1-1-1991

Decision Support Systems Process Tracing Using Hypermedia

Robert P. Minch
Boise State University

This document was originally published by IEEE in *Proceedings of the Twenty-Fourth Annual Hawaii International Conference.* Copyright restrictions may apply. DOI: 10.1109/HICSS.1991.184173

DECISION SUPPORT SYSTEMS PROCESS TRACING USING HYPERMEDIA

Robert P. Minch

Computer Information Systems & Production Management
College of Business
Boise State University
Boise, ID 83725
208-385-3491
Bitnet: RISMINCH@IDBSU

Abstract

Two main alternative approaches to analyzing decision processes--implicit input/output inference models and explicit tracing of observable decision process manifestations--are reviewed, with emphasis on explicit tracing methods. An emerging technology, hypermedia, is then examined as to how it may facilitate the process tracing method of decision making analysis. Examples are presented of mappings between hypermedia computer/user interface functions (such as mouse movements and mouse clicks) and underlying decision process functions. Issues of data quality, breadth of application, and implementation cost are discussed. Hypermedia process tracing is compared with other process tracing methods, including monitoring of eye movements, verbal protocols, and non-hypermedia computerized logging. Advantages and disadvantages of the hypermedia approach are identified. Further directions for the application of hypermedia process tracing include areas related to information retrieval, use of models, study of user interfaces, and the potential for using the techniques to identify and compare cognitive processes of decision makers.

Introduction

Decision support systems (DSS) research has emphasized decision-related inputs and outputs, neglecting the *process* of decision making. Inputs typically focus on characteristics of the information, models, technology, decision maker, decision task, and decision environment. Outputs are measured though variables such as economic gain, quality of and time taken for decisions, and confidence in decisions made. Unfortunately, the input/output approach does not adequately capture the decision *process* occurring during decision making, whether aided by computer or not. The regression approach and other input/output approaches attempt to predict outcomes from input cues without considering time-phased decision-related activities, and hence do not afford a detailed look at the decision processes themselves. Process tracing models [Einh79] are responses to this problem which capture at least some aspects of the decision process. The process tracing approach utilizes techniques including information boards, eye movement tracing, computer logs, and verbal protocols to allow a more complete monitoring of decision activities. Input/output and process tracing approaches are compared in [Einh79], while a review of process tracing methods applicable to DSS appears in [Todd87].

The purpose of this paper is to provide a discussion of the important issues involved in using a relatively new user interface, hypermedia, for process tracing. Rapidly gaining popularity in the marketplace, hypermedia has many applications in DSS [Minc89]. Hypermedia has the potential to be a promising vehicle for process tracing studies. With its highly interactive user interface, which may be easily and unobtrusively monitored, hypermedia will be shown to offer certain advantages over previously developed process tracing techniques in DSS environments.

The following sections will review process tracing methods and hypermedia technology. Use of hypermedia to facilitate process tracing will then be discussed, including how user inputs might be mapped to decision processes, particular hypermedia features which facilitate process tracing, comparisons to other methods on a number of criteria, and prescriptions for use including potential application areas. A summary and conclusions section completes the paper and includes suggestions for further research.

Process Tracing

Process tracing methods have been developed to study the decision processes which transform decision inputs to decision outputs and occur between receipt of inputs and issuance of outputs in time. These processes include perceiving and recognizing stimuli, remembering and searching for information, inducing rules, recognizing patterns, formulating concepts, and applying these in sensing, formulating, and solving problems [Rama87, p. 140]. Because decision making cannot be directly observed in the human mind, it is important to note that *all* process tracing methods developed to date rely on analyzing *observable artifacts* from the decision making process. Four common process tracing methods are briefly reviewed below: information display boards, eye movement tracing, verbal protocols, and computer (non-hypermedia) logging. More complete discussions and comparisons of these process tracing methods can

0073-1129/91/0000/0442$01.00

be found in [Todd87] and [Russ78]. Choice of one method need not exclude others--multi-method approaches using several process tracing techniques simultaneously have been proposed [Todd87] and used [Payn76].

Information Display Boards

Information display boards arrange pieces of decision-related information in an organization appropriate for the problem at hand, and require the decision maker to take some explicit action to access each piece of information. For instance, one study [Payn76] used an information board with rows corresponding to alternatives in the search for an apartment and columns corresponding to attributes (such as noise level), respectively. Subjects were required to open envelopes to obtain desired information pieces, which was again hidden after each use. Information display boards allow the monitoring of the first and subsequent uses of information, but do not capture the way information is used. Furthermore, the mechanism itself may interfere with and slow the decision making process.

Eye Movement Tracing

Eye movement tracing also monitors information used by decision makers through the surrogate measurement of tracking their eye movements with various devices. Some researchers distinguish eye movements from eye fixations in that fixation research emphasizes sequence rather than density or duration of fixation [Russ75]. Compared to information display boards, this method reveals more detail but is more difficult to apply in a wide range of settings and involves more costly equipment [Russ78]. Sophisticated monitoring equipment can produce very detailed maps and scanning patterns allowing dynamic analysis which also has been used in non-DSS research such as how algorithms are read [Cros90].

Verbal Protocols

Verbal protocols involve the audio and sometimes video recording of subjects verbalizing their thought processes while engaged in decision making. The transcripts are decomposed into short phrases representing assertions or references, which are sometimes subsequently coded into formal categories [Payn78]. There is some subjectivity involved in the coding process. Verbal protocols are viewed as possibly the most informative method with a wide range of applicability, but may suffer from problems of validity and obtrusiveness [Russ78]. Primarily because of their data richness, verbal protocols are viewed by some as the most powerful of the current process tracing tools for use in DSS research [Todd87]. A review of issues involved in collecting, coding, and analyzing verbal reports can be found in [Eric80].

Computer Logging

Computer logging is the recording of human/computer interaction artifacts while a computerized decision aid is used. The range of computer logging is from relatively simple automated versions of information display boards to more complex systems which record virtually every aspect of user interaction with a variety of tools. Computer logs can show not only what data is accessed, but what models, algorithms, and transformations are applied to that data. The unobtrusiveness of this method makes it particularly relevant to DSS research [Todd87], but it may offer only limited insight into the decision maker's thought processes. A further weakness of current computer logging techniques is that the log indicates only actions actually chosen, not those considered but eliminated [Todd87].

Hypermedia

Hypermedia systems are computer-implemented networks of nodes (database or media objects) and links (arcs used to traverse between nodes). Although various media including audio and video are included under the general term hypermedia, the most common medium currently employed is text, hence the more specific term hypertext is used to describe these systems. This paper will use the term hypermedia when it is desirable to imply generality, and hypertext where no multi-media extensions are addressed.

In hypertext systems the database objects are collections of text which are displayed to the user as screen windows. Links between the objects appear to the user as selectable tokens (also called buttons or link icons), forming anchors in source and destination nodes and corresponding to pointers in the database. Links are activated by selecting a link icon through the use of a pointing device such as a mouse or a touchscreen. The operationalization of machine-supported links allowing traversal of a non-linear network of nodes is central to the hypertext paradigm. Hypertext appears to the user as a system of imbedded menus (as opposed to explicit menus) that allows selection of items in context within a current window for further investigation or manipulation [Kove86].

The most prevalent use of hypermedia technology has been to facilitate browsing of (exploration and information retrieval from) textual databases. A hypertext database may be browsed in at least four distinct ways: (1) spontaneously and iteratively by choosing any desired link from the present node; (2) in a directed manner by following a pre-defined path through the network; (3) in search mode by invoking a mechanism to locate a node having some desired characteristic; and (4) in quasi-direct access mode by viewing a graphical browser which displays a visual representation of all or part of the network and allows the user to select a node to visit. Usability of hypertext for information browsing and other purposes may be affected by the level of *granularity* (average node size relative to total database size), as well as a number of other factors.

In addition to browsing of textual databases, hypertext systems commonly have been used in three additional ways [Conk87]: (1) as macro-literary systems supporting on-line collaborative writing; (2) as problem exploration tools such as outliners; and (3) as general experimental vehicles most often related to textual applications. More recent applications of hypertext with relevance to DSS include computational hypertexts where nodes may be executable models or programs [Minc89] and multi-user hypertexts supporting collaborative work [Conk88].

Readers seeking background information on hypermedia will find the following sources valuable. A recently published book [Niel90] comprehensively treats hypertext and hypermedia, and includes a very useful annotated bibliography--it would be an excellent starting place for any researcher entering the subject area. A survey of hypermedia applications in academia appears in [Beck88]. Hypertext and hypermedia bibliographies include [Fran88], [Legg89], and [Yank87]. A comparison of ten hypertext systems which examines their general, node, link, authoring, and browsing characteristics is reported in [Schn88]. Applications and research areas involving hypertext in decision support systems are discussed in [Minc89].

Hypermedia Process Tracing

Hypermedia process tracing is proposed as a general approach for how user inputs may be mapped onto and interpreted as decision processes. Features of hypermedia which make it particularly applicable to process tracing are discussed, followed by comparisons on a number of criteria to other process tracing methods, and prescriptions for use and identification of potential application areas.

User Input Mappings

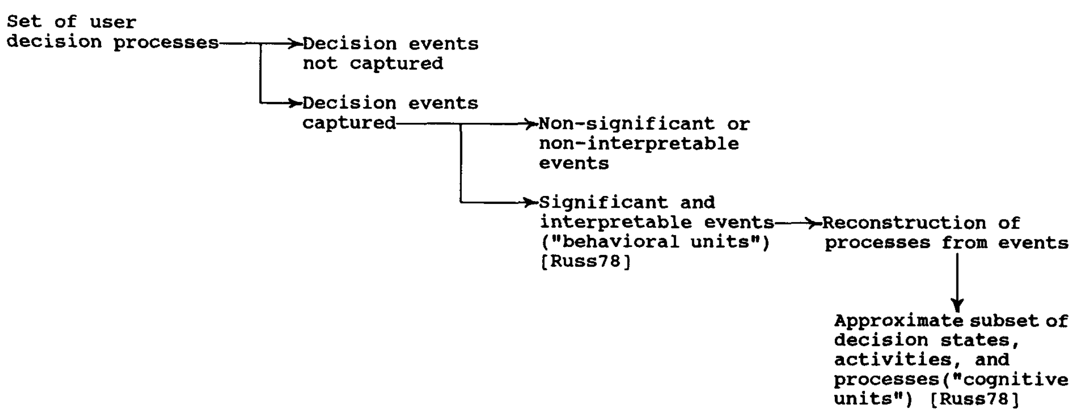
Mappings of user inputs onto decision processes are crucial in the process tracing approach to studying decision making. Useful concepts and nomenclature which can describe these mappings are found in the field of discrete-event system simulation [Bank84]. The *state* of a system is the collection of variables which describe the system at any time, relative to the objectives of the study. In decision process tracing this would include the set of alternatives currently under consideration. An *event* is an instantaneous occurrence that may change the state of the system (for instance, a decision maker initiates access to a particular piece of information or discards an alternative). An *activity* represents a time period of determinate length, normally bounded by two events (e.g., the events opening and closing a text window bound the activity of accessing, and perhaps reading, the text inside the window). Finally, a *process* is a time-ordered collection of events, activities, and delays which are somehow related.

There are three fundamental assumptions behind the proposed hypermedia process tracing method:

1. While using a hypermedia-based DSS, the user's decision processes will at least partly manifest themselves as machine-readable actions (events) at the user interface level.
2. These events can be captured, identified, and stored in real-time.
3. Appropriate analysis of the user interface event-log data will allow at least a partial inferential reconstruction of decision states, activities, and processes associated with the user's decision making.

Some information loss and interpretation error is possible at each stage in the process, as shown in the model presented in Figure 1.

Figure 1:
Hypermedia User Input to Decision Process Mappings



The tasks of hypermedia process tracing which must be performed are to collect user interface events and map these onto inferred decision states, activities, and processes. This method is consistent with the construction of a Problem Behavior Graph (PBG) [Newe72] in which PBG nodes represent states of knowledge and discrete operators (like events) form transitions between states. An example of a few of the possible mappings will illustrate the process in hypertext. This example assumes: (1) a hypertext system (nodes may be text or two-dimensional display graphics, but hypermedia features such as audio and full-motion video are not included); (2) a mouse input device; and (3) a discrete, multi-criteria decision making context for which monitoring the accessing of information (rather than models, etc.) is a primary emphasis of the research. The example mappings, shown in Table 1, relates captured user interface events to states, activities, and processes with event/interpretation pairs.

Table 1
Example of User Input Mappings

EVENT	INTERPRETATION	
While in node n, the mouse pointer is moved into the link icon area which links to node m (assume both nodes relate to the same alternative A_i).	State:	Changes in list of "touched" or "considered" nodes.
	Activities:	Decision maker is contemplating access to additional information.
	Processes:	Decision maker is in the process of collecting information about alternative A_i.
While in node n with mouse pointer in the link icon area which links to node m, mouse button action o is taken.	State:	Changes in list of accessed nodes and current active node.
	Activities:	Decision maker has had access to the information in node n for time length t_n.
	Processes:	Decision maker is in the process of collecting type o information about alternative A_i.

Varying levels of analytical sophistication can be brought to bear upon the problem of transforming user interface events into inferred decision processes. The simplest involves a mere tallying of decision inputs such as the list of information sources accessed, rate or frequency of access, etc. This allows analysis similar to the input/output approach mentioned earlier. More sophisticated approaches involve the use of time-phased sequences of inputs, such as that employed in sequential lag analysis [Bake86]. In this method, event sequences of various lengths are examined for significance (e.g., if a node visitation sequence of A, B, A is observed, there may be some importance attached to the user returning to node A after visiting one intermediate node). Perhaps the most sophisticated analysis might involve logic modeling of the system, inferring decision processes from large numbers of implicitly stored interaction patterns. A simple example with fairly general applicability is the logical inference contained in Figure 2, the Prolog code (with natural language translation). An operational system for user input mapping might have a number of such rules which detect and interpret significant patterns of DSS use. Thus if appropriate assumptions are made it is possible to draw conclusions about the decision making process using sophisticated inference mechanisms.

Hypermedia Features Facilitating Process Tracing

Many hypermedia systems employ a variety of link types [Trig83] [Hala87]. For instance, the commercially available Guide system [Owl87] has "replacement" links which supplant the link icon with a replacement text, "reference" links which traverse to another node in the network, and "note" links which temporarily overlay the contents of another node on the current screen as long as the mouse button is held down. Because the different link types have different meanings to the user, hypermedia process tracing may be used to record the link types used along with other information so that the proper interpretation can be assigned. In the Guide system, for example, use of a note link may indicate that the user is unsure about the meaning of a term and wishes a definition (a common use of note links). Use of a replacement link normally would connote a different purpose such as pursuing a related or even different line of information gathering or reasoning.

Hypermedia process tracing is an intrinsically unobtrusive, concurrent, and neutral-probing recording of movement through the network by the user. Concurrent versus retrospective and structured-proving versus neutral-probing methods are discussed in [Todd87], where it is proposed that concurrent, neutral-probing techniques are most desirable for DSS process tracing. Concurrent process tracing avoids problems of inadequate or distorted memory recall, which may occur with retrospective methods. Neutral-probing methods involve asking the user only to describe their general processes as they progress, not to answer specific

Figure 2
Prolog Code with Natural Language Translation

```
used(Information,Decision_phase):-          (Information was used in this DM phase if
    occurred(Decision_phase,_,T1),          it ended at time T1 and
    contains(Node,Information),             a particular node contained that information
    displayed(Node,_,T2),                   whose display ended at time T2
    T2 <=T1.                                and time T2 came no later than time T1).
```

questions about their problem solving strategy which might require separate attention and affect the process itself. Hypermedia process tracing is neutral-probing because the process is recorded without specific inputs requested from the user for the sake of the process tracing alone.

In some cases, structured-probing methods may be desired. Hypermedia is particularly well suited to efficiently implement structured probing. For instance, if certain intermediate decision activities are anticipated by researchers but not captured by neutral-probing methods (regardless of technique such as verbal protocols or hypermedia), automatic traversal to a special data-gathering node may be scheduled by the system to ask the user for the desired information. This might involve the user clicking on one or more of a number of buttons indicating possible activities in which he or she is engaged, after which there would be an immediate return to the problem at hand. Such "pop-up" nodes could be scheduled in such a way as to randomly sample the decision process, or as part of a more sophisticated experimental design.

Hypertext systems often allow the user to modify or establish new nodes and links. For instance, the user may wish to attach an annotation to an existing link with an interpretation or personal note. This may be a valuable key to the user's use of working storage to supplement short-term memory. Users also may create new links between nodes to create a structure which more closely resembles their own cognitive organizations. Both of these capabilities are extremely important to decision process tracing. Indeed, construction of cognitive maps [Axel76] using hypertext facilitates gaining insight into the user's cognitive processes [Minc90].

Computational hypertexts allow nodes to contain models or programs whose execution is initiated by the selection of the link leading to that node. In this case we are able to monitor not only user information accesses, but also the use of algorithms, decision aids, and other decision support system components. At fine levels of granularity, computational nodes might be as elementary as an arithmetic operator in an equational network, while at courser levels nodes might include self-contained optimization or simulation models. Tracing the patterns of access, modification, and use of models may provide valuable insights about the decision maker's use of those decision aids.

Hypermedia features such as its audio and video capabilities could prove useful in partially automating the coding tasks associated with process tracing. Because both audio and visual information are stored in digital form, they can easily be linked to other events monitored by the computer. Thus user utterances can be conveniently linked with nodes visited just before or after the utterance. This allows systematic analysis of verbalizations which correspond to the content and type of nodes as well as the type of links. For example, a researcher might want to retrieve all verbal protocols which occurred just before visiting a particular node in the network. Similarly, video information which might include eye fixation data can be linked with important events.

Comparison of Hypermedia Process Tracing to Other Methods

Table 2 presents a comparison between hypermedia process tracing and three other methods (eye movement tracing, verbal protocols, and non-hypermedia computer logging) on seven previously established criteria (detail revealed, informativeness, and validity of data quality; range of settings and unobtrusiveness related to breadth of applicability; and ease of use and equipment price related to research cost). The choice of these other process tracing methods and criteria represent an adaptation and synthesis of selected relevant methods and criteria compared in [Todd87] and [Russ78]. Specifically, the table follows the format of [Russ78], which used the same column criteria but included three additional methods: chronometric analysis, information boards, and input/output analysis. These methods were appropriate to the consumer information processing research environment of that study, but are less relevant to DSS process tracing. For example, manual information boards would have little applicability to computerized decision support systems, particularly when their functionality may be easily incorporated into an on-line system.

The first two rows of Table 2 are taken directly from [Russ78], where each method is rated poor, fair, good, very good, or excellent on each of the seven performance attributes. Each cell entry is justified in the context of consumer information processing research in that paper (the justifications for these entries are included in [Russ78] and are not repeated here). These

Table 2 Comparison of Hypermedia and Other Process Tracing Methods[1]							
	Performance Attributes						
	Data Quality			Breadth of Acceptability		Cost	
Method	Detail Revealed	Informa-tiveness	Validity	Range of Settings	Unobtru-siveness	Ease of Use	Equipment Price
Eye movement tracing	Excel.	Fair	Excel.	Fair	Good	Fair	Poor
Verbal protocols	V. Good	Excel.	Fair	Excel.	Poor	Good	V. Good
Non-hypermedia computer logging	Good (1)[2]	Fair (3)	Excel. (5)	V. Good (7)	Excel. (9)	Good (11)	Excel. (13)
Hypermedia process tracing	V. Good (2)	Good (4)	Excel. (6)	V. Good (8)	Excel. (10)	V. Good (12)	Excel. (14)

[1] Adapted from [Russ78].
[2] Numbered items are discussed separately in the text of the paper.

entries are suitable and appropriate for DSS research as well.

The third and fourth rows of Table 1 are relevant to DSS process tracing using non-hypermedia computer logging and hypermedia process tracing, respectively. These have not been previously discussed in a method/performance attribute matrix as in Table 2, thus each cell in these rows is numbered to serve as a reference for discussion in the following paragraphs.

(1). Detail revealed by computer logging is typically good, but varies considerably according to the DSS software used. The particular user interface and other implementation specifics will determine the level of recording detail, which may be considerable in volume for systems designed with process tracing in mind. Macro-level commands issued by the user in pre-existing software systems may limit the level of detail.

(2). Detail revealed by hypermedia process tracing can be expected to be very good. For comparison, eye movement tracings can detect many fixations per second [Cros90], and verbal protocols have encoded on the order of 19 elementary cognitive units per minute [Russ78]. Hypermedia interactions with mice or other input devices easily can meet or exceed the rate of events recorded by verbal protocols, particularly by experienced users. Furthermore, the level of detail revealed is dependent on the granularity of the hypertext or hypermedia system, which is controllable by the system designer.

(3). Informativeness is defined as the cognitive interpretability of the process tracing data collected [Russ78]. Informativeness of non-hypermedia computer logging is only fair, due primarily to the limitation of recording only choices actually *made* by the user rather than also including those *considered* but not made. Also, little insight is gained as to *why* a particular choice was chosen.

(4). Informativeness of hypermedia process tracing may be good in a well-designed system for several reasons. First, when choices are made by the user, we have considerable context in which to place those choices. We know which nodes were previously visited, the order of visitation, and the node contents. We also know the contents of all nodes, windows, and other currently displayed screen information. Second, information on paths or choices *considered* may be available by recording mouse movements even when selection is not made (this may be imperfect, however, as it may reveal only choices which the user intended to make but reconsidered just before selection). Finally, in generalized hypermedia systems the use of both information and models may be captured, whereas techniques such as eye tracing capture use of information only.

(5) and (6). Validity is defined as freedom from subject censorship or distortion of the process tracing data [Russ78]. Because both non-hypermedia and hypermedia computer systems have the ability to collect user interaction without the user's knowledge or involvement, validity for both techniques is excellent.

(7) and (8). The range of settings to which both computerized logging/process tracing methods may be applied in DSS is very good. Applications software may require modification to collect desired process tracing data, or systems software utilities may be used. For instance, microcomputer software presently exists which

will record every user keystroke to a file while using any application under the operating system. Facilities have been easily added to popular hypertext software such as hypercard which will record and time-stamp node visitation during a user's session [Niel88].

(9) and (10). Unobtrusiveness for both non-hypermedia computer logging and hypermedia process tracing are excellent. Recording can occur in real-time with little or no delay to the user, who may not even be aware that the interaction is being recorded.

(11). Ease of use for non-hypermedia computer logging (from the researcher's perspective, not the user's) is good because most DSS environments may incorporate the necessary recording features, but will require modifications to implement them.

(12). Ease of use for hypermedia process tracing is very good because the logging facilities are available with essentially no extra effort on the part of the designer [Niel88]. Furthermore, some aspects of the logging may be made available to the user as an aid to system use [Niel88].

(13) and (14). Equipment price for both non-hypermedia computer logging and hypermedia process tracing is excellent in an existing computerized DSS environment. No additional equipment is needed assuming adequate on-line storage space for the log itself is available.

Although the importance of the measures used in Table 2 may vary for particular researchers and application areas, we can summarize a few important points. Verbal protocols probably remain the most informative method, although sophisticated approaches to computerized logging such as those involving hypermedia can narrow this gap and provide advantages on other criteria (hypermedia process tracing has the second best informativeness and is superior to verbal protocols on four other criteria). Eye movement tracing provides the most detailed data and has excellent validity, but has performance inferior to other methods on several other criteria. Non-hypermedia computer logging is essentially dominated by the hypermedia approach, due to a number of intrinsic hypermedia features discussed earlier. Thus for many purposes researchers will find that hypermedia process tracing not only offers new and unique capabilities but also compares very favorably with other existing process tracing methods. Furthermore, a combination of process tracing methods (discussed later) may prove useful as well.

Prescriptions for Use and Potential Application Areas

The fullest benefit from DSS hypermedia process tracing lies in studying the problem solving behavior of decision makers using a system which is intrinsically structured as hypertext or hypermedia. Although research has been done into "usability" of such systems, this has largely focused on non-decision making tasks such as information retrieval [Niel90]. We need to progress to the point where we study actual decision making in realistic situations using the tools provided by hypermedia.

A second mode of application which also may be fruitful involves retrofitting existing decision-aiding systems with new hypermedia user interfaces to gain the process tracing benefits therein. This could be done as a type of front-end system which might not only make the underlying system easier to user, but allow process tracing data collection as well. Computational hypertexts and other extensions of the basic model are sufficiently general to enclose many DSS tools in a kind of "process tracing envelope."

Potentially Fruitful Application Areas for Hypermedia Process Tracing

Hypermedia is particularly well suited to DSS in general, due to its support of associative memory, non-linear modes of decision making, and support of ancillary activities such as problem structuring, communication and information gathering [Minc90]. This intrinsic suitability, combined with the process tracing capabilities of hypermedia, suggests a number of potentially valuable application areas in DSS.

With the primary goal of process tracing being to better understand the <u>process</u> (not merely the inputs and outputs) of decision making, many research questions immediately present themselves. Does decision making in fact proceed in stages such as intelligence, design, and choice [Simo77]? Is it a linear process, or a non-linear one in which some steps might be repeated several times? When do decision makers employ techniques such as problem reduction [Bonc81], or other strategies from the list of over seventy such strategies which have been suggested [MacC76] to deal with complexity? Do experts solve problems differently from novices? How do decision support needs and usage patterns differ according to the degree of system formality, organizational level, and decision making mode [Phil88]? Although each of these questions has been addressed before, hypermedia process tracing technology should allow us better insight and a better opportunity to resolve some of these issues. Indeed, perhaps we will be able to develop entirely new and more useful models of decision making.

In DSS where information retrieval is a main research emphasis, hypermedia process tracing can unobtrusively monitor what information is accessed as well as the rates and patterns of access. We might investigate issues such as what forms of information retrieval mechanisms (structured database query, free-text search, etc.) are better for supporting particular types of or phases in decision making. For large classes of users (such as with public information services), the results can be used to better index and otherwise organize the data base for future users. Studies of smaller groups in group decision making [Krae88] might gain insight into what information tends to be accessed and shared at what stages in group processes. For individual users, findings might allow the customization of data base linkages and other features (perhaps even in real-time) to better match their individual preferred patterns of use. Researcher control of data base granularity in studies involving subjects solving specific problems allows the possibility for significant insights into how and when decision makers use information.

In DSS where the use of models is a primary emphasis, hypermedia process tracing can assist in studying the level and pattern of modeling features utilized, addressing questions about which modeling features are most used, most closely linked to particular items of information, etc. Model management systems for DSS are often proposed (e.g., [Lian85]) but infrequently subjected to empirical tests regarding their usefulness. Again the level of analysis (e.g., public, group, or individual), type of problem, and other factors would suggest a number of relevant research questions. A further model-related application is similar to that proposed by Payne for the use of verbal protocols--that of building and testing models [Payn78]. In this case, hypermedia process tracing of a human expert using an advanced set of computerized tools could perform a knowledge acquisition function, for example. By observing patterns of expert use, a model which simulates that expertise could be built and tested.

It may be possible to design better future user interfaces based on knowledge obtained from hypermedia process tracing. Previous user interface studies examining user performance with command, menu, and iconic interfaces have found significant differences among interface modes [Whit85]. Unfortunately, the majority of the user interface studies are input/output analysis type (i.e., they tend to measure whether users perform better or worse on a particular outcome assessment based on the type of user interface used) rather than process tracing. Process tracing may offer much more insight into the actual functioning of user interfaces. In addition to studying a small number of distinct interface techniques such as command, menu, and iconic interfaces, hypertext may be used to manipulate several design variables over a virtually continuous range. For instance, the level of directedness (availability and degree of enforced use of pre-defined network paths) may be continuously varied from minimally directed to strongly directed control of user browsing and tool selection. With efficient monitoring and interpretation of user input, process tracing could be useful in real-time for purposes such as assisting the user if necessary when certain interaction patterns are detected.

Finally, a possible application area might be to design a new operational measure of a person's cognitive processing type based on the observed interaction patterns between the user and a hypermedia system. Controversies over cognitive style in DSS research [Hube83] seem to have shifted research emphasis toward more attention to decision processes rather than individual characteristics [Rama87]. Measures of cognitive processing type which could be easily obtained through the administration of an on-line procedure could prove useful in tailoring DSS to specific user needs. In fact, such an instrument might actually be an integral part of a process tracing user interface and be quite unnoticeable to the user.

Potential Difficulties

The primary obstacle in hypermedia process tracing (like most other process tracing methods) is that we have only incomplete links between a user's actual decision making processes and their detectable outward manifestations. In areas where hypermedia process tracing is not as effective as other methods, however, parallel joint methodologies may be a promising approach. For instance, verbal protocols might be used simultaneously with hypermedia techniques to compensate for the weaknesses of each individual approach. This also provides a technique for cross-validation of approaches which increases confidence in the subsequent use of either approach individually (for example, once a particular interpretation of hypermedia interaction has been validated through verbal protocols, that interpretation may be deemed more reliable in future research involving hypermedia process tracing alone).

A second impediment to hypermedia process tracing in DSS is that only structured or (at best) semi-structured problems are supported. This stems more from the characteristics of a DSS [Spra82] than from the hypermedia technology. In fact, issue-based hypermedia systems such as gIBIS [Conk88] have been used to support relatively free-form discussions between multiple participants for a variety of problems with little initial structure. So it seems in this case we are more constrained by the technology whose use is to be traced than by the tracing technology itself.

Summary, Conclusions, and Suggestions for Further Research

Recognizing the need to study decision processes rather than just input/output relationships in DSS, we have reviewed common process tracing methods including information boards, eye movement tracing, verbal protocols, and computer logging. A promising user interface and information system paradigm, hypermedia, is proposed as a candidate for supporting process tracing in decision support systems. By mapping machine-detectable behavioral events to inferred decision states, activities, and processes, hypermedia provides an effective mechanism to facilitate DSS process tracing.

Hypermedia has many characteristics which make it suitable for process tracing, such as typed links, controllable granularity, and the ability to unobtrusively monitor many significant user actions. Compared to other methods suitable for DSS process tracing, it offers advantages on a number of criteria. Whether used in a native mode or as an envelope user interface to existing

DSS, it is suitable for a number of application areas. The primary shortcoming of the approach, limited informativeness or insight into some decision processes, is shared by other process tracing techniques but may be lessened through multi-method approaches.

Further research is called for in several areas. First, methods for mapping user interface events to decision processes should be developed and validated for various decision and technological contexts. Second, carefully designed empirical studies should be conducted which test the usefulness of the hypermedia process tracing approach--perhaps in both laboratory and field experiments. Finally, the technology should be implemented in operational systems where, over time, it can be further studied and used to improve the performance of human-computer decision making systems.

References

[Axel76] Axelrod, R. M. (ed.). The Structure of Decision. Princeton, NJ: Princeton University Press, 1976.

[Bake86] Bakeman, R. and Gottman, J. M. Observing Interaction: An introduction to Sequential Analysis. Cambridge, Cambridge University Press, 1986.

[Bank84] Banks, J. and Carson, J. S. II. Discrete-Event System Simulation. Englewood Cliffs, NJ, Prentice-Hall, 1984.

[Beck88] Beck, J. and Spicer, D. Hypermedia in academia. Academic Computing, (February 1988), 22-50.

[Bonc81] Bonczek, R. H.; Holsapple, C. W.; and Whinston, A. B. Foundations of Decision Support Systems. New York: Academic Press, 1981.

[Conk87] Conklin, J. Hypertext: An Introduction and Survey. IEEE Computer, (September 1978), 17-41.

[Conk88] Conklin, J. and Begeman, M. gIBIS: A Tool For All Reasons. Microelectronics and Computer Technology Corporation, MCC Technical Report STP-252-88, July 1988.

[Cros90] Crosby, M. E. and Stelovsky, J. How Do We Read Algorithms? A Case Study. IEEE Computer, 23, 1 (January 1990), 24-35.

[Einh79] Einhorn, H., Kleinmuntz, D., and Kleinmuntz, B. Linear Regression and Process-Tracing Models of Judgment. Psychological Review, 86, 5 (1979), 465-485.

[Eric80] Ericsson, K. A. and Simon, H. A. Verbal Reports as Data. Psychological Review, 87, 3 (May 1980), 215-251.

[Fran88] Franklin, C. An annotated hypertext bibliography. Online, 12, 2 (March 1988), 42-46.

[Hala88] Halasz, F. Reflections on NoteCards: Seven issues for the next generation of hypermedia systems. Communications of the ACM, 31, 7 (July 1988), 836-852.

[Hube83] Huber, G. Cognitive style as a basis for MIS and DSS designs: Much ado about nothing? Management Science, 29, 5 (May 1983), 567-579.

[Kove86] Koved, L. and Shneiderman, B. Embedded menus: Selecting items in context. Communications of the ACM, 29, 4 (April 1986), 312-318.

[Krae88] Kraemer, K. and King, J. Computer-based systems for cooperative work and group decision making. ACM Computing Surveys, 20, 2 (June 1988), 115-146.

[Legg89] Leggett, J., Kacmar, C., and Schnase, J. Working Bibliography of Hypertext. Department of Computer Science Technical Report No. TAMU 89-005, Texas A&M University, College Station, TX, March 1989.

[Lian85] Liang, T. Integrating model management with data management in decision support systems. Decision Support Systems, 1, 3 (September 1985), 221-232.

[MacC76] MacCrimmon, K. and Taylor, R. Decision making and problem solving. in M. D. Dunnette (ed.), Handbook of Industrial and Organizational Psychology. Santa Monica, CA: Rand Corporation, 1976, 1397-1453.

[Minc89] Minch, R. P. Application and research areas for hypertext in DSS. Journal of Management Information Systems, 6, 3 (Winter 1989/90), 119-138.

[Minc90] Minch, R. P. Hypermedia Knowledge Management for Intelligent Organizations. Proceedings of the Twenty-Third Annual Hawaii International Conference on Systems Sciences, Kona, Hawaii, January 1990.

[Niel88] Nielson, J. Trip Report: Hypertext '87. ACM SIGCHI Bulletin 19, 4 (April 1988), 27-35.

[Niel90] Nielson, J. Hypertext & Hypermedia. Boston: Academic Press, 1990.

[Newe72] Newell, A. and Simon, H. A. Human Problem Solving. Englewood Cliffs, NJ: Prentice-Hall, 1972.

[Owl87] Owl International. Guide: Hypertext for the PC. 1987.

[Payn76] Payne, J. W. Task Complexity and Contingent Processing in Decision Making: An Information Search and Protocol Analysis. Organizational Behavior and Human Performance, 16 (1976), 366-387.

[Payn78] Payne, J. W., Braunstein, M. L., and Carroll, John S. Exploring Predecisional Behavior: An Alternative Approach to Decision Research. Organizational Behavior and Human Performance 22, (1978), 17-44.

[Phil88] Philippakis, A. S. and Green, G. I. An architecture for organization-wide decision support systems. Proceedings of the Ninth International Conference on Information Systems, Minneapolis, (1988), 257-263.

[Rama87] Ramaprasad, A. Cognitive process as a basis for MIS and DSS design. Management Science, 33, 2 (February 1987), 139-148.

[Russ75] Russo, J. E. and Rosen, L. D. An Eye Fixation Analysis of Multialternative choice. Memory and Cognition, 3 (1975), 267-276.

[Russ87] Russo, J. Eye Fixations Can Save the World: A Critical Evaluation and a Comparison Between Eye Fixations and Other Information Processing Methodologies. Advances in Consumer Research, 5 (1978), 561-570.

[Schn88] Schnase, J.; Leggett, J.; Kacmar, C.; and Boyle, C. A Comparison of Hypertext Systems. Hypertext Research Lab, Technical Report TAMU 88-017, Texas A&M University, September 1988.

[Simo77] Simon, H. The New Science of Management Decision. Englewood Cliffs, NJ: Prentice-Hall, 1977.

[Spra82] Sprague, R. and Carlson, E. Building Effective Decision Support Systems. Englewood Cliffs, NJ: Prentice-Hall, 1982.

[Todd87] Todd P. and Benbasat, I. Process tracing methods in decision support systems: Exploring the black box. MIS Quarterly, (December 1987), 493-512.

[Trig83] Trigg, R. G. A Network-based Approach to Text Handling for the Online Scientific Community. PhD. Thesis, University of Maryland, 1983 (University Microfilms #8429934).

[Whit85] Whiteside, J.; Jones, S.; Levy, P. S.; and Wixon, D. User performance with command, menu, and iconic interfaces. Proceedings of Human Factors in Computing Systems, San Francisco, (April 1985), 185-191.

[Yank87] Yankelovich, N. Hypermedia bibliography, Unpublished report, Brown University, Providence, RI, Institute for Research in Information and Scholarship, 1987.