-reference.html. Accessed on 24 November 2004.
• University of Minnesota. 2004b. *Mapserver Software*. Available online at: http://mapserver.gis.umn.edu/dload.html. Accessed on 20 November 2004.
• University of Minnesota. 2004c. *Mapserver Test Suite*. Available online at: http://maps.dnr.state.mn.us/mapserver_demos/tests/. Accessed on 25 November 2004.

---

[1] The term "MapServer system" is used to identify the network (or any portion thereof) of the user, the webserver, the mapserver, the mapfile, and the html-file(s) involved.