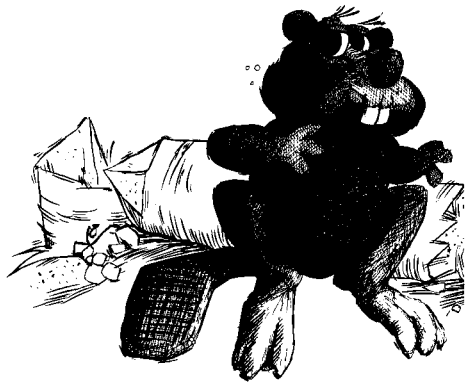
UNIVERSITY OF WATERLOO
COMPUTER SCIENCE DEPARTMENT



Measuring the Effectiveness of Personal Database Structures

Darrell R. Raymond
Alberto J. Cañas
Frank Wm. Tompa
Frank R. Safayeni

CS-86-60

November 1986

Measuring the Effectiveness of Personal Database Structures†

Darrell R. Raymond
Alberto J. Cañas
Frank Wm. Tompa
Frank R. Safayeni

University of Waterloo
Waterloo, Ontario, Canada
N2L 3G1

ABSTRACT

The increasing prominence of online databases and electronic billboards necessitates the design of effective tools for personal data structuring. An experiment was conducted to investigate subjective processes involved during structuring an online database. Subjects organized two hundred proverbs into hierarchical structures over four sessions and used their structures to solve queries. Structuring and retrieval activity in the online environment was markedly different to that obtained by subjects in a previous manual experiment, but in both cases retrieval performance was correlated to the level of distinction employed in the construction of categories.

November 14, 1986

† This research was funded by grant G1154 of the Natural Sciences and Engineering Research Council of Canada.

Measuring the Effectiveness of Personal Database Structures

Darrell R. Raymond
Alberto J. Cañas
Frank Wm. Tompa
Frank R. Safayeni

University of Waterloo
Waterloo, Ontario, Canada
N2L 3G1

1. Introduction.

Computerized public databases need tools for personal information structuring. The paradox of these databases is that the greater their ability to store information and be contacted by large groups of users, the greater their tendency to impede access to information. This hindrance is in part due to the difficulty in completely and consistently indexing large corpora of information, but a more significant factor is that users typically conduct bounded searches. If the majority of the database is irrelevant to a user's needs (as is the case with very large databases), then searches are largely consumed in rejecting unwanted choices. This situation is commonly experienced by subscribers of large electronic billboards and networks; as the billboard increases in size, more of a subscriber's time with the billboard is spent in discarding useless mail or postings than in reading and absorbing relevant items. Since restricting information flow from the producer is often a politically unattractive policy, we must provide more effective means for the information consumer to organize and process this flow.

Though methods for automatic classification of information have some potential, it is still too early to consider these techniques for application on a large scale. Methods for centralized manual classification are common; indeed, the vast majority of research in information organization proceeds on the assumption that professionals can best determine information structure. However, people habitually restructure information simply because it is so often provided in ways they find unsatisfactory for their use. Paper documents are often subject to underlining, photocopying, clipping, "dog-ears", highlighting, and so on — each a means of restructuring the document to provide quicker access to sections of interest. A more modern example of this kind of activity is found in programmable videocassette recorders and the phenomenon of "time-shifting" [1] The ability to restructure the television networks' broadcast schedules has been a significant factor in the popularity of the videocassette recorder.

Structuring tools are found in several advanced systems for computer-based information manipulation.[2,3,4] However, such systems are typically designed for professionals, are oriented towards creation rather than rapid processing of existing information, and are designed to support formalized, impersonal information structures. Personal databases exhibit several interesting characteristics, of which we emphasize two. The first is the *subjective* nature of personal categories and organizations. A subjective organization is an information structure based on personal estimates of the use, value, or meaning of information. Thus a personal library is typically not organized according to Library of Congress cataloging rules, but instead according to criteria such as cost, format, age, or status (e.g., "borrowed"). The second characteristic is the *flexibility* with which personal databases are manipulated. A flexible organization is one which can be changed (or whose interpretation can be changed) with minimal effort. Flexibility is facilitated by a lack of formalism, consistency and completeness, since this permits multiple interpretations of a structure. Thus while a file folder labelled *July 23* has a well-defined (and hence inflexible) content, an unlabelled stack of papers can be thought of as "unimportant work" at one point in time, and "my overdue assignments" somewhat later. Similar issues are discussed in Malone (1983).[5]

Current structuring tools are based on simple adaptations of existing information systems (e.g., relational databases, file systems) or analogues of a familiar physical environment (e.g., the "desktop" metaphor). Raymond (1984) [6] questions whether current tools are satisfactory for dealing with the coming explosion in online information. In order to evaluate the effectiveness of these and future tools for information structuring, we must develop an adequate understanding of structuring behaviour. The aspect of structuring we have investigated is a measure of structure which reflects the subjective and flexible nature of personal databases. This measure is also a useful predictor of retrieval performance on such databases, and hence can be used as a guideline in the design of future structuring tools.

2. A model of structuring.

Knowledge of structuring implies both an appreciation of the activities common in a structuring task, and a model of the internal mechanisms governing these activities. In an online environment such as an electronic news network, users receive (and sometimes submit) large quantitites of information in small packages or units. Information is processed in many sessions over a long period of time, in which the needs and activities of users may change. In each session with the online system, new information is processed and either rejected or integrated with the existing personal database. Though only a small fraction of the total information is entered into the personal database, much of it is candidate material whose suitability is judged by the same mechanisms that would be employed to store it. Such mechanisms should be the focus of a study which attempts to explain the structuring task.

There are few studies which consider the subjective organization of data in an online environment, so observational or anecdotal results can be of significant value. However, in addition to qualitative observations, it is desirable to obtain a quantifiable description of both the subjects' structures and the appropriateness of the structures for a retrieval task. Experimental studies of computer-based tasks have tended to concentrate on measures of low-level or mechanical activity such as keystrokes.[7] Similarly gross aspects of structure such as the number of categories or the average size of a category can be easily measured, but these measures are not directly related to the subjective judgements which are the key issue in development of a structure. Another possibility is to analyze the labels chosen by subjects for their categories. [8] While such a study would be indicative of subjective judgements, it is hard to produce a uniform quantifiable comparison of labels. Furthermore, the assignment of labels to categories is still one step removed from the activity involved in category generation. It is desirable to measure this structuring activity as directly as possible.

Ordinary objects can be distinguished from one another in varying degrees. For example, fermented grape beverages may be classified into a single category ("wines"), or they may be split into a few major sets ("red", "white", "rose"). Further distinction can be obtained by considering the type of grape, year, bouquet, country of origin, vintner, container, cost, and combinations of these or many other factors. Organization of electronic billboards or databases also requires distinctions to be drawn between units of information, and these distinctions are almost always highly subjective. *Good information structuring requires the establishment of an appropriate level of distinction.* We will assume that online databases are organized around information in compact, self-contained units known as *items*; clusters of similar items will be called *categories*. A collection of categories over a given set of items will be called a *structure*.

The choice of a given level of distinction (and hence the structure) depends upon several factors. One factor is limited knowledge; for example, people who are unfamiliar with wine may not know the difference between bordeaux and burgundy, and hence they are incapable of including such a distinction within their structure. A second factor is limited resources; people do not often select the maximal level of distinction of which they are capable because of the cost of doing so. Generally there is an implicit task or purpose perceived for the structure, which affects the selection of a level of distinction. The marginal cost of potential extra distinction is balanced against the marginal value of that distinction, in order to arrive at an acceptable solution.

It is important to note that the level of a distinction employed in producing a structure cannot be inferred from physical properties of that structure. Intuitively, the number of categories and the average category size seem indicative of the level of distinction, since well-defined categories often contain few members, and a high level of distinction tends to produce many categories. However, for a given set of physical properties there are typically many possible structures, not all of which are equally

appropriate. For example, there are more than 1.2 billion ways of choosing 4 categories of 5 items from a set of 20, but these are not all subjectively equivalent. We must measure the subjective level of distinction employed in the creation of the structure more directly.

The measurement of the level of distinction between two items can be made by asking for a spatial approximation. The person responsible for the distinction is requested to place the items close together if they seem similar, and far apart if they seem different. If a scale is provided, the relative distance can be given a numerical value; we call this the *subjective distance* between the two items.

A category inherits a level of distinction based on the accumulation of the pairwise subjective distances between its members. The closer its members seem to be to each other, the more well-defined is the category, and the lower is its *variability within the category*, denoted as V. Determining the subjective distance between several items simultaneously is somewhat difficult. A spatial indication of the level of distinction may be clumsy, impractical, or impossible if it includes many items. However, it is possible to approximate V for a category by measuring the subjective distance between the "most representative" member of the category and "least representative" member.†

Similarly, a set of categories inherits a level of distinction based on the subjective distance between its members (which are categories, rather than objects). The more dissimilar the member categories are relative to each other, the more well-defined is each individual category. We refer to such a set as having high *variability between categories*, denoted D. We can approximate D by measuring the subjective distance between the most representative elements of the categories in the structure.

We can combine V and D to arrive at a measure of the overall level of distinction used to construct the structure, which is called R or *variability ratio*. R is defined as $R=\frac{\bar{V}}{D}$, where $\bar{V}$ is the mean of V for the component categories of the structure. A small variability ratio corresponds to succinct, well-defined categories which are quite distinguishable from one another. A large variability ratio corresponds to loose, ambiguous categories that are less distinguishable from one another. We expect R to be less than one for good structures, since the average variability within categories should be less than the variability between categories. We conjecture that for a given task (i.e., class of data and class of queries) there is a range of R that will result in the best retrieval performance. Structures with a smaller R than optimal will generally have categories that are more discriminating than the queries. Structures with a larger R than optimal consist of categories with many irrelevant or unrelated items. In either case retrieval performance will be reduced.

---

† The determination of which members are most and least representative is to be made by the same experimental subject who provides the subjective distance.

3. The structuring experiment.

We wanted to observe people performing a structuring task that closely simulated the processing of information from an online database. Several factors were important:

- online environment
- a retrieval task
- avoid memory effects
- evolution of structures
- emphasis on subjective characteristics

The first criteria was that subjects should perform their tasks in a working online system. Though paper-based simulations are important indicators and useful for comparison, we considered the use of an online system to be essential in capturing unknown variables and problems in the online structuring task. Furthermore, the system employed should simulate or represent some class of existing systems, partly in order to obtain useful information about these systems, but also because we felt that existing systems were inadequate and wanted to see how subjects would perform in an adverse environment. The next most important criteria was that the subjects should solve a non-trivial retrieval task with their structures. Solution of a retrieval task would provide a means to judge the effectiveness of the structures and a way to interpret measurements of R. Furthermore, we would ensure that the subjects were actively attempting to produce useful structures by providing a concrete purpose or goal. To avoid the possibility that subjects might use memory rather than their structures to solve the retrieval task, the number of stimuli to be structured should be large. Many stimuli would require several experimental sessions per subject, but this would also have the advantage of simulating the repetitive access common in online situations. Multiple sessions would permit us to observe the structures as they evolved. In addition, a large number of stimuli would effectively limit the time subjects would spend structuring, as is the case in realistic situations. This bound on time would emphasize the tradeoffs involved in the choice of a level of distinction.

An important issue was the choice of stimuli to be structured. We wished to avoid structures derived by simple mechanical classifications (i.e., chronological, alphabetic, functional), and wanted to select stimuli that encouraged flexible, subjective distinctions. At the same time, it was necessary to choose concise stimuli so that a large number could be accommodated without overly taxing the subjects. It was also necessary to be reasonably confident that subjects had equal knowledge of the stimuli. We rejected *recipes* [9] and *office documents* because they tend to be organized along simple, previously learned dimensions. Alternatively, pilot studies showed that *famous quotations*, while being short, were so thought-provoking that subjects had difficulty in choosing satisfactory categories. *Newspaper articles*[10] require a significant amount of reading and are susceptible to classification by key words or phrases.

We decided that the subjects should organize *proverbs*. Pilot studies showed that proverbs are easily comprehensible during a session, but are flexible enough to permit various categorizations. For example our subjects interpreted *He laughs best who laughs last* as belonging to categories labelled *silence*, *triumph*, *winning*, and *wisdom*. Subjects were asked to play the role of "proverb manager" for a newspaper. In each of four sessions they would receive proverbs online, add them to an existing organization, and then be required to find solutions for queries such as *Find a proverb which points out that hindsight is always better than foresight.*

4. The online system.

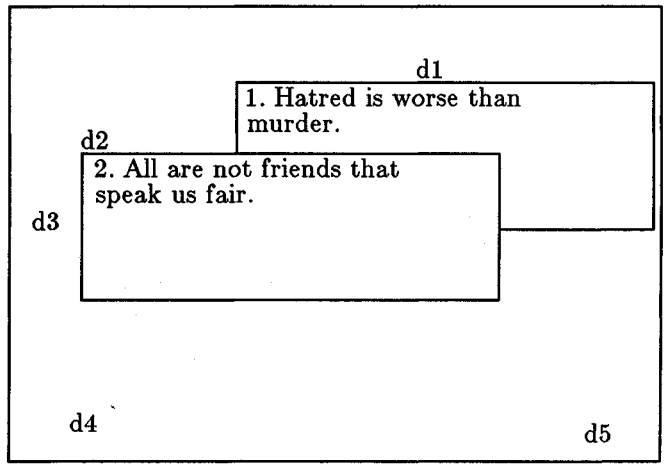
We required an online system with three characteristics: it should be capable of presenting unstructured stimuli; it should support flexible structuring; and it should maintain a detailed record of structuring activity. Several existing systems were rejected because they concentrated on aspects other than structuring or because of the difficulty of adding log features. Instead, we implemented a simple *structure editor* in order to retain close control over the system.

Previous pilot experiments and a full-scale manual experiment conducted by Cañas [11,12] had shown that people rely heavily on spatial strategies to organize proverbs. As subjects processed proverbs, they arranged them on the desks or floor, clustering related proverbs and categories via spatial proximity. Large categories were often overlapped so that important items were more visible than less important ones. Spatial organization is an important component of several systems that also commonly exploit the use of icons to represent items.[13,14,15,16,17] We chose to construct the editor in this popular "desktop" style.† Our "icons" were short strings of text, with proverbs represented by strings of the form **d***i*, where *i* ranged from 1 to 200. Proverbs could be spatially arranged by moving the appropriate icon with a mouse. The subject could examine the proverb by pressing a button on the mouse; this would open a small window and display the proverb's content. Figure 1 shows the initial display employed to familiarize subjects with the editor. Proverbs *d1* and *d2* are visible in windows below their icons.

The only other objects in the space were categories created by the subjects. Each category was represented by a short string of the subject's choice, and was spatially manipulated just as the proverbs were. Subjects could move proverbs (or other categories) into a category by positioning them on top of the destination category's icon. Subjects could view the contents of the category by "entering" it (moving the cursor to the icon and pressing a mouse button); this action would display a new desktop in which proverbs could be organized and more categories could be created. We refer to the initial desktop as the *root* category of the structure. By

† While there is significant intuitive weight to such designs, it should be noted that some experiments[10,18] throw doubt on the efficacy of spatial organizations.

Figure 1. Demonstration Session Display
(containing five proverb icons and two windows)



permitting nesting of desktops, the editor facilitated construction of arbitrary depth and breadth hierarchies which were spatially organized at each node. A maximum of 1700 characters could be displayed at any one time.

Non-root categories (i.e., those created by the subject) contained initially the system-supplied category *back* which enabled the subject to return to the parent category (i.e., towards the root). *back* also served as a "tunnel" through which categories and proverbs could be moved to other parts of the hierarchy. *back* always appeared in the lower lefthand corner of the "desktop", enabling users to move quickly to the root with repeated clicks of the appropriate mouse button.

Proverbs that were moved to a category were not displayed directly on the "desktop", but appended to the category's cyclic list of objects waiting to be organized. The current member of this cyclic list was displayed in the lower righthand corner. Before each session, the proverbs to be organized were appended to the root list by the experimenter; hence subjects would peruse the root list as the set of proverbs received online. In Figure 1, *d5* is the current member of the list. Using function keys, subjects could rotate the list forward or backward, or view the content of the current member without removing it from the list. The root's cyclic list enabled us to present the experimental stimuli with minimal spatial bias.

In addition to these features, subjects could make an unlimited number of copies of each proverb (but not copies of categories) at any time. Copies had the same label as the original. Subjects could also close all open proverb windows (leaving just the icons visible) with a special function key.

One of the key questions in designing the structure editor was when to stop adding features. For example, it seemed reasonable to provide subjects with a "trash can" or other means by which unwanted objects

could be removed. Similarly, the ability to re-label proverbs is a natural one. Since these facilities duplicate existing capabilities†, require more training, complicate experimental measures, and increase the development time of the prototype, we decided to make the editor as simple as possible and note any suggestions for improvement.

Two types of data were automatically collected in addition to recording the subject's structure. First, the editor maintained a detailed log of the subject's activity that enabled us to examine each session in detail. The log consisted of timestamped records of the invocation of every function other than simple cursor motion. Second, special facilities enabled the experimenter to insert data about performance in the subject's session log during retrieval.

The editor was developed on an IBM PC/XT running Waterloo PORT, a multi-process message-passing operating system. The display was produced with an Electrohome QUICKPEL board generating NAPLPS graphics displayed on a 19" Sony KX1901-A monitor. A three-button Hawley mouse was used as a pointing device.

5. The experiment.

Ten undergraduate students at the University of Waterloo were paid for their participation in the experiment. All subjects had English as their mother tongue, and none had expertise in computer or library science. Each subject played the role of "proverb manager" for a newspaper, organizing a set of proverbs over four sessions and then solving queries. Two hundred proverbs were extracted randomly from references[19,20] and split into sets of 50, 75, and 75 for classification in the first three sessions.

Session 1 began with a short training session to familiarize the subject with the features of the editor. The training session included examples of possible structuring behaviour on a small set of proverbs not included in the experimental stimuli, and examples of the retrieval task. Subjects were allowed to practise until they felt confident in using the editor. The remainder of Session 1 was spent organizing the first 50 proverbs. Subjects were allowed to keep notes on paper if they wished during the sessions, and were free to ask questions about the use of the editor at any time.

Session 2 began with a retrieval task performed on the structure created during Session 1. The experimenter asked 10 queries one at a time; for each query the subject located any and all proverbs thought to be useful answers. The retrieval task of Session 2 was followed by classification of 75 new proverbs.

Session 3 was identical to Session 2 except that 15 new queries were solved (on the structure as created in Session 1 and modified in Session 2) and 75 new proverbs were given for further classification. Session 4

† Users could easily create a category called "junk" and move unwanted objects there; moving proverbs to a singleton category has the effect of "re-labelling" the proverb.

consisted of 30 new queries for solution and measurements of subjective distances for randomly selected categories. At the end of Session 4 subjects answered a general questionnaire about the editor. The duration of a session was controlled by the subject, typically requiring two to three hours.

During Session 1, the experimenter suggested to each subject that a category named *junk* be created so that errors could be removed if necessary. The experimenter added categories *1-50* and *1-125* to each subject's structure before Sessions 2 and 3, respectively. These categories contained only cyclic lists with the proverbs encountered up to (but not including) the respective session. The subjects were told that these categories need not be examined, but would enable a quick look at previously categorized proverbs if it was thought that some previous proverbs might belong in newly created categories.

Queries and solutions were developed by a person not otherwise participating in the experiment; example queries and their solutions are shown in Table 1. *Rewording* queries had a single solution whose words were slightly modified to produce the query. *Situation* queries required a single solution and presented a situation for which that proverb seemed most appropriate. *Multiple response* queries were situation queries that permitted a solution set of size larger than one. *Non-existent* queries were based on proverbs not contained in the stimulus set.

The measurement of R was carried out during the last session. Subjects were asked to choose the most and least representative proverb for a set of randomly selected categories. Subjects were provided with copies of the proverbs on 3 by 5 cards; these were to be placed along a 1-meter scale with ten gradations. The experimenter placed one of the proverbs at the extreme left of the scale; the subject placed the other at a point that would indicate the relative similarity of the two proverbs.

During the retrieval part of the sessions, the experimenter logged the time at which the query started, the times at which solutions were located, and the time that subjects indicated that no solution existed or no further solution could be located. Retrieval performance was calculated as the hit rate (percentage of correct answers) multiplied by 100 divided by elapsed time in seconds.

6. Results.

Variability and performance measures. Table 2 gives variability and performance measures for each subject. The variability within categories was less than the variability between categories for all except one subject. This exception had very poor retrieval performance as might be expected when categories are not well-defined.

With the exclusion of subject 4, variability ratio seems to be a good predictor of retrieval performance. This subject reported headaches during the last session, and the experimenter observed that she was not able to concentrate while providing results. Both her retrieval performance and measurements of subjective distance are suspect. Exclusion of this

Table 1. Example Queries (**bold**) and Responses (*italic*)

1. Rewording queries:

Find the proverb that says something like: You shouldn't judge a man until you have tried walking in his shoes.
Don't judge any man until you have walked two moons in his moccasins.
Find a proverb that says something like: If one keeps one's mouth shut, one won't say anything wrong.
Silence never makes mistakes.

2. Situation queries:

A proverb is needed which addresses the importance of desire or want in the accomplishment of goals.
Where there is a will, there is a way.
Your editor is writing an article about overeating and wants a proverb which stresses its serious consequences.
The glutton digs his grave with his teeth.

3. Multiple-response queries:

Find all proverbs about old age.
Even if we study to old age we shall not finish learning.
Age is a bad traveling companion.
Find all proverbs about the importance of sleeping.
Sleep is a priceless treasure; the more one has of it the better it is.
The beginning of health is sleep.

Table 2. Variability and Performance Measures

Subject	Variability			Retrieval		
#	*V*	*D*	*R*	Hit %	Time	Performance
1	6.8	7.02	0.97	0.85	82.9	1.03
2	6.5	7.76	0.84	0.69	71.5	0.97
3	6.6	7.20	0.92	0.65	78.6	0.83
4	4.5	7.07	0.64	0.61	133.0	0.46
5	4.6	6.07	0.76	0.61	74.8	0.82
6	5.8	7.41	0.78	0.71	77.4	0.92
7	4.6	6.38	0.72	0.79	68.6	1.15
8	4.5	7.64	0.59	0.68	64.8	1.05
9	2.6	7.84	0.33	0.77	52.7	1.46
10	7.5	6.67	1.12	0.74	116.1	0.64
Mean	5.4	7.12	0.74	0.70	82.1	0.93

subject results in a strong inverse linear relationship *(r=0.86, F(1,7)=19.23, p<0.01)* shown in Figure 2. Inclusion of this subject would result in a correlation which is not significant *(linear: r=0.55, F(1,8)=3.39, p=0.10).*

Figure 2. Retrieval Performance vs. Variability Ratio

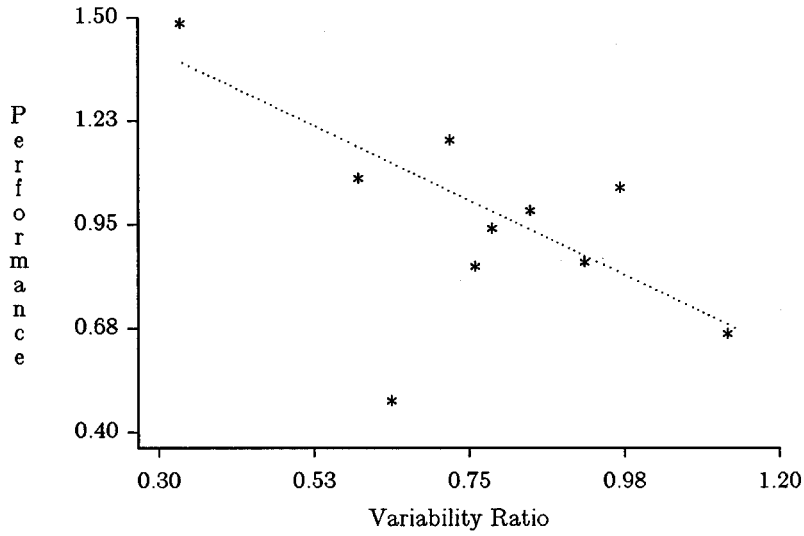


Table 3 gives some simple objective measures of structure i.e., those measures pertaining to the physical size of structures and categories. We did not include the category *junk* or the categories *1-50* and *1-125* in our totals. The most interesting result is the large range of all three measures; the total number of categories ranged from 240 to 20, mean category size ranged from 1.02 to 17.60, and number of root categories ranged from 7 to 54. Our subjects clearly had different ideas about what was an appropriate structure. No subject created a structure more than three levels deep.

Use of space. A mean of 93.8% *(s=7.27)* of subject's categories created in Session 1 still occupied the same "desktop" position in Session 4. At the root level, eight subjects organized their categories in columnar order, starting at the top left corner. One subject organized in row order starting at the top left; one appeared to place categories randomly. A mean of 85.7% *(s=14.45)* of all categories and proverbs were left on the cyclic list of the category to which they belonged. These results indicate that subjects generally did not use spatial clusters or otherwise manipulate space to represent subjectives aspect of the structures.

Comparison with manual systems. Table 4 contrasts the current experiment with the previous manual experiment reported by Cañas. [12] All comparisons in Table 4 are significant at the 0.01 level with the exception of mean category size. Note that retrieval performance was

Table 3. Objective Measures

Subject	Mean Category Size	Number of Categories (*total*)	(*root*)
1	9.23	22	22
2	6.79	29	7
3	7.03	32	32
4	3.71	107	33
5	3.13	62	54
6	7.85	27	19
7	1.02	240	27
8	6.11	35	20
9	15.25	60	30
10	17.60	20	19
Mean	6.77	63.4	26.3

significantly better in the manual experiment, with percentage of correct answers higher and elapsed time smaller. The difference in elapsed time is reflected in the number of nodes visited (a node is a level in the subject's hierarchy). Generally, subjects in the manual experiment went directly to the subcategory containing the desired proverb without looking at intermediate nodes, a procedure not permitted by the editor's design.

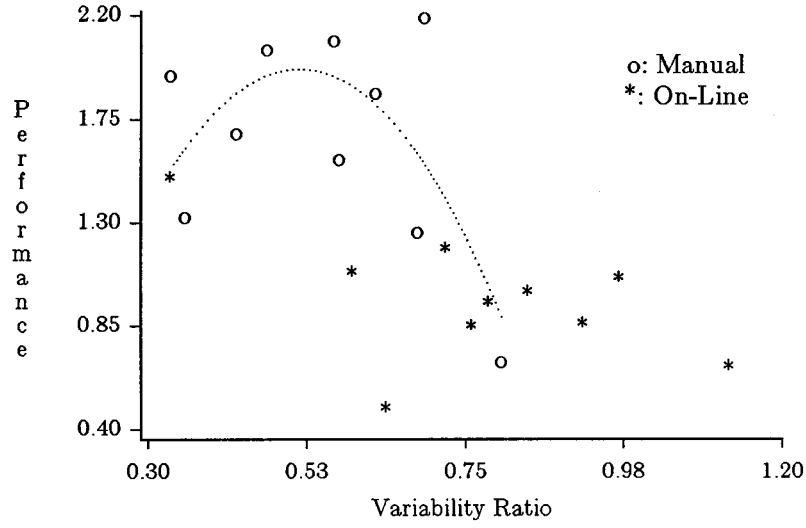
Structures were more well-defined in the manual experiment, as reflected in the smaller mean V and larger mean D. It is interesting to note that the mean category size was somewhat larger in the manual experiment, reinforcing our claim that category size is not directly related to either retrieval performance or variability. More copies were used in the manual system, despite the ease with which copies could be generated in the editor. This suggests that either a need for copies was not perceived, or that subjects found it more difficult to keep track of copies in the editor's structures and hence restricted their use.

Table 4. Comparison of Manual and Editor Experiments

	Experiment (Mean, *s*) *Manual*		*On-line*
% Hits	0.81 (*0.20*)	$>$	0.71 (*0.22*)
Elapsed time	54.16 (*28.48*)	$<$	82.05 (*34.24*)
Nodes visited	1.58 (*0.46*)	$<$	2.56 (*0.90*)
V	4.01 (*2.09*)	$<$	5.40 (*2.54*)
D	7.45 (*2.18*)	$>$	7.11 (*2.08*)
Number of categories	52.80 (*45.89*)	$<$	63.40 (*67.50*)
Category size	6.51 (*6.25*)	$>$	4.08 (*5.47*)
Number of copies	17.31 (*2.71*)	$>$	13.15 (*1.71*)

Figure 3 contrasts retrieval performance and R for both experiments, showing the optimum range of R in the inverse quadratic relationship obtained in the manual experiment.

Figure 3. Retrieval Performance Comparison



It is not possible to treat subjects in both experiments with the same correlation because of differences in experimental procedure. In particular, subjects in the manual experiment were asked to provide subjective distances in each of the four sessions, and were also asked to give short descriptions of each of their categories. The experimenter observed that as subjects performed these tasks, they realized that their structures could be improved and proceeded to make the necessary changes.

Another important difference was that subjects in the manual experiment often ordered the proverbs in their categories from most to least typical. Such an ordering is not possible in the editor-based structures without extensive reorganization on the "desktop". Furthermore, this ordering is not captured by variability measures, since an ordered category has the same value for V as an unordered one. We observed that ordering resulted in better retrieval, as subjects often knew the approximate position of the solution proverb within the category if it was ordered. These differences lead us to believe that better performance of the manually-produced structures is at least partly a result of the subject's greater knowledge about the R of their structures.

Function usage. Table 5 shows the subjects' usage of the editor's functions. These functions can be organized into three groups: *list* functions (forward, back, display current proverb in list), *spatial* functions (position on "desktop", enter a new category, and show a proverb), and *categorization* functions (move object to a category, create a category,

copy a proverb). The table shows the *normalized* mean number of invocations, standard deviation and percentage. The normalized mean is the mean number of invocations per proverb; it gives some indication of the effort expended to organize a single proverb independent of the session. Normalized figures indicate that use of list functions decreased over the sessions, use of the spatial functions remained relatively constant, and use of categorization functions increased.

The total number of function invocations and their distribution becomes more meaningful if one considers a hypothetical "lazy" categorizer, who would expend minimal effort. Such an organizer would merely look at a proverb (forward and display), occasionally make a category (make), and move the proverb to the category (move). The lazy categorizer would invoke a maximum of 4 functions per proverb, of which two would be list functions and two categorization functions.

Subjects averaged more than three times as many function invocations as the lazy categorizer, showing that they invested significant effort in manipulating their structures. However, by Session 3 the distribution of function usage was approaching that of the "lazy categorizer"; most effort was concentrated in list functions and moving objects to categories, with spatial manipulation used very infrequently. This suggests that subjects did not experiment with various types of spatial organization while developing categories.

Table 5. Function Usage

	Session1		Session2		Session3	
Function	norm†	%	norm†	%	norm†	%
list	*11.67(8.11)*	*73.11*	*8.64(3.33)*	*64.94*	*7.82(6.18)*	*59.56*
forward	5.71(5.46)	35.75	4.40(2.75)	33.05	3.36(3.49)	25.63
display	5.17(2.57)	32.36	3.55(1.25)	26.65	3.71(2.03)	28.28
back	0.80(0.73)	5.00	0.70(0.36)	5.24	0.74(0.87)	5.65
spatial	*2.05(2.43)*	*12.87*	*1.83(2.36)*	*13.72*	*2.10(2.95)*	*15.97*
position	0.59(1.25)	3.68	0.29(0.47)	2.18	0.44(1.21)	3.34
enter	1.29(0.99)	8.11	1.40(1.68)	10.54	1.64(1.85)	12.47
show	0.17(0.36)	1.08	0.13(0.29)	1.00	0.02(0.04)	0.15
category	*2.24(0.89)*	*14.02*	*2.84(1.70)*	*21.34*	*3.21(1.69)*	*24.47*
move	1.63(0.58)	10.20	2.15(1.00)	16.13	2.52(1.13)	19.23
create	0.39(0.15)	2.43	0.39(0.65)	2.97	0.31(0.43)	2.40
copy	0.22(0.28)	1.39	0.30(0.40)	2.24	0.37(0.41)	2.84
total	15.60(7.75)	100.00	12.70(3.40)	100.00	12.60(7.43)	100.00

† Invocations per proverb (Mean, *s*)

Questionnaire results. Subjects (none of whom were computer specialists) rated themselves average in computer experience. They found the editor easy to learn and gave it a high overall rating. Display of proverbs seemed the easiest function to use, with list manipulation, copying, and spatial positioning about equal. Category creation was rated the most difficult activity. Subjects claimed they almost never wanted to remove categories. Subjects thought they spent equal amounts of time exploring categories and looking at the list, with half as much time in spatial positioning and fixing mistakes.

7. Discussion.

The complexity and duration of the experimental design meant that we could not test a large number of subjects, as would have been desirable. Our subjects provided interesting and consistent results which clearly indicated a correlation between variability and performance. However, there were not enough subjects to establish the exact nature of this correlation.

The editor contained three different types of structuring tools. These were the "desktop" or spatial dimension provided at each node, the cyclic list at each node, and the hierarchy of nodes. We expected that the "desktop" would be used for experimenting with temporary categories which would eventually become explicit members of the hierarchy, since we had observed this type of behaviour in the manual experiment. In particular, we expected subjects to group related proverbs spatially without explicit categorization until groupings exceeded a threshold size or complexity. At this point the group would coalesce into an explicit, labelled category. The cyclic list was intended merely as a convenient means with which to present stimuli and as a holding place for objects that were being moved around the hierarchy. However, our subjects had other ideas.

The training session included examples of overlapping and clustering strategies, since we were convinced that these were the best structuring possibilities within the limitations of our simple editor. Despite this bias, subjects made very little use of either clustering or overlapping strategies; instead they exploited the spatial dimension in a more subtle fashion.

Subject 7 showed evidence of subjective clustering at the internal nodes of his structure. His was the largest structure, with the root organized in alphabetic columnar order as shown in Figure 4. This subject's performance in the first retrieval session was quite dismal; recognizing this, he spent a great deal of time re-organizing his structure. After re-organizing, his retrieval improved dramatically and was second-best overall. Figure 5 shows the subcategory *judgement*; note that *notbylooks* and *appearance* are close together and separate from *inhisshoes* and *experience*. Subject 7's retrieval performance benefited from restructuring and the use of space to reflect subjective relationships, indicating that these techniques have some importance in development of a good structure.

Figure 4. Root Categories for Subject 7

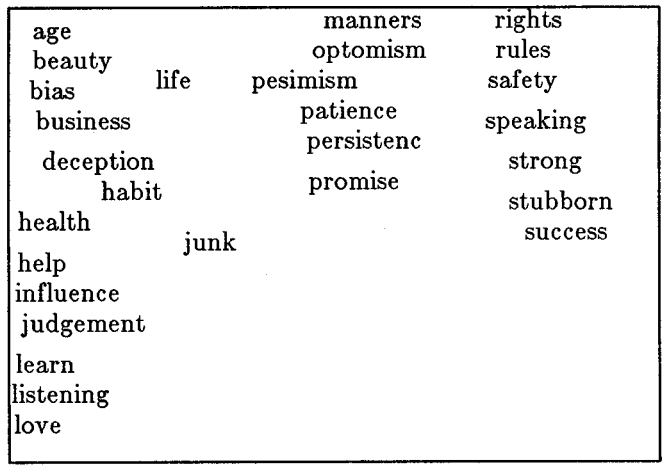
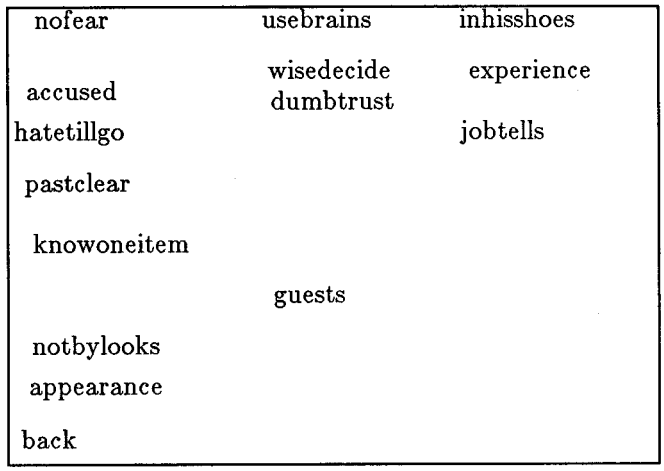


Figure 5. Subcategories of *judgement* for Subject 7



More subtle evidence of the spatial dimension can be found in the structures of subject 8 and subject 3. Subject 8's root level contained two categories labelled *home*, while Subject 3's root level contained two categories labelled *senses*. These identically labelled categories had no proverb in common, and subject 8 in particular was unaware of the "collision" until the experimenter pointed it out. The spatial position of the identically labelled categories must have been an important index into a non-trivial memory pattern of the structure.

People unfamiliar with the structures of subject 8 or subject 3 would not be able to distinguish between the identically labelled categories without investigating them extensively. Such an investigation would serve to create a connection between memory and spatial index similar to that originally established by the subjects themselves. Our model does not

consider the importance of memory or the role of the structure as a cue to memory. We conjecture that this is a significant use of space in both the manual and on-line environments.

Subjects typically categorized by following an interesting procedure that we call *hierarchical extraction*. This method consists of refining a category by choosing some closely-knit subset of its members as a subcategory. This process is carried out iteratively on the initial category as long as is deemed necessary, and then recursively on the new subcategories.

The root's cyclic list was employed as an initial "temporary" category for the hierarchical extraction; subjects would examine this list without moving its contents to the desktop. After some small number of passes through the list, the subject would create one or more categories and move proverbs directly from the list to the category: in effect, directly from one cyclic list to another. Subjects continued to reduce the root list until it contained only miscellaneous, hard-to-categorize proverbs.

The heavy use made of the cyclic lists is evidenced by the fact that 85% of the proverbs and categories remained in some list and were not moved to the "desktop". We did not expect that such a large fraction of objects would be considered "miscellaneous" at some level of distinction, or that the lists would so facilitate hierarchical extraction that they would replace the use of temporary categories in the form of spatially clustered proverbs. We conjecture that the driving motive behind hierarchical extraction is to avoid structuring ambiguous objects.

We were curious to know if objective measures such as mean category size and "depth/breadth" parameters could be of some use in evaluating the structures created by our subjects. Our study differs from previous work in that we encouraged the use of copies, and did not provide the subjects with pre-existing structures, but we did attempt to find correlations between performance and mean category size, total number of categories, and number of root categories. No significant correlation existed.

Subjects 2 and 5 provide an illuminating example of the extreme range in objective measures; the roots of their structures are shown in Figures 6 and 7, respectively. Each subject's root "desktop" was essentially a menu to a hierarchical data base. The root "menus" for subjects 2 and 5 are in fact the most extreme ones constructed, in the sense that all other users had "menus" that contained more items than subject 2 but fewer than subject 5. The great difference in the appearance of their root menus might lead us to predict that performance would also be quite different, yet these subjects had essentially equal performance. Their structures were also quite close in R value.

While these results do not invalidate work on "depth/breadth tradeoff" or studies of other objective measures, they do indicate a limited range of applicability for such results. Objective models evaluate the *mechanical* effort involved in using a structure; it appears that this effort is in some cases less important than the *mental* effort required.

Figure 6. Root Categories for Subject 2

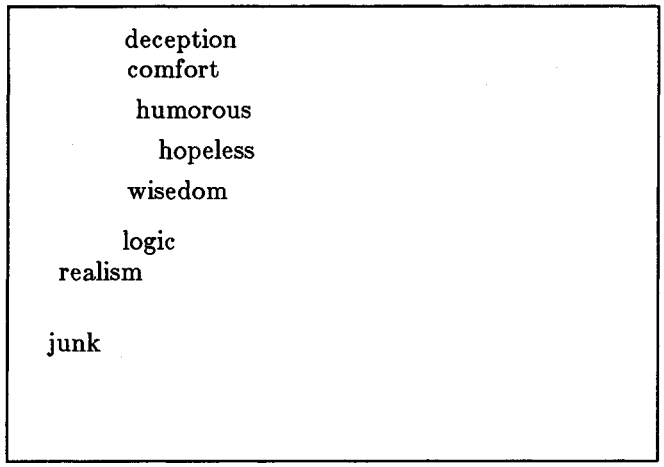
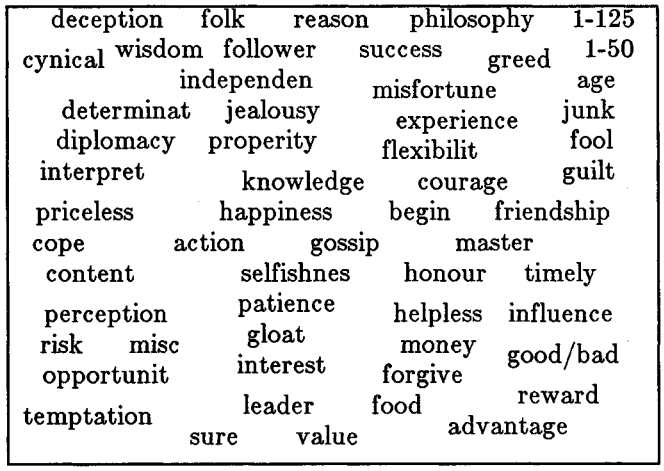


Figure 7. Root Categories for Subject 5



What kind of fundamental limitations are faced when using a system that employs a "desktop" metaphor? Contrasting the results of this experiment with the results of the manual experiment indicates that online structures tend to a larger variability ratio. We must therefore explain how the editor interfered with our subjects' ability to make adequate variability judgements.

Subjects had limited ability to see and evaluate their environment compared to the manual experiment. The editor permitted subjects to display at most two or three proverbs simultaneously without overlapping windows. If the "desktop" contained several objects, subjects would avoid covering them with windows, further reducing the amount of space available for structuring. Subjects could see only the immediate descendants of a node unless they navigated through the structure, a time-consuming

process.

The subjects' ability to manipulate the environment was also greatly limited compared to the manual experiment. Since subjects could only manipulate what was on the screen, reduction in vision constitutes a reduction in manipulation capability as well. Furthermore, subjects were effectively limited to manipulation of single items. In the manual experiment, a simple sweep of the hand would suffice to move a spatially contiguous temporary category to a new location. A similar task in the editor would require a tedious process of moving objects one by one to the new location. As one pilot subject observed, "moving objects on the screen is similar to using a magnet to move objects kept under glass".

Since subjects could only evaluate a small part of their structure, the subjective quality of their structures would tend to a local rather than global optimum. Since subjects could manipulate their structures only with difficulty, the cost of temporary categories exceeded their perceived marginal value, and hence they were not often employed.

Our implementation does not employ the most advanced hardware. While we expect that a higher resolution display and a faster processor would make the interface more pleasant, we do not think such modifications would result in a fundamental difference unless several orders of magnitude of improvement were possible. A real desktop provides a space continuum that is qualitatively distinct from a *discrete display* device employing several virtual screens for presentation of one or more dimensions. The subject's perception of the continuum undergoes continual visual *refresh* as the subject scans the structure. By contrast, a discrete display device requires explicit, conscious action for refresh. The desktop permits arbitrarily fine adjustments to be made to the spatial contiguity of various parts of the structure so that it matches the subjective contiguity; however, information that is on different screens in a discrete display seems "separate" no matter how closely the screens may be linked in the overall hierarchy.

Perhaps more importantly, the manual environment also includes highly-developed manipulative tools (i.e., hands) with powerful group-oriented functions. Using one's hand to push some proverbs to the side of the table is a simple manual activity, but it has complex structuring implications. Its most visible purpose is to render the moved set less important by moving it out of the foveal area, but a more subtle factor is the increased clustering of the members of the set. This clustering reinforces both the increased variability between categories (the set is spatially more distinct from its neighbours) and the decreased variability of the category (the members are seen as more alike in their unimportance). At the same time, the clustering preserves much of the relative spatial organization within the category and thus it can be "reconstituted" at a later date if the decision to move it was too hasty. Finally, the clustering increases the amount of overlap in the set and hence reduces the amount of information that must be evaluated when considering further structuring moves.

8. Conclusions and future directions.

Our subjects learned the editor very quickly and gave it a high rating for "user-friendliness", confirming the general notion that "desktop" interfaces are pleasant, fun to use, and quickly learned. However, we have identified a significant, quantifiable distinction between such interfaces and the real desktops they attempt to emulate, namely the added interference in making and preserving variability judgements. This result has important implications for design: improvements in the interface will not result in improvements in the task performance unless they address directly the assessment of the subjective quality of the personal database.

We suggest two general approaches to more effective structuring interfaces, which we refer to as the *manipulation* and *evaluation* approaches.

The manipulation approach concentrates on augmenting simple structuring tools with powerful group-oriented ones that encourage the use of temporary categories and permit a wide base of comparison. The user can be provided with tools to select groups of elements (e.g. with a "lasso"), which can then be spatially clustered automatically. Halasz and Moran (1982)[21] recommend care when using analogy; it may be possible to avoid analogy entirely. For example, Raymond (1984)[6] suggests that simple menus be replaced by *multi-menus*, which permit multiple (rather than single) choices at each node and show more than one level of descendants. Multi-menus reduce the the number of discrete steps needed to explore the environment by permitting larger steps and by displaying a structured view of the environment. Proper implementation of multi-menus is highly dependent on advanced picking tools such as the mouse and intelligent use of limited display space, but completely independent of analogies to a manual environment.

The evaluation approach concentrates on automating the measurement of R and presenting it to the user so that the structure can be appropriately adjusted. This approach is based on our observation that improved performance is partly a result of extra feedback about the variability of the structure. Evaluation could be carried out during the structuring process by automatic selection of appropriate elements of the structure for comparison to the element to be structured. Conversely, evaluation could be conducted by an off-line tool that might resemble an English style-checker — a *structure checker*. The user would indicate which parts of a structure were doubtful, and perhaps give some indication of the precision with which checking should be performed. The structure checker would then obtain subjective distance measurements from the user in order to compute R for the structure. The structure checker would indicate a relative measure of subjective goodness, and might also suggest where improvements are most necessary or could be most beneficial.

Finally, what of variability and the model of categorization? Our measures were incomplete pictures of the structures, since they did not include the effects of ordering within categories, nor were they expressive

of the nature of the hierarchies constructed. These and various other inadequacies could be rectified in a future experiment in order to obtain a more precise correlation of R with retrieval performance. Even at the current level of investigation, however, we observe the importance of subjective quality of information structure, and this observation is invaluable in setting the direction for design of more advanced computer structuring tools.

9. Acknowledgements.

We would like to thank Bert Bonkowski of the Software Portability Group at the University of Waterloo for his unfailing courtesy and support during development of the structure editor. Our thanks also to Mert Cramer for his suggestions on an early draft of this paper, and to Dave Conrath for his involvement with the manual experiment.

10. References

1. CIT, *Videodisc Opportunites and Options,* Communications and Information Technology Research Ltd., London, England (June 1984).
2. Engelbart, Douglas C., Richard W. Watson, and James C. Norton, The Augmented Knowledge Workshop, *Proceedings of the 1973 AFIPS National Computer Conference,* pp. 9-20 (1973).
3. Feiner, Steven, Sandor Nagy, and Andries Van Dam, An Experimental System for Creating and Presenting Interactive Graphical Documents, *ACM Transactions on Graphics* **1**(1) pp. 59-77 ACM, (January 1982).
4. Robertson, G., D. McCracken, and A. Newell, The ZOG Approach to Man-Machine Communication, CMU-CS-79-148, Department of Computer Science, Carnegie-Mellon University, Pittsburgh, Pennsylvania (October 23, 1979).
5. Malone, Thomas W., How Do People Organize Their Desks? Implications for the Design of Office Information Systems, *ACM Transactions on Office Information Systems* **1**(1) pp. 99-112 (January 1983).
6. Raymond, Darrell R., Personal Data Structuring in Videotex, CS-84-7, Department of Computer Science, University of Waterloo, Waterloo, Ontario (February 1984).
7. Card, Stuart K., Thomas P. Moran, and Allen Newell, The Keystroke-Level Model for User Performance Time with Interactive Systems, *Communications of the ACM* **23**(7) pp. 396-410 (July 1980).
8. Jones, W.P. and T.K. Landauer, Context and Self-Selection Effects in Name Learning, *Behaviour and Information Technology* **4**(1) pp. 3-17 (1985).
9. Furnas, G.W., T.K. Landauer, L.M. Gomez, and S.T. Dumais, Statistical Semantics: Analysis of the Potential Performance of Keyword Information Systems, *The Bell System Technical Journal* **62**(6) pp. 1753-1806 (July-August 1983).
10. Dumais, Susan T., A Comparison of Symbolic and Spatial Filing, *CHI '85 Conference Proceedings,* pp. 127-130 (April 14-18, 1985).
11. Cañas, Alberto J., Frank R. Safayeni, and David W. Conrath, A Conceptual Model and Experiments on How People Classify and Retrieve Documents, *8th International ACM SIGIR,* (June 1985).
12. Cañas, Alberto J., Variability as a Measure of Semantic Structure for Document Storage and Retrieval, *Ph.D. Thesis,* Department of Management Science, University of Waterloo, (August 1985).
13. Negroponte, Nicholas, Books Without Pages, *IEEE International Conference on Communications,* pp. 56.1.1-56.1.8 IEEE, (June 10-14, 1979).

14. Negroponte, Nicholas, Media Room, *Proceedings of the Society for Information Display* **22**(2) pp. 109-113 (1981).

15. Herot, Christopher F., Spatial Management of Data, *ACM Transactions on Database Systems* **5**(4) pp. 493-514 (December 1980).

16. Smith, David Canfield, Charles Irby, Ralph Kimball, and Eric Harslem, The Star User Interface: An Overview, *Proceedings of the AFIPS National Computer Conference,* pp. 515-528 (June 7-10, 1982).

17. Shneiderman, Ben, Direct Manipulation: A Step Beyond Programming Languages, *Computer,* pp. 57-68 IEEE Computer Society, (August 1983).

18. Jones, W.P. and S. Dumais, The Spatial Metaphor for User Interfaces: Experimental Tests of Reference by Location versus Name, *ACM Transactions on Office Information Systems* **4**(1) pp. 42-63 (January 1986).

19. Fergusson, R., *The Penguin Dictionary of Proverbs,* Penguin Books Inc., Markham, Ontario (1983).

20. Tripp, R.T., *The International Thesaurus of Quotations,* Thomas Y. Crowell Co., New York, N.Y. (1970).

21. Halasz, Frank and Thomas P. Moran, Analogy Considered Harmful, *Proceedings of the CHI '82 Conference on Human Factors in Computing Systems,* pp. 383-386 (1982).

11. Appendices.

There are three appendices to the report. The first appendix contains the experimental instructions as they were read to the subject in each of the four sessions. The second appendix contains figures for each subject showing the root of their structure after each of the first three sessions (only retrieval was performed in the fourth session, so the structure did not change). The third appendix contains composite figures showing the usage of space at the root by all subjects in each of the first three sessions. These composite figures show clearly that subjects generally arranged their categories to the left and upper sides of the space, avoiding the area where proverbs were displayed.

Appendix 1. Experiment Instructions.

Session 1

Introduction

Objective of the experiment

The purpose of this experiment is to find out how people organize and retrieve information.

General instructions

For this experiment, assume that you have a job at a newspaper. Your boss, the editor of the newspaper, has decided that you are to be in charge of handling a filing system of proverbs. The idea is that every time he needs a proverb he will come to you and ask for it. Of course, you have to find it fast and give it to him. It might be that he wants a nice, cute proverb to put in the front page as *The proverb of the day*, or he may seek a proverb that is appropriate for an article he is writing. To keep track of the proverbs you will be using a computer program. Using the program, you can arrange the proverbs in whatever way you want, as long as you can meet his needs. Sometimes he will know exactly which proverb he wants, but other times he will ask you if you have a proverb that will fit some idea he has. The following are sample requests he might have:

> Find me the proverb that says "the person who laughs last laughs best" – or something like that.
>
> I'm writing an article about the Spanish people as lovers – do you have a Spanish proverb about love?

During the course of the experiment you will be given approximately two hundred proverbs. You won't have to classify them all in one single session; they will be given to you throughout the sessions. You will be asked questions like the ones above for which you will seek the appropriate proverb.

Practice

First of all I will teach you how to use this program. Let's commence with a small group of proverbs. Feel free to ask any questions you may have.

Use of the Program

This program allows you to handle the different proverbs on this large screen and classify them. Most of the actions you can perform use this little device here called a *mouse*. As you move the mouse around, you can point to different parts of the screen. The place being pointed on the

screen is indicated by this **cursor** (demonstrate). Here, try moving the mouse around.

Displaying a proverb

These *d1*, *d2*, *d3*, etc., each represent a proverb. To display the contents of the proverb, first move the mouse so that the cursor is over the proverb. For example, move the mouse so it is over *d1*. Now press the button on the right of the mouse. (This button is called the display button.) Notice that the content of the proverb is displayed. Each proverb consists of a unique number, possibly an origin which is usually a country, and the proverb itself. Now display the contents of *d2*, *d3*, and *d4*. Notice that the displays of some of the proverbs will overlap. To clear the contents of the proverbs from the screen press the key **CL**. You can go back and display the content of any proverb at any time by moving the cursor to it and pressing the displaying, right button. Go back and display *d2*. Notice that it overlaps with *d3* and *d4*.

Selecting a proverb

There is always one proverb that is *selected*, and is shown in red (show the selected proverb). To select another proverb, move the cursor to it and press the left button on the mouse. Select *d2*. Notice how now *d2* is shown in red. (The left button of the mouse is called the *selection* button). Now go and select *d3*.

Moving a proverb

To move a proverb to another position on the screen you must do three things. First you must select the proverb (by moving the cursor to the proverb and pressing the selection, left button of the mouse – you will know it is selected when it turns red). Second, you move the cursor to the new position where you want to place the proverb and third, press the middle button of the mouse (usually called the positioning button). Try moving *d1*. Notice that the proverb turns yellow for an instant before moving. Try moving some proverbs around the screen.

Summary of the use of the mouse

As a summary, the mouse is used for moving around on the screen. Its selection, left button is used to select a proverb; its middle, positioning button to move a proverb, and its right, display button to display a proverb.

Initial set of proverbs

The proverbs that your editor will give you to classify will be located in a special position on the lower right hand corner of the screen. This is really a *Stack* of proverbs, of which you can only see the top, in this case *d5*. You can display the proverb on top of the Stack pressing the **DS** key. The **DS** always displays the proverb on top of the Stack, independent of where your cursor is. Try displaying the proverb on top by using the **DS** key. To move to the next proverb on the Stack you use the + key. Try it. The + key always moves you forward on the Stack. If you press the + key when at the end of the Stack, you will go to the first element on

the Stack. Try moving forward through the Stack. To move backwards on the Stack use the – key. This way, by combining the **DS**, +, and – key you browse through the Stack. This way you can skim rapidly through the set of proverbs and get and idea of what its contents are. Try displaying a few of the proverbs on the Stack. You can clear the screen at any moment by using the **CL** key as before.

Moving proverbs out of the Stack

Moving a proverb out of the Stack is done in the same way as you moved proverbs around the screen: select the proverb on top of the Stack (use the left, selection button), move the cursor to the desired position on the screen and press the middle, positioning button. Try it. Notice that the next proverb on the Stack is now, automatically, the selected proverb. You can now position the cursor in another location on the screen and move this proverb out of the Stack too. Try it also. This way you can move proverbs out of the Stack without having to go and select each one of them – just move the the cursor to the desired location and press the middle, positioning button each time.

Making a Category

If you want to give a name to a proverb, or you want to group a number of proverbs together and give them a label or name, you can create a category. A category is created by positioning the cursor in the desired location and pressing the **F9** function key. A small box will be displayed on the screen on which you can type the name you want to give to the proverb or category. The number of letters or numbers you can use in the name is limited. If you make a mistake while typing the name, you can erase what you have typed by pressing the ← (*left arrow*) key. When you are finished typing the name of the category, press the positioning, middle button of the mouse. Try making a category.

Moving proverbs to a category

To move a proverb into a category, the procedure is the same as for moving it around the screen: select the proverb (using the left, selection button), move the cursor to the desired category and position the proverb there (by using the middle, positioning button). Try it. Notice that the proverb turned yellow for a moment and then "disappeared." What happened is that the proverb is now *within* the category. You can also move proverbs directly from the Stack into categories. Try it. **Make sure always that the correct proverb is selected before moving it into a category.** Try making a couple other categories and moving proverbs into them.

Going into a category

To find out what the contents of a category are, *display* the category: position the cursor over the category and press the display (right) button. Try it. Notice that the screen is blank except for a new Stack on the lower right hand corner and a "category" called *back* on the lower left hand corner. The Stack contains all the proverbs that you moved into the category, waiting to be positioned somewhere on the screen. The *back*

category is the way to move up to the main screen from where we just came. To move up, position the cursor on the *back* category and *display* it (press the display button on the right of the mouse). Try it. You are now back in the "original" category. Now lets go into the category again.

Actions within the category

To move the proverbs out of the Stack of this new category, you follow exactly the same procedures as before. In fact, anything that you were able to do in the main category can be done here: displaying proverbs, moving proverbs around, even creating new categories within this category. They are all done in the same way. Try moving a few proverbs out of the Stack onto the screen. Now create a new category and move a couple of proverbs into it. Go into the category, display those two proverbs and come back. Notice that this way you can create a hierarchy or tree of categories. You can also move the category's location on the screen in the same way you moved the proverb: select the category, position the cursor and press the positioning, middle button. Whenever you move out of a category and come back, the subcategories and proverbs are displayed in exactly the same position you left them. The only difference will be that the screen will be clean from displayed proverbs (as if the **CL** key had been pressed). Try it.

Moving proverbs up a category

Just as you moved a proverb down into a category you can move a proverb up the hierarchy. All you have to do is select the proverb and position it in the *back* category. It will appear on the Stack of the category one level up the hierarchy. Try it.

Moving a category into a category

You can take a category with all its proverbs and subcategories and move it into a subcategory or up the hierarchy (through *back*). Moving one category into another is exactly the same as moving a proverb into a category: select the category, position the cursor on the destination category, and press the positioning (middle) button – the category will appear in the Stack of the destination category and can be moved into the screen (or moved into another subcategory!) in the same way as proverbs are moved.

Copies of proverbs

In some cases you may want to file a proverb under two different categories. For this you can make a *copy* of a proverb. Making a copy of a proverb is almost the same as moving the proverb around, except you use the **F10** function key instead of the positioning button. In other words, you first select the proverb, you position the cursor in the location where you want the copy, and you press the **F10** function key. The copy of the proverb will appear in the desired location. Try making a few copies of proverbs. You can make a copy of a proverb that is in the Stack in the same way as for other proverbs.

What if nothing happens when you press the buttons?

If you press the positioning button for moving a proverb, the function keys for making a copy of a proverb or creating a category, and nothing happens, it means that there is not enough space on the screen where you have the cursor to place the name or proverb, either because it is too close to another proverb or category or to the border of the screen. Move the cursor a bit to where there is more space and try again.

Things you cannot do

There are a few thing you cannot do in this system. One of them is to get rid of a proverb (more probably, of a copy of a proverb) or of a category which you don't need any more. A suggestion is to create a category called *junk* and move everything you don't need in there. Another is to make a copy of a category with all its contents. You can, however, have two categories with the same name.

Summary

As a summary, with this system you can move proverbs around by the use of the mouse. Initially, proverbs will be located on the Stack at the right hand corner of the screen. You can browse through them move them around and/or make copies of them. You can create categories, move proverbs into them and move the categories around just as you moved the proverbs.

The mouse is used to move the cursor around. Its left, selection button is used to select a proverb or category, its middle, positioning button for positioning the selected proverb or category, and the display button on the right for displaying the contents of a proverb or category.

The different function keys are:

F9: make a new category;

F10: copy a proverb;

CL: clear the screen;

DS: display the proverb on top of the stack.

+: move forward to the next item on the stack.

–: move backward to the previous item on the stack.

Classification

Now that you know how to use the system, lets try using it with a small group of proverbs as practice. On the stack here is a set of 20 proverbs. They are called *s1*, *s2*, ..., instead of *d1*, *d2*, ..., but everything is essentially the same as before. Now, let's suppose these are the first set of

proverbs your boss gives you. You know there will be lots more coming so you want to organize them so that you can find the proverbs he wants easily. You probably will want to group together those that sort of seem to go together. If you feel that you want to place a proverb in two or more groups, make copies of it. Feel free to make as many categories as you want. Remember the main thing is to be able to give the proverbs to your boss as quickly as possible when he asks for them. Now, please try going through the proverbs and organizing them.

You will be doing a lot of classification of proverbs. The way you organize them is by no means fixed, you can change it around at any time.

Retrieval

The main purpose of classifying the proverbs is to be able to provide any proverb that your boss requests. Before we try a few sample requests let's look at a few details of the procedure we will follow.

I will give you each request on an index card, one at a time. As soon as you receive the request you may start looking for the appropriate proverbs. Once you have found a proverb that you think satisfies the request, **display it**, and say so. The proverb that was last displayed will be considered to be your response. If there isn't a proverb that satisfies the request, or if you know there is a proverb or group of proverbs that satisfy the request but you can't find it, say so outloud. Indicate also when you have finished your search for a request.

The responses will be recorded by means of the Function Keys. However, I will take care of pressing the appropriate keys when you say outloud the response.

Here are five examples of requests. (Execute the retrieval, one by one).

The requests you will get during the experiment are similar to those you just fulfilled. There will be requests for which it's clear which proverb is required, others for which it is not so clear because your boss doesn't really know which proverb he wants, and occasionally he will ask you for proverbs you don't have.

What you will be doing in the experiment is the same thing as what you have just done, only with a larger number of proverbs. It is therefore very important that you understand what the procedures are. Are there any questions? Is there any part that you don't understand and would like to go over again? If not, we will start with the experiment itself.

At this point, we make sure that the subject has used each of the following actions during his classification. If not, we ask him/her to perform it and confirm he/she knows how to do it.

1. *Browse through the stack.*

2. *Clear the screen.*

3. *Move proverbs on the screen.*

4. *Create a category.*

5. *Move proverbs into the category*

6. *Move into and from a category.*

7. *Move proverbs up the hierarchy.*

8. *Make copies of proverbs.*

9. *Move a category up and down the hierarchy.*

Experiment

Remember, you are working in a newspaper managing a collection of proverbs. It is important to be able to give your boss the proverbs he needs when he needs them.

Classification

The first set of 50 proverbs is now on the Stack. They are called *p1, p2, p3, ..., p50.* You will be given more proverbs in the following sessions but for this one you will classify just 50. Scan them before beginning so you get an idea of the topics included. If you find that some proverbs are hard to classify, leave them on the side and come back to them later. You can reorganize or make any changes you wish to the categories you create on the way. Now go ahead and start classifying them.

End

This finishes the first session of the experiment. We will continue in the next session.

Session 2
Introduction

General instructions

Remember that in this experiment you are working for a newspaper managing a collection of proverbs. Every time your boss, the editor of the newspaper, needs a proverb to use somewhere in the newspaper he will come to you and ask for it. Of course you have to find it fast and give it to him.

During the last session you classified 50 proverbs. Notice that the categories and proverbs are on the screen in the same way you left them.

Experiment

Retrieval

I will now give you 10 requests from your boss, one at a time. You can look around through the categories in search of the proverbs. Once you have found the proverb that you think satisfies the request, make sure you display it and say so outloud. The proverb that was last displayed will be considered to be your response. In the same way, if there is no proverb, or if you think there is a proverb that satisfies the request but you can't find it, say so outloud. I will record your responses by means of the function keys. (Execute the retrieval, giving the queries one by one.)

Classification

On the Stack now is the second set of proverbs in the collection: there are 75 of them. You must add these proverbs to your original set. Scan them before beginning so you get an idea of the topics included and how they relate to the previous 50. If you find that some proverbs are hard to classify, leave them on the side and come back to them later. Notice that in the upper right hand side of the screen there is a category called *1-50*. In the Stack of this category are the 50 proverbs from the previous session in case you need them.

You can reorganize, create new categories, modify existing categories, in general make any changes you wish to the categories. Now go ahead and start classifying.

End

This finishes the second session. We will continue in the next session.

Session 3
Introduction

General instructions

Remember that in this experiment you are working for a newspaper managing a collection of proverbs. Every time your boss, the editor of the newspaper, needs a proverb to use somewhere in the newspaper he will come to you and ask for it. Of course you have to find it fast and give it to him.

During the last session you classified 75 proverbs (making the size of the collection 125). Notice that the categories and proverbs are on the screen in the same way you left them.

Experiment

Retrieval

I will now give you 15 requests from your boss, one at a time. You can look around through the categories in search of the proverbs. Once you have found the proverb that you think satisfies the request, make sure it is displayed and say so outloud. The proverb that was last displayed will be considered to be your response. In the same way, if there is no proverb, or if you think there is a proverb that satisfies the request but you can't find it, say so outloud. I will record your responses by means of the function keys. (Execute the retrieval, giving the queries one by one.)

Classification

On the Stack now is the third set of proverbs in the collection: there are 75 of them. You must add these proverbs to your original set. Scan them before beginning so you get an idea of the topics included and how they relate to the previous 125. If you find that some proverbs are hard to classify, leave them on the side and come back to them later. Notice that in the upper right hand side of the screen there are two categories called *1-50* and *51-125*. In the corresponding Stacks of these categories are the 50 proverbs from the first session and the 75 proverbs from the second session in case you need them.

You can reorganize, create new categories, modify existing categories, in general make any changes you wish to the categories. Now go ahead and start classifying.

End

This finishes the third session. We will continue in the next session.

Session 4
Introduction

General instructions

Remember that in this experiment you are working for a newspaper managing a collection of proverbs. Every time your boss, the editor of the newspaper, needs a proverb to use somewhere in the newspaper he will come to you and ask for it. Of course you have to find it fast and give it to him.

During the last session you classified 75 proverbs making the size of the collection 200. The categories and proverbs you organized are on the screen in the same way you left them.

Experiment

Retrieval

I will now give you 30 requests from your boss, one at a time. You can look around through the categories in search of the proverbs. Once you have found the proverb that you think satisfies the request, make sure it is displayed and say so outloud. The proverb that was last displayed will be considered to be your response. In the same way, if there is no proverb, or if you think there is a proverb that satisfies the request but can't find it, say so outloud. I will record your responses by means of the function keys. (Execute the retrieval, giving the queries one by one.)

Semantic Distances – Practice

Introduction

During the rest of this session, we will take some *measures* of the filing system you made. First I will show you the type of measures and how to take them.

Most typical proverb

Take this category from your filing system. Assume you have to show somebody what type of proverbs go in it, but can only do so by showing him one of the proverbs in it. In other words, you have to pick the proverb from the category that best shows what type of proverbs are included in the category. Which one would you choose to show him so that he will have the best understanding of what is in the category? Just to give it a name by which to refer to it, let's call the proverb that you chose the *most typical* proverb of the category.

Now try to choose, in the same way, the most typical proverb of these other two categories.

Least typical proverb

Just as you chose the most typical proverb of the category, there are usually proverbs that don't fit very well within the category. Or at least they don't fit as well as do some of the other proverbs. You may think of them as the opposites of the most typical proverb. Let's take this first category again. Suppose that you could remove one proverb that fits least well within the category, which one would you remove? Now let's repeat this exercise for the other two categories. We call these proverbs the *least typical* proverb of the category.

Similarity distances

Some proverbs are quite similar to each other. Take the following two for example, their content is very similar.

A. Proverb.
Experience is the mother of wisdom.

B. Proverb.
Trouble brings experience and experience brings wisdom.

While others may be quite different, for example these two:

C. Proverb.
A maid marries to please her parents, a widow to please herself.

D. Proverb.
You can lead a horse to the water but you can't make him drink.

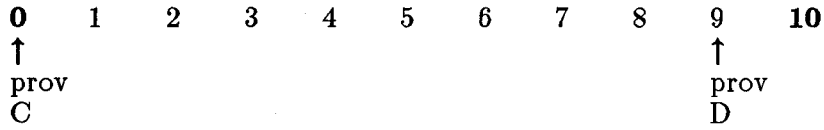
By similar proverbs we mean those that go together according to the criteria you used to create the categories. One way we can show how similar two proverbs are is by measuring their difference on a scale from **0** to **10**. In this scale we put similar proverbs close together and dissimilar proverbs far from each other. The more similar the proverbs are, the closer we place them to each other. The more different they are, the further apart we place them, one on **0** and the other on **10** being the extreme. Take again the two similar proverbs. We could represent how similar they are as follows:

0	1	2	3	4	5	6	7	8	9	**10**
↑		↑								
prov		prov								
A		B								

It's easier if we always place one of the two proverbs in the position **0** and the other proverb somewhere on the scale to the right, so that the number where we put the other proverb indicates how dissimilar they are:

the larger the number, the less similar they are.

The two dissimilar proverbs could be placed like this:



Let's take this category again. You have already chosen the most typical and the least typical proverbs in the category. Here are index cards with those proverbs. Place the most typical proverb in position **0**, and place the least typical on the scale according to how similar you consider them to be. Let's try it also with the most and least typical of these other two categories. Always place the most typical proverb in position **0**.

In the same way, we can compare how similar the most typical proverbs of any two categories are. Try it for these two categories – take the most typical proverbs from each and place them on the scale showing how similar they are. Now compare one of these with the typical proverb representing the third category.

Semantic Distances – Measurement

Most and least typical proverb

For each of these ten categories, select the most typical and least typical proverbs. Remember that the most typical proverb is the one you would show somebody as the best example of the proverbs included in the category. The least typical proverb is the one that is the worst fit within the category – the one you would get rid of so that the others are as much alike as possible. (Take index card copies of the selected proverbs.)

Similarity distances

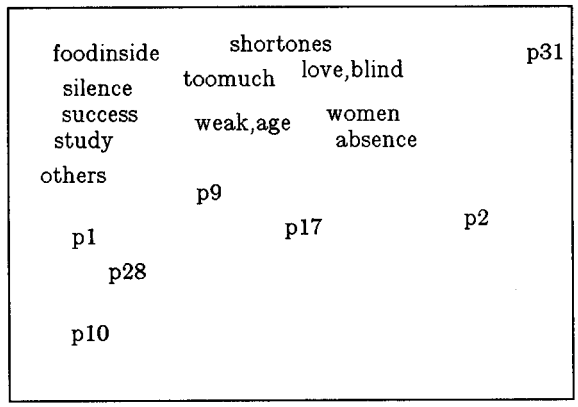
You just selected for each of the ten category the most and least typical proverbs. You will now use this scale from **0** to **10** to indicate, for each category, how similar the most and least typical proverbs are. We will use these copies on index cards of each of the proverbs to place on the scale. Remember to place the most typical proverb always in position **0** and the least typical somewhere on the scale to the right, so that the number where you put the least typical proverb indicates how dissimilar they are: the larger the number, the less similar they are. I will write down on this sheet of paper the result for each pairwise comparison. At any time, feel free to go back and change any of the values you have indicated earlier.

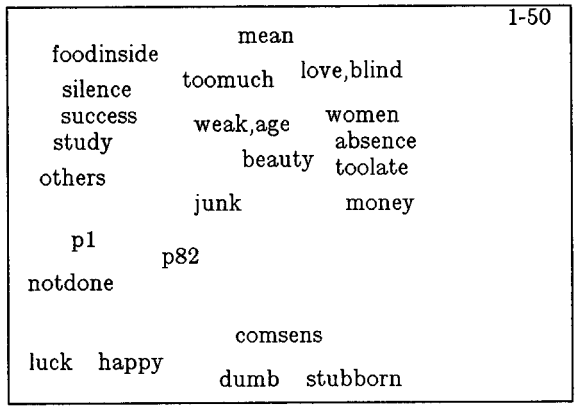
In the same way, compare the most typical proverbs for each pair of the ten categories. Use the same scale from **0** to **10**, with one of the proverbs always on the position **0**. I will record the result of each comparison. Again, at any time you can change any of the values you have given previously.

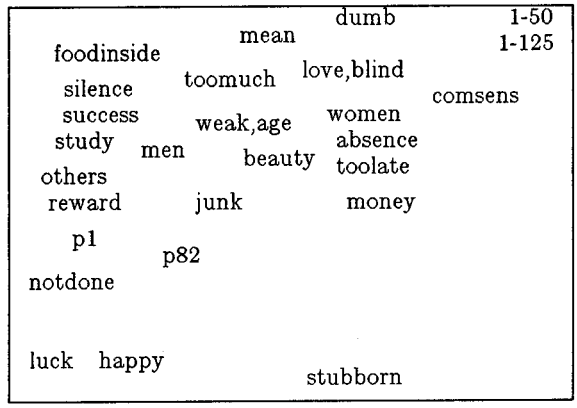
End

This finishes the fourth session. Thank you for participating in the experiment.

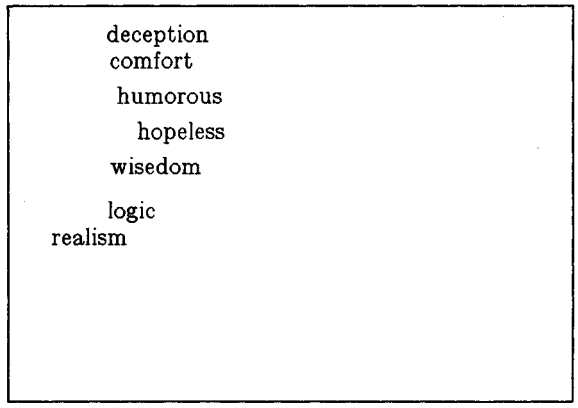
Appendix 2. Roots of Structures.

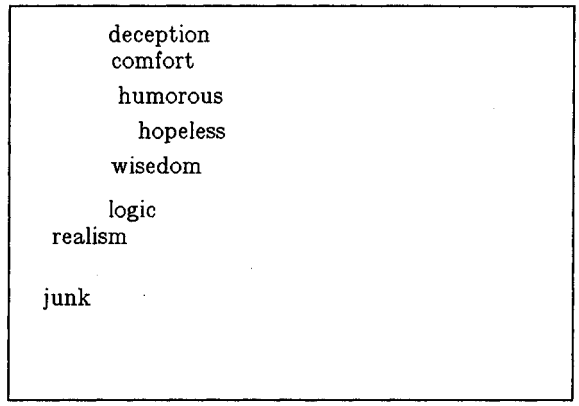


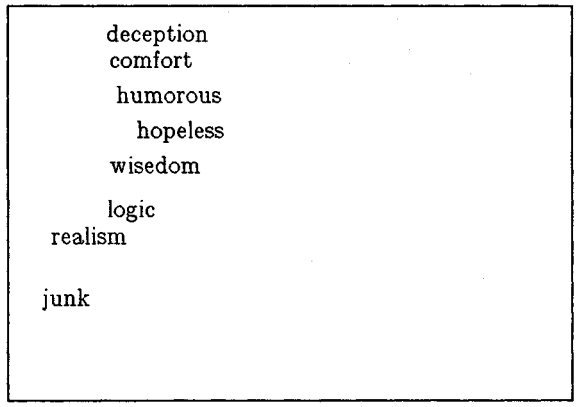




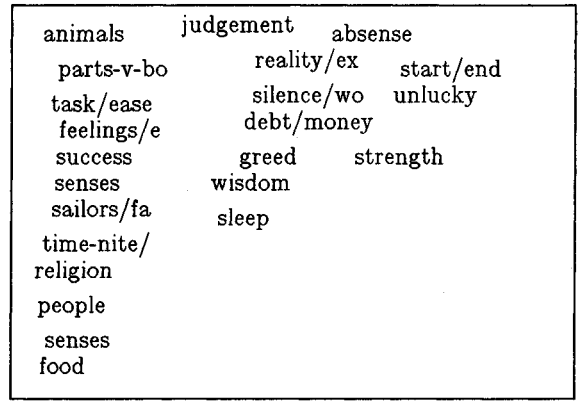
Subject 1

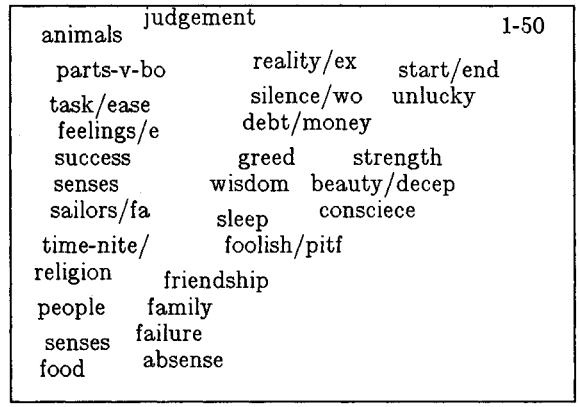

deception
comfort
humorous
hopeless
wisedom
logic
realism

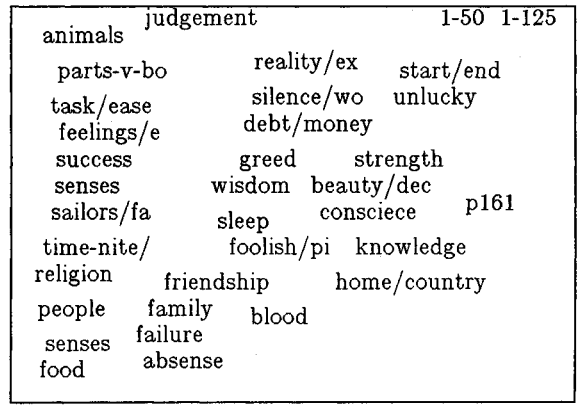

deception
comfort
humorous
hopeless
wisedom
logic
realism
junk


deception
comfort
humorous
hopeless
wisedom
logic
realism
junk

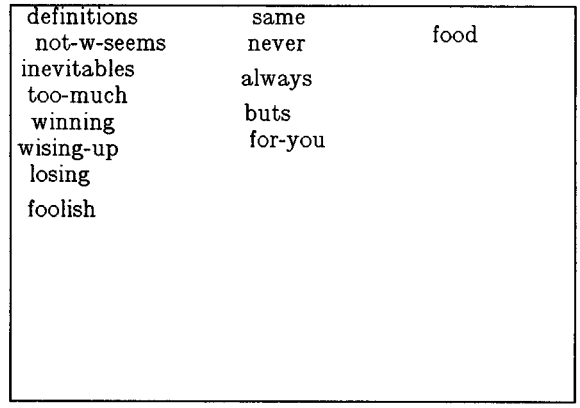
Subject 2

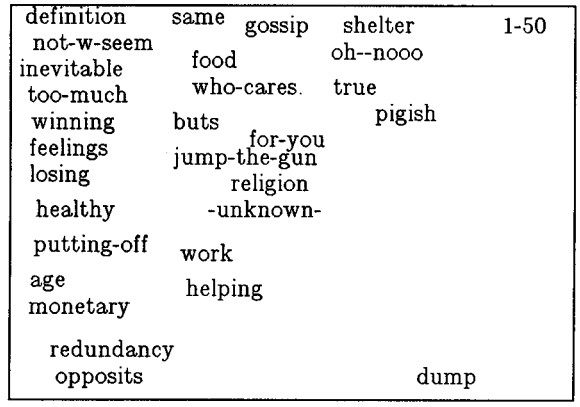

animals
judgement
absense
parts-v-bo
reality/ex
start/end
task/ease
silence/wo
unlucky
feelings/e
debt/money
success
greed
strength
senses
wisdom
sailors/fa
sleep
time-nite/
religion
people
senses
food

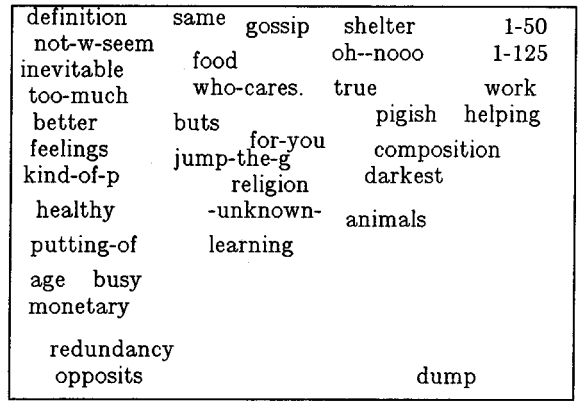

animals
judgement
1-50
parts-v-bo
reality/ex
start/end
task/ease
silence/wo
unlucky
feelings/e
debt/money
success
greed
strength
senses
wisdom
beauty/decep
sailors/fa
sleep
conscience
time-nite/
foolish/pitf
religion
friendship
people
family
senses
failure
food
absense


animals
judgement
1-50
1-125
parts-v-bo
reality/ex
start/end
task/ease
silence/wo
unlucky
feelings/e
debt/money
success
greed
strength
senses
wisdom
beauty/dec
sailors/fa
sleep
conscience
p161
time-nite/
foolish/pi
knowledge
religion
friendship
home/country
people
family
blood
senses
failure
food
absense

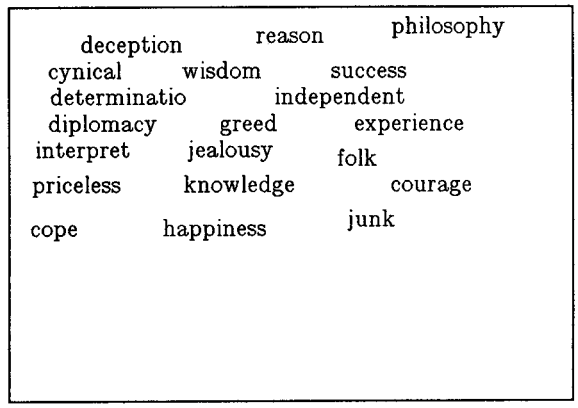
Subject 3

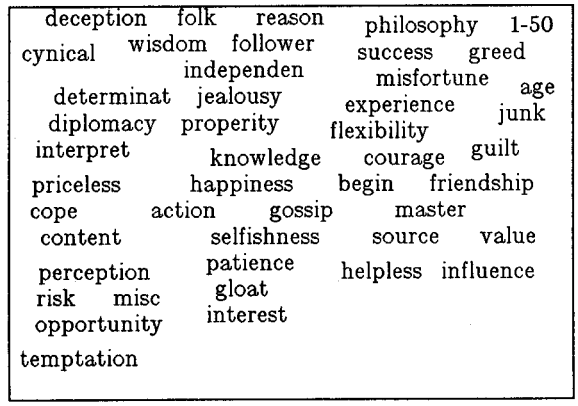

definitions
not-w-seems
inevitables
too-much
winning
wising-up
losing
foolish
same
never
always
buts
for-you
food

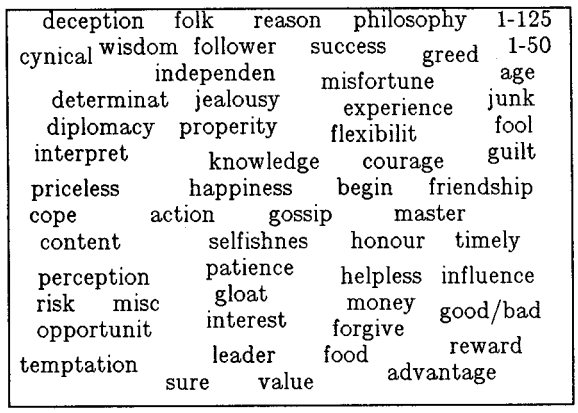

definition
not-w-seem
inevitable
too-much
winning
feelings
losing
healthy
putting-off
age
monetary
redundancy
opposits
same
gossip
food
who-cares.
buts
for-you
jump-the-gun
religion
-unknown-
work
helping
shelter
oh--nooo
true
pigish
1-50
dump


definition
not-w-seem
inevitable
too-much
better
feelings
kind-of-p
healthy
putting-of
age
busy
monetary
redundancy
opposits
same
gossip
food
who-cares.
buts
for-you
jump-the-g
religion
-unknown-
learning
shelter
oh--nooo
true
pigish
composition
darkest
animals
1-50
1-125
work
helping
dump

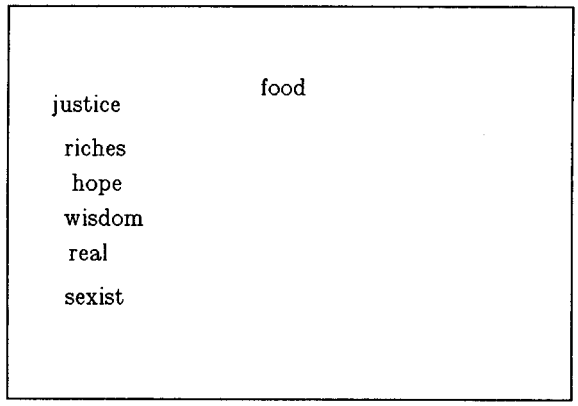
Subject 4

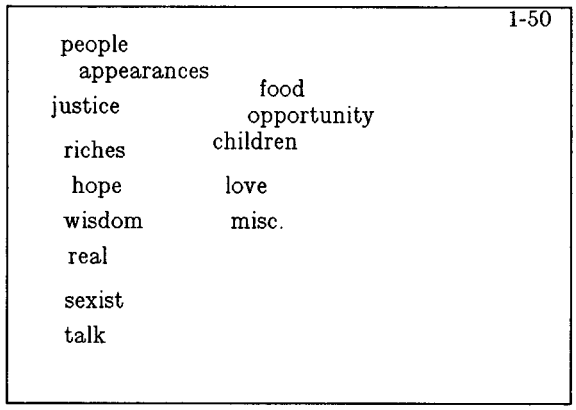

deception
reason
philosophy
cynical
wisdom
success
determinatio
independent
diplomacy
greed
experience
interpret
jealousy
folk
priceless
knowledge
courage
cope
happiness
junk

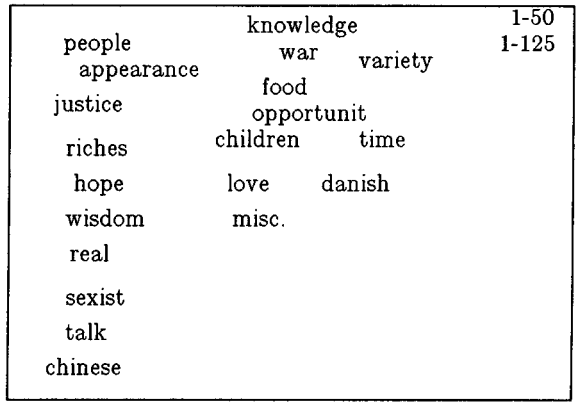

deception
folk
reason
philosophy
1-50
cynical
wisdom
follower
success
greed
independen
misfortune
age
determinat
jealousy
experience
junk
diplomacy
properity
flexibility
interpret
knowledge
courage
guilt
priceless
happiness
begin
friendship
cope
action
gossip
master
content
selfishness
source
value
perception
patience
helpless
influence
risk
misc
gloat
opportunity
interest
temptation


deception
folk
reason
philosophy
1-125
cynical
wisdom
follower
success
greed
1-50
independen
misfortune
age
determinat
jealousy
experience
junk
diplomacy
properity
flexibilit
fool
interpret
knowledge
courage
guilt
priceless
happiness
begin
friendship
cope
action
gossip
master
content
selfishnes
honour
timely
perception
patience
helpless
influence
risk
misc
gloat
money
opportunit
interest
forgive
good/bad
temptation
leader
food
reward
sure
value
advantage

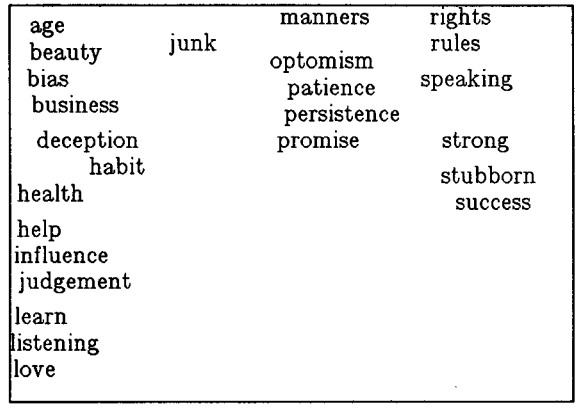
Subject 5

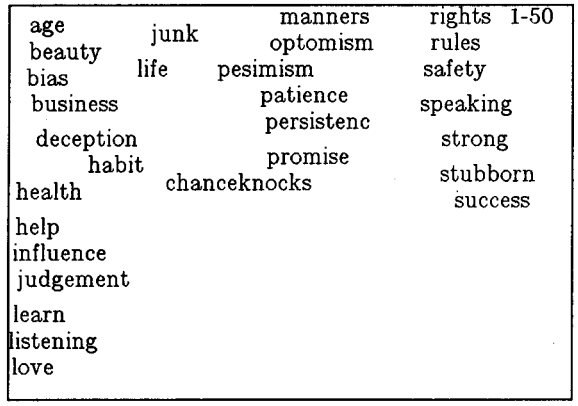

food
justice
riches
hope
wisdom
real
sexist

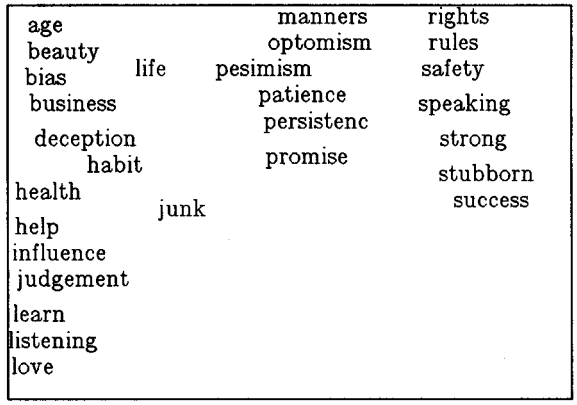

1-50
people
appearances
food
justice
opportunity
children
riches
hope
love
wisdom
misc.
real
sexist
talk


knowledge
1-50
1-125
people
war
variety
appearance
food
justice
opportunit
children
time
riches
hope
love
danish
wisdom
misc.
real
sexist
talk
chinese

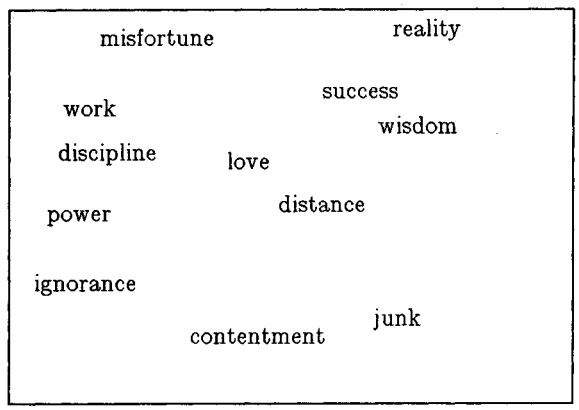
Subject 6

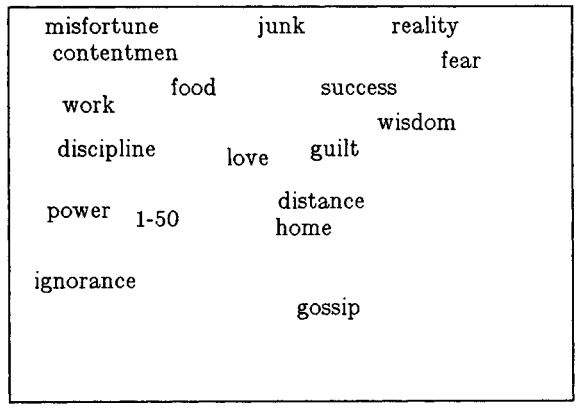

age
beauty
bias
business
deception
habit
health
help
influence
judgement
learn
listening
love
junk
manners
optomism
patience
persistence
promise
rights
rules
speaking
strong
stubborn
success

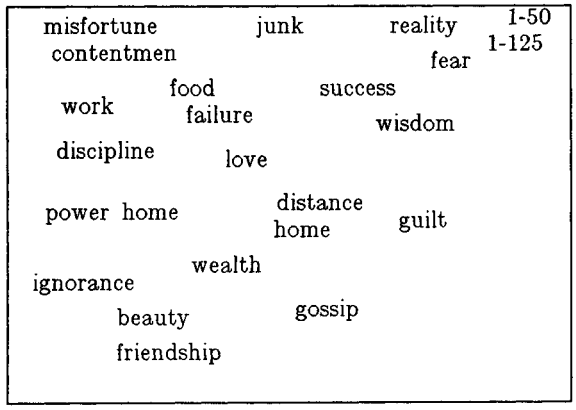

age
beauty
bias
business
deception
habit
health
help
influence
judgement
learn
listening
love
junk
life
chanceknocks
manners
optomism
pesimism
patience
persistenc
promise
rights
rules
safety
speaking
strong
stubborn
success
1-50


age
beauty
bias
business
deception
habit
health
help
influence
judgement
learn
listening
love
life
junk
manners
optomism
pesimism
patience
persistenc
promise
rights
rules
safety
speaking
strong
stubborn
success

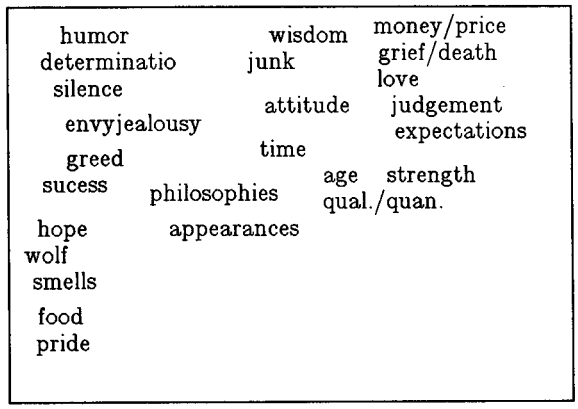
Subject 7

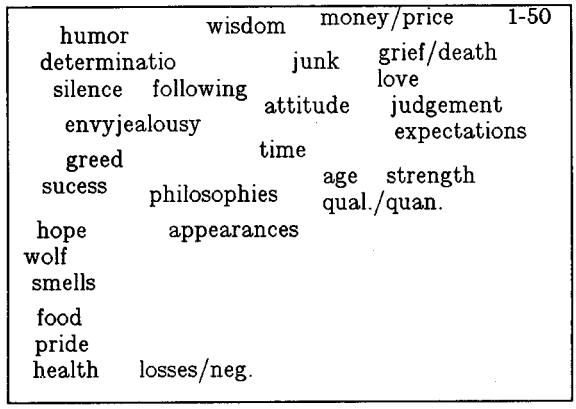

misfortune
reality
success
work
wisdom
discipline
love
power
distance
ignorance
junk
contentment

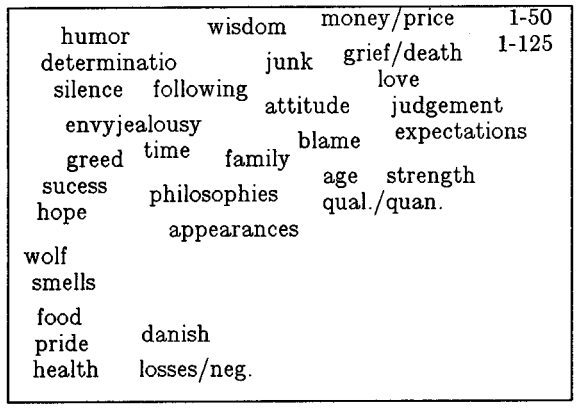

misfortune
junk
reality
contentmen
fear
food
success
work
wisdom
discipline
love
guilt
power
1-50
distance
home
ignorance
gossip


misfortune
junk
reality
1-50
contentmen
1-125
fear
food
success
work
failure
wisdom
discipline
love
power
home
distance
home
guilt
wealth
ignorance
beauty
gossip
friendship

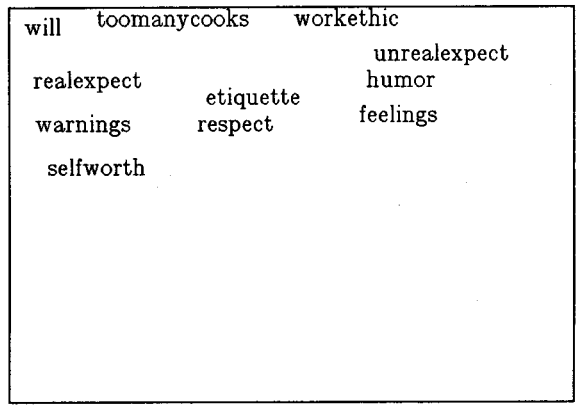
Subject 8

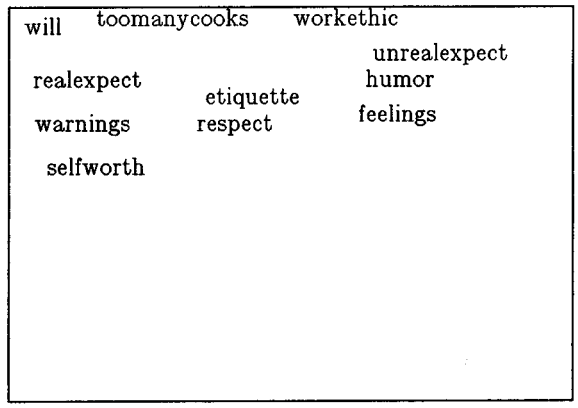

humor
wisdom
money/price
determinatio
junk
grief/death
silence
love
attitude
judgement
envyjealousy
expectations
greed
time
sucess
philosophies
age
strength
qual./quan.
hope
appearances
wolf
smells
food
pride

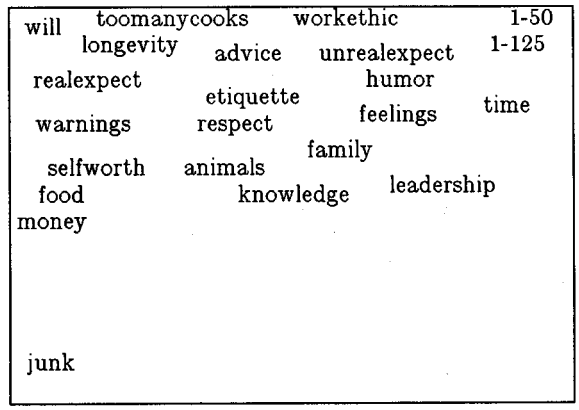

humor
wisdom
money/price
1-50
determinatio
junk
grief/death
silence
following
love
attitude
judgement
envyjealousy
expectations
greed
time
sucess
philosophies
age
strength
qual./quan.
hope
appearances
wolf
smells
food
pride
health
losses/neg.


humor
wisdom
money/price
1-50
determinatio
junk
grief/death
1-125
silence
following
love
attitude
judgement
envyjealousy
blame
expectations
greed
time
family
sucess
philosophies
age
strength
hope
qual./quan.
appearances
wolf
smells
food
pride
danish
health
losses/neg.

Subject 9


will
toomanycooks
workethic
unrealexpect
realexpect
humor
etiquette
warnings
respect
feelings
selfworth


will
toomanycooks
workethic
unrealexpect
realexpect
humor
etiquette
warnings
respect
feelings
selfworth


will
toomanycooks
workethic
1-50
longevity
advice
unrealexpect
1-125
realexpect
humor
etiquette
warnings
respect
feelings
time
family
selfworth
animals
food
knowledge
leadership
money
junk

Subject 10

Appendix 3. Root Space Usage over all Subjects.