

Design Requirements of Argument Mapping Software for Teaching Deliberation

MATTHEW W. EASTERDAY, JORDAN S. KANAREK, AND MARALEE HARRELL

1 Introduction

Argument mapping software can be used to teach the (much needed) argumentation skills required for deliberation (van Gelder 2003, 2001; Harrell 2005b; Kirschner, Buckingham Shum, and Carr 2003; Kuhn 2005, 1991). But widespread usability problems among current tools have prevented teachers from using these tools in their classrooms (Harrell 2005a). In a comparative usability evaluation of argument mapping software in an introductory university philosophy course, we found that even the most popular tools fail to meet six key criteria: correct representation, flexible construction, ‘visual logic’, automation of nonsemantic operations, simultaneous display of multiple diagrams, and cross platform compatibility. Using our prototype, we show how a tool can satisfy these requirements by achieving the proper balance of drawing-based interaction and automation.

2 The Problems With Argument Mapping Software

The following scenarios illustrate the most common problems (with example packages exhibiting each problem noted):

Online Deliberation: Design, Research, and Practice.
Todd Davies and Seeta Peña Gangadharan (eds.).
Copyright © 2009, CSLI Publications.

1. A teacher wants to diagram an argument from a text but the software does not run on the school's operating system (Omnigraffle, Reason!Able, Argutect).
2. The teacher attempts to diagram the first claim in the text, but the tool does not allow the claim to be added unless the conclusion has been specified (Araucaria, Reason!Able).
3. The teacher reads the remaining text for the conclusion and enters it but has now forgotten the reasons supporting the conclusion and must start rereading. The teacher abandons the tool and draws the diagram on paper.
4. After copying the paper diagram into the tool, the teacher wants to represent a 'linked' reason: but the tool does not allow representation of linked reasons. (Argutect, Athena, Belvedere, Inspiration).
5. The teacher wants to move a reason a little to the left but moving the reason requires redrawing every arrow connected to the reason. (Illustrator).
6. The teacher tries to move the conclusion to the upper-left and enlarge the entire diagram so that it is legible when projected but the tool does not allow the size, layout or text format to be manipulated (Araucaria, Argutect, Athena Standard, Belvedere, Inspiration, Reason!Able).
7. Finally, the teacher tries to display two diagrams side-by-side for comparison, but the tool only displays one diagram at a time, (Araucaria, Belvedere, Reason!Able).

3 Design Requirements

The usability breakdowns illustrated in the previous scenario suggest six criteria that even the best argument mapping tools often overlook:

1. **Correct representation:** Tools must provide visual representations of structures unique to arguments such as 'linked reasons' and 'rebuttals'.
2. **Flexible construction:** Tools must allow input of elements in any order, e.g., claims before conclusions.
3. **'Visuallogic':** Teachers need some control over the visual properties including size, layout, and typeface because they have semantic connotations, e.g. size may indicate importance, layout may in-

dicating order, or type might simply need to be larger to be visible to students when projected on a screen.

4. **Automation of nonsemantic operations:** General purpose drawing programs often require operations unnecessary for argument mapping, for example in Adobe Illustrator, moving a claim (box) might require redrawing all the arrows connected to the claim.
5. **Multiple diagrams:** Tools should allow simultaneous presentation of multiple diagrams for comparison.
6. **Cross platform compatibility** Programs that satisfy all of the above criteria still might not run on school computers.

The table below summarizes the requirements satisfied by different argument mapping tools:

Tool	Represent.	Construct.	Visualogic	Automation	Multiple	Platform
<i>Special purpose argument mapping tools</i>						
Araucaria	*			*		*
Argutect		*		*	*	PC
Athena		*		*		*
Belvedere		*		*		*
Inspiration		*		*	*	*
Reason!Able	*			*		PC
<i>General purpose drawing tools</i>						
Omnigraffle	*	*	*	*	*	Mac
Illustrator	*	*	*		*	*

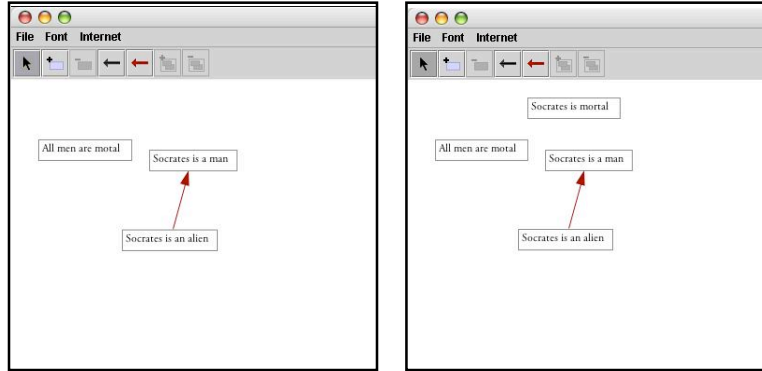
Table 1: Design requirements met by different tools

4 The iLogos Prototype

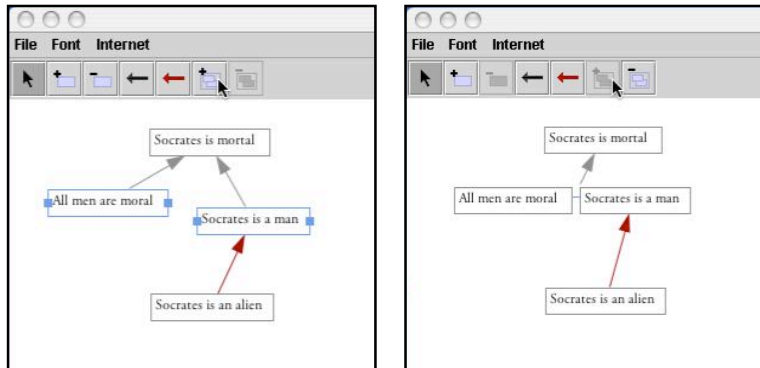
We developed a prototype that satisfies all of the requirements by balancing the flexibility, interaction, and ‘visualogic’ of a general purpose drawing program with the automation and specialized representations required for argument mapping. The following scenario illustrates the improvements, by reiterating the teacher’s workflow, this time using the prototype (with the satisfied criterion noted at each step):

1. The teacher opens the argument mapping program. iLogos runs on Windows, Macintosh, and Linux (*cross platform*):

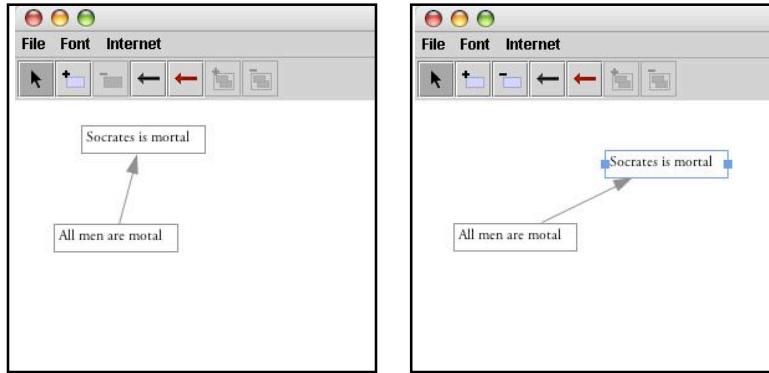
2. The article begins with the claim before stating the conclusion. The prototype allows the teacher to enter claims in any order (*flexible construction*):



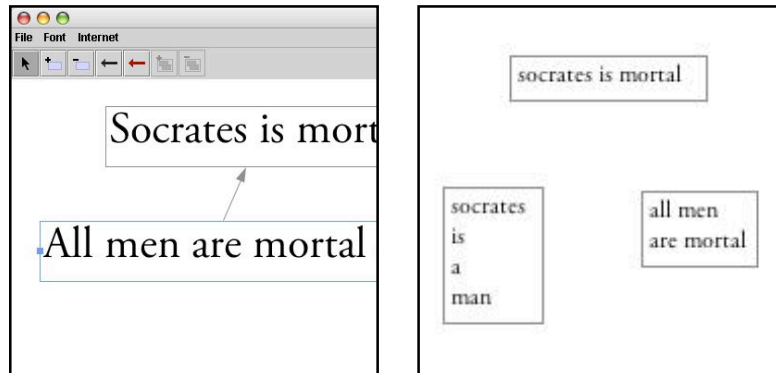
3. After the argument is entered into the tool, the teacher wants to show that some reasons are 'linked'. The prototype allows the teacher to create the desired argument representation by clicking on the 'linked reasons' button (*correct representation*):



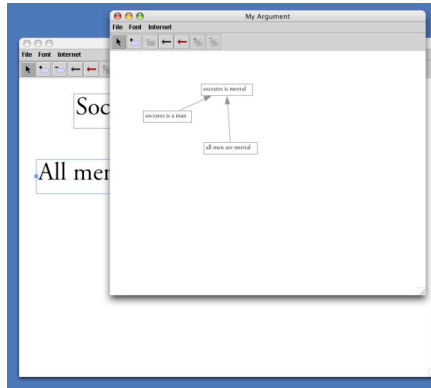
- The teacher moves a reason a little to the left. Arrows between reasons are automatically redrawn (*automation*):



- The teacher wants to move the conclusion nearer to the top of the screen, making it large enough to be seen when projected. The prototype allows manipulation of visual properties such as layout, text, and size using standard drawing conventions (*'visualogic'*):



6. Finally, the prototype allows the teacher to compare diagrams by displaying two of them simultaneously (*multiple diagrams*):



5 Conclusion

By balancing the interaction of a drawing program with automation and support for visual representation of argumentation structure, iLogos satisfies the six design requirements allowing argumentation teachers to successfully use the software in the classroom. By demonstrating how to overcome the functional and usability obstacles of argument mapping software, we hope it will lead to improved usability in other tools and open the door to widespread use of argument mapping software for teaching deliberation.

References

- Adobe Illustrator 9 (Version CS2) [Computer software]. 2005. San Jose, CA: Adobe Systems Incorporated.
- Arguetect (Version 3.0) [Computer software]. 2002. Pittsburgh, PA: Knosis.
- Araucaria (Version 3.0) [Computer software]. 2005. Scotland: University of Dundee, Department of Applied Computing.
- Belvedere (Version 4.0) [Computer software]. 2002. University of Hawai'i at Manoa, Laboratory for Interactive Learning Technologies.
- Kirschner, P. A., S. J. Buckingham Shum, and C. S. Carr, eds. 2003. *Visualizing Argumentation: Software Tools for Collaborative and Educational Sense-making*. New York: Springer.
- Kuhn, D. 1991. *The Skills of Argument*. New York: Cambridge University Press.
- Kuhn, D. 2005. *Education for Thinking*. Cambridge: Harvard University Press.

- Harrell, M. 2005a. Using Argument Diagramming Software in the Classroom. *Teaching Philosophy* 28(2): 163-177.
- Harrell, M. 2005b. *Using Argument Diagrams to Improve Critical Thinking Skills in 80-100 What Philosophy Is*. (Tech. Rep. CMU-PHIL-176). Pittsburgh, PA, Carnegie Mellon University.
- Inspiration (Version 7.6) [Computer software]. 2005. Portland, OR: Inspiration Software.
- Omnigraffle (Version 4) [Computer software]. 2005. Seattle, WA: Omni Group.
- Reason!able (Version 1.1c) [Computer software]. 2001. Australia: University of Melbourne.
- Rolf, B. and Magnusson, C. 2005. Athena (Version 2.5) [Computer software]. Karlskrona, Sweden: Blekinge Institute of Technology.
- van Gelder, T. 2001. How to Improve Critical Thinking Using Educational Technology. *Meeting at the Crossroads: Proceedings of the 18th Annual Conference of the Australian Society for Computers in Learning in Tertiary Education*, eds. G. Kennedy, M. Keppell, C. McNaught, and T. Petrovic, 539-548. Melbourne: Biomedical Multimedia Unit, University of Melbourne.
- van Gelder, T. 2003. Enhancing Deliberation through Computer Supported Visualization. *Visualizing Argumentation: Software Tools for Collaborative and Educational Sense-making*, eds. P. A. Kirschner, S. J. Buckingham Shum, and C. S. Carr, 97-115. New York: Springer.

