## : What technical and human factors influence DSS success?

by Dan Power

Editor, DSSResources.com

A decision support system is a success when it is used and when its use actually makes a positive difference in decision making. Targeted users must perceive the DSS does something worthwhile and in general the system must be easy to learn and use. User interface design is critical to success. To improve usability, design rules (cf., Power, 2007) should guide DSS user interface construction. According to Larson (1982), user interface design success is influenced by 11 factors. Technical system factors include: execution time; the system's versatility; quality of help provided; adaptability; and uniformity of the commands and interface. Human factors include: learning time; ease of recall; errors made by end users; concentration required; fatigue from using the system; and the fun the user has while operating the system. Let's examine each factor for modern DSS.

**Execution time.** Although the time it takes to use a DSS impacts future use, faster is not always better. Only a fraction of task time is determined by actual DSS execution time. Developers have some control over execution time by choosing appropriate software tools and hardware that are fast enough to provide the desired responsiveness. Also, a well designed user interface minimizes wasted time on the user's part.

**Versatility.** Most DSS are used to perform a variety of tasks. A specific DSS must be versatile enough to accommodate the full range of tasks for which a decision maker will want to use it. Once a system becomes widely used, users may want new functionality incorporated in the DSS. While it is reasonable for these new tasks to require development work, an effective user interface framework should accomodate new functionality. Versatility and flexibility in design for a DSS are however less of a problem for modern DSS (Power, 2008).

**Quality of Help.** Does the system provide appropriate help when the end user has trouble? Online help facilities are the norm for DSS. More and more development tools make it easy to incorporate online help into a system. Wherever possible, help should be context sensitive. The Help facility should recognize what the user is trying to do, or at least what screen the user is looking at, and provide help that is tailored as closely as possible to the current need.

**Adaptability.** Does the system adjust to the DSS user as he or she becomes more experienced? Does it tailor itself to the habits and styles of different users? It is generally sufficient to let more experienced users select an "expert user" mode in which prompts are minimized.

**Uniformity of commands and interface.** Are the commands of this DSS identical to equivalent commands in other systems? It is difficult to learn and remember widely differing commands and interfaces and switch frequently among them. A DSS developer must be aware of other systems that the intended DSS users are familiar with. Also, using familiar terminology is a factor in making commands look uniform.

**Learning time.** How long does it take a novice to learn the DSS? A design that provides for rapid learning must take into account what the user knows and how the user's mind fits that knowledge together. The DSS should seem intuitive to targeted users.

**Ease of recall.** How easy is it to recall how to use the system after not using it for some time? This is a more important factor for DSS than for data processing systems because managers often return to a DSS after a long interval of nonuse. A user interface that facilitates recall will reduce the time it takes to "get back up to speed" with the DSS.

**Errors.** How many errors does the DSS user make, and how serious are those errors? The most serious errors lead to wrong decisions as a result of system misuse. Following closely behind are errors that corrupt a corporate database. Then come errors that "crash" the computer, followed finally by errors that waste the user's time but have no other bad effects. Fortunately, most user errors will be in the last category. Understanding the users' usual decision making process can help minimize errors.

**Concentration required.** How many things must a user keep in mind while using the DSS? Most people have difficulty keeping more than six or seven active facts in mind at any one time. One way to reduce the memory load is to label screens and output appropriately.

**Fatigue.** How quickly does the user tire while using the DSS? Physical fatigue is seldom a factor with DSS because usage frequency is usually low. However, mental fatigue can occur. Minimize mental fatigue by reducing the concentration required and by asking for information once and in a systematic manner.

**Fun.** Does the user enjoy interacting with the DSS? This means designers should minimize frustrations that come from using a a poorly designed DSS. Fun means "not aversive" and easy to use.

Design guidelines are intended to impact these factors and these factors create the need for guidelines. Building an effective DSS remains challenging. Improving awareness of what factors impact success of a DSS and especially of the user interface can lead to better, more effective DSS.

References

Ambysoft, URL http://www.ambysoft.com/essays/userInterfaceDesign.html

Constantine, L., L. A.D. Lockwood, Software for Use: A Practical Guide to the Models and Methods of Usage-Centered Design, Addison-Wesley Professional, 1999.

Larson, J. End User Facilities in the Nineteen Eighties. IEEE Computer Society, Los Alamitos, CA., 1982.

Power,D. J., "How important is flexibility for a DSS?" DSS News, Vol. 9, No. 14, July 13, 2008.

Power,D. J., "What rules should guide DSS user interface design?" DSS News, Vol. 8, No. 19, September 23, 2007.

Power, D., Decision Support Systems: Concepts and Resources for Managers, Greenwood/Quorum Books, 2002.

Shneiderman, B. Designing the User Interface: Strategies for Effective Human-Computer Interaction. (2nd edition) Reading, MA: Addison-Wesley, 1992.

Wikipedia, User interface design, http://en.wikipedia.org/wiki/User_interface_design

Author: Daniel Power
Last update: 2008-07-23 09:07