Research Paper ■

The PEN–Ivory Project: Exploring User-interface Design for the Selection of Items from Large Controlled Vocabularies of Medicine

ALEX D. POON, LAWRENCE M. FAGAN, MD, PHD, EDWARD H. SHORTLIFFE, MD PHD

Abstract **Objective:** To explore different user-interface designs for structured progress note entry, with a long-term goal of developing design guidelines for user interfaces where users select items from large medical vocabularies.

Design: The authors created eight different prototypes of a pen-based progress-note–writing system called PEN-Ivory. Each prototype allows physicians to write patient progress notes using simple pen-based gestures such as circle, line-out, and scratch-out. The result of an interaction with PEN-Ivory is a progress note in English prose. The eight prototypes were designed in a principled way, so that they differ from one another in just one of three different user-interface characteristics.

Measurements: Five of the eight prototypes were tested by measuring the time it took 15 users, each using a distinct prototype, to document three patient cases consisting of a total of 63 medical findings.

Results: The prototype that allowed the fastest data entry had the following three user-interface characteristics: it used a paging rather than a scrolling form, it used a fixed palette of modifiers rather than a dynamic "pop-up" palette, and it made available all findings from the controlled vocabulary at once rather than displaying only a subset of findings generated by analyzing the patient's problem list.

Conclusion: Even simple design changes to a user interface can make dramatic differences in user performance. The authors discuss possible influences on performance, such as positional constancy, user uncertainty, and system anticipation, that may contribute significantly to the effectiveness of systems that display menus of items from large controlled vocabularies of medicine.

■ JAMIA. 1996;3:168–183.

Physicians have traditionally recorded clinical findings in the form of handwritten progress notes. The enduring use of handwriting and paper-based medical records is not surprising. Paper is easy to use, portable, and facilitates the jotting of quick notes and simple sketches. But the use of pen and paper is not without problems. Handwritten patient charts are often illegible, poorly formatted, and even missing.[1,2]

Many physicians now use audio recording devices to dictate their notes, which are later transcribed using typewriters or word processors. While this method of documentation saves time for the physician, the tran-

Affiliation of the authors: Section on Medical Informatics, Stanford University, Stanford, CA.

Mr. Poon was supported by fellowships provided by the Department of Defense and IBM. Dr. Shortliffe was supported by Fellowship #84R-2459-HPE from the Henry J. Kaiser Family Foundation. Additional support for the project was provided by the CAMIS project (LM-05305), by Eli Lilly, Inc., and by Apple Computer, Inc.

Presented in part at the Eighteenth Annual Symposium on Computer Applications in Medical Care in November 1994.

Correspondence and reprints: Alex D. Poon, MSOB X-215, Stanford University School of Medicine, Stanford University, CA 94305. e-mail: poon@camis.stanford.edu

Received for publication: 4/20/95; accepted for publication: 11/17/95.

scription process introduces not only the potential for transcription errors, but also a delay between the time a physician generates a note and the time the note becomes accessible.[1]

But perhaps the most important drawback of handwritten and dictated information is that it is neither structured nor coded, and therefore difficult to process with computers. The advantages of coded data entry are well documented.[1] For example, coded medical data can be used as input for medical expert systems, systems for outcomes analysis, billing systems, and research databases.

Because of the problems of illegibility, chart inaccessibility, and noncoded data, researchers have created computer systems that can be used to record medical findings in a structured way. Many of the earliest systems to use structured data entry were created for writing radiology reports.[3,4] However, computer systems for clinical data entry do not enjoy widespread use. They are generally difficult to use[1,5,6] and are less time-efficient[7,8] and less flexible[1,2,9] than paper and pen. Furthermore, large computer terminals have been shown to interfere with the physician–patient encounter.[5,9]

The need for systems that allow and encourage direct data entry by physicians is well known. The Institute of Medicine's Committee on Improving the Patient Record calls the development of such systems "the single greatest challenge in implementing the computerized patient record."[1] We have argued elsewhere that awkward human–computer interfaces are perhaps the biggest factor inhibiting the clinical use of computers.[2]

One of the goals of the PEN-Ivory project was to develop a computer system that would realize the benefits of coded data entry and take advantage of the convenience of paper and pen. This idea is not a new one; there are at least five other pen-based computer systems for medical charting.[7,10–13] PureMD, developed by Développement Purkinje Inc., is perhaps the most comprehensive, allowing physicians and dentists to enter information from a database of over 50,000 possible clinical observations.[11] However, none of these projects has published evaluations of the merits of alternative user-interface characteristics for such systems.

A second goal of our work was, accordingly, to evaluate alternative user-interface designs through controlled experiments. Though the experiments we describe in this paper focus on the PEN-Ivory system, one of our key long-term aims is to develop specific design guidelines that can be applied generally to any medical system that uses structured data entry for choosing items from large controlled vocabularies.

In this paper, we describe the design and evaluation of PEN-Ivory, a pen-based system using structured data entry for writing patient progress notes. First, we give a general description of the system and its user interface. Then, we describe how we designed the system in a principled way by creating multiple working prototypes that differ in just one user-interface characteristic. Next, we present the results of an experiment in which we evaluated the different prototypes in a controlled laboratory setting. Finally, we outline areas for further exploration that might lead to general design guidelines that are useful to all designers of systems that use large controlled vocabularies of medicine.

System Description

PEN-Ivory is derived from an earlier system called Ivory.[14] Like Ivory, PEN-Ivory uses a SOAP (subjective, objective, assessment, and plan) format for its progress notes.[15] However, while Ivory was designed for use with a mouse-based, non-mobile, graphic user interface, PEN-Ivory is designed for a mobile, pen-based system. Currently, PEN-Ivory employs structured data entry for only the subjective and objective sections of the note. We are presently extending the vocabulary to include the assessment and plan portions.

PEN-Ivory runs on an Apple Macintosh computer connected to a digitizing tablet integrated with a backlit LCD display. We developed PEN-Ivory using C++ with the Metrowerks Codewarrior development environment and the Apple MacApp application framework. The MacApp framework provides standard Macintosh-style interface widgets such as scrollbars, buttons, and dialog boxes, but lacks support for pen-based input. Therefore, we developed our own set of tools for processing the pen input signal, including tools for digital ink capture and gesture recognition.

Figure 1 shows that PEN-Ivory's user interface is divided into two main areas. The left side represents the encounter form, on which the names of medical findings are listed; the right side represents the attributes palette, used to embellish findings with specific modifiers. A text translation of the finding and modifiers currently being entered is shown at the top right of the screen. In between the encounter form and the attributes palette are page tabs used for moving among the different pages of the encounter form.

Users interact with the encounter form and attributes palette with a set of three simple gestures: circle, line-out, and scratch-out.

Circle

Circling a finding in the encounter form signifies that the patient has that particular finding. Multiple findings may be circled with a single stroke of the pen.

When a finding is first circled, the circle appears in bold and the user may enter more information about that particular finding by circling items on a predefined lists of modifiers in the attributes palette (Figure 1). The contents of the palette change depending on the finding that was just circled on the encounter form.

Line-out

Drawing a horizontal line through a finding or modifier signifies that the patient does not have that particular finding or and that the modifier does not apply. As with circling, multiple findings may be lined-out with a single stroke.

Scratch-out

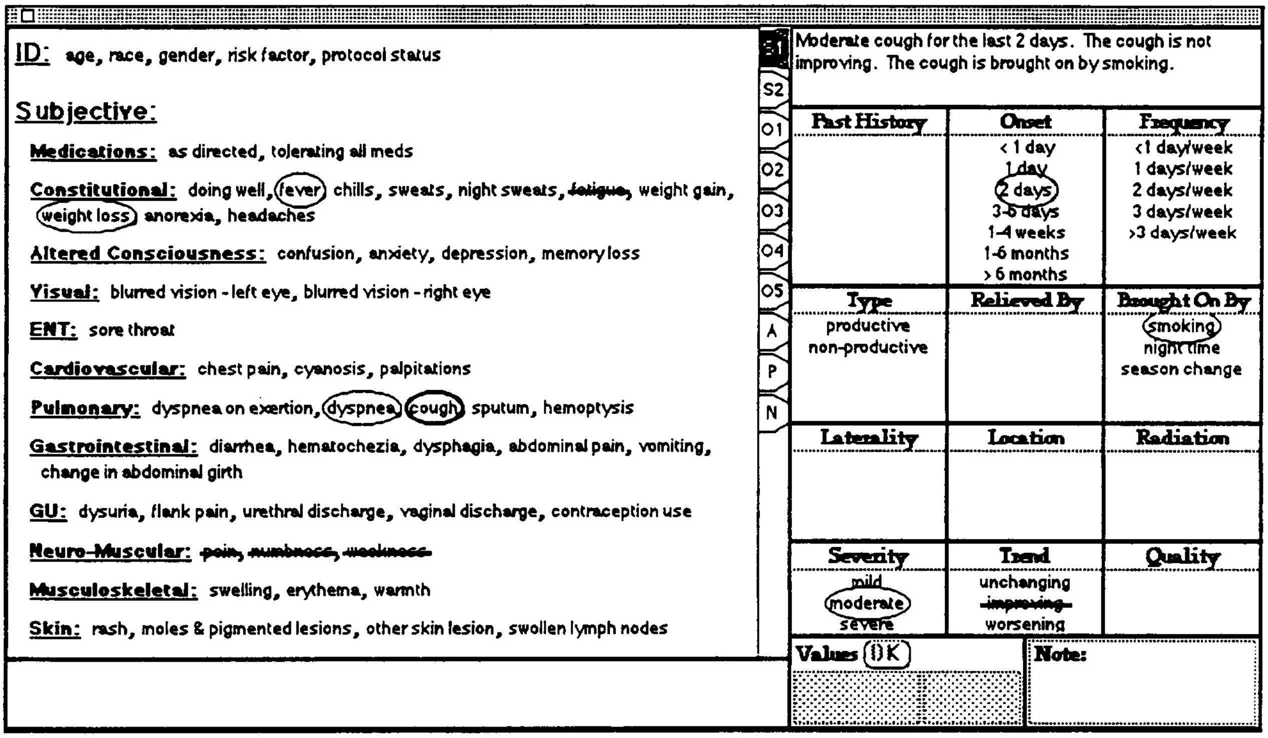
Leaving a finding untouched signifies that a finding's status is either nonassessed or unknown. Scratching out a finding or modifier that has previously been assessed returns its status to the nonassessed or unknown state (Figure 2). Multiple findings or modifiers may be scratched out with a single stroke.

Handwritten Notes

Users may also make handwritten, free-form notes that are stored as electronic ink (Figure 3). This allows the user not only to draw sketches and diagrams that are printed out along with the note, but also to record information that cannot easily be expressed in PEN-Ivory's controlled vocabulary. PEN-Ivory does not attempt to translate the handwritten notes into ASCII text.

Progress-note Text Generation

As the user interacts with the encounter form and the attributes palette, PEN-Ivory generates English-language text based on the information that the user has



ID: age, race, gender, risk factor, protocol status

Subjective:

Medications: as directed, tolerating all meds

Constitutional: doing well, [✓ fever] chills, sweats, night sweats, ~~fatigue,~~ weight gain, [✓ weight loss] anorexia, headaches

Altered Consciousness: confusion, anxiety, depression, memory loss

Visual: blurred vision - left eye, blurred vision - right eye

ENT: sore throat

Cardiovascular: chest pain, cyanosis, palpitations

Pulmonary: dyspnea on exertion, [✓ dyspnea] [✓ cough] sputum, hemoptysis

Gastrointestinal: diarrhea, hematochezia, dysphagia, abdominal pain, vomiting, change in abdominal girth

GU: dysuria, flank pain, urethral discharge, vaginal discharge, contraception use

Neuro-Muscular: ~~pain, numbness, weakness~~

Musculoskeletal: swelling, erythema, warmth

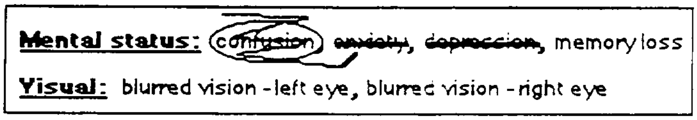
Skin: rash, moles & pigmented lesions, other skin lesion, swollen lymph nodes

Moderate cough for the last 2 days. The cough is not improving. The cough is brought on by smoking.

Past History	**Onset**	**Frequency**
	< 1 day ~~1 day~~ [✓ 2 days] ~~3-6 days~~ 1-4 weeks 1-6 months > 6 months	<1 day/week 1 days/week 2 days/week 3 days/week >3 days/week
Type	**Relieved By**	**Brought On By**
productive non-productive		[✓ smoking] ~~night time~~ season change
Laterality	**Location**	**Radiation**
Severity	**Trend**	**Quality**
~~mild~~ [✓ moderate] ~~severe~~	unchanging ~~improving~~ worsening	

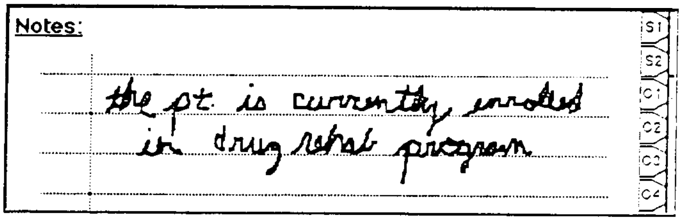
Note:

Figure 1 PEN-Ivory's user interface. The left side of the screen represents the encounter form on which the names of medical findings are listed. The right side represents the attributes palette, used to augment findings with specific modifiers (in this case, modifiers refer to "cough," which is circled in bold on the encounter form). Users circle, line out, and scratch out words to interact with the system. A text translation of the selected finding and its attributes is displayed at the top right. The page tabs located between the encounter form and the attributes palette are used to move among the pages of the encounter form.



Mental status: confusion, ~~anxiety~~, ~~depression~~, memory loss
Visual: blurred vision - left eye, blurred vision - right eye

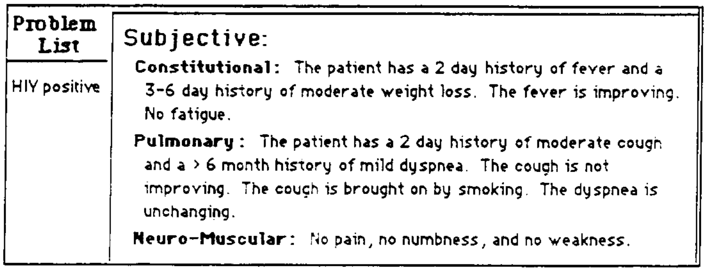
Figure 2 Scratching out a finding name returns that finding to the nonassessed or unknown state.



Notes:

the pt. is currently enrolled in drug rehab program

Figure 3 Users can handwrite notes in free-text form.



Problem List	Subjective:
HIV positive	**Constitutional:** The patient has a 2 day history of fever and a 3-6 day history of moderate weight loss. The fever is improving. No fatigue. **Pulmonary:** The patient has a 2 day history of moderate cough and a > 6 month history of mild dyspnea. The cough is not improving. The cough is brought on by smoking. The dyspnea is unchanging. **Neuro-Muscular:** No pain, no numbness, and no weakness.

Figure 4 A PEN-Ivory–generated progress note. Users may interact with the note using the same circle, line-out, and scratch-out gestures.

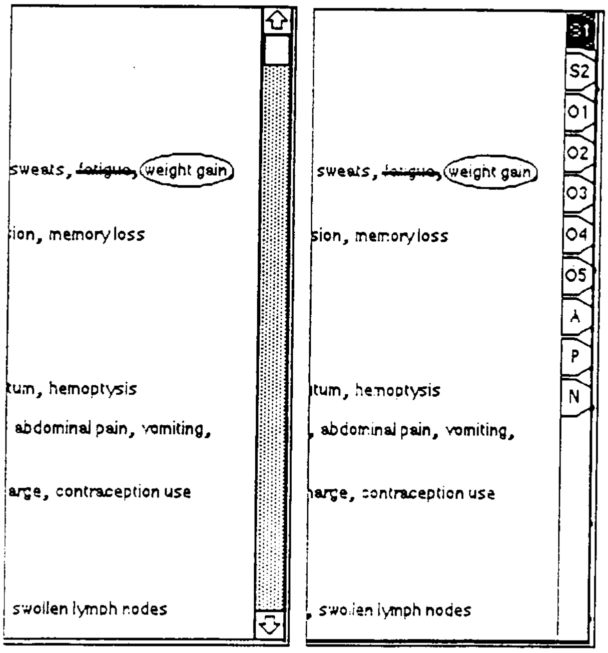
entered. The user may toggle between viewing the encounter form and viewing the full generated progress note. Figure 4 shows an example of a generated progress note based on the entries shown in Figure 1. The user may directly edit the finding and modifier terms in the progress note with the same set of gestures used in the encounter form.

User-Interface Characteristics

As was stated earlier, PEN-Ivory is derived from a mouse-driven program called Ivory. Based on informal observations of problems that physicians had navigating through Ivory's menu interface, we identified three major *user-interface characteristics* that we could change to enhance the performance of PEN-Ivory. Rather than assume that all three changes should be made, however, we implemented the changes as options and designed controlled experiments to evaluate the effect of each change, singly and in combination. First, we allowed changing the metaphor of the encounter form from that of a long scrolling list of findings to a series of notebook pages with tabs for moving among them (Figures 1 and 5). Second, we built two forms of attributes palettes. The first one is like Ivory's, in that it is dynamic; it pops up only when needed, and includes only those attribute categories that are relevant for the particular finding being modified (Figure 7). The second palette option is fixed; it is always displayed on the screen (Figures 1 and 6), and contains all attribute categories, including those that are irrelevant for the particular finding being modified. Third, we tried two different methods for determining which finding terms to display in the encounter form. One design uses Ivory's methods of displaying on its encounter form only those findings that are related to the patient's problem list, while the second displays all of the findings from the vocabulary, highlighting those that are related to the patient's problem list (Figure 8).

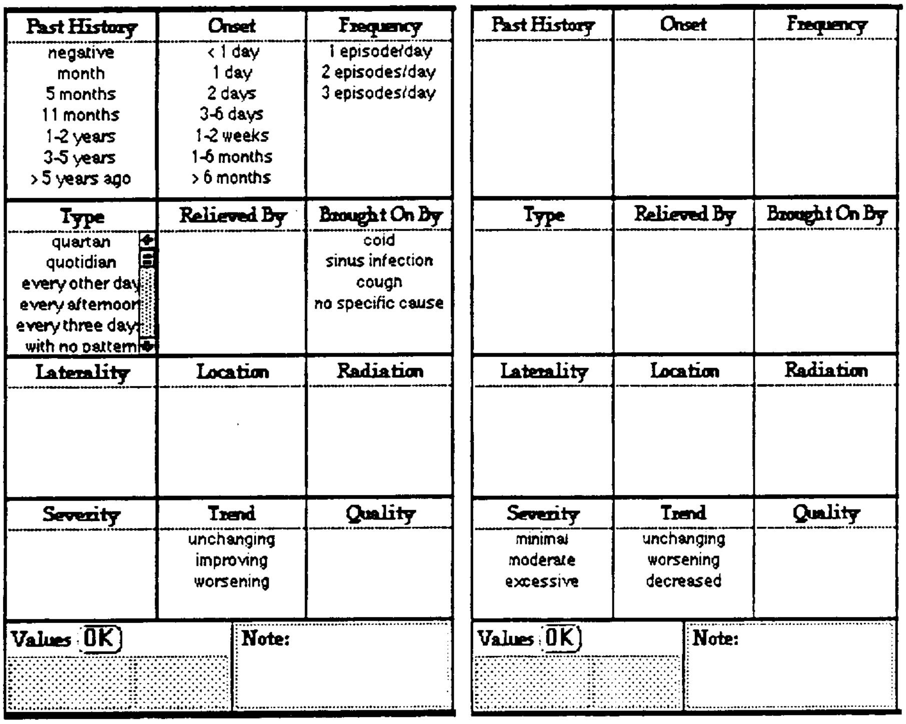
To test whether any of these changes would make PEN-Ivory easier and more efficient to use, we created eight different PEN-Ivory prototypes that had different combinations of the three interface characteristics. Each prototype used a pen-based interface, but they differed from one another in just one interface characteristic.

The following is a more detailed description of each



sweats, ~~fatigue~~, weight gain
ion, memory loss
tum, hemoptysis
abdominal pain, vomiting,
arge, contraception use
swollen lymph nodes

Figure 5 Findings can be presented on either a scrolling form (left) or a paging form (right).



Past History	Onset	Frequency
negative month 5 months 11 months 1-2 years 3-5 years > 5 years ago	< 1 day 1 day 2 days 3-6 days 1-2 weeks 1-6 months > 6 months	1 episode/day 2 episodes/day 3 episodes/day
Type	Relieved By	Brought On By
quartan quotidian every other day every afternoon every three days with no pattern		cold sinus infection cough no specific cause
Laterality	Location	Radiation
Severity	Trend	Quality
	unchanging improving worsening	
Values OK	Note:	

Past History	Onset	Frequency
Type	Relieved By	Brought On By
Laterality	Location	Radiation
Severity	Trend	Quality
minimal moderate excessive	unchanging worsening decreased	
Values OK	Note:	

Figure 6 Fixed-attributes palettes for two different findings, fever (left) and alcohol use (right). Notice that in both palettes, the positions of the attribute categories are the same.

of the three interface characteristics that we investigated.

Scrolling vs Paging (Form Type)

The encounter form in which the findings are displayed can be listed either as one long *scrolling* list of findings, or as a series of *pages* of findings (Figure 5). The advantage of a scrolling form is that it allows for greater flexibility with regard to which findings can be displayed on the screen at once. The advantage of a paging form is that it enables the user to remember the location of particular findings by both their page numbers and their positions on the pages. For instance, a user might remember that subjective gastrointestinal problems are always listed on the bottom of page S2.

We hypothesized that the paging encounter form would be faster and easier to use because of its greater ability to allow users to remember findings by screen location. This hypothesis relied on the premise that users can become sufficiently familiar with the forms that they can take advantage of such positional memory.[16–20]

Dynamic vs Fixed Palette (Palette Type)

Users can elaborate upon a particular finding by selecting items from the attributes palette. For instance, "headache" is a basic finding, while the attributes "every day," "worsening," and "no past history" are attributes that can embellish "headache."

A *fixed attributes palette* is one that is always visible and whose location and size on the screen are fixed. In addition, the categories of attributes have fixed absolute positions within the palette. Figure 6 shows two examples of fixed palettes for two different findings, fever and alcohol use. Notice that the attribute categories "Severity," "Trend," and "Quality" are always listed at the bottom of the palette, regardless of which finding is being modified. This requires that all of the 14 attribute categories be shown for each finding, regardless of which ones are relevant for the particular finding.

A *dynamic attributes palette* is one that pops up on demand and whose location and size on the screen are not necessarily fixed. In addition, the absolute positions of the attribute categories may change, depending on which categories are relevant for the particular finding being modified. Figure 7 shows two examples of dynamic palettes for two different findings, fever and alcohol use. Notice the attribute category "Trend" is in a different position for the two different findings, because only those attribute categories that are relevant for the particular finding are shown.

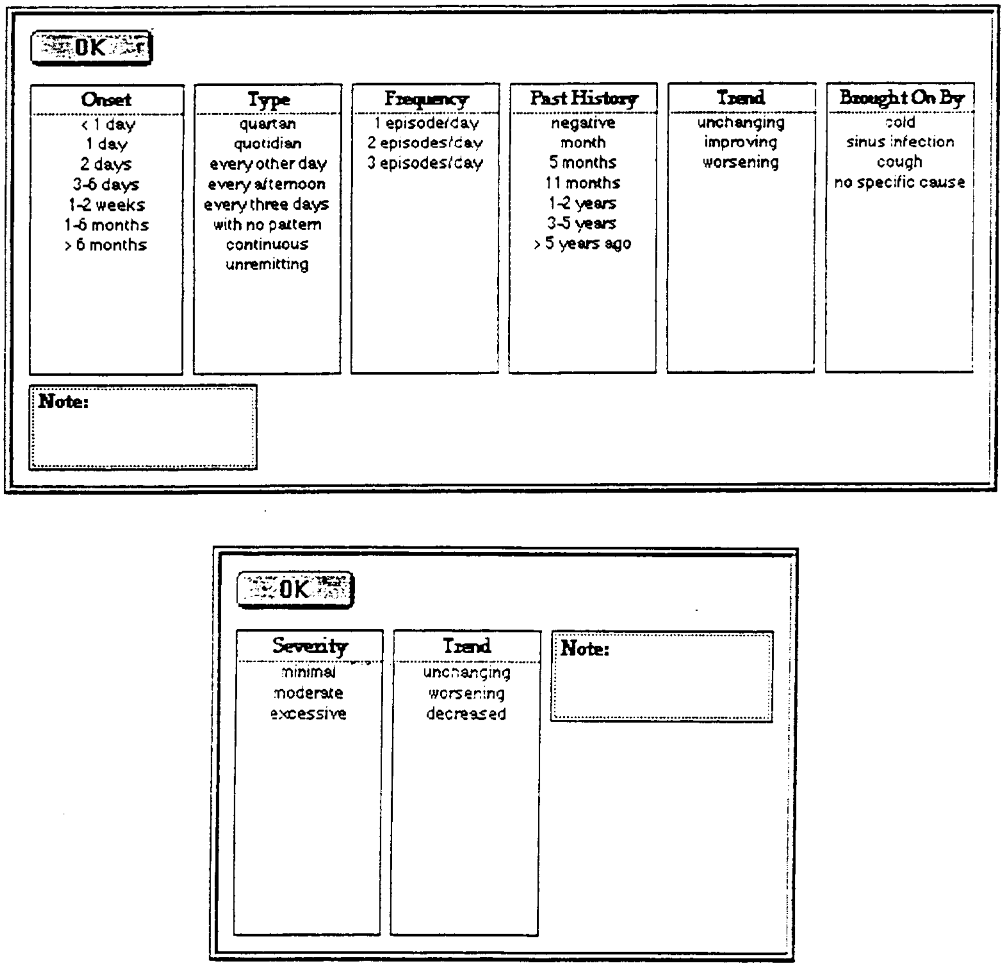
There are three potential advantages of a fixed palette. First, because a fixed palette is always displayed on the screen, the user need not wait for the palette to pop up after selecting a finding. Second, a fixed palette does not have to be explicitly dismissed by the user (in the dynamic palette, the user must hit the "OK" button to dismiss the pop-up palette), and therefore the palette can be easily ignored if desired. Finally, because the absolute positions of the attribute categories remain fixed, the user can more easily memorize the location of each category in the palette.

Still, the dynamic palette is not without its own set of advantages. First, because a dynamic palette pops up only when needed, more of the tablet can be devoted to displaying findings in the encounter form, reducing the length of the scroll or number of pages needed to hold the findings. Second, because the dynamic palette can be larger than the fixed palette, the individual windows for the attribute categories can be larger as well, thereby reducing the need for scroll bars in the attribute categories. Finally, because the dynamic palette automatically filters out the attribute categories that are irrelevant for a particular finding, it reduces the amount of information that the user must see at once.

Our hypothesis was that the fixed palette would be faster to use than the dynamic one, because it permits the user to memorize the absolute location on screen of the attribute categories. Again, we relied on the assumption that users are able to take advantage of positional memory.[16-20] We also suspected that users would prefer the fixed palette to the dynamic one because the fixed palette does not require an explicit dismissal step.

All Findings vs Subset of Findings (Completeness)

The Ivory vocabulary consists of over 1,000 basic findings. Findings can be grouped logically by either problem or organ system. A problem group consists



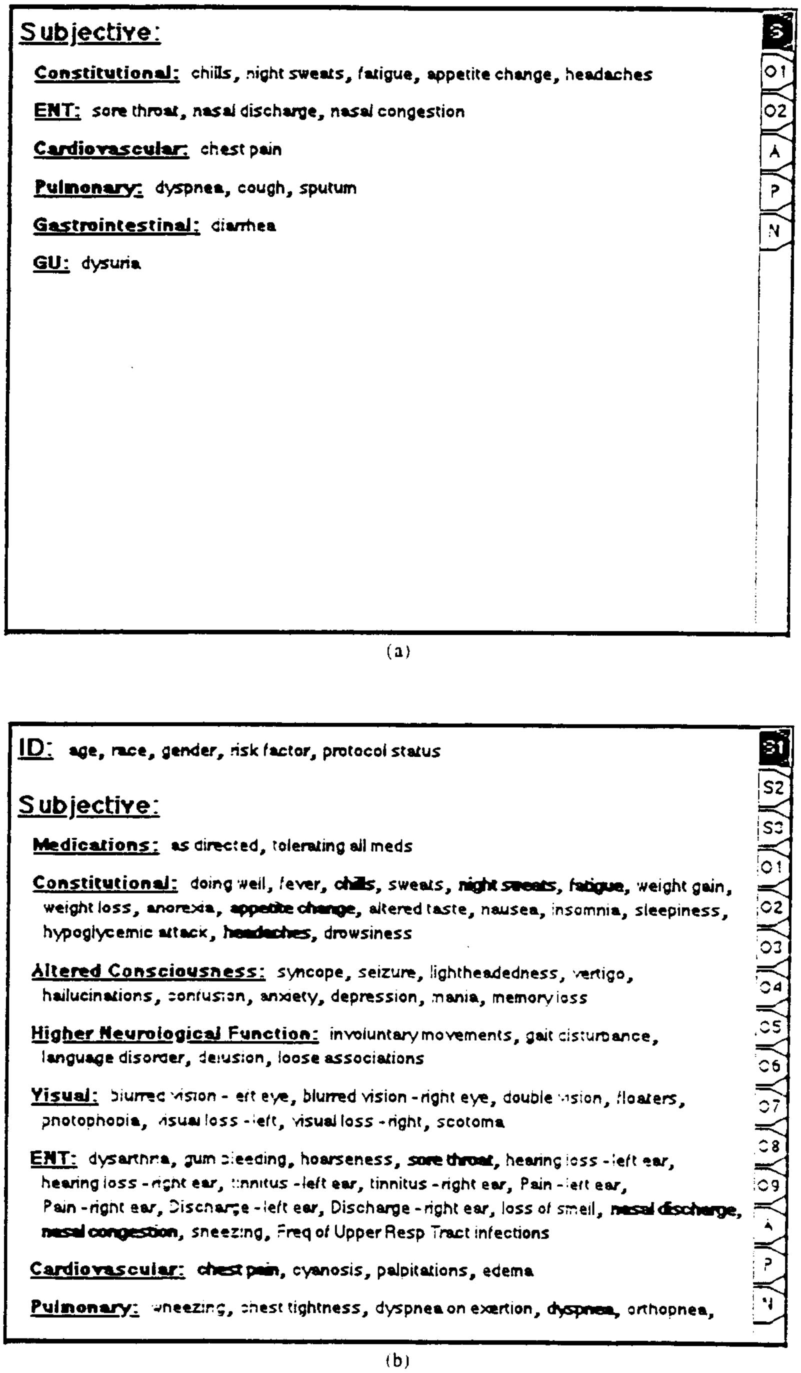
Onset	Type	Frequency	Past History	Trend	Brought On By
< 1 day	quartan	1 episode/day	negative	unchanging	cold
1 day	quotidian	2 episodes/day	month	improving	sinus infection
2 days	every other day	3 episodes/day	5 months	worsening	cough
3-6 days	every afternoon		11 months		no specific cause
1-2 weeks	every three days		1-2 years		
1-6 months	with no pattern		3-5 years		
> 6 months	continuous		> 5 years ago		
	unremitting				

Severity	Trend
minimal	unchanging
moderate	worsening
excessive	decreased

Figure 7 Dynamic attributes palettes for two different findings, fever (top) and alcohol use (bottom). Notice that the positions of the attribute categories differ depending on the finding.

of only those findings that are relevant for patients with that particular problem. For instance, "AZT Intolerance" is a problem group that consists of 83 findings that are likely to be evaluated in patients who are AZT-intolerant. The problem groups used in our studies were created by a physician who was a member of the Ivory project team. The problem groups were AIDS-related, as the physician helped develop Ivory's vocabulary for a project on decision support for AIDS-protocol care. An organ-system group consists of findings that belong to a particular organ system, such as all findings related to the cardiac exam. We refer to problem groups and organ-system groups more generally as finding groups.

We developed two different methods for using finding groups. The first displays only a *subset of findings* based on a particular finding group. The second displays *all findings* all the time, while highlighting in boldface those for a relevant finding group.



Subjective:

Constitutional: chills, night sweats, fatigue, appetite change, headaches

ENT: sore throat, nasal discharge, nasal congestion

Cardiovascular: chest pain

Pulmonary: dyspnea, cough, sputum

Gastrointestinal: diarrhea

GU: dysuria

(a)

ID: age, race, gender, risk factor, protocol status

Subjective:

Medications: as directed, tolerating all meds

Constitutional: doing well, fever, **chills**, sweats, **night sweats**, **fatigue**, weight gain, weight loss, anorexia, **appetite change**, altered taste, nausea, insomnia, sleepiness, hypoglycemic attack, **headaches**, drowsiness

Altered Consciousness: syncope, seizure, lightheadedness, vertigo, hallucinations, confusion, anxiety, depression, mania, memory loss

Higher Neurological Function: involuntary movements, gait disturbance, language disorder, delusion, loose associations

Visual: blurred vision - left eye, blurred vision - right eye, double vision, floaters, photophobia, visual loss - left, visual loss - right, scotoma

ENT: dysarthria, gum bleeding, hoarseness, **sore throat**, hearing loss - left ear, hearing loss - right ear, tinnitus - left ear, tinnitus - right ear, Pain - left ear, Pain - right ear, Discharge - left ear, Discharge - right ear, loss of smell, **nasal discharge**, **nasal congestion**, sneezing, Freq of Upper Resp Tract infections

Cardiovascular: **chest pain**, cyanosis, palpitations, edema

Pulmonary: wheezing, chest tightness, dyspnea on exertion, **dyspnea**, orthopnea,

(b)

Figure 8 Subset of Findings (a) vs All Findings (b) for the problem group "fatigue." In (a), only findings related to fatigue are displayed. In (b), all Ivory findings are displayed, and findings related to fatigue are shown in boldface.

In the prototypes that display only a subset of findings at once, the system decides which findings to display based on the patient's problem list (which is previously specified). This allows users to work with a manageable subset of findings. For instance, in Figure 8a, only those findings that are related to the problem group "fatigue" have been loaded into the encounter form, and only a few pages are needed to display them. Users can use pull-down menus to load in additional finding groups if the currently loaded groups do not contain all of the findings that they wish to enter.

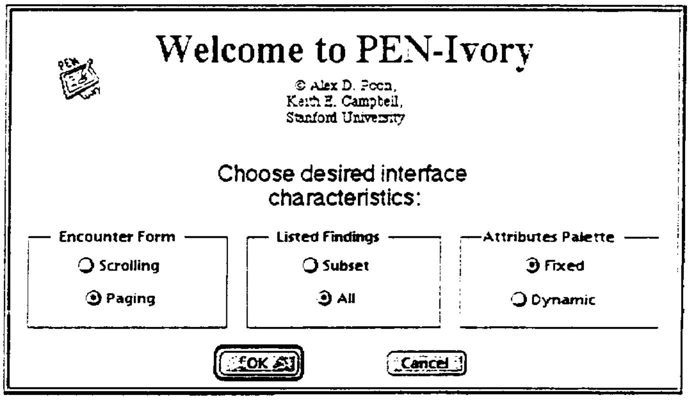
In the prototypes that display all findings at all times, finding names from relevant finding groups are highlighted in boldface on the screen for emphasis. Thus, although all 1,000 findings are made available at once, only a small subset are highlighted, allowing users to ignore the findings that are less relevant. For instance, in Figure 8b, all 1,000 findings are presented in 12 pages, but only those related to the problem group "fatigue" are highlighted.

There are two potential advantages in displaying all findings. First, because all findings in the database are always shown, the position of each finding on the form is constant, regardless of which problem or organ-system groups are loaded. Second, if a desired finding is not listed on the form, the user knows that the finding does not exist in the database of descriptive terms. If only a subset of findings is loaded at once, and a desired finding is not listed, the user cannot easily know whether the finding is simply not part of the loaded finding groups, or if it does not exist in the vocabulary at all.

We hypothesized that displaying all findings would be superior to displaying only a subset because of its ability to allow users, once again, to remember findings by position. Our hypothesis also relies on the premise that users can easily ignore extraneous information presented in a display, an ability that Nygren has observed in studies of computer users.[20]

Selection of Prototype

Changing the user-interface characteristics to test a different prototype was easy, as we designed the system so that we could choose the desired characteristics using a control panel that appears at system startup (Figure 9). By designing the system this way, we were able to ensure that each prototype differed from another by just one controlled characteristic, and did not have to worry about possible subtle differences that might have arisen if we had maintained a separate, individual source code for each prototype.



Welcome to PEN-Ivory

© Alex D. Poon,
Keith E. Campbell,
Stanford University

Choose desired interface characteristics:

Encounter Form: ○ Scrolling ● Paging

Listed Findings: ○ Subset ● All

Attributes Palette: ● Fixed ○ Dynamic

Figure 9 A control panel presented at startup allowed selection of the different user-interface characteristics.

Experimental Design

The goal of the experiment was to discover empirically the better design for each of the three interface characteristics. We accordingly conducted an experiment in which we timed users entering progress notes with one of the PEN-Ivory prototypes. Our user group consisted of 15 students from our laboratory, five of whom were medical doctors. The non-physician subjects were all graduate students in medical informatics and were therefore familiar with the clinical environment and substantial medical terminology. We felt it acceptable to use a convenience sample of non-physicians in our study, as our goal was not to test domain-specific aspects of design such as vocabulary organization and work flow, but rather to concentrate on the lower-level, domain-independent aspects of human–computer interaction, such as menu layout and navigation controls. We divided the physician users in our study evenly across the prototype groups, but found, in the end, that there was little performance difference between the physicians and the non-physicians in our study.

We faced four major decisions in designing the experiment: which of the eight prototypes to test, the number of prototypes each user would test, the number of cases each user would enter, and the nature of the stimulus material used to present the cases.

Prototypes to Test

Because there were three different interface characteristics to study, each with two possible designs, there was a total of eight PEN-Ivory prototypes to test in order to conduct a completely crossed, 2 × 2 × 2 factorial design. However, in order to reduce the number of prototypes to test, we decided to conduct a frac-

Table 1 ■

Average Times and Standard Deviations for Entering a Finding Using Each of the Five PEN-Ivory Prototypes Tested

Prototype	Time per Finding Average (sec)	Standard Deviation (sec)
PFA (paging, fixed, all)	19.85	13.87
SFA (scrolling, fixed, all)	24.24	18.70
PFS (paging, fixed, subset)	28.87	24.75
SFS (scrolling, fixed, subset)	26.90	25.81
PDA (paging, dynamic, all)	22.34	17.03

tional factorial design, in which we studied only five of the possible eight prototypes. The disadvantage of using a fractional rather than a full factorial design is that the former does not allow for testing all possible interactions among the three interface characteristics. However, we had no reason to believe that there would be any interactions with the palette-type interface characteristic, as the choice of palette type intuitively seemed orthogonal to the choices of form type and completeness. Therefore, we designed the fractional factorial design so that we could study the potential interaction between form type and completeness, while sacrificing the ability to test the interactions involving palette type. The five prototypes we included in our study were:

1. PFA (paging, fixed palette, all findings)
2. SFA (scrolling, fixed palette, all findings)
3. PFS (paging, fixed palette, subset of findings)
4. SFS (scrolling, fixed palette, subset of findings)
5. PDA (paging, dynamic palette, all findings)

Again, notice that because the five prototypes we tested included every combination of form type and completeness, the design allowed us to test for an interaction between form type and completeness. In contrast, this set of five prototypes did not allow us to test interactions involving palette type, as we assumed that such interactions did not exist.

Number of Prototypes per User

We chose to have each user test exactly one of the five prototypes, as opposed to having each user test multiple designs. We felt that this would not only allow each user to become more familiar with the workings of a single prototype, but also simplify the study design, in that it avoided the need to consider cross-over effects that use of one prototype might have on use of another. Therefore, each of the five prototypes was tested by three of the 15 subjects.

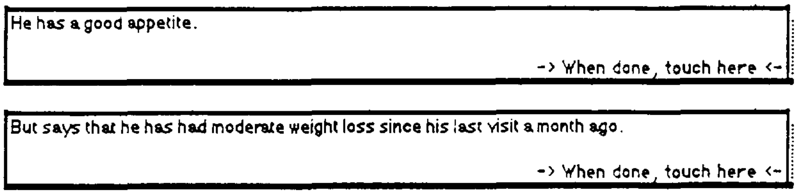
Stimulus Material

The stimulus material is the method used to present the cases. The most realistic approach would have been to have each user interview actual patients in a clinical setting. However, we felt that we needed a more controlled environment in which we could ensure that all users entered the same cases. Our goal in selecting an appropriate stimulus material was to choose the simplest way to present the cases without artificially favoring one prototype over another. We chose to use an automated method that was integrated into the user interface itself. Findings from patient cases created by a physician familiar with the PEN-Ivory vocabulary were presented one by one in a message box on the computer tablet. Many of the findings were intentionally phrased in generic text that did not necessarily correspond to the precise terminology used on the encounter forms, thereby forcing subjects to find alternate ways of expressing the concepts. Users were instructed to touch a button on the screen when they were finished entering the current finding along with its attributes and ready to proceed to the next one (Figure 10). This method allowed us not only to automate the collection of time data points, but also to measure elapsed times per finding.

The same presentation method was used to guide the participants through a 15-minute tutorial before each session. The tutorial guided users in entering 11 finding terms along with their attribute terms. However, for the tutorial, a participant was not allowed to continue until the system detected that the participant had completed the task correctly. By using the automated approach, we were able to ensure that each participant received the same level of training. We also took advantage of the tutorial as a chance to instill into the participants' memories the locations of particular finding terms that we knew would later be presented in the actual cases, thereby helping us to test the effects of positional constancy of items on user performance.

Number of Cases per User

We created three patient cases for each subject to enter. Each case had on average 21 findings, and therefore each user entered 63 findings. The first case presented a patient complaining of diarrhea; the second an HIV-infected patient with a history of CMV retinitis; and the third a patient with severe headaches.


He has a good appetite.

-> When done, touch here <-

But says that he has had moderate weight loss since his last visit a month ago.

-> When done, touch here <-

Figure 10 Patient information was presented to the participants one by one in a message box. Participants touched the bottom right corner of the box to tell PEN-Ivory to continue. This figure shows two separate instances of instructions.

Experimental Results

We describe two sets of analyses. The first, represented by Table 1, shows the average times for entering a finding using each of the five PEN-Ivory prototypes in the study. The times include both the time to select the basic finding term and the time to choose appropriate modifier terms from the attribute palette. The average times were calculated by first computing averages across the three users of a particular prototype for each individual finding (resulting in 63 averages per prototype), and then taking the averages of the individual finding-time averages. Therefore, the standard deviations of Table 1 reflect the variability in time for entering the different findings using a given prototype. The fastest prototype, PFA, used a paged encounter form, had a fixed attribute palette, and made available all findings. Notice that, in general, those prototypes that used a paging encounter form and showed all findings performed the best, while those that showed only a subset of findings performed the worst.

A second, more detailed analysis of variance is shown in Tables 2–6. We used multifactored ANOVA with type 1 sum of squares using the SAS statistical analysis package's GLM (general linear models) procedure. The unit of analysis was the time to enter each case finding, and the four factors in the ANOVA were the form type (scrolling or paging), the palette type (fixed or dynamic), the completeness of the form (all or a subset of findings), and the finding number entered (1–63).

Table 2 shows that of the four factors, finding number and completeness were the largest sources of variance, at the $p < 0.0001$ and $p < 0.0009$ levels of significance, respectively. It is not surprising that the finding number was a large source of variance, as we designed the patient cases to consist of a wide variety of patient findings with varying degrees of difficulty with respect to entry time. For instance, the finding "fever" was consistently located by users more quickly than "leukoplakia," as "fever" had been used in the tutorial phase of the experimental session.

A more interesting discovery is completeness' large contribution to variance, indicating that showing all findings from the PEN-Ivory vocabulary was a much more efficient design than showing only a subset of findings. We were surprised to find that completeness' effect on user performances was larger than both form type's and palette type's effects. We noticed that the main reason for the slowness of prototypes that

Table 2 ■

ANOVA Data Using the Model Time = Form Type + Palette Type + Completeness + Finding # + Form Type*Completeness, Where Form Type Represents Paging vs Scrolling, Palette Type Represents Fixed vs Dynamic Palette, Completeness Represents All vs Subset of Findings, and Finding # Represents Which of the 63 Case Findings the User Entered*

Source	DF	Type I SS	Mean Square	F Value	Pr > F
Form Type	1	705.28	705.28	1.40	0.2363
Palette Type	1	311.98	311.98	0.62	0.4308
Completeness	1	5,588.75	5,588.75	11.13	0.0009
Finding #	62	231,131.86	3,727.93	7.42	0.0001
Form Type*Completeness	1	1,605.20	1,605.20	3.20	0.0742

*Notice the significance of the interaction Form Type*Completeness, demonstrating that Form Type and Completeness are not independent. See Tables 3 and 4 to see Form Type evaluated again with fixed values held for Completeness and Tables 5 and 6 to see Completeness evaluated with fixed values for Form Type.

Table 3 ■

ANOVA Data Using the Model Time = Form Type + Finding #, Where Form Type Represents Paging vs Scrolling and Finding # Represents Which of the 63 Case Findings the User Entered*

Source	DF	Type I SS	Mean Square	F Value	Pr > F
Form Type	1	1,345.35	1,345.35	5.64	0.0180
Finding #	62	111,890.06	1,804.68	7.56	0.0001

*This analysis was run on only those data from which participants used prototypes that showed all findings (Completeness set to "all").

Table 4 ■

ANOVA Data Using the Model `Time = Form Type + Finding #`, Where `Form Type` Represents Paging vs Scrolling and `Finding #` Represents Which of the 63 Case Findings the User Entered*

Source	DF	Type I SS	Mean Square	F Value	Pr > F
Form Type	1	272.761	272.76	0.39	0.5345
Finding #	62	203,431.67	3,281.16	4.65	0.0001

*This analysis was run on only those data from which participants used prototypes that showed all findings (`Completeness` set to "subset").

showed only subsets of findings (PFS and SFS) was that users were often unsure whether a particular target finding was already loaded in as part of the current finding group. Even for those findings that were already displayed on the encounter form as part of the subset, users often *mistakenly* believed that the desired finding was not loaded, and wasted time loading in additional finding groups. In fact, of the 63 findings that users were asked to enter, only seven actually needed to be loaded in. The remaining 56 were already loaded into the encounter form as part of the current subset, yet those using the subset design still spent 36% more time entering those findings. These instances also account for the particularly high standard deviations seen in prototypes PFS and SFS —in several cases, users spent well over a minute trying to load in finding groups. In contrast, those using PFA and SFA never had to spend time loading in additional findings.

Though form type was not by itself a significant source of variance, we can see from Table 2 that it interacted with completeness (p < 0.0742). Tables 3 and 4 are ANOVA tables that show form type's contributions to variance with completeness fixed to "all" and "subset," respectively. In other words, Table 3 shows form type's effect on performance using only data from prototypes that showed all findings, while Table 4 shows its effect on performance using only data from prototypes that showed a subset of findings. As we can see from the tables, form type was a significant source of variance ($p < 0.0180$) when showing all findings, but was not ($p < 0.5345$) when showing only a subset.

In a similar fashion, Tables 5 and 6 separate out completeness' effect on performance when using a paging form from its effect when using a scrolling form. The tables show that when using a paging form, completeness had a significant effect on performance ($p < 0.0001$), but when using a scrolling form, its effect was much less pronounced and was not statistically significant ($p < 0.3335$).

Palette type did not have a statistically significant effect on user performance ($p < 0.4308$). We can compute from Table 1 that the fixed palette was on average only 2.49 seconds faster per finding than the dynamic palette. Furthermore, upon close examination of the time points, we found that much of the 2.49-second difference was due to the time that the system needed to create the dynamic pop-up palettes on the fly. The delay appeared to range from 3/4 of a second to one and a half seconds, and was a result of the window-management routines of the MacApp application framework.

Discussion

The experiments described in the previous section are specific to the context of writing patient progress notes using the PEN-Ivory system. However, one of our key aims is to broaden the context in future studies. To that end, we have chosen to focus our discussion on the task of selecting menu items from large controlled vocabularies, not only because that task has been insufficiently modeled in the past, but also because it is prevalent in a variety of biomedical domains in addition to progress-note generation.

Prior Work on Menu Selection

Many researchers have examined the task of menu selection.[21] Lee and MacGregor created a model for users accessing large videotext systems, and found that search time varies linearly with the number of menu items.[22] An assumption in their model is that users inspect menu items serially, as if reading text. Card proposed that users do not typically perform a serial inspection of items, but rather perform a ran-

Table 5 ■

ANOVA Data Using the Model `Time = Completeness + Finding #`, Where `Completeness` Represents All vs Subset and `Finding #` Represents Which of the 63 Case Findings the User Entered*

Source	DF	Type I SS	Mean Square	F Value	Pr > F
Completeness	1	6,548.55	6,548.55	14.91	0.0001
Finding #	62	124,101.13	2,001.63	4.56	0.0001

*This analysis was run on only those data from which participants used prototypes that used a paging encounter form (`Form Type` set to "paging").

dom inspection with replacement, especially when using command menus.[16] He also found search time to be a linear function of the number of items in the menu. Still others have created more complex mathematical models of menu selection, incorporating the breadth and depth of menu trees into their formulas.[22-25]

Many researchers have studied other aspects of menu selection, including the organization of menu items, positional constancy, and learning effects. In general, they have found that alphabetical and categorical ordering of menu items is preferable to random ordering, and that menu items whose positions remain constant are more easily learned than those that move randomly or even by frequency of use.[16,18,26,27]

Though the background literature is extensive, none of the preceding studies adequately explains the results of our studies. For example, all of the past models predict that search time increases with the number of menu items, either linearly for flat menus or in log fashion for hierarchical menu trees. Our experimental results do not follow the past models. We found that the design that displayed only a subset of the controlled vocabulary rather than all of it (and that therefore displayed many fewer items) produced much slower performance times, even when the displayed subset of terms contained the terms that the user wished to choose.

It is unclear whether past studies involving the positional constancy of menu items can adequately explain the results of our studies, particularly the two studies having to do with the paging vs scrolling forms and the fixed vs dynamic palettes. Whereas the past studies have typically involved menus of 100 items or less, so that users could easily remember the spatial locations of the menu items, the PEN-Ivory vocabulary contained a little over 1,000 basic terms. In fact, MacGregor, Lee, and Lam have argued that for videotext systems in which the number of items may approach 10,000, it is unlikely that users can learn the locations of the items.[28]

Interfaces to Large Controlled Vocabularies of Medicine

Perhaps the reason that our PEN-Ivory results do not conform to models put forth by past researchers is that our experiments involved a large controlled vocabulary for which small, traditional menus are inadequate for navigating to the choices. In past studies, either the number of total choices was sufficiently small to fit all the items into one menu frame, or the items were arranged hierarchically in menu trees so that each menu frame contained a small number of items. Our study involved a large medical vocabulary displayed sequentially either on several pages or on a long scrolling form.

Table 6 ■

ANOVA Data Using the Model `Time = Completeness + Finding #`, Where `Completeness` Represents All vs Subset and `Finding #` Represents Which of the 63 Case Findings the User Entered*

Source	DF	Type I SS	Mean Square	F Value	Pr > F
Completeness	1	595.05	595.05	0.94	0.3335
Finding #	62	124,034.93	2,000.56	3.15	0.0001

*This analysis was run on only those data from which participants used prototypes that used a scrolling encounter form (`Form Type` set to "scrolling").

Another fundamental difference between a controlled medical vocabulary and the kinds of items displayed in typical menuing systems is that an interface to a large controlled medical vocabulary is unlikely to contain all of the appropriate terms and concepts that a user might want to express, either because the vocabulary itself does not contain the concept or term or because the system can make available only a manageable subset of the vocabulary at once. On the other hand, the menus used in previous studies have typically been either command menus or artificially created menus that contain all the items that a user might wish to choose. Therefore, we believe that user uncertainty (see below), while playing perhaps only a small role in the prior studies of small command menus, contributes significantly to the performance of any system that requires users to select items from large controlled vocabularies, such as those encountered in medical applications.

Because prior work by others has not focused on selecting items from large controlled vocabularies, the models and guidelines resulting from that work cannot be easily applied to interfaces designed to deal with the selection of items from large controlled vocabularies. Data-entry tasks such as structured progress-note writing,[3,8,14,29-37] entry of ICD-9 and CPT codes for automated billing, and input of findings to medical decision-support systems[38] such as QMR (Quick Medical Reference)[39,40] are all examples of tasks that require users to select terms from large controlled vocabularies. Many data-retrieval tasks, such as selecting MeSH (Medical Subject Headings) terms for searching using the Grateful MED interface,[41] also involve the selection of terms from large controlled vocabularies. It is clear that the task is prevalent in medical computing systems—what is unclear is how to design interfaces for such systems so that the task can be completed with optimal ease and efficiency.

Areas for Future Exploration

As stated earlier, one of the key goals of our work is to develop general design guidelines for systems that involve the selection of items from large controlled vocabularies. Though the data presented in this paper show clearly that a paging form, a fixed palette, and showing all finding terms at once were more efficient than their counterparts, we are left to our intuitions to explain the reasons behind the differences. Accordingly, we suggest a set of psychological* and mechanical factors that we believe may have caused the differences we found among the PEN-Ivory prototypes we studied.

Psychological Factors

- *Positional Constancy.* The paging form enabled users to memorize over time the locations of findings by both their page numbers and their positions on the pages. For instance, a user might have remembered that the term "vomiting" was always listed at the bottom of page S2. In the scrolling design, the screen locations of findings changed depending on the distance the user had scrolled, and therefore did not allow the user to take good advantage of spatial memory.

  In the all-findings (AF) design, because all findings in the database were always shown, the position of each finding on the form was constant, regardless of which problem or organ-system groups were loaded. In contrast, the locations of findings in the subset (SS) design changed depending on the finding groups loaded.

- *User uncertainty.* In the AF design, if a desired finding was not listed on the form, the user knew that the finding did not exist in the database. In the SS design, if a desired finding was not listed on the form, the user could not easily know whether the finding was simply not part of the loaded finding groups, or if it did not exist in the controlled vocabulary at all. We noticed that even for those findings that were already loaded into the system, users who could not immediately locate a finding would often wrongly assume that the finding was not loaded and then waste efforts trying to load it.

  Notice that the level of user uncertainty is likely to be highly influenced by the nature of the subset used in the SS design. In our study, the subsets were based on problem groups hand-crafted by physicians related to the project. Had users been able to customize the problem groups themselves, it is likely that the level of user uncertainty would have been significantly lower, and perhaps the SS design would have fared better. One of the main goals of our future work will be to test how different levels of user uncertainty and familiarity with the vocabulary affect user performance.

- *System anticipation.* The AF design and the SS design used different implementations for anticipating which finding terms would most likely be needed by the user to describe the patient. The SS design made available only those terms that were related to the patient problem list, while the AF design made available all terms in the vocabulary and merely highlighted those terms related to the patient problem list. It is likely that the degree of system anticipation interacted tightly with the degree of user uncertainty. As the system more boldly anticipated the terms that users might choose, users may have felt less certain about whether a term was being hidden by the system.

Mechanical Factors

- *Page indexing.* With the paging design, users were able to move among the pages of findings in a random-access fashion by touching page tabs that served as indices into the notebook of pages. The scrolling design, on the other hand, had no such indices, as it used a standard Macintosh-style scroll bar for navigation.

- *Physics of page tabs vs scroll bars.* The page tabs of the paging design might have been physically easier to operate than the scroll bar of the scrolling form design, especially with the electronic stylus that is used by PEN-Ivory in lieu of a mouse.

- *Palette delays.* With the dynamic palette, the user had to wait up to one and a half seconds for the palette to pop up before being able to select attributes. The fixed palette, on the other hand, was constantly displayed and thus the user needed to

*We use the term "psychological" not to refer to personality characteristics of individual users, but rather to describe cognitive factors that affect the performances of users. Our usage is consistent with that of other writings in human-computer interaction[21,42] and does not imply that we intend to study variations among users (i.e., *user modeling*)—additional important issues, but ones that are outside the scope of our current work.

wait only about half a second for the attributes to appear in the palette before being able to select them. This allowed immediate anticipatory focus on the portion of the palette of interest.

- *Palette dismissal.* After the dynamic palette popped up, the user had to dismiss the palette explicitly by touching on the "OK" button. This was less efficient, especially in the cases when the user did not wish to add attributes and wanted to ignore the palette. With the fixed design, the user could easily ignore the palette, as it never had to be dismissed.
- *Loading additional finding terms.* In the AF design, because all of the terms in the database were always shown, there was never a need for the user to load in more terms manually. In the SS design, if a desired finding was not listed on the form, the user had to load in the finding term by loading in a corresponding finding group using a pull-down menu.

Future Research Directions

We intend to test those factors that not only have significant effects on performance, but also are more generalizable to other settings; For these reasons, we will focus on the psychological rather than the mechanical factors. Though mechanical factors are a necessary and important aspect of a user interface, we feel that it is the psychological factors in particular that distinguish the task of selecting terms out of a large medical vocabulary from other menu-selection tasks that have been studied in the past. Furthermore, we believe that the psychological factors have a more significant effect on user performance than do the mechanical factors, especially in light of the rapid evolution of computing devices with respect to computational power, mobility, connectivity, and modes of interaction.

Limitations of the Study

As stated earlier, our results are specific to the context of writing patient progress notes with the PEN-Ivory system, and not necessarily generalizable to other settings. However, there are several other limitations of our study that affect its generalizability, namely the use of first-time users only, the artificial conditions in which we conducted our experiments, and the use of data-entry time as the sole evaluation metric.

Novice Users

Our studies involved only novice users of PEN-Ivory, and therefore the conclusions drawn from the results are not necessarily applicable to individuals with more experience using the system.[43] It is possible that the optimal settings for the interface characteristics may vary depending on the level of the user. For instance, novice users may prefer to have all of the choices shown in the encounter form, whereas expert users may prefer to see only a small number of choices that are tailored to their needs. We chose to study only novice users because we felt that it would be unrealistic to identify volunteers who could spend sufficient time with the prototypes to become expert users. Furthermore, one of the main hindrances to the use of clinical data entry systems is the difficulty in attracting new users to the systems by allowing them to be efficient and comfortable with little or no training. Thus, an initial focus on novice users seemed appropriate for our purposes.

Artificial Experimental Conditions

Meister[44] draws a clear distinction between what he terms *system-effectiveness testing* and traditional *controlled experimental research.* We describe work that falls more appropriately into the latter category. System-effectiveness testing involves experiments that attempt to mimic closely the conditions in which the tested system would eventually be used, while sacrificing the ability to run a tightly controlled experiment. System-effectiveness testing usually involves the collection of data on the macro level (minutes). On the other hand, controlled laboratory research aims to collect experimental data from a highly controlled environment, thereby enabling the collection of highly accurate data at the micro level (seconds). The goal of our research was to determine which combination of three interface characteristics allowed the most efficient entry of medical terms. Because we anticipated that efficiency differences among the different prototypes would be on the micro level, we decided early on to conduct our tests in a highly controlled environment.

Though using a controlled, experimental setting for our tests allowed use to gather precise data with minimal disturbance from external factors, the artificial setting potentially reduces the external validity of our results.[43] As stated earlier, the stimulus material consisted of short sentences describing medical findings presented in a message box on the tablet and presented to the users one by one. In real clinical settings, physicians generate finding terms on their own, based on an interactive experience with a patient, rather than have them fed to them in a prescribed order. It is possible that the ideal interface characteristics for progress-note writing in a real clinical setting with actual patients would differ from the results of our controlled study. However, there is no a priori reason to assume this is the case, and controlled experiments are a prudent first step in creating and evaluating new tools prior to clinical implementation and testing.

Speed as the Sole Evaluation Metric

We used data-entry time as the metric for evaluating the different user-interface characteristics. However, there are other usability metrics that are also important, such as error rate and subjective satisfaction. We chose not to consider errors in our study, not only because very few errors were made by our users, but also because it was not clear what constituted an error. Occasionally, users selected findings other than the ones that we had intended for them to select, but which nevertheless seemed appropriate. For our future studies, we are considering several ways of identifying errors. For instance, we could enlist a third-party observer to rate the quality of the selections made by the users, or even ask the users themselves to comment on how well the terms they selected matched the concepts they wished to enter. We also chose not to evaluate the interfaces on subjective satisfaction, but rather asked users informally how they felt about the particular interface they tested. We used this information only to help formulate our intuitions, outlined above, for the reasons that particular characteristics fared better than others.

Conclusions

The PEN-Ivory experiments have taught us that even simple design changes to a user interface can make dramatic differences in user performances. Subtle, yet powerful psychological factors such as positional constancy, user uncertainty, and system anticipation appear to contribute significantly to the effectiveness of systems that display menus of items from large controlled vocabularies of medicine.

The authors thank Keith Campbell for creating the Ivory system from which this work was derived. This work would not have been possible without his contribution. They also thank Dr. Jeanette Sison for helping to create the patient cases and problem groups for the PEN-Ivory experiments, and Tom Moran and Terry Winograd for their enlightening advice and continued support. They are also grateful to the anonymous reviewers whose comments on an earlier version of the report led to greatly improved paper. Special thanks to Mike Walker for his generous and invaluable statistics help. And finally, thanks to those who graciously lent their time and effort by participating in the PEN-Ivory experiments.

References ■

1. Dick R, Steen E, eds. The Computer-Based Patient Record: An Essential Technology for Health Care. Washington, DC: Institute of Medicine, National Academy Press, 1991:190.
2. Shortliffe E, Barnett G. Medical data: their acquisition, storage, and use. In: Shortliffe E, Perreault L, eds. Medical Informatics: Computer Applications in Health Care. Reading, MA: Addison-Wesley, 1990:37–69.
3. Greenes R, Barnett G, Klein S, Robbins A, Prior R. Recording, retrieval, and review of medical data by physician–computer interaction. N Engl J Med 1970;282:307–15.
4. Pendergrass H, Greenes R, Barnett G, Poitras J, Pappalardo A, Marble C. An online computer facility for systematized input of radiology reports. Radiology. 1969;92:709–13.
5. O'Dell D. Increasing physician acceptance and use of the computerized ambulatory medical record. Proceedings of the Symposium on Computer Applications in Medical Care. Washington, DC: McGraw Hill, 1991:848–52.
6. Hammond W, Stead W. Adapting TMR for physician/nurse use. Proceedings of the Symposium on Computer Applications in Medical Care. Washington, DC: McGraw Hill, 1991:833–7.
7. Gelman M. The right system for the write reason. Proceedings of the AMIA Spring Congress. San Francisco, CA: American Medical Informatics Association, 1994:83.
8. Urkin J. A computerized medical record with direct data entry for community clinics in Israel. Proceedings of the Symposium on Computer Applications in Medical Care. Washington, DC: McGraw Hill, 1991:838–42.
9. Luff P, Heath C, Greatbatch D. Tasks-in-interaction: paper and screen based documentation in collaborative activity. Proceedings of Computer Supported Cooperative Work, 1992:163–70.
10. Essin D, Lincoln T. Design criteria for event-driven pen-based user interfaces. Proceedings of the AMIA Spring Congress. San Francisco, CA: American Medical Informatics Association, 1994:100.
11. Lussier Y, Maksud M, Desruisseaux B, Yale P, St-Arneault R. A computerized patient record software for direct data entry by physicians using a keyboard-free pen-based portable computer. Proceedings of the Symposium on Computer Applications in Medical Care. Baltimore, MD: McGraw Hill, 1992:261–4.
12. Rich C. Pen computer clinical encounter recorder. Proceedings of the AMIA Spring Congress. San Francisco, CA: American Medical Informatics Association, 1994:62.
13. Swearingen R, Brown T. Requirements and benefits of a successful pen-based data entry system. Proceedings of the AMIA Spring Conference. San Francisco, CA: American Medical Informatics Association, 1994:101.
14. Campbell K, Wieckert K, Fagan L, Musen M. A computer-based tool for generation of progress notes. Proceedings of the Symposium on Computer Applications in Medical Care. Washington, DC: McGraw-Hill, 1993:284–8.
15. Weed L. Medical Records, Medical Education, and Patient Care. Cleveland, OH: The Press of Case Western Reserve University, 1969.
16. Card SK. User perceptual mechanisms in the search of computer command menus. Proceedings of Human Factors in Computer Systems. New York: Association for Computing Machinery, 1982:190–6.
17. Teitelbaum RC, Granada RE. The effects of positional constancy on searching menus for information. Proceedings of Human Factors in Computing Systems. New York: Association for Computing Machinery, 1983:150–3.

18. Somberg BL. A comparison of rule-based positionally constant arrangements of computer menu items. Proceedings of Human Factors in Computing Systems and Graphics Interface. New York: Association for Computing Machinery, 1987:255–60.
19. Mitchell J, Shneiderman B. Dynamic versus static menus: an exploratory comparison. SIGCHI Bulletin 1989;20(4):33–7.
20. Nygren E. Reading documents in intensive care. I: Pattern recognition and encoding of characteristics of the information media. Center for human computer studies (CMD), Tech report number 21/91, University of Uppsala, Uppsala, Sweden, 1991.
21. Norman KL. The Psychology of Menu Selection. Norwood, NJ: Ablex Publishing Company, 1991.
22. Lee E, MacGregor J. Minimizing user search time in menu retrieval systems. Human Factors. 1985;27:157–62.
23. Kiger JI. The depth/breadth trade-off in the design of menu-driven user interfaces. Int J Man–Machine Stud. 1984;20:210–3.
24. Landauer TK, Nachbar DW. Selection from alphabetic and numeric menu trees using a touch screen: breadth, depth, and width. Proceedings of Human Factors in Computing Systems. New York: Association for Computing Machinery, 1985:73–8.
25. Papp KR, Roske-Hofstrand RJ. The optimal number of menu options per panel. Human Factors. 1986;28:377–85.
26. McDonald JE, Stone JD, Liebolt LS. Searching for items in menus: the effect of organization and type of target. Proceedings of the Human Factors Society—27th Annual Meeting. Santa Monica, CA: Human Factors Society. 1983: 834–7.
27. Parkinson SR, Sisson N, Snowberry K. Organization of broad computer menu displays. Int J Man–Machine Stud. 1985;23:689–97.
28. MacGregor J, Lee E, Lam N. Optimizing the structure of database menu indexes: a decision model of menu search. Human Factors. 1986;28:387–99.
29. Bell D, Greenes R. Evaluation of UltraSTAR: performance of a collaborative structured data entry system. Proceedings of the Symposium on Computer Applications in Medical Care. J Am Med Informatics Assoc. 1994;1:216–22.
30. Ginneken A, van der Lei J, Moorman P. Toward unambiguous representation of patient data. Proceedings of the Symposium on Computer Applications in Medical Care. Baltimore, MD: McGraw-Hill, 1992:69–73.
31. Gouveia-Oliveira A, Raposo V, Salgado N, Almeida I, Nobre-Leitao C, Galvao de Melo F. Longitudinal comparative study on the influence of computers on reporting of clinical data. Endoscopy. 1991;23:334–7.
32. Huff S, Pryor A, Tebbs R. Pick from thousands: a collaborative processing model for coded data entry. Proceedings of the Symposium on Computer Applications in Medical Care. Baltimore, MD: McGraw-Hill, 1992:104–8.
33. Kuhn K, Gaus W, Wechsler J, et al. Structured reporting of medical findings: evaluation of a system in gastroenterology. Meth Inform Med. 1992;31:268–74.
34. Naeymi-Rad F, Almeida F, Trace D. IMR-Entry (Intelligent Medical Record Entry). Proceedings of the Symposium on Computer Applications in Medical Care. Baltimore, MD: McGraw-Hill, 1992:783–4.
35. Nowlan W, Rector A, Kay S, et al. PEN&PAD: a doctor's workstation with intelligent data entry and summaries. Proceedings of the Symposium on Computer Applications in Medical Care. Bethesda, Maryland, American Medical Informatics Association, 1989:941–42.
36. Poon AD, Fagan LM. PEN-Ivory: The design and evaluation of a pen-based computer system for structured data entry. Proceedings of the Symposium on Computer Applications in Medical Care. J Am Med Informatics Assoc. 1994;1:447–51.
37. Wheeler P, Simborg D, Gitlin J. The Johns Hopkins radiology reporting system. Radiology. 1976;119:315–9.
38. Shiffman S, Detmer W, Lane C, Fagan L. A continuous-speech interface to a decision support system: I. Techniques to accommodate for misrecognized input. J Am Med Informatics Assoc. 1995;2:36–45.
39. Miller RA, Pople HE, Myers JD. INTERNIST-1: an experimental computer-based diagnostic consultant for general internal medicine. N Engl J Med. 1982;307:468–76.
40. Miller RA, McNeil MA, Challinor SM, Maserie FW, Myers JD. The Internist/Quick Medical Reference project status report. West J Med. 1986;145:816–22.
41. Siegel E, Cummings M, Woodsmall R. Bibliographic-retrieval systems. In: Shortliffe E, Perreault L, eds. Medical Informatics: Computer Applications in Health Care. Reading, MA: Addison-Wesley, 1990:434–65.
42. Card S, Moran T, Newell A. The Psychology of Human-Computer Interaction. Hillsdale, NJ: Lawrence Erlbaum Associates, 1983.
43. Nielsen J. Usability Engineering. Cambridge, MA: AP Professional, 1993.
44. Meister D. System effectiveness testing. In: Salvendy G, ed. Handbook of Human Factors. New York: John Wiley & Sons, 1987:1271–97.