Citation: Proctor, Patrick; Heath, Linda S.; Van Deusen, Paul C.; Gove, Jeffrey H.; Smith, James E. 2005. COLE: A web-based tool for interfacing with forest inventory data. In: McRoberts, Ronald E. and others, eds. Proceedings of the fourth annual forest inventory and analysis symposium; 2002 November 19-21; New Orleans, LA. Gen Tech Rep NC-252. St Paul, MN; U.S. Department of Agriculture, Forest Service, North Central Research Station: 167-172.

COLE: A Web-based Tool for Interfacing with Forest Inventory Data

Patrick Proctor[1], Linda S. Heath[2], Paul C. Van Deusen[1], Jeffrey H. Gove[2], and James E. Smith[2]

Abstract.—We are developing an online computer program to provide forest carbon related estimates for the conterminous United States (COLE). Version 1.0 of the program features carbon estimates based on data from the USDA Forest Service Eastwide Forest Inventory database. The program allows the user to designate an area of interest, and currently provides area, growing-stock volume, and carbon pool estimates for states east of the Great Plains. The COLE program can be accessed at http://ncasi.uml.edu.

The Forest Inventory and Analysis (FIADB) program of the USDA Forest Service provides the most scientifically credible and comprehensive data on the amount and condition of forest resources in the United States. The Forest Inventory and Analysis Database has the potential for a wide array of applications (Miles *et al.* 2001). High-quality tools for online access to the FIADB will assure that this potential is realized. The goal of the Carbon Online Estimation (COLE) project is to develop an online tool to provide access to the FIADB using a versatile user interface (UI) while maintaining a fast response time. With minor revision, the code for COLE will also be useful for providing access to spatial databases and for analyzing the data.

In the first year of development, we have primarily focused on creating an interface and backend that will embody the versatility and speed described above. Additionally, we have created sample queries that calculate carbon levels based on the Eastwide Forest Inventory Database (EWDB) data. The interface has been designed to make it easy to incorporate the FIADB when it becomes available. Due to security concerns, version 1.0 of COLE is linked to a version of the EWDB, which is a forerunner of the FIADB (Hansen *et al.* 1992). We have augmented the EWDB with estimates of carbon for forested plots. It is anticipated that COLE version 2.0 will be linked directly to the FIADB after security issues can be overcome.

Technologies

We want the tool to provide online access with a fast response time while featuring an easy-to-use user interface. The merits of a fast computation time are obvious and easy to quantify, but the idea of a "versatile" UI is less so. UI's are probably the most difficult component of software development. They require the users to familiarize themselves with how each new program works, along with its particular idiosyncrasies. Users can become productive more easily if they are not required to learn a new interface for each piece of software they use. In designing a tool to access the FIADB, it would be fruitful to create a powerful interface that could be easily recycled for various applications. This increases user familiarity and the usefulness of the FIADB.

In approaching the COLE project, the first step was to decide which technologies we would enlist. Our primary goals were to minimize cost and assure operating-system portability. To minimize the cost, the clear choice is open-source software. In today's market, open-source software is a stable and valid option with no licensing cost. We also wanted operating-system portable software that would work on the Windows, Linux, and Macintosh platforms. The solution to this problem was to utilize a cross-platform programming language that could be compiled and/or ported to all three systems.

Keeping these principles in mind, we chose a Linux platform to run the COLE engine. The Linux operating system is a free, stable, open-source platform that is widely recognized as one of the preeminent alternatives to Microsoft Windows and Macintosh. We also needed a database to run from our own

[1]Programmer and Principal Research Scientist, respectively, National Council for Air and Stream Improvement, 600 Suffolk Street Fifth Floor, Lowell, MA 01854. E-mail: paul_vandeusen@uml.edu.

[2]Project Leader, Research Forester, and Research Plant Physiologist, respectively, U.S. Department of Agriculture, Forest Service, Northeastern Research Station, P.O. Box 640, Durham, NH 03824. E-mail: Lheath@fs.fed.us.

local server. We chose MySQL because it is an open-source database query engine compatible with SQL. Since MySQL strictly adheres to the SQL standard, the database we use, and any modifications made to it, can be easily ported to other SQL compatible applications such as Oracle or Sybase.

After establishing our operating and database platforms, we had to decide on a programming language to develop COLE. We wanted a cross-platform language that could be used on all three major operating systems. We also wanted the language to be fast and Web accessible. The two available options were an HTML/CGI/Perl system or a Java platform. Perl offers less than Java in graphical client-side conveniences that would lead to a familiar, easy-to-use interface. Therefore, we selected Java with the accompanying applet/servlet technology. Java is easily compiled on all three major platforms, offers a wide array of UI design tools, and is built to handle Web access. Having decided on a Java platform, we enlisted the open-source Jakarta Tomcat Servlet engine (also available on all platforms) to handle the client-server communications.

User Interaction

We had to select the most logical way for the user to interact with the EWDB. Some existing technologies use drop-down menus and other text-based input to define EWDB queries. However, the EWDB is largely a geographical database, and it would be both intuitive and logical for the users to begin their query by selecting an area on a map.

While the idea of a map-based selection tool was a good one, it was also necessary to decide how such a tool would work. To have a great deal of user power and flexibility, it was necessary to use a map file-format that would allow the users to define their own polygons or use predefined polygons. Furthermore, a display of latitude and longitude would be needed such that the user could easily define accurate and relevant shapes.

Users also want to modify their query with easy-to-use text-based filters and sort variables. Inputs for these factors could be easily designed using the EWDB's predetermined fields. Combining map-based area selection with text-based query modifications provides the user with a powerful query-building tool that is also easy to use.

Development

The first problem in developing COLE was how to make the map interface accessible and dynamic. The obvious answer was to utilize the ESRI Shapefile® format, since the format is publicly available. This allowed us to read shapefiles directly with Java by using the specifications found in the ESRI document. Additionally, the format is widely used and there is a large library of available shape files that could be utilized for COLE.

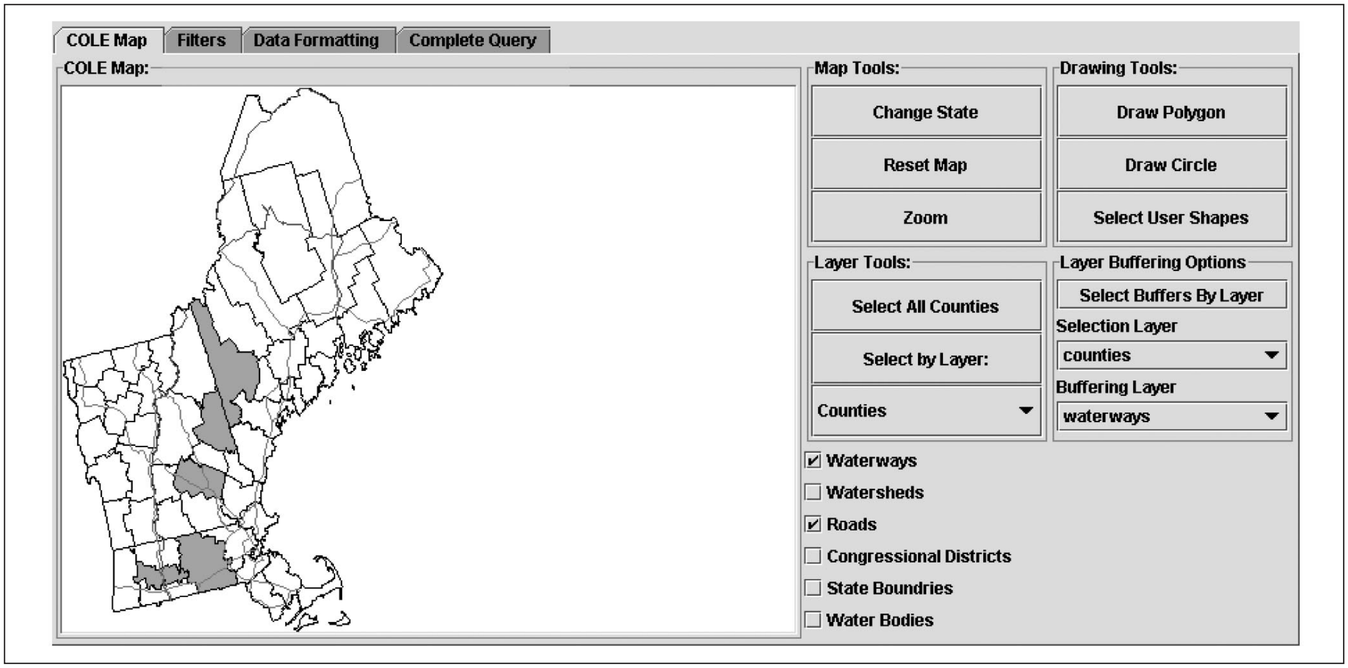
Once we had decided to use the ESRI shape format, we had to make the format interact with Java. While there are available commercial Java/ESRI tools, we decided instead to design our own parser and use it in conjunction with the Java2D Graphics. This would allow us to display the shapes on a Java Canvas object in the interface and enable the user to interact with the map using the properties inherent to Java2D Shape objects. Using these methods, we created a working Canvas-based Java class to facilitate our map display.

With the capacity to display our maps as desired using a standard format, we moved on to defining filters and sorting variables. While the actual sorting and filtering would take place server-side, it was necessary to make a user-friendly representation of those sort and filter variables such that the users could successfully modify their query. To make COLE as flexible as possible, we decided to have the filters and sort variables defined in regular text files. This would allow a COLE administrator to modify the text files and, in turn, modify COLE's capabilities easily. The text files contain both an SQL representation of variables and an English representation. This allows for a link between the user interface and SQL database when submitting a query.

The client side of the program allows the user to pick an area of interest on a map and then submit a query. It is then necessary to develop the server-side mechanism to handle the computations. For our purposes, we created a Java servlet that submits queries to the database, analyzes them for spatial relevance, applies computational algorithms, and returns a data table to the user. By modifying this servlet, one could potentially add more computational functions and increase the number of data formats for the return table.

In completing this server-side functionality, the two primary tasks of the COLE engine are defined. What remains is to build a

Figure 1.— *The tabbed interface of COLE.*



user interface that could leverage these functional technologies and present them to the user in a simple graphical interface.

The COLE Interface

The COLE interface has now reached a reasonably stable and usable layout. The interface enlists the use of Java "tabbed" panes (fig. 1). This allows the user to easily separate submitting a query to COLE into a series of distinct tasks, which combine to generate a dynamic query. COLE currently utilizes a four-tabbed system that addresses the following query-building steps: map-based area selection, data filters, output formatting, and complete query.

The first COLE tab (fig. 1) allows users to select the area relevant to their query. There are several different ways to do this in the interface. First, the user can create a polygon based on coordinate points selected with the user's mouse. Second, the user can create a circle based on a center and a radius. Third, the user can select areas based on predefined polygons imported from a shape file. This selection currently includes counties, watersheds, U.S. Congressional districts, and State boundaries. Finally, the user can select line-based shape files and run a buffering query. The buffering distance is defined on the final tab. Automatically importing a customized shape file from the user is not supported currently; however, users defining their own polygon serves a similar purpose. Following the area selection, the user can then move to the second tab.

The second COLE tab contains the filters defined in a server-side text file. Currently, the interface accommodates up to nine filters. To select a filter, the user must check the checkbox. Once this has been done, the user can select one or many filters from the given filter category. The data will then be filtered in accordance with the user selection.

The third COLE tab allows the user to modify the formatting and data retrieval parts of the query. This includes choosing sort variables (which define table rows and columns), units, query variable, buffering distance, and analysis function to be applied to these data, such as sum and mean. Sort variables, as mentioned above, are based on a server-side text file. Next are the formatting and unit fields. Units currently returned include English and metric. Table formatting and units are both done server-side and require manipulating the servlet to enact further standards. Data computations also are defined server-side. Currently, one can summarize data using sum, average, median, standard error, standard deviation, or sample size (plots or trees). Finally, if users choose to use the buffering feature, they can select the buffering distance on this tab.

On the fourth and final COLE tab, the user can select the format to receive the retrieved data. Currently the options include four types of tables (Jtable, HTML, Spreadsheet, and Tab Delimited) and the graph option. If the users select the graph option, they must then select which type of graphs to retrieve. By checking any combination of the three graph checkboxes, the users can obtain scatter, bar, or pie charts of their data. Once all of these variables have been selected, the user is ready to submit a query. This can be done by selecting the “submit query” button at the base of the fourth tab.

Using the 4-tab interface COLE offers an easy-to-use, progressive interface for accessing a spatial database. The interface has the advantage of being customizable without actually changing compiled code. Furthermore, the types of queries and databases are highly generalized, allowing for later expansion of COLE capabilities.

Carbon Estimates

The carbon estimates in COLE are pre-computed values based on inventory data from the EWDB. Carbon pools are estimated from plot data, such as growing-stock volume and forest type. Aboveground and belowground tree carbon is estimated using the equations presented in Smith *et al.* (2003). The equations for estimating forest floor carbon based on plot data are taken from Smith and Heath (2002). The down dead wood carbon pool is based on an approach similar to that described in Chojnacky and Heath (2002). Forest soil carbon is determined only by forest type and region and is based on estimates in Johnson and Kern (2003). That is, we are using an estimate of soil carbon based on broad regions and forest types of average conditions; previous land use and management were assumed to not affect soil carbon. This approach to estimating carbon is also discussed further in Heath *et al.* (2003).

Data Visualization Tools

The tabular and spreadsheet data retrieval options described above are useful, but COLE provides other options for viewing the data as well. A graphing option allows the user to display results as graphs. A separate, but similar, interface is available to display estimates as maps.

Graphs

By offering graphing options to the user, COLE allows for visual representation of the data. The scatter plot gives details about individual records without revealing what might be considered sensitive information, such as plot location or estimates on land of a specific owner (see fig. 2). The bar and pie options offer a different perspective of the data by giving an easy-to-read relationship between the totals and their sort categories.

When selecting the graphing option, the COLE engine creates graphs based on the user-selected sort variables. The graphs are dynamically built and labeled based on those sort variables, and then returned to the user. This dynamic graph-building ability means that COLE can offer the users a potentially endless selection of sorted graphs based on their area-based query.

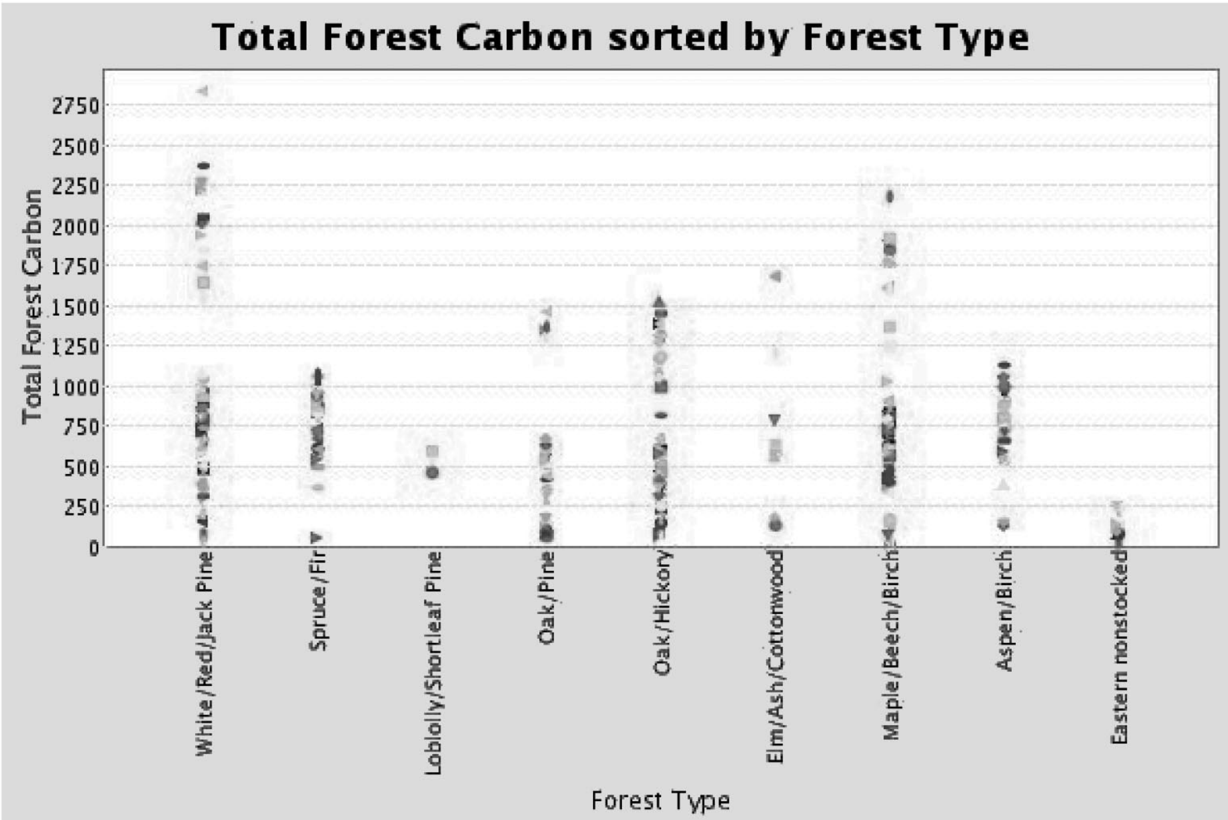
In addition to providing a different perspective on the retrieved data, COLE’s graphing option also increases its functionality and usefulness as a data-reporting tool. The graphs are returned in the standard JPEG file format and can be easily saved or cut and pasted in any PC. The resulting files can then be used in presentations, papers, or other media-based interpretations of COLE related data.

COLE-Map

COLE presents a map that defines an area for which data are retrieved, but users also want a tool that would create a map using the retrieved data. This could easily be done with off-the-shelf GIS software but requires more access to the data than most users would be allowed. Adding an online mapmaking capability was the primary goal behind the creation of the COLE-Map as an applet to accompany COLE.

Because COLE was designed as a dynamic toolset, COLE-Map is able to leverage a great many of the technologies used in COLE. COLE-Map utilizes the same tabbed interface as COLE as well as many of the same query-building and submittal classes. The primary differences in COLE-Map occur server-side, and in the absence of any actual map in the user interface.

Figure 2.— *Example scattergraph output from COLE.*



The first difference is largely self-evident. The servlet for COLE-Map performs a different task than that in COLE, and therefore had to be modified. COLE retrieves the data, sorts them, and builds a corresponding table. By contrast COLE-Map retrieves data, sorts them, and then creates and colors a map based on an ESRI shape file. The second difference is also clear: if COLE-Map is making a map, there is no reason for the user to view a map in the interface. Therefore, the map component is removed and replaced with a customizable color palette for the output map. Other than these two minor differences, COLE-Map implements the COLE operations precisely.

Conclusions

In the first year of development, the joint NCASI/USDA Forest Service team has created a powerful and dynamic tool for accessing a version of the EWDB. We have met our initial goal of developing a highly extensible user interface with which to access the EWDB. We have leveraged this user interface to create two separate applications, both of which rely on the extensive data of the EWDB to answer user queries.

This proof-of-concept has opened the door for creating any number of applications of the COLE interface. Other graphical and statistical analysis could be added along with a greater number of GIS-related tools. COLE can readily be linked to additional spatial databases, including the FIADB, with estimation performed on-the-fly. Additional variables, such as the down woody material data from Phase 3 plots, can also be included in linked databases. The usefulness of this tool will only increase as more features are included.

Acknowledgment

This work was made possible by funding from the USDA Forest Service, Forest Health Monitoring program.

Literature Cited

Anonymous. 1998. ESRI Shapefile technical description: an ESRI white paper. Copyright 1997. 34 p.

Chojnacky, David C.; Heath, Linda S. 2002. Estimating down dead wood from FIA forest inventory variables in Maine. Environmental Pollution. 116(1001): S25–S30.

Hansen, Mark H.; Frieswyk, Thomas; Glover, Joseph F.; Kelly, John 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.

Heath, Linda S.; Smith, James E.; Birdsey, Richard A. 2003. Carbon trends in U.S. forestlands: a context for the role of soils in forest carbon sequestration. In: The potential of U.S. forest soils to sequester carbon and mitigate the greenhouse effect. Boca Raton, FL: CRC Press: 35–45.

Johnson, Mark G.; Kern, Jeffrey S. 2003. Quantifying the organic carbon held in forested soils of the United States and Puerto Rico. In: The potential of U.S. forest soils to sequester carbon and mitigate the greenhouse effect. Boca Raton, FL: CRC Press: 47–72.

Miles, Patrick D.; Brand, Gary J.; Alerich, Carol L.; *et al.* 2001. The forest inventory and analysis database: database description and users manual version 1.0. Gen. Tech. Rep. NC-218. St. Paul, MN: U.S. Department of Agriculture, Forest Service, North Central Research Station. 130 p.

Smith, James E.; Heath, Linda S. 2002. A model of forest floor carbon mass for United States forest types. Res. Pap. NE-722. Newtown Square, PA: U.S. Department of Agriculture, Forest Service, Northeastern Research Station. 37 p.

Smith, James E.; Heath, Linda S.; Jenkins, J. C. 2003. Forest volume-to-biomass models and estimates of mass for live and standing dead trees of U.S. forests. Gen. Tech. Rep. NE-298. Newtown Square, PA: U.S. Department of Agriculture, Forest Service, Northeastern Research Station. 57 p.

United States
Department of
Agriculture

Forest Service

GTR–NC–252
February 2005

Proceedings of the Fourth Annual Forest Inventory and Analysis Symposium

Meeting jointly with the Southern Forest Mensurationists

Edited by
Ronald E. McRoberts
Gregory A. Reams
Paul C. Van Deusen
William H. McWilliams
Chris J. Cieszewski

New Orleans, Louisiana
November 19–21, 2002

Sponsored by
Forest Inventory and Analysis Program, USDA Forest Service
National Council for Air and Stream Improvement
Southern Forest Mensurationists
University of Georgia

McRoberts, Ronald E.; Reams, Gregory A.; Van Deusen, Paul C.; McWilliams, William H.; Cieszewski, Chris J., eds. 2005. Proceedings of the fourth annual forest inventory and analysis symposium; 2002 November 19–21; New Orleans, LA. Gen. Tech. Rep. NC-252. St. Paul, MN: U.S. Department of Agriculture, Forest Service, North Central Research Station. 258 p.

Documents contributions to forest inventory in the areas of sampling, remote sensing, modeling, information management and analysis for the Forest Inventory and Analysis program of the USDA Forest Service.

KEY WORDS: annual forest inventory, estimation, sampling, modeling, remote sensing, assessments