ASV3 Dial-In Interface Recommendation for the Repository Based Software Engineering (RBSE) Program

RICIS Preface

This research was conducted under auspices of the Research Institute for Computing and Information Systems by SofTech, Inc. Dr. E. T. Dickerson served as RICIS research coordinator.

Funding was provided by the NASA Technology Utilization Program, NASA Headquarters, Code C, through Cooperative Agreement NCC 9-16 between the NASA Johnson Space Center and the University of Houston-Clear Lake. The NASA research coordinator for this activity was Ernest M. Fridge III, Deputy Chief of the Software Technology Branch, Information Technology Division, Information Systems Directorate, NASA/JSC.

ASV3 Dial-In Interface Recommendation

For the Repository Based Software Engineering (RBSE) Program

ADANET-FD-R&T-160-1

March 30, 1992

University of Houston-Clear Lake Subcontract No. 107
NASA/JSC Cooperative Agreement NCC-9-16
RICIS Project No. RB.6

Submitted to:
University of Houston Clear Lake
2700 Bay Area Boulevard
Houston, TX 77058

Prepared by:
SofTech, Inc.
1300 Hercules Drive
Suite 105
Houston, TX 77058

Table of Contents

CUI/Dial-In Interface Design Report

1.0 Introduction

The purpose of this report is to provide insight into the approach and design of the Cooperative User Interface (CUI). The CUI is being developed based on Hypercard technology and will provide the same look and feel as is provided by NASA Electronic Library System (NELS) X-Window interface. The interaction between the user and ASCII-LIB is presented as well as the set of Hypercard Cards which the user will work with.

2.0 The AdaNET Cooperative User Interface (CUI) Design

The AdaNET cooperative user interface (CUI) design consists of:

a) a prototype Hypercard stack incorporating a set of screens (cards) for review and serving as a basis for defining the interface behavior,
b) a description of the CUI behavior, and
c) a preliminary recommendation for an ASCII-Lib command line interface

The provision of a cooperative user interface for dial-in users is an important part of AdaNET service for users with desktop personal computers. The cooperative user interface will provide a graphical user interface (GUI) which simplifies AdaNET interaction yet requires only a low-bandwidth communication link (1200 baud modem) and simplified command-line interpreter on the host system. Without the cooperative user interface, dial-in users (the majority of current AdaNET users) would have available only the command-line interface.

The design effort quickly focused on the use of Hypercard for the Macintosh and Toolbook for the IBM PC and compatibles with Microsoft Windows. The decision was based on cost and schedule constraints, the technical merits of utilizing a rapid prototyping environment and the desire to port the development solution to both the Macintosh and IBM PC systems. An initial investigation of Microphone (a telecommunication package available on both systems) proved unacceptable because of the high end-user cost. Both Hypercard and Toolbook allow for the distribution of the developed CUI at no cost to the end-user. A separate utility called Convert-It is available to aid in the porting of a Hypercard system to Toolbook. Since the utility exists for conversion in this direction only, and also because of our familiarity with Hypercard, the initial development will be in Hypercard.

The basis for layout and interaction in the CUI is defined by the NELS X-Window implementation. This implementation provides an easily duplicated user interface and is the primary definition of NELS functionality. The "back-end" of the CUI is the command-line driver to the host. NELS will provide a command-line interface here referred to as "ASCII-Lib". Our design efforts have determined that within the restrictions of the Hypercard system (as compared to X-Windows) a nearly one-to-one matching of cards to windows is possible. (Hypercard is based on the layout of interactive fields on "cards". These may be considered the similar to interface screens, windows or window-panes defined in other interactive systems). This is based on the ability of Hypercard to model the X-Windows system and on the fact that ASCII-Lib is anticipated to be very closely derived from the X-Windows system. This latter consideration allows for a relatively simple translation of user interactions with cards to ASCII-Lib commands.

The design centers around the definition of four classes of interface cards: "List Browsers", "Forms", "Query Generators", and "Displays". These four classes correspond to the four styles of interaction provided for. List Browsers allow the user to scroll through lists, highlight one or more items as a selection and initiate activity from menu commands. Forms consist of a list of fields, with the distinction that selecting the field brings up a panel for the user to change the contents of the field. Queries are similar to forms but include the interaction with ASCII-Lib in the development and eventual submittal of a user-defined search query. Displays are cards or panels brought up for

display only, with no interaction other than to have them put away when the user has read the information. It is clearly our recommendation that the design be implemented in Hypercard, then ported to Toolbook. While this does not serve all AdaNET users, it will address a reasonable number of users, and the two implementations will be available to serve as a basis for extending the CUI to other users as funding permits.

Attached are three sets of pages providing details to this design. First is a more detailed discussion of CUI cards behavior. This establishes the functionality of programming scripts which implement the behavior for each card. The second attachment is an early proposal for an ASCII-Lib command language. This was developed in an assessment of feasibility for translation of user interaction to ASCII-Lib commands. We assume that ASCII-Lib will be similar to what we propose here as the basis of our confidence in being able to complete the job. The last attachment is the set of prototype Hypercard Cards for review. These cards are a preliminary definition showing how the X-Window interface will map to Hypercard. It is expected that all of these early definitions will change as NELS is revised and ASCII-Lib defined.

3.0 The Bridge Between User Actions and ASCII-Lib Interaction

The CUI interface is based on "cards" which display certain information and embody a specific mechanism for interaction.

The common paradigm for interaction is one of possible selection or input by the user, a menu item selection to initiate interaction with ASCII-Lib, a possible switch of active card, followed by display of ASCII-Lib responses. Each card will have a general display field in which all output from ASCII-Lib is displayed. The capture of output from ASCII-Lib will move from active card to active card as different cards are selected. It is intended that output thus captured will be specifically relevant to the card in which it is displayed.

Just as in a simple terminal interface, at any point in time the user will see any characters being sent to and from ASCII-Lib (including noise on the line). It will appear quite different to the user however, since the display field will shift from one card to another. When activity shifts from one card to another, the previous relevant information from ASCII-Lib will remain. It is important that only an echo of ASCII-Lib commands and information in response is sent in non-verbose mode, without excessive prompting or menu display of commands available.

It is our expectation that ASCII-Lib commands will correspond nearly one-for-one with menu items in the X-Windows version of NELS 1.2.

Card Classes:
- list browsers
- query generators
- forms
- displays

3.1 List Browsers:

In a list browser a list of items for selection has been sent by ASCII-Lib and captured by the CUI in a list browser card. Interaction will consist of selecting an item from the list and selecting a menu command. The response will be to extract from the selected line the number of the item selected (always the first word on the line) and include that in the ASCII-Lib command to be sent. Selecting a list item and initiating a command generally implies shifting command modes, so the CUI will correspondingly activate a different and appropriate card for displaying the information requested.

3.2 Forms:

In a form, a list of fields and their current values will be sent by ASCII-Lib and captured by the CUI in a form card. Forms will behave initially just like lists: the user will select one line and initiate processing to inspect further or change the value. Again the number of the field will be sent to ASCII-Lib to indicate desired processing. However, in form processing a pop-up field will allow for the modification of the desired item and insertion of the new value in the previously displayed list. The pop-up field will extend the interaction of the user with ASCII-Lib to allow for display of a list

of allowable values sent from ASCII-Lib with selection by the user, or to allow for the user to type in the desired new value and have this sent to ASCII-Lib.

- Note that the CUI can handle updating the local display of the form, but that a refresh form command is necessary to allow the user to verify that the updates have been sent and received correctly by ASCII-Lib.

3.3 Queries

Queries can be handled very similarly to forms, the differences being in the display of the query being built and menu items to initiate the search. Each query card will be unique to the type of query being built. Building the query, for example the boolean query, will require the sending of a list of possible extensions for the user to select from, with corresponding selection by the user and pop-up fields for further interaction. Again, a display query command is necessary to allow the user to verify the query being generated.

Appendix A - ASV3 Command Line Interface to "ASCII-Lib" v1.2

Note: To support CUI access, both mute mode (non-verbose) and command catenation are valuable.
Command catenation means multiple commands strung together on one line.

Main Menu: (default super-collection shown)

list [<start> [<end>]]	-- list collections, where <start> and <end> are index integers
description <index>	-- show collection description
objects <index>	-- browse objects from collection
classes <index>	-- browse classes from collection
search ...	-- natural language, pattern match, boolean
view ...	-- by index, by alpha, show nested, show related
previous level	
top level	
next level <index>	
profile ...	-- user customization
application ...	-- mail, accounts

exit, (?) help

.v: View Menu

hierarchy	-- sort collection list by Collection ID (no subcollections shown)
outline	-- sort collection list by Collection ID - show subcollections inline
alphabet	-- sort collection list by alphabet (no subcollections or related items shown)
related	-- sort collection list by Collection ID - show related items inline

exit, (x) return, (?) help

.p: Profile Menu

(o) collection ...	-- prompt for default collection to view
view ...	-- set default view option (leads to above),
(t) strategy ...	-- set search strategy option (c, s, r)
directory ...	-- prompt for directory path name for local copies
(r) script ...	-- prompt for path name to script for copy
print ...	-- prompt for path name to script for print
max list size ...	-- prompt for max list size
(u) cutoff ...	-- prompt for cutoff ?!

(n) save profile
list settings
exit, (x) return, (?) help

```
.o:             Object Browser
   list [<start> [<end>]]          -- display current object list
   description <index>             -- show object description (abstract)
   meta-data  <index>              -- show object meta-data
   object              <index>     -- show object itself
   file request <index>            -- output request (copy, download, media support)

   natural language                -- refine browse list through natural language search
   comparable <index>              -- revise browse list to include objects comparable to
      object <index>

   exit,  (x) return,  (?) help

.c:             Class_Browser
   list [<start> [<end>]]          -- list classes, where <start> and <end> are index
                                      integers
   meta-data <index>               -- show meta-data for class <index>
   objects <index>                 -- enter object browser for class <index>

   exit,  (x) return,  (?) help

.a:             Applications
   ?

..n: ?  Notify (send message)
   notify mode                     -- ...all or single
   to  (user_id)                   -- ...prompt for user_id
   retain                          -- ...prompt for retain until date
   message                         -- ...prompt for message
   send
   exit,  (x) return,  (?) help

..v: ?          View (messages)
   delete
   (n) view
   exit,  (x) return,  (?) help

.s:             Search
   natural language
   boolean index
   index
   exit,  (x) return,  (?) help

..n:            Natural Language Search
   input query
   start
   view query
   clear query
   strategy
```

..b: Boolean Search on Fields

f irst field -- ... select index field and match desired
a nd -- ... select index field and match desired
o r -- ... select index field and match desired
n ot -- ... select index field and match desired
u ndo
(k) clear
v iew query

s t rategy
s tart
e xit, (x) return, (?) help

...f, a, o, n: Field Selection

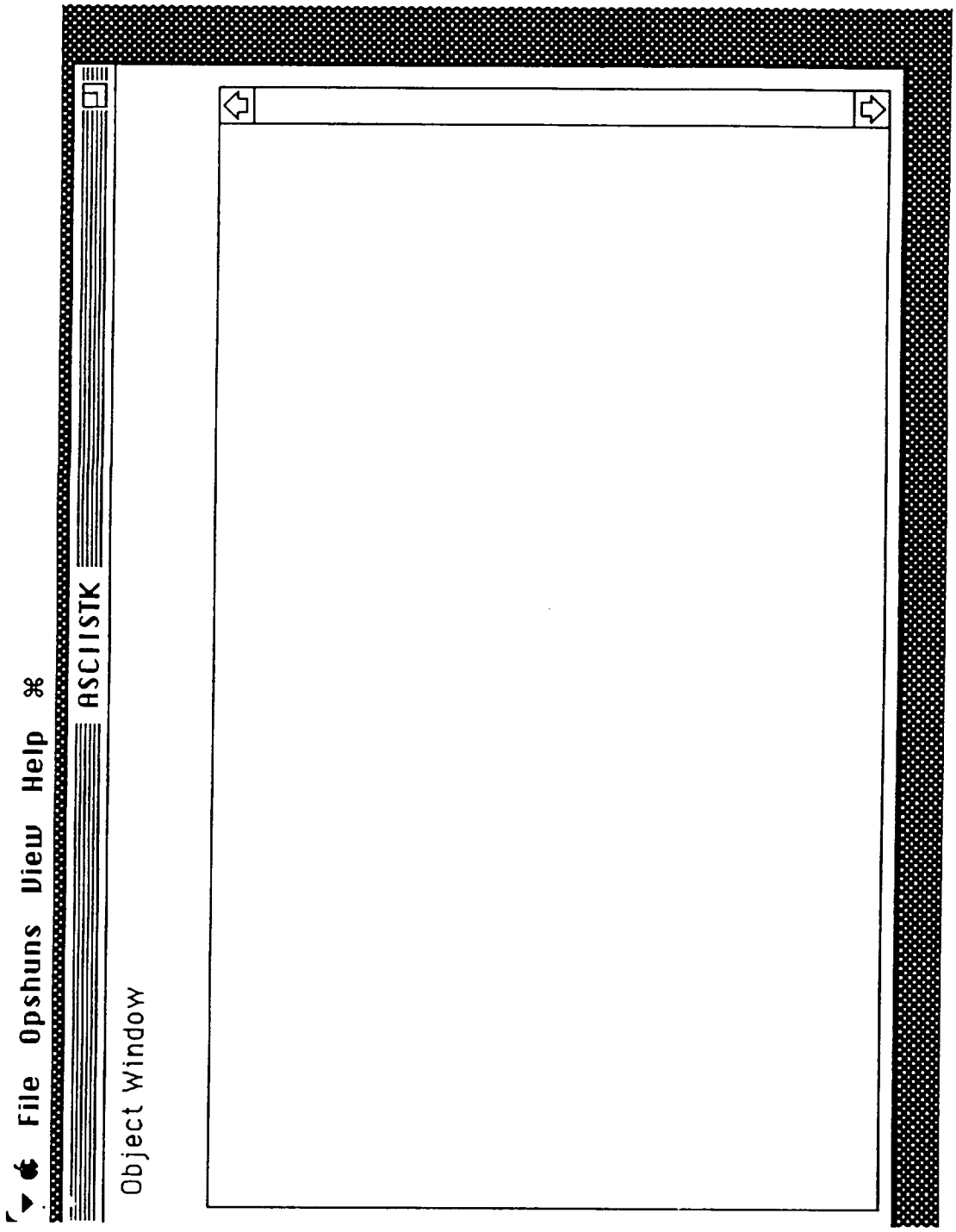
1 -	<undefined>	2 -	Object Name
3 -	Format	4 -	Class Id
5 -	Address	6 -	Version
7 -	Library Entry Date	8 -	Title
9 -	Author	10-	Subject Terms

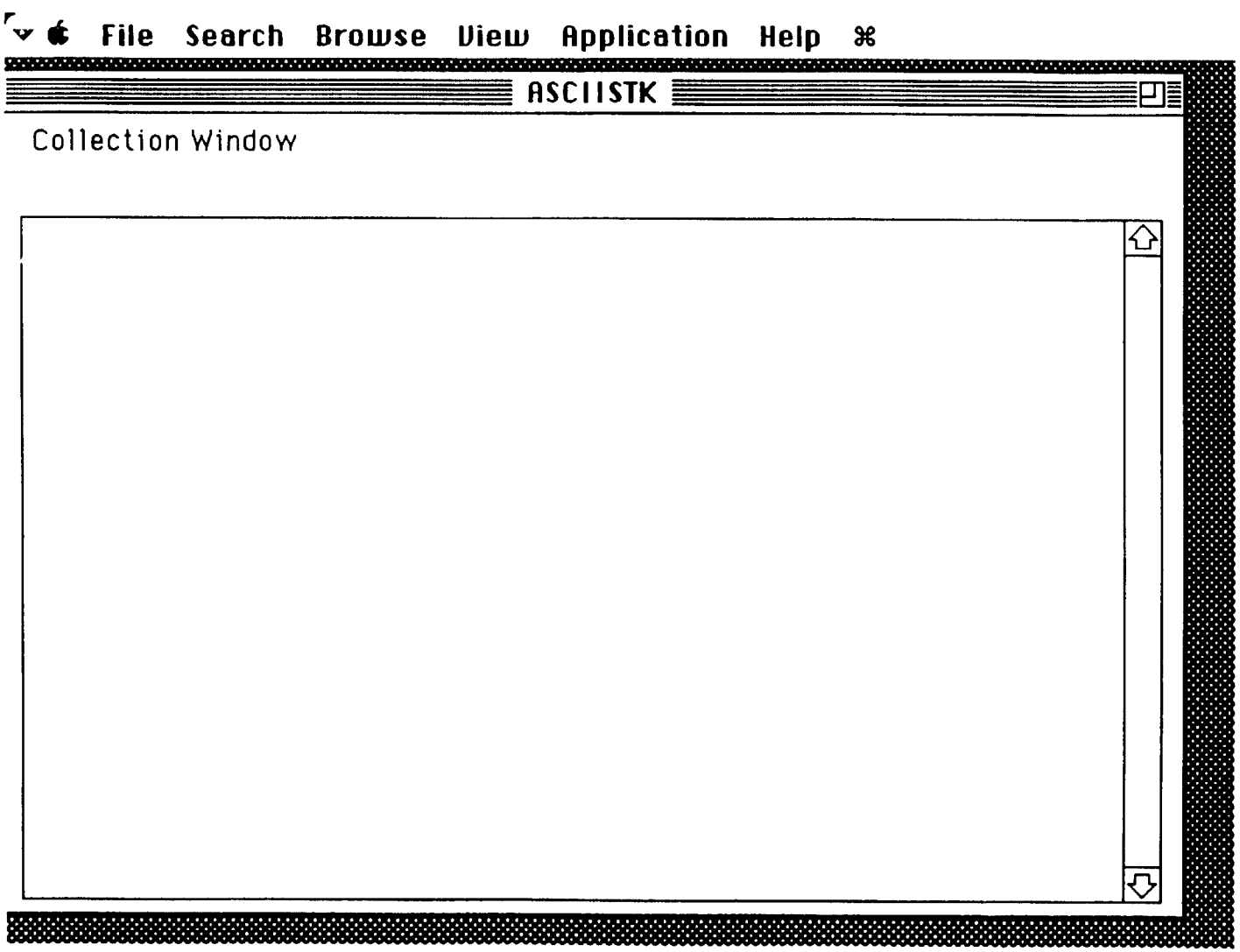
..i: Index Search on Fields

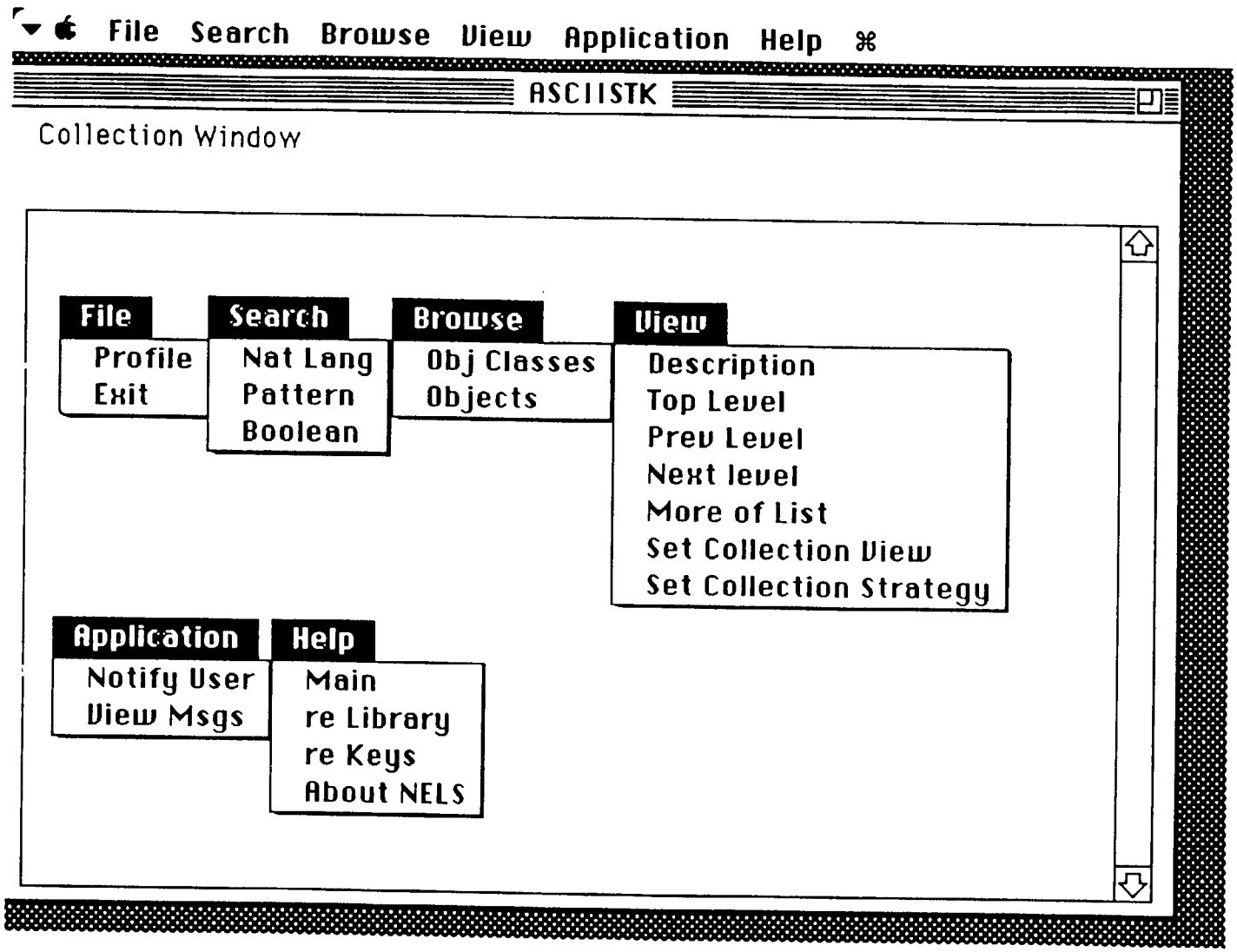
f, u, k, v, s, t -- as for Boolean Search on Fields
e xit, (x) return, (?) help

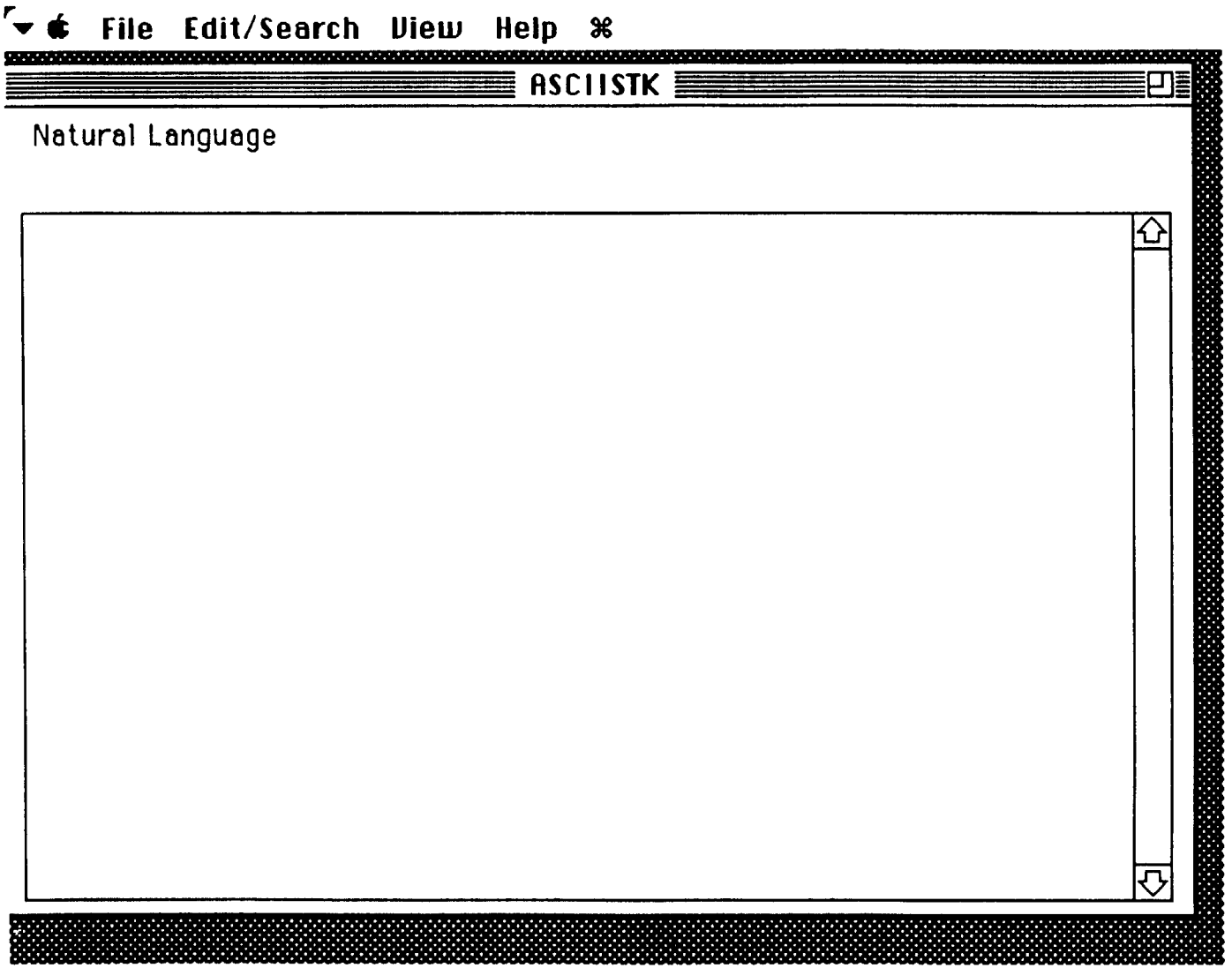
Dots (.) preceding letters indicate the number of menu levels there are above this prompt. Trailing ellipses (...) indicate linkage to sub menus.

Appendix B - Hypercard Cards




File Search Browse View Application Help ⌘
ASCIISTK
Collection Window


File Search Browse View Application Help ⌘
ASCIISTK
Collection Window
File
Profile
Exit
Search
Nat Lang
Pattern
Boolean
Browse
Obj Classes
Objects
View
Description
Top Level
Prev Level
Next level
More of List
Set Collection View
Set Collection Strategy
Application
Notify User
View Msgs
Help
Main
re Library
re Keys
About NELS


File Edit/Search View Help ⌘
ASCIISTK
Natural Language

File View Browse Help ⌘
ASCIISTK
Object Classes

File Edit/Search View Help ⌘
ASCIISTK
Natural Language