

Date: 15 February 1981

To: Distribution

From: Larry Tesler

Subject: Proposed Addenda to the LISA User Interface Standards, round 2

CONFIDENTIAL

Attached are proposed additions and revisions to the LISA User Interface Standard (Bill Atkinson, 24 September 1980). The January 19 proposals have been revised according to feedback from the January 22 review meeting. In addition, several new changes are proposed.

Text in [brackets] explains how the change fits in to the original document. Text in {braces} explains why the change is needed; often, it just lists applications that require it. "OK" marks changes that were provisionally accepted at the January 22 review meeting.

The terminology used herein is that of the September 20 document. More descriptive and appropriate terminology is under active development. I request that people do not object to the changes solely on the grounds of inconsistent terminology (e.g., folders containing scroll bars)

There will be a meeting on Tuesday, February 24 from 3-4 p.m. in the Taco Room to discuss these proposals.

Distribution:

Bruce Daniels	Wallace Judd
Steve Jobs	Wendell Henry
Phil Spira	Rich Page
Wayne Rosing	Barry Smith
Jeb Eddy	Pat Marriott
Phyllis Cole	Nellie Connors
Andy Averill	Gail Pilkington
LISA Software (1 copy per person)	

2. MOUSE AND CURSOR (to avoid too many shapes) OK

[replace middle paragraph about cursor changes by:]

The cursor may take on different shapes to indicate its current function. To give the user additional warning of what will happen when the button is pressed, additional feedback may be given as long as it is definitely noticeable without being annoying. For example, when the cursor is over a scroll arrow (section 19), the box around the arrow could fill in black while the button was up, and the arrow itself could fill in black while the button was down.

The Terminology Committee (a joint committee of Publications, Marketing, and Software) will specify the cursor shapes as well as other user feedback. The soundness of their choices will be verified by user tests.

[delete last sentence about motion of the mouse with the button up]

3. POINTING; CLICKING; DRAGGING WITH THE MOUSE (clarification) OK

[add:]

The 1/2 second is measured from the release of one click to the press of the next click.

5. MEMORY (correction) OK

A LISA computer will have at least 256 [not 128] Kbytes of main memory, of which 32K is dedicated to the display.

14. THE ACTIVE FOLDER (to be more obvious to user) OK

One folder is distinguished by its title tab being displayed in inverse video [not a thicker outline], and by having two or more [not just two] light gray scroll bars. [rest unchanged]

15. MAKING A FOLDER ACTIVE (to clarify and to agree with section 29) OK

[add:]

When a folder first becomes active, it is brought in "front" of all other folders so that nothing obscures it. Later, however, it may be obscured by the Dialog Box or by inactive folders that appear to report asynchronous events (see Section 29).

## 20. SPLIT VIEWS (to clarify and to support LisaCalc Titles)

[add:]

There is a minimum view size determined by the application; typically about 0.5 inches in each direction.

There is an icon in the scroll bar at every view boundary. In the vertical scroll bar, the icon at the top of the folder is called the "create-split" icon; the icon at the bottom of the folder is called the "grow icon"; any view-boundary icons in between are called "adjust-split" icons. The same terms are used for the analogous icons in the horizontal scroll bar.

The grow icon was described in section 17. When it is manipulated, the application decides how much of the growth or shrinkage to allocate to each pane.

The create-split icon is used to create new splits. A new split is created by dragging from that icon to the coordinate of the desired split. While dragging, feedback is given by showing a line across the folder where the split would occur. When the button is released, the view beside the cursor is split along that line.

The adjust-split icon is used to move or eliminate a view boundary. For example, if there is an adjust-split icon in the vertical scroll bar, then dragging it upwards expands the view below and shrinks the view above. If the moving boundary passes the next higher boundary, then the shrinking view disappears altogether. If the moving boundary continues onward, then it splits another view just as if it were the "create-split" icon. It stops moving at the edge of the folder.

### 20A. PANELLED FOLDERS (LisaCalc; Business Graphics; Browser; Data Entry)

In many applications, the user must deal with several different but strongly related information sets and must move her (or his) attention back and forth between them as frequently as she moves her attention to different parts of the same document. Requiring the information to be in different folders would add steps, time, and confusion to such applications. On the other hand, split views are inappropriate because the information sets are disjoint.

In these cases, a folder may be divided by the application into "panels." Panels can not be created or deleted by manipulations in the scroll bars, but the boundaries between them can be adjusted using adjust-split icons.

All the panels in one folder must be oriented in the same direction. Suppose the panels are stacked one above another. Then the bottom panel is scrolled horizontally by the horizontal scroll bar at the bottom of the folder. Any other panel may be scrolled horizontally (1) by the same scroll bar as the one below it, or (2) by its own scroll bar along its bottom edge, or (3) not at all. In (case 2), the bottom boundary of the

panel is as wide as a scroll bar, and a scroll bar appears in it when the folder is active. In the other cases, lines narrower than a scroll bar are used. In order from widest to narrowest those lines are: (case 3), (case 1), (the line dividing a split view).

There is at most one selection (as defined in section 21) in any folder at a given time. In general, that selection is in one and only one panel. However, it is possible for an application to display the same information in more than one panel such that if it is selected in one it is also selected in the others (this is also true of split views).

An application can allow the user to split a panel into views as described in section 20.

## 28. MAKING MENU CHOICES FROM THE KEYBOARD (to accommodate section 47A)

[add to universals:] APPLE-R: Redo

## 29. THE DIALOG BOX (to unify menu commands with parameters)

[add:]

The dialog box carries out dialog using a form. Here is a made-up example to demonstrate several points. The menu command "Find a document..." brings up a form saying:

Find a document ...  
... labelled [ ]  
... in the [ ] file for [ ]  
... in the [ ] drawer  
... created between [ ] and [ ]

Clear Form Do it Forget it
----------------------------------

The user fills in the form and then clicks "Do it" to make it happen and to get rid of the dialog box. Clicking "Forget it" just gets rid of the dialog box without doing the command.

Note that the name of the menu command is an abbreviation of the text in the dialog box, and that the characters "... " appear in the name. This is a mandatory convention. Note also that the two menu commands "Do it" and "Forget it" are present in the dialog box without further pull down, as is "Clear Form."

The idea here is that there are two kinds of menu commands. Menu commands requiring no parameters have no "... " in their name and execute as soon as the button is released in the menu. Menu commands requiring parameters have a "... " in their name and execute only after the "Do it" is clicked.

"Do it" can be invoked on the keyboard by (???).

The form that comes up can have default field values determined by the application. Often they come from the previous value supplied for the same or a similar parameter. For example, a number-formatting command

called "...decimals" might bring up:

[3] decimal places

where the value 3 is a standard or a previous value. For another example, a set of "Find" commands could be constructed as follows:

Find...  
Change to...  
Change to...& Find...  
Find all...& Change to...

The third command might put up the following form:

Change to [ ] and Find [ ]

The default value of the first field is the last "Change" value entered into either this form or the "Change to ..." form. (Those worried about keyboard invocation in the Word Processor, see section 47A.) To get rid of defaults, use "Clear form."

29. THE DIALOG BOX (to deal with asynchrony in Clock; Electronic Mail) OK

[add:]

The Dialog Box is an extension of the active folder and its menus, and may only be used to communicate with the user in a synchronous fashion. For example, if the user is drawing a picture and a message arrives over the network or an alarm goes off that the user wanted to hear about, notification may not be given in the Dialog Box. It should be given in an inactive folder that appears in front of all other folders including the active folder; a beep should sound as well.

29A. ASSISTED FORM FILL-IN (for Filer)

When filling in a form, the application can choose to help the user by displaying a list of possible values for the field in which the active selection resides. The choices are listed in a panel of the active folder if the form is in a folder, or in a panel that pops down below the dialog box if the form is in the dialog box.

The list may be displayed in multiple-column format to increase the number that can fit on the screen. The list may have a scroll bar to provide access to more choices than can fit in its panel. The list is generally alphabetized.

Clicking one of the entries in the list fills that value into the selected field, replacing the previous value, and moves the selection to the next field in fill-in order.

### 31. TYPING PRINTING CHARACTERS

[add:]

Characters deleted by typing do not go into the Scrap (section 49), but they may be retrieved by the Undo command (section 47).

### 36. CODE (more general and accommodates proposed Apple standard kbd)

[Delete this section and supply instead an "expand" facility in certain applications. Typing a sequence of letters such as "co" and then invoking "expand" in the Edit menu or by APPLE-E will look up the abbreviation "co" in the Abbreviations Folder and replace it by the expansion "company".]

### 38A. ABORTING COMMANDS IN PROGRESS (Word Processor and others)

The "STOP" command is invoked by APPLE-period. It immediately empties the typeshead buffer and notifies the active folder to abort the operation in progress as soon as it can (if it can). This command can be used to terminate a long search or to reduce the damage that will be done by a series of typed-ahead APPLE-key commands. The application should be sure it is reasonably clear to the user what did and did not get done before the abort. A standard way to give such notice is in the Dialog Box.

### 40A. CLEAR (LisaCalc; Data Entry; and others) OK

The CLEAR key on the numeric pad is only meaningful in a field of a form or a cell of a table. Elsewhere, it sounds a beep.

CLEAR blanks the entire field or cell and leaves a caret there.

### 42B. SPACE (clarification) OK

SPACE is a "printing" character (section 31) produced by typing the space bar on the keyboard. Its representation on the display is blank space. It can be selected and operated upon (e.g., deleted) like any character (as can RETURN and TAB in documents).

### 47. UNDO (to help the user predict its effect)

[add:]

The Edit menu shows the Undo command in the form: "Undo XXX" where "XXX" is a hint of what will be undone by it, e.g., "Undo typing", "Undo Cut", "Undo Undo", "Undo CLEAR". If Undo would do nothing, it is just "Undo" and is dimly highlighted.

"Undo Cut" restores the Scrap to its previous state. "Undo typing" restores the characters that were selected before typing began (if any).

47A. REDO (to make section 29 changes palatable to the word processor)

An application may also offer a Redo command. Like Undo, it tells what it will Redo, e.g., "Redo Find" or "Redo Change&Find".

52. DEFINITION OF A "WORD" (was undefined)

For purposes of "backward" (section 39) and word selection (section 23), but not for "word-wrap" at right margins, a document or other text unit is considered a sequence of "words" defined as follows.

A word consists of a series of letters and/or digits. An apostrophe between two successive letters is part of the word. A decimal point or comma between two successive digits is part of the word. A dollar sign preceding a digit is part of the word. A percent sign following a digit is part of the word. Any character that does not qualify for word-membership according to these rules is considered to lie between the words on either side of it.

Example

In the sentence:

My friend's wife said: "In a state without tax-free food, a \$2.00 'burger' would sell for \$2+6%=\$2.12."

the words are:

My	friend's	wife	said
In	a	state	without
tax	free	food	a
\$2.00	burger	would	sell
for	\$2	6%	\$2.12