



Bell Laboratories

Cover Sheet for Technical Memorandum

1548

The information contained herein is for the use of employees of Bell Laboratories and is not for publication (see GEI 13.9-3)

Title: Description of a Menu Creation and Interpretation System

Date: February 21, 1980

Other Keywords: Menus

Human Factors

State Transition Graph

TM: 80-3166-2

Author(s)
M.J. Heffler

Location
HO 1E-232

Extension
7316

Charging Case: 49408-129
Filing Case: 40324-2

ABSTRACT

Menu systems are useful learning aids for users unfamiliar with a system. This memo describes the Menu Creation and Interpretation System (MCIS) that can provide the user interface and the high-level control structure for any menu-driven software system. The Menu Creation and Interpretation System provides a friendly environment, including extensive help facilities. MCIS is written in the C programming language under the UNIX* operating system. It can be used in any environment that includes a C compiler and the standard I/O library.

MCIS is designed to be application independent. It provides a single tool for both system development and user training thus simplifying these tasks. MCIS's power comes from the modular linking of the menu items and the actions they invoke. The choice of any menu item can either display another menu, execute a program or a subroutine, or display a menu that collects parameters for program or subroutine execution.

Human factors consideration have been stressed as an important part of the system design. The user is able to jump from one menu to another, obtain help whenever it's needed, and easily recover from typing or logical errors. Menu systems created with MCIS are extensible and easily modified, enabling them to meet changing user needs. Other features include interactive menu definition and system customization through the specification of default parameters.

* UNIX is a trademark of Bell Laboratories

Pages Text: 6 Other: 6 Total: 12

No. Figures: 0 No. Tables: 0 No. Refs.: 0

DISTRIBUTION
(REFER GEL 13.9-3)

COMPLETE MEMORANDUM TO	COMPLETE MEMORANDUM TO	COVER SHEET ONLY TO	COVER SHEET ONLY TO
CORRESPONDENCE FILES	GLASER, CHING-MEI S YANG, CHUNG-SHU TATES, GREGORY S ZAWADZKI, J S 5 NAMES	BECKER, RICHARD A BERHMAN, J S BERNCO, DAVID S BERNICK, JEAN BERNITT, RAYMOND W BENNETT, WILLIAM C BERGBE, A A BERGLAND, G D BERK, DONALD A BERNHARDT, RICHARD C BERNSTEIN, DANIELLE B BERNSTEIN, L BEEZER, ALEXANDER B BEIER, JEAN-DAVID BIANCHI, M H BICKFORD, NEIL B BILLINGTON, MAJORIE J BILLOWES, B M BIRN, IRMA B BISHOP, THOMAS P BISHOP, VERONICA L BITTNER, B B BLAKE, GARY D BLAZIER, S D BLEZIER, JOSEF BLEWETT, C DOUGLAS BLINK, J C BLOCCESI, PATRICK A BLUM, MARION BOCKUS, ROBERT J BODER, F J BOEHM, KIM B BOGART, F J BOGART, THOMAS G BOIVIE, RICHARD H BONANNI, L E BORDON, EUGENE P BORNSON, ZELMA A BOTAKOS, MICHAEL A BOURNE, STEPHEN B BOWERS, RICHARD J BOWYER, L RAY BOYCE, W M BOYER, PHYLLIS J BRADLEY, M HELEN BRADLEY, B BRADY, PAUL T BRANDAUER, C M BRANDT, RICHARD B BRAUNZ, DAVID P BREIN, R S BREITHAUP, A R BRIGGS, GLORIA A BROAD, MARTHA M BROCS, JEFFREY D BROWMAN, INNA BROWN, ELLINGTON L BROWN, LAURENCE MC PEE BROWN, M B CICINSKI, STEVEN CICON, J P CHUNG, MICHAEL CHU, MARISA N CHU, PAUL H N CHU, SAI-CHEONG CIEMINSKI, DEBRA F CIRILLO, C CLARK, DAVID L CLARK, EDNA L CLAYTON, D P CLINE, LAUREL M I CLOUTIER, J	COATES, KAREN E COFFIN, STEPHEN COHEN, HARVEY COKE, PETER G <<COLE, LOUISE M COLL, MARILYN O COLLICOTT, R B COLTON, JOHN B CONKLIN, DANIEL L CORNERS, RONALD B COOPER, ARTHUR F COOPER, MICHAEL H COPP, DAVID H COREY, D N COSTELLO, PETER E COTTRELL, JENNIE L COTTINGTON, RALPH L CRAGUN, DONALD W CRAGUN, JOAN CRAIG, JOHN B CRAIG, N E CRANE, B A CRAWFORD, JOHN B CRISTOFOR, EUGENE CRUME, I L CRUPI, JOSEPH A DALBYMPLE, FREDERICK L DAVEY, DOUGLAS A DAVIDSON, CHARLES LEWIS DAVISON, JOSEPH W DAVISON, THOMAS C B DAVIS, C L JR DAVIS, B DREW DE FAZIO, M J DE GRAAF, D A DE TREVILLE, JOHN D DEAN, JEFFREY S DELLNER, W J DENES, P B DENNY, MICHAEL S DENSMORE, SUSAN DEVLIN, SUSAN J DI PIETRO, B S DIB, GILBERT DICKMAN, BERNARD N DIMARCO, ROBERT T DINEEN, THOMAS J DOLATOWSKI, VIRGINIA M DOLOTTA, T A DONNELLY, MARGARET M DONOHUE, D P, 3RD DOUGHERTY, B J DOWDEN, DOUGLAS C DRAKE, LILLIAN DREYFUSS, SYDNEY J DRISCOLL, P J D'ANDREA, LOUISE A DUCHARME, ROBERT LAWRENCE DUDICK, ANTHONY
OFFICIAL FILE COPY PLUS ONE COPY FOR EACH ADDITIONAL FILING CASE REFERENCED	DATE FILE COPY (FORM E-1328)	COVER SHEET ONLY TO	COVER SHEET ONLY TO
10 REFERENCE COPIES	CORRESPONDENCE FILES	4 COPIES PLUS ONE COPY FOR EACH FILING CASE	
ACKACOFF, JOHN M APPULINGAN, SUBRAMANIAM ASENLD, PHYLLIS A ASTHANA, ABBAYA BELLUMONT, ISLAND A BECKER, CURTIS A BEDNAR, JOSEPH J JR BENNELL, RICHARD L BURKE, THERESA M CHANG, CHUNG W COHEN, NEIL B COEDLINE, BARBARA ANN DOMANSKI, BERNARD DUNKIN, PATRICIA ENGLAE, BRUCE WYATT FORSON, SENRY M FRANKLIN, JAMES W GOMEZ, JIRENE HAYDEN, DONALD F, JR HEFFLER, MICHAEL JASCHA, E G JENSEN, CRAIG WILLIAM KAPLAN, FRANK KOBAN, ALAN S LEVIE, IRMELIN G LIU, HUEI-CHI RICHARD MACEI, F P MATTEC, PETER J MAUNSELL, H I MC DONALD, C F MILALOVIC, B F MITCHELL, J C OTT, PETER A PILLA, M A RAMAKRISHNAN, K G REBBERY, J L REFKIN, BWARD M SCHEIBEL, JOHN C SEREDA, BOBAN SILVERBERG, S M SCOLECKI, MARY L STREETER, LYNN A STENSON, JOHN S TAYLOR, GLEN ALAN TEABEE, CSCAS W, III TEEME, FRANCIS E WALKER, PATRICK DEAN			

* NAMED BY AUTHOR > CITED AS REFERENCE < REQUESTED BY READER (NAMES WITHOUT PREFIX
WERE SELECTED USING THE AUTHOR'S SUBJECT OR ORGANIZATIONAL SPECIFICATION AS GIVEN BELOW)

952 TOTAL

MERCURY SPECIFICATION.....

COMPLETE MEMO TO:
3166 3161COVER SHEET TO:
316-SUP

CCCSHE = HUMAN ENGINEERING OF COMPUTER SYSTEMS
CCOLGP = COMPUTING/PROGRAMMING LANGUAGES/GENERAL PURPOSE
COSA = COMPUTING/SPECIAL APPLICATIONS/SURVEY PAPERS ONLY
LSITPT = LIFE SCIENCES/INSTRUCTION, TRAINING/PERFORMANCE, TRAINING AIDS

NO CORRESPONDENCE FILES
HC 1A127TM-80-3166-2
TOTAL PAGES 12

PLEASE SEND A COMPLETE

() MICROFICHE COPY () PAPER COPY
TO THE ADDRESS SHOWN ON THE OTHER SIDE.

TO GET A COMPLETE COPY:

- BE SURE YOUR CORRECT ADDRESS IS GIVEN ON THE OTHER SIDE.
- FOLD THIS SHEET IN HALF WITH THIS SIDE OUT AND STAPLE.
- CIRCLE THE ADDRESS AT RIGHT. USE NO ENVELOPE.
- INDICATE WHETHER MICROFICHE OR PAPER IS DESIRED.



Bell Laboratories

subject: **Description of a Menu Creation and Interpretation System**
Case: **49408-129**
File: **40324-2**

date: **February 21, 1980**
from: **M.J. Heffler**
HO 3166
1E-232 x7316

TM: **80-3166-2**

ABSTRACT

Menu systems are useful learning aids for users unfamiliar with a system. This memo describes the Menu Creation and Interpretation System (MCIS) that can provide the user interface and the high-level control structure for any menu-driven software system. The Menu Creation and Interpretation System provides a friendly environment, including extensive help facilities. MCIS is written in the C programming language under the UNIX* operating system. It can be used in any environment that includes a C compiler and the standard I/O library.

MCIS is designed to be application independent. It provides a single tool for both system development and user training thus simplifying these tasks. MCIS's power comes from the modular linking of the menu items and the actions they invoke. The choice of any menu item can either display another menu, execute a program or a subroutine, or display a menu that collects parameters for program or subroutine execution.

Human factors consideration have been stressed as an important part of the system design. The user is able to jump from one menu to another, obtain help whenever it's needed, and easily recover from typing or logical errors. Menu systems created with MCIS are extensible and easily modified, enabling them to meet changing user needs. Other features include interactive menu definition and system customization through the specification of default parameters.

MEMORANDUM FOR FILE

1. OVERVIEW

1.1 General Description

A menu is text which may provide the user with information, but usually presents several choices. These choices, or menu items, are frequently numbered. The user interacts with the menu by choosing among the items, each of which either provides information or causes some computerized action to occur. The key feature of MCIS is its modular linking of the menu items and the actions they invoke. The choice of any menu item can either display another menu, execute a program or a subroutine, or display a menu that collects parameters that are

* UNIX is a trademark of Bell Laboratories

passed to a program or a subroutine. Given this capability, MCIS can be used to create any menu-driven system, no matter how general or special purpose. It provides a structure upon which a menu-driven system can be built to meet a wide range of user needs.

1.2 Structured, modular design

The separation of the items displayed on the menu and the actions caused by their selection provides the modularity necessary for a very flexible and extensible system. It is highly desirable to have a system capable of adapting to evolutionary changes in its functional requirements and, eventually, to changing user needs. Menu systems created with MCIS are easy to modify. The number of actions needed to modify a system are limited by MCIS's modular design. The menu editor, supplied with the system to aid user's in menu system modification, is described in greater detail below.

1.3 State transition graph model

The menu system structure can be thought of as a network or state transition graph. Each menu corresponds to a node of a graph or a state of an automaton. Transitions are well defined, corresponding to a user's choice and the action it invokes. Each choice brings the user to a new state. Generally, the actions associated with a state are the display of a menu or the execution of a load module. This system is very flexible for the user because each menu (state) is named. An inexperienced user can traverse the menu system to a desired point by a succession of choices. An experienced user can move directly to a desired menu (state) by supplying the system with the menu's name.

1.4 Human factors considerations

Human factors considerations are incorporated in the design in the form of extensive help and error recovery facilities. MCIS enables menu designers to supply help facilities at every step a user takes. Help menus are created like any other general purpose menu. In addition, when a user is prompted for specific data in a parameter collection menu, a special type of menu described below, help is available to the user for each system prompt. When a user makes a typing or a logical error in responding to a general purpose menu or a parameter collection menu it is simple to correct. For example, the parameter collection menus have an option that allows the user to verify his responses and change any of them.

1.5 System customization

MCIS also includes a feature that enables menu designers to customize menu-driven systems to provide various users a unique view of the system. A table containing default values for parameters in parameter collection menus can be provided to the system at run time. This feature is utilized to tailor the system to the limitations and needs of each user or user group.

2. GENERAL DEFINITIONS

There are two types of menus. The first type is a general purpose menu which contains a heading, optional help or descriptive text, menu items and a prompt asking the user to choose a menu item.

Here is an example of a general purpose menu.

THIS IS THE MENU HEADING

This is the optional descriptive text that can describe the menu to the user or provide some general help information.

1. First menu item - Display a menu
2. Second menu item - Run a program
3. Third menu item - Collect parameters

Choose an item number

The second type of menu collects parameters that are passed to a load module. Parameter collection menus are unlike general purpose menus in that the user is not given a list of menu items from which a single choice is made. Parameter collection menus are line-by-line interactive dialogues between the system and the user. The user is prompted for specific information, as in the example below, and either responds with that information or asks for help. A help routine provides information that should enable the user to respond more knowledgeably when repromted for the information. Help routines are described in more detail below.

When the user has responded to the last prompt for data on a parameter collection menu, the responses can be verified and re-entered if the user so desires. An opportunity is then given to re-enter any of the responses.

At present, the parameter collection menus are designed for character mode, as opposed to block mode, terminals. There are no accommodations for tabbing between fields or any of the other features of block mode terminals. The ability to handle block mode terminals is a desirable enhancement which will be addressed in the future.

An example of a parameter collection menu in which a user asks for help, verifies responses and changes one response is given below. The user's responses are in capital letters to distinguish them from the menu system's prompts. The symbol "c/r" stands for the carriage return. This response terminates the "item number - change to" prompting cycle.

THIS IS THE PARAMETER COLLECTION MENU HEADING

Message name? HELP

The current message name is "banana".

BANANA

Directory where message can be found? MY STORAGE

Where would you like to send the message? HELP

Possible destinations include:

Yankee Stadium	Bronx, N.Y.
Eskimo Pies	Circle, Alaska
Home Goods, Inc.	Apple Pie, U.S.A.
ABC Trucking	Bayonne, N.J.

Where would you like to send the message? HOME GOODS, INC.

Do you wish to verify your responses? (y/n) Y

1. Message name? banana
2. Directory where message can be found? my storage
3. Where would you like to send the message? Home Goods, Inc.

Do you wish to change any responses? (y/n) Y

Do you wish to change some or all? (some/all) SOME

Item number? 3

Change to? ESKIMO PIES

Item number? c/r

MCIS consists of three phases: creating and defining menus, using the menu system once it's created, and modifying the menu system to meet any changes in user needs. The first and third phases are used by the menu designer. The second phase is used by the menu user.

3. MENU CREATION

In the creation phase the menu designer defines the menu system structure. An interactive program prompts the menu designer for the contents of the menus. This proceeds from heading to descriptive text to the menu items. For each menu item, the item as it will be displayed is entered along with the action which occurs in response to selection of the menu item. The actions that can occur are:

- a. display another menu.
- b. execute a load module and, after it has completed execution, display another menu.
- c. display a parameter collection menu, pass the parameters to a load module, execute the load module and, after the load module has completed execution, display another menu.

After a load module has completed execution (in cases b and c above) either type of menu can be displayed. An example of a menu creation session and the menus produced by the session is in the appendix to this memo.

4. USING THE MENU SYSTEM

In this phase a user is able to navigate through general purpose and parameter collection menus. A user will usually respond to a prompt from a general purpose menu with an item number. Two other responses are also recognized. The user can request the list of all general purpose menu names or he can supply the name of a general purpose menu, causing that menu to be displayed. Adding this flexibility to the system allows the user to:

1. return to menus he has already seen during this session to correct errors or review the menu system
2. save time descending through the menu hierarchy when a specific known menu is desired
3. jump to menus that contain help information, then return to the original menu or another menu.

The help facilities are tailored to the two types of menus. Help menus are available from general purpose menus and help routines are available from parameter collection menus. A menu designer can provide a hierarchy of general purpose help menus. For example, one help menu, named 'help', could give an overview of the menu system, including the help facilities available, while others could explain specific functions in more detail. For instance, a menu named 'help commands' could explain and assist in the use of the different commands available to the user. In a parameter collection menu a user can get help in responding to the prompt for a specific parameter. Help routines, supplied by the menu designer in the creation phase, are linked to the prompts in parameter collection menus. Unlike help menus, which are context independent and invoked by their unique names, help routines are context dependent. A help routine is known only at the response point, or points, where it is invoked. Consequently, users can invoke help routines in response to prompts in parameter collection menus with the keyword "help".

The flexibility of the help facilities provides a friendly environment that is easily modified to meet changing user needs. These changing needs might require either the contents or the nature of the help routines to be modified. The modularity of the system enables any help routine changes to be incorporated into the system by simply editing the outdated routines and recompiling the system.

The user can supply a table, described in the overview section, which contains defaults for items in parameter collection menus. Static and dynamic defaults are available:

1. A static default suppresses the prompt for the parameter and sends the default value to the load module. For example, a clerk who is constrained to send messages to one company will not be prompted for a message's destination. The name of the company, as specified in the default table, will be passed to the routine that sends the messages.
2. A dynamic default is displayed with the parameter collection prompt. The user can choose to override or retain this default. For example, a clerk who usually sends all messages to one company but occasionally sends messages elsewhere would normally retain the default, overriding it when required.

5. MENU MODIFICATION

Rather than recreate an entire menu system a menu editor will be provided so that the menu designer can make modifications to the system. This allows a user to change, delete and add entire menus or any parts of menus in an existing menu structure. The modification phase works like the creation phase. It prompts for information and will display, change, add or delete any of the information previously created. The menu editor edits the data files that contain menu text, the sequence in which actions are invoked, and the names of the programs and subroutines, including help routines, that the menu system uses. Some editing, such as correcting typing errors in menu text, will not require recompilation of the menu system. Any changes to the programs and subroutines the system uses or any additions or deletions of programs or subroutines requires that the menu system be recompiled.

6. CONCLUSION

This system was written in the C programming language under the UNIX* operating system. The size of the creation phase load module is 6k, the interpretation phase without any user-provided modules is 14k, and the menu editor is approximately 20k.

MCIS was built as a general purpose tool to provide the user interface and high level control structure for any menu-driven software system, including computer-aided instruction systems. It significantly simplifies the task of building menu-driven systems and help facilities as well as making them modular, extensible and easily modifiable.

Acknowledgements

I would like to thank Joe Bednar, Peter Ott, Lynn Streeter, Jerry Spivack, and Marty Welt for their comments and assistance. My fullest thanks to Glen Taylor and Greg Baran for their contributions in time, energy, ideas and inspired editing.



M.J. Heffler

HO-3166-MJH-mjh

Att.
Appendix

APPENDIX

This is an example of a menu system creation session. The session is initiated with the command:

collect aa bb

The keyword "collect" is the command the user will issue to initiate the session. The parameters "aa" and "bb" are names of data files where the data produced by the menu creation session will be stored. These data files, a C-language subroutine, and a command procedure file are the outputs of the menu creation session. The data files and the C-language subroutine are inputs to the menu generation phase which is invoked, after executing the command procedure file, by the command "generate". The "generate" command has to be supplied with the two data files and, optionally, with the default table used to customize the system.

MENU CREATION

You have just initiated a menu system creation session. In this session you will be prompted for :

1. The menu name
2. The type of menu you are creating. The choices are:
 - a. General purpose menu (m)
 - b. Parameter collection menu (p)
3. The menu header
4. Any descriptive text that will be displayed with the menu
5. The menu items displayed on the menu
6. The actions invoked by the choice of each menu item

A menu item can be one of four possible types depending on the action it invokes and the type of menu it is associated with. A menu item on a general purpose menu can either:

1. display another menu (m)
2. run a load module (!)
3. display a parameter collection menu (p)

There are a different series of prompts that the system expects you to respond to depending on the action associated with the general purpose menu item. The fourth type of menu item is associated with items displayed on parameter collection menus. These items are collectors (c). When a menu item is a collector you are prompted for the name of the help routine that will be invoked when a user requests help in responding to the prompt for this menu item.

If you have any further questions about menu creation please consult your user manual.

Menu name?
first menu

Menu type (m or p)? m

Menu header?
FIRST DEMO MENU

Descriptive or supplementary text?

This is a test menu that demonstrates some of the capabilities of the
Menu Creation and Interpretation System.

S

Item?

Display the second demo menu

What type of action will the choice of this item invoke (m,l,p,c) ? m

What is the name of the invoked entity?
second menu

Item?

Run the editor

What type of action will the choice of this item invoke (m,l,p,c) ? l

What is the name of the invoked entity?
ex_edit

What is the type of menu that will be displayed
after the load module has completed its execution (m or p)? m

Menu name? first menu

Item?

Collect parameters to pass to a program

What type of action will the choice of this item invoke (m,l,p,c) ? p

What is the name of the invoked entity?
pcollect

What is the name of the load module to call
after the parameters are collected? runner

What is the type of the menu that will be displayed
after the load module has completed its execution (m or p)? m

Menu name? second menu

Item?

c/r

Menu name?

second menu

Menu type (m or p)? m

Menu header?

THE SECOND DEMO MENU

Descriptive or supplementary text?

c/r

Item?

Display the first menu

What type of action will the choice of this item invoke (m,l,p,c) ? m

What is the name of the invoked entity?
first menu

Item?

Collect parameters for another program

What type of action will the choice of this item invoke (m,l,p,c) ? p

What is the name of the invoked entity?
pcollect

What is the name of the load module to
call after the parameters are collected? run2

What is the type of the menu that will be displayed
after the load module has completed its execution (m or p)? m

Menu name? first menu

Item?

c/r

Menu name?
pcollect

Menu type (m or p)? p

Menu header?

PARAMETER COLLECTION MENU

Descriptive or supplementary text?

This menu collects arguments for the programs runner and run2.
These programs take a file name and a key for their arguments. Runner
encrypts the file using the key, run2 decodes the file using the key.

\$

Item?

File name:

What type of action will the choice of this item invoke (m,l,p,c) ? c

Name of the help routine to execute
if the help is requested for this item? hfile

Item?

Key:

What type of action will the choice of this item invoke (m,l,p,c) ? c

Name of the help routine to execute
if help is requested for this item? hkey

Item?

c/r

Menu name?

help

Menu type (m or p)? m

Menu header?

HELP MENU

Descriptive or supplementary text?

This menu is a help menu that is being supplied with the system to assist the user if he types the menu name "help" in responding to the prompt at the end of one of the general purpose menus.

On this menu a description of the menu system could be provided to answer some of the questions a user might have. This menu could also explain how the help facilities work on the parameter collection menu, pcollect. It might also describe the keyword "list" that will display the names of all the general purpose menus. It might also inform the user that he can respond to the prompt at the end of a general purpose menu with one of those names to display that menu.

A help menu with information more specific to any of the general purpose menus can also be created with this system to provide more specific types of information. This menu might be named "help ??", where the two question marks are replaced by the keyword that warrants the systems assistance.

S

Item?

Display first menu

What type of action will the choice of this item invoke (m,l,p,c) ? m

What is the name of the invoked entity?

first menu

Item?

Display second menu

What type of action will the choice of this item invoke (m,l,p,c) ? m

What is the name of the invoked entity?

second menu

Item?

c/r

Menu name? c/r

These are the menus that were created by the example menu creation session.

FIRST DEMO MENU

This is a test menu that demonstrates some of the capabilities of the Menu Creation and Interpretation System.

1. Display the second demo menu
2. Run the editor
3. Collect parameters to pass to a program

Choose an item number _

THE SECOND DEMO MENU

1. Display the first menu
2. Collect parameters form another program

Choose an item number _

PARAMETER COLLECTION MENU

File name:

Key:

HELP MENU

This menu is a help menu that is being supplied with the system to assist the user if he types the menu name "help" in responding to the prompt at the end of one of the general purpose menus.

On this menu a description of the menu system could be provided to answer some of the questions a user might have. This menu could also explain how the help facilities work on the parameter collection menu, pcollect. It might also describe the keyword "list" that will display the names of all the general purpose menus. It might also inform the user that he can respond to the prompt at the end of a general purpose menu with one of those names to display that menu.

A help menu with information more specific to any of the general purpose menus can also be created with this system to provide more specific types of information. This menu might be named "help ??", where the two question marks are replaced by the keyword that warrants the systems assistance.

1. Display first menu
2. Display second menu

Choose an item number