POINT-&-CLICK VS. COMMANDS: HOW STUDENTS LEARN GIS IN DIFFERENT INTERFACES

C. Victor Wu

Abstract: This paper examines and discusses how students learn GIS in two user-interfaces, namely the menu-driven graphic environment as in ArcToolbox, and the command-line environment as in ArcGIS Workstation. Central to the question is that if menu-driven environment really facilitates learning effectiveness, or it merely creates a perception of user-friendliness. Given that the menu-driven interface seems more appealing to students, how can educators avoid some of the pitfalls associated with the learning in such an environment?

It is unquestionable that the point-&-click style dominates the user interface of today's geographic information systems (GIS) software. One has to look hard to find a few packages that feature a command-line interface. It was only ten or fifteen years ago when text entry was the primary mode for interacting with a GIS. Those days are long gone! To date, a majority of college level introductory GIS courses start with a point-&-click style software package, such as ArcView or ArcGIS Desktop. Fewer and fewer courses cover packages that can only be interfaced through text commands (as in ArcGIS Workstation). The switch to a point-&-click style package reflects the changes in software landscape and the potential improved environment of GIS learning.

The appeals of this point-&-click paradigm are understandable. A well-designed menu system helps a user navigate through a library of commands and identify the proper ones to use. It can also free a user from having to memorizing the syntactical and parametric details of a command. Aesthetically a menu-driven user interface makes the look of the software environment more interesting, inviting, and attracting. Thus, it helps broaden the user base. Without menu-driven, graphic user interfaces, it would have been difficult to involve casual users such as library patrons, and K-12 level students to use GIS.

While welcoming those potential advantages, we educators should be vigilant about these advancements in the software environment. Such vigilance naturally begs the question if these

advancements help students learn more effectively. If the answer is no, how can we make up the shortfalls in this paradigm, given that the point-&-click interface is here to stay? On the other hand, the body of students who are taking today's GIS classes is much bigger in size and diverse in academic preparedness, compared to the GIS class fifteen years ago. Moreover, they are the generation that grew up with the point-&-click style, graphic user interface. Does this comfort level with the point-&-click interface plays a positive or a negative role in their learning of GIS concepts.

This paper will answer the above questions by examine and discuss how students learn in ArcGIS's ArcToolbox (point-&-click) and ArcGIS's Workstation (text-based) environments. Foci of this paper are the strategies employed by students and learning effectiveness in the two environments. The author makes no attempts to argue if one interface is superior to the other, or educators should favor one interface to the other. The appeals of the point-&-click design are obvious and powerful. The interface is here to stay. The text-based interface will gradually fade into the background but never disappear entirely. However, the discussions may help educators to rethink their educational objectives and strategies, for as this paper will point out that there are learning problems with this user-friendly design. We would like to make sure that students stay away from those problems when learning GIS.

For the convenience of discussion, the point-&-click style ArcGIS Desktop environment (including ArcToolbox) will be referred to as Desktop hereafter. The term ARCINFO will be used to mean the text-based ArcGIS Workstation.

The Subjects and the Setting

The subjects are 12 students in the author's Advanced GIS class at a large size state university. There are 3 females and 9 male students in the class; all but one are undergraduate students.

Prior to this course, they all had successfully completed an introductory course, in which their software experience included ArcView 3.x and ArcGIS Desktop. A survey in the beginning of the semester indicates that none of them had any experience with ARCINFO. Two of the students took courses that covered the Basic or Pascal programming before. Neither claim to be an avid programmer. Furthermore, five students check up identify themselves to have experience with text-based operating systems, mostly MS-DOS. One of the five students mentioned a brief exposure to UNIX operating system on his job.

Subjects are informed at the beginning of the semester that they will be learning ARCINFO. Considering subjects' limited exposure to text-based interfaces, the first week is devoted to basics of the ArcInfo environment ranging from syntax, parameters, variables, to online help. Student also receive a tutorial that walks them through the basic operation in ARCINFO. In the following two months students had three laboratory modules, each of which consists of two exercises based on the ARCINFO and the Desktop platform respectively. The two exercises involve exactly the same GIS tasks, such as data management, overlay, and map projection. Upon completing each exercise students also have to fill out a survey designed to find out how they feel and their learning difficulties in the two exercises. During laboratory exercises, the author makes numerous random observations by circulating in the classroom.

In the second stage, six students are recruited based on their class performance for in-depth observation. They are asked to perform one at a time under the author's eye on two laboratory exercises without detailed procedural instructions. As in the previous stage, the two exercise involve exact the same GIS tasks. But, one of the exercise requires subjects to work in the Desktop environment and the other in the ARCINFO environments. Each exercises runs for 50 minutes. A subject takes a ten minute break between two exercises. Subjects are asked to

speak out their decision or intend at very step during the exercises. At times, a subject forgets to announce the thoughts behind an act, the instructor would deliberately ask the subject to state his/her mind. It is only natural that subjects seek the author's assistance when they encounter difficulties. In no case does the author give any response or hint related to the GIS operations.

The Ugly Text Scenes

In general, subjects fare better in the Desktop environment. Four subjects are able complete the Desktop exercise in the 50 minute time frame (averaging 42 minutes), while only two subjects finish the ARCINFO exercise (averaging 37 minutes). Please be reminded that subjects have already performed the same tasks in the ARCINFO environment prior to the exercise based on Desktop. The completion time may not truly reflect how efficiently a subject perform in an exercise due also to the protocol announcement and sometimes the interference of the observer. For those reasons, the time factor is removed from further discussion.

The higher performance in the Desktop environment may be attributed to the subjects' familiarity, preference, and confidence with one particular interface. It is difficult to assert that students learn more effectively in the Desktop environment as one takes a closer look at how students struggle in the exercises. More than often, subjects complete a task by trial and error. Such a strategy has a higher success rate in the menu-driven, point-&-click Desktop environment. The text-based ARCINFO on the other hand is less forgiving. Exact words and syntax of a command are required! To successfully complete a task students have to possess knowledge that is formal (in spelling and syntax) and precise. On the other hand, subjects are less likely to name the exact commands in the Desktop environment. The following quote is a typical response in the Desktop environment: "Mmmmm….I don't remember what it is called.

But I think it is here…" as a subject announces when he navigate into the imbedded menu in ArcToolbox.

Thus, it is no surprise that students in general prefer the point-&-click interface. Poor typing skills, spelling problems, and a lack of understanding of a command's parameters are the most common blames for the text-based interface. They had great difficulties in memorizing a few commands. For example, the 'listcov' command, that stands for list coverages, does not make much sense to students. For those who had the MS-DOS experience often try to list coverages by applying the 'dir' command. Even those who had no MS-DOS experience had problem remembering the 'list coverage' act. They enjoy, rather, the ease of listing the content of a directory in ArcCatalog. The other frequently cited example is the 'project' command, which is in fact for re-projecting a coverage. To them, the word 'project' when used as a verb has a different meaning from setting map projection properties.

Complaints are also made on ARCINFO's 'workspace' mechanism, which sets up a working environment for ARC coverages. Confusions run deep in the distinction between the 'cd' (in MS-DOS) and the workspace (in ArcGIS Workstation) commands. Those who have no MS-DOS experience feel particularly disadvantaged. They have hard time navigating to different directories in the DOS environment, let alone in the Workstation environment.

Many of the difficulties with the text-based interface arise from the unfamiliarity with the text-based interface and failure to read system messages. Students pay little attention to the text messages resulting from executing a command. During the in-depth observation, low performing students tend to disregard the system message entirely. They cannot tell whether a command is successfully executed or not; they, instead, are overly dependent upon human feedbacks for validation. Additionally, students at all levels suffer from inabilities to understand

the system message. 'I never bother to read those messages because I never understand them.' puts one student. It is only after the exercise is over, that through the author's explanations a few students are able to relate the system message to the concepts covered in lectures. They also blame some messages being too long or scrolling up too fast on the screen. High performing students are more inclined to scroll back to examine a long message. Such an examination often leads to asking many more questions. Low performing individuals upon receiving the message cares only about if the execution works.

Pitfalls in a Perfect Picture

Even if subjects succeed in the Desktop environment by trial and error, it is still a less intimidating environment for GIS learners. The Desktop is a much better design thanks not only to the point-&-click and interactive manu features, but also to many small improvements that offter timely and context-sensitive information to a user. For example, a well-designed 'wizard' will walk a user through an array of parameters and choices associated to a command.

However, the extent to which a wizard assists a user varies. One of the well designed wizard can be found in the GeoProcessing Wizard that is designed to merge, dissolve or split a database. The wizard features easy-to-understand illustrations of those operations as part of the dialog. Subjects usually make the correct choice at the first attempt. Despite that those illustrations are also available in the ARCINFO's online help, students rarely use it even when in doubt. The Project or Define Projection Wizards do not fare that well. Subjects were confused by the distinction between the two wizards in the first place. Their confusion deepens when they see in each operation there are a wizard for ARCINFO coverages and the other for shapefiles and geodatabases. They also quickly lose their confidence as they encounter a seemingly huge number of projection parameters. Facing the parameters they do not fully understand, many

subjects hesitate for a moment and proceed by not really finding out what those parameters are. They blindly hit the Next button to the end of the wizard. In many cases subjects succeed in the operation, due not to their mastery of the operation but to the smart wizard design which figures out projection parameters correctly based on the information in a coverage. Their successful execution of the operation surprises even themselves. This concurs with the observation mentioned earlier that it is easier to hit the jack pot in the Desktop environment.

Discussions

The smart design of user interface in today's software is both a blessing and a curse to learning GIS. On one hand, it helps organizes a library of commands into a logic menu system, and alleviates users' cognitive burden by allowing pointing and clicking, instead of typing the exact words, and makes complex operations more accessible. On the other, it is easy to 'cheat' in this interface without a firm understanding of the conceptual basis. Despite the user-friendly environment, the fuzzy nature of point-&-click interface and other smart design make it difficult to validate learning outcomes. Observations in this research paint a conflicting picture between students' satisfaction with the Desktop environment and their failure to understand many key concepts to the GIS operation. Students may not grasp those concepts better in the ARCINFO environment, which requires the form of knowledge that has to be more formal and precise. Thus, it is easier to identify failures to learn as they are working in a text-based environment.

The author does advocate for abandoning a software package simply because it features a smart user interface. Rather, the author argues that educators should be aware of the shortcomings of learning GIS in a well-designed software environment, and find ways to stay away from these shortcomings. Nonetheless, before we go on, we should examine what do we expect students to accomplish through learning these laboratory exercises.

There are two general goals for GIS laboratory education: enforcing concepts (Unwin & Dale, 1990; Kemp & Goodchild, 1992; Raper & Green, 1992; Dibiase, 1996) and technical training for system operation and applications (Kemp & Goodchild, 1991; Raper & Green, 1992). While a majority of undergraduate do not come to a GIS class with the first goal in mind, it is vital for students to engage to the conceptual basis of the technology. Furthermore, educators (Denham, 2001, Marshall, 1993) now recognize that college education is limited in time and scope. College graduates to date have to develop skills for problem solving and learning, in order to cope with the rapid changing technological landscape and fast expansion of knowledge. Which skills do we help our students to develop in such a software environment where one can complete a complicated GIS task by chance?

The bottom line is to hold students accountable for what they are supposed to learn. It is crucial to make students understand that GIS work is not just keyboard or mouse exercises. GIS is actually a small piece in the puzzle of a problem solving. Comprehension of GIS's conceptual basis is only a natural prerequisite to problem solving. But, how do we hold validate if students learn both the technique and the concepts in a point-&-click environment? A few common strategies can be useful:

- Invite students to probe and explore. Do not always give students every piece of information they need to know to complete an exercise. An exercise can be a much more effective tool if students are required to write a summery of the exercise at the end.
- Provide timely feedbacks to students. Laboratory exercises are typically designed to be self-paced with a minimum interaction between the instructor and the students. In this mode, learning is left unchecked for a long stretch of time. Instructors should find a

way to validate students' learning. The author employs three or four in-class computer-based tests in a semester. During these tests, students are asked to complete a small GIS project under time constraint. They can use notes and online help but are forbidden to talk to each other. These tests provide opportunity to evaluate students' learning of GIS operation.

- Encourage or make students to use manuals, online or offline. Manuals or other documents contain a wealth of information. Unfortunately, most students do not make use of this wonderful resource. On the other hand, instructors should spend time explaining technical terms and concepts in these documents.
- Develop students' self-learning skills, including active acquisition of information, goal setting, problem solving, evaluation, and time management (Denham, 1986; Unwin & Dale, 1990). Given the rapid rate of technological advancement, college graduate had to regularly update their trade tool and knowledge through out their professional career. It is, therefore, important to develop students self-learning skills so that they can succeed and prosper in their future learning (Denham, 1986; Marshall, 1993).

To summarize, this research investigate the effects of user interface style on students' learning based on observation. It concludes that the improved user-interface in ArcGIS presents a learner-friendly environment to learn GIS. However, the user interface sometimes provides too much information and makes it too easy to execute an operation without a good understanding of how it works. Furthermore, it is flexibility and smart design in ArcGIS environment make it difficult to validate students' learning. The traditional text-based interface demands for knowledge that is more formal and allows a smaller margin of error.

While the point-&-click style Desktop is an improved user interface, it is a treacherous learning environment. In such an environment, instructors should not equate a student's completion of a task to a manifestation of a firm grasp of the concept and the technique. Instead, instructors should find ways to circumvent the pitfalls in the point-&-click environment and ensure of students' learning effectiveness.

References

Denham, M. 2001. Information technology at degree level. Journal of Information Technology. 1(2):7-13.

DiBiase, D. 1996. Rethinking laboratory education for an introductory course on geographic information. Cartographica. 33(4): 61-72.

Kemp, K. K. & Goodchild, M. F. 1992. Developing a curriculum in GIS: the NCGIA Core Curriculum Project. cartographica. 28(3):38-54.

Marshall, K. P. 1993. Preparing for change in computer education for business. Journal of Education for Business. 68(6):376-381.

Raper, J. & Green, N. 1992. Teaching the principles of GIS: lesson form the GISTutor project. International Journal of Geographic Information Systems. 6(40:279-290.

Unwin, D., & Dale, P. 1990. An Educational view of GIS: some educational and sociological concerns. Journal of Geography in Higher Education. 14(2): 166-172.

Author Information

C. Victor Wu, Ph.D. is an assistant professor of Geography at California State Unviersity, Sacramento. Dr. Wu is interested in studying how students learn GIS. He can be reached at email address cvwu@csus.edu.