CNGBOCHS: An Integrated Ingres-Email-Interleaf System for Processing Change Requests Associated with GEMBOCHS, EQ3/6, and other Research Groups

Stephanie A. Daveler
Suzanne R. Lundeen
James W. Johnson

Publication Date: June 1993

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED
DISCLAIMER

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial products, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

Prepared by Yucca Mountain Site Characterization Project (YMP) participants as part of the Civilian Radioactive Waste Management Program. The YMP is managed by the Yucca Mountain Site Characterization Project Office of the U.S. Department of Energy, Las Vegas, Nevada.

Work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract W-7405-Eng-48.
Table of Contents

1.0 Introduction 1

2.0 The Email–CNGREQ Interface 2
   2.1 Initiating a Change Request Using Email 2
   2.2 Evolution of a Change Request Through the CNGBOCHS System 3
       2.2.1 “Review” Status 4
       2.2.2 “Assigned” Status 5
       2.2.3 “Verification” Status 5
       2.2.4 “Completion” Status 6

3.0 The CNGBOCHS–CNGREQ Interface 6
   3.1 CNGBOCHS Access Requirements 6
   3.2 CNGBOCHS Start–up Procedure 6
   3.3 CNGBOCHS Run–time Options 8
       3.3.1 QUERY (R4): Interactive Specification of CNGREQ Queries 9
       3.3.2 GO (R3): Directing Queries to CNGREQ 9
       3.3.3 LIST: Writing Selected Change Requests to an ASCII File 10
       3.3.4 SAVE (R5): Saving Modified or New Requests to CNGREQ 11
       3.3.5 WRITE (R6): Printing Selected Change Requests 11
       3.3.6 NEW (R5): Entering a New Change Request 12
       3.3.7 CLEAR (R4): Clearing the Displayed Change Request Form 12
       3.3.8 HELP (R2): Getting On–line Help 13
       3.3.9 QUIT (R1): Exiting CNGBOCHS 13
   3.4 Use of CNGBOCHS by Local Research Groups 14
       3.4.1 Task Leaders 14
       3.4.2 Technical Reviewers 15
       3.4.3 Programmers 16
       3.4.4 Verifiers 16

4.0 Concluding Remarks 17

APPENDIX A: CNGBOCHS Change–Request Form 18
APPENDIX B: CNGBOCHS Interleaf Document 20
1. Introduction

Over the years, users of the GEMBOCHS thermodynamic database (GEMBOCHS is an acronym for Geologic and Engineering Materials: Bibliography Of Chemical Species) and the EQ3/6 software package have periodically forwarded suggestions for improvement to the GEMBOCHS and EQ3/6 research groups. These correspondences typically report database or software errors that have been discovered during application, solicit the inclusion of additional or improved thermodynamic data, or suggest incorporation of new or alternate algorithms for performing various geochemical calculations. Traditionally, these suggestions have been made in person, over the phone, or via surface mail, and their resolution has often required additional correspondence. Hence, it has been difficult to maintain a complete, readily-accessible database of such transactions.

This difficulty has been eliminated by development of the CNGBOCHS software package (CNGBOCHS is an acronym for ChaNGe Bibliography Of CHemical Species). This system permits GEMBOCHS and EQ3/6 users to submit suggestions for improvement using the standard electronic mail system (hereinafter referred to as email); facilitates timely review, status notification, and resolution of these suggestions; and greatly simplifies the task of maintaining a comprehensive change-request database that can be relationally searched and selectively printed in rapid fashion.

The CNGBOCHS system consists of a dedicated Ingres database, CNGREQ (an acronym for ChaNGe REQuest database), which stores the change requests, review comments, and other information regarding their resolution history; an interactive Ingres application, CNGBOCHS, which provides a form-driven interface between local staff and this database, facilitates printing of the requests as Interleaf documents, and automatically disperses email notification to relevant parties when requests arrive and each time their status is upgraded (from "review" to "assigned" to "verification" to "completion"); and the background CKMAIL utility (CKMAIL is an acronym for ChecK eMAIl), which monitors incoming email messages and files those addressed to CNGBOCHS in the CNGREQ database. CNGBOCHS currently supports the GEMBOCHS and EQ3/6 research groups; additional groups can be readily included. The CNGBOCHS system is maintained together with the GEMBOCHS database and software library on a dedicated sun SPARCstation2 (node s60 of the local sun network), and utilizes Ingres version 6.4.

The purpose of this document is to describe the procedure for submitting email change requests to CNGREQ and to present a succinct summary of the CNGBOCHS--CNGREQ interface: its local access, operation, and run-time options.
2.0 The Email–CNGREQ Interface

The email–CNGREQ interface facilitates use of the standard email system to submit change requests and carry out all subsequent exchange of information between change requesters and those responsible for reviewing and resolving these requests. In this section we describe the procedure for submitting change requests via email, the evolution of such requests through the CNGBOCHS system, and the sequence of email correspondence that tracks this evolution from “review” to “assigned” to “verification” to “completion” status.

2.1 Initiating a Change Request Using Email

Anyone having an email account and access to the “gov” domain can forward a change request in the form of a standard email message to any research group supported by CNGBOCHS (at present, the GEMBOCHS and EQ3/6 groups). The text format of such a message is completely unrestricted; however, the message must be addressed correctly, contain specific keywords in the subject field, and include certain critical information in the text field. Specifically, the message must be addressed to cngbochs@s60.es.llnl.gov, its subject field must be group request, where group must be one of the designated group identifiers (currently, gembochs or eq3/6), and its text must contain the sender's full name, preferred email address (only if different than the return address of the current message), and phone number, the version number of the relevant database and/or software, and a complete description of the request. (Note: If hardcopy attachments (e.g., journal articles) are to accompany the request, this should be noted in the request description, and the hardcopies forwarded to the appropriate party. The presence of these attachments will be noted in CNGREQ, and they will be appended to the filed hardcopy of the change request itself.)

An example illustrating the use of email on a unix system to forward a change request to the GEMBOCHS group is given below. In this and all subsequent examples, em represents the standard email utility, <cr> denotes a carriage return, and all boldface entries are (system) or must be (user) specified exactly as shown.

EXAMPLE

user ........ em <cr>
system ..... To:
user ........ cngbochs@s60.es.llnl.gov <cr>
system ..... Cc:
user ........ <cr>
system ..... Subject:
user ........ gembochs request <cr>
system ..... the default editor is displayed (vi in this example)
2.2 Evolution of a Change Request Through the CNGBOCHS System

When a new change request arrives in the cngbochs mailbox, it is filed in the CNGREQ database, assigned an CR id number, its status is set to "review", and the appropriate task leader is notified of its arrival; this is all done automatically. The complete evolution of any given request through the CNGBOCHS system is schematically illustrated in figure 1.

In this figure, the boldface entries specify current status of the request and the arrows indicate a change of status or review solicitation, both of which trigger automatic email notification of the relevant parties. The status categories are defined as follows.
2.2.1 "Review" Status

As noted above, each incoming change request is initially assigned "review" status, and the appropriate task leader is notified. During this initial stage, the task leader reviews the request and has the option of soliciting additional technical reviews. If this option is selected, the task leader uses CNGBOCHS (see 3.4.1 for procedural details) to enter some initial comments regarding the request and assign additional reviewers from a current list of those available. The assigned reviewers then receive email notification that contains their login name (as currently filed in CNGREQ), the CR id number and description of the change request, and the task leader's initial comments. The technical reviewers examine the request and email their comments to cngbochs@s60.es.llnl.gov, using the procedure exemplified below.

EXAMPLE

```
user ....... em <cr>
system ..... to:
user ....... cngbochs@s60.es.llnl.gov <cr>
system ..... Cc:
user ....... <cr>
system ..... Subject:
user ....... review name group number <cr>
   (name    = your login name, as currently filed in CNGREQ)
   (group   = the change request group)
   (number  = the change request id number)
   NOTE: name, group, and number will be specified in the email message that
   solicits your input. You must respond to the subject prompt exactly as shown
   or your email message will be returned.

system ....: the default editor is displayed
user ....... enter your comments here, then exit the editor
system ....: abort, display, help, include, justify, message, modify, receipt,
            save, send, userinfo, whoinfo:
user ....... send <cr>
system ....: em>
user ....... quit <cr>
```

Alternatively, technical reviewers can use CNGBOCHS to enter their comments directly into CNGREQ (see 3.4.2).

When all reviewers have returned their responses, the task leader is notified automatically, then uses CNGBOCHS (see 3.4.1) to review these responses and classify the request as either "invalid", "error", "new data" or "other". If the request is classified as "invalid", its status is set to "completion"; otherwise, the task leader uses CNGBOCHS to enter an appropriate resolution strategy, assign one or more programmers to implement this strategy, and select a verifier to ensure that it has been
implemented correctly. At this point, the request's status is upgraded to "assigned" and the programmers are notified of their assignment via email.

A change request will revert back to "review" from "assigned" status if any of the assigned programmers encounter insurmountable difficulties while attempting to implement the proposed modifications. In such cases, the task leader has two options: (1) provide further instructions that resolve the problem, then reset the classification of the request (to "error", "new data", or "other"), or (2) reclassify the request as "invalid", which automatically sets its status to "completion".

2.2.2 "Assigned" Status

During "assigned" status, the assigned programmers attempt to implement the resolution strategy proposed by the task leader. After completing this assignment, they use CNGBOCHS (see 3.4.3) to input the status of their work. If the modifications have been entered successfully, they set the programmers' status field for the request to "finished" and enter whatever comments they might have in the programmers' comments field; the request is then automatically set to "verification" status, and the verifier is so-notified via email. If one or more of the programmers run into difficulties that require clarification, they set the programmers' status field to "problem found" and enter explanatory comments in the programmers' comments field; the request automatically reverts to "review" status, and the task leader is so-informed.

A change request reverts back to "assigned" from "verification" status if the assigned verifier discovers a problem with the programmer's implementation. In this case, the programmers have two options: (1) resolve the problem, then reset their status field to "finished", or (2) enter explanatory remarks in their comments field, then solicit advice from the task leader by resetting their status field to "problem found".

2.2.3 "Verification" Status

During "verification" status, the verifier reviews the modifications incorporated by the programmers to ensure their correctness. If this review is successful, the verifier uses CNGBOCHS to activate the "finished" column in the verifier's status table field; the status of the change request is then automatically set to "completion". If the verifier feels that something is incorrect, they activate the "problem found" column and explain the problem in the comments field; the status of the change request then automatically reverts to "assigned", and the programmers are so-informed via email.
2.2.4 "Completion" Status

When a change request achieves "completion" status, it has been resolved; the task leader and requester are so-notified via email.

3.0 The CNGBOCHS–CNGREQ Interface

*CNGBOCHS* is a form-based FORTRAN77-EQUEL program that facilitates direct user interaction with the *CNGREQ* database. *CNGBOCHS* permits those having a login account on node *s60.es.llnl.gov* of the local *Sun* network and appropriate permissions to view, relationally search, list, and print selected change requests, and to update those fields of the requests for which they have write access. In this section, we describe the access requirements and start-up procedure for using *CNGBOCHS*, provide a complete summary and examples of all run-time options, and detail those option sequences most frequently used by local task leaders, technical reviewers, programmers, and verifiers.

3.1 CNGBOCHS Access Requirements

To use *CNGBOCHS* you must have access to a login account on node *s60.es.llnl.gov* of the local *Sun* network, and this account must be designated as a valid *Ingres* user (for help with these matters, see the *GEMBOCHS* database administrator). In addition, it is necessary to have several *unix* environment and shell variables defined appropriately in the *~login* file of this user account. Specifically:

```
set path=( ... must include /ingres/ingres/bin and /usr/local/bin)
setenv ING_HOME /pubusr/ingres
setenv TERM_INGRES sunkwin
setenv II_SYSTEM /ingres
setenv MYPATH ... must include /ingres/ingres/bin and /usr/local/bin
setenv PATH ... must include /ingres/ingres/bin and /usr/local/bin
setenv CNGBOCHS_INTERLEAF_DIR full_path_of_desktop
                        (e.g., s05:/us/johnson/desktop)
```

3.2 CNGBOCHS Start-up Procedure

Presuming that all of the access requirements described above have been met, *CNGBOCHS* can be run by logging onto *s60.es.llnl.gov* from any Shell-Tool (but not Command-Tool!) window and typing "*cngbochs*" at the system prompt. The blank change-request form that appears on the screen is the user interface to the *CNGREQ* database; because of size constraints, only a portion of this form can be displayed at
any one time. The top portion, which appears when one first invokes CNGBOCHS, is illustrated in Figure 2; the complete form is given in Appendix A.

The displayed portion of the form will change as required by movement of the cursor.

The fields that you are allowed to edit (enter, delete, or modify data) depend upon who you are (more specifically, what login account you're using!), what you're trying to do, and when you're trying to do it. Change requesters, task leaders, technical
reviewers, programmers, verifiers, and CNGBOCHS itself all have access to different sets of fields at different times during the evolution of a given change request. Those individuals having general access to specific fields are listed in those fields on the sample change request form given in Appendix A. The basic rule of thumb is that if your responsibilities require you to edit a field, you can; if they don’t, you can’t. If you try to modify the contents of a field for which you lack permission, CNGBOCHS will respond with the message “not a valid command for browse mode”.

Several commonly used editing keys are listed below:

- **delete** ............... deletes character to the left of the cursor
- **<cr>** ................ clears characters from the cursor to the end of the line
- **<cntrl>d** .............. deletes character under the cursor
- **<cntrl>e** ............ toggles from insert to overstrike mode
- **<cntrl>g** ............. prints displayed menu to a standard ASCII file.

Whenever possible, your input data are checked for validity. CNGBOCHS will recognize most attempts to enter invalid data, inform you as to the nature of acceptable data, then prompt you for such data.

At the very bottom of the screen is a single line that consists of single-word commands separated by blanks; this is the menu of CNGBOCHS run-time options. When the character “>” is shown to the right of the option menu, more options exist than can be displayed at one time. To view these additional options, use the function key R11; press it once to move the cursor to the menu bar, then additional times to scroll through the menu bar line by line. Scrolling in this manner, the last line of options is distinguished by the first appearance of the character “<” at the start of the line; additional scrolling now moves in reverse.

Any of these run-time options may be selected from anywhere on the form by pressing the appropriate function key (except for the HELP and LIST options; see below), which is defined parenthetically to the right of the option name on the menu bar. Alternatively, one can press the R11 key once, then type the option name (or minimum number of unique characters) at the “:” prompt (currently, this must be done for the HELP and LIST options). After a menu-bar option has been selected, a new form and associated menu bar will be displayed; these differ depending upon which option has been selected.

### 3.3 CNGBOCHS Run-time Options

In this section, we describe all of the CNGBOCHS run-time options, and provide examples that illustrate their effective use.
3.3.1 QUERY (R4): Interactive Specification of CNGREQ Queries

When CNGBOCHS is first invoked and the blank change request form illustrated in figure 2 is displayed, the program is in QUERY mode. This mode allows the user to define relational queries that can be subsequently directed to the CNGREQ database (using the GO option, as described in 3.3.2). These queries are specified by scrolling through the form (as described above in 3.2), and entering the values of interest for each relevant data field. For fields that contain numeric or calendar data, search values may be prefixed with one of the following comparison operators: "=" (equal to, which is presumed implicitly with the exception of the value "blank", as described below), "!=" (not equal to), "<=" (less than or equal to), "<" (less than), ">" (greater than), and ">=" (greater than or equal to). These operators can also be used in any logical combination; e.g., the first quarter of FY93 could be represented as ">30-sep-1992 and <1-jan-1993". When specifying values for character data, one may use the usual wildcard characters: "*" (matches any character string) and "?" (matches any single character).

If a data field is left blank (no value specified) when defining a query, then this query will match all values for the field except for the blank entries themselves. These blank entries can be matched by specifying "=" in the field; similarly, all non-blank entries can be matched by specifying "!= ".

In the following example, the user defines a query that will match all GEMBOCHS change requests that were submitted during the last quarter of FY92, and have not yet been resolved.

EXAMPLE.

user ....... cngbochs <cr>
system ..... in query mode
user ....... (in the field research group) gembochs
user ....... tabs cursor to the field date received
user ....... >30-jun-1992 and <1-oct-1992
user ....... tabs cursor to the field date completed
user ....... enters "= ", which completes definition of the query

NOTE: This query can be directed to CNGREQ by following the example provided in 3.3.2.

3.3.2 GO (R3): Directing Queries to CNGREQ

Selection of the GO option initiates a search of CNGREQ for all change requests that match the set of data field values currently displayed; these values were specified by the user while the program was in QUERY mode (see 3.3.1). The matching requests are stored in a match queue, the first entry of which is displayed, and a new menu of run-time options is activated: QUIT, HELP, GO, QUERY, SAVE, WRITE, and LIST.
QUERY returns the program to QUERY mode (see 3.3.1), each subsequent selection of GO displays the next request in the match queue, and the other options are described later in 3.3.

If no change requests in CNGREQ match the specified query, the message "no change requests selected [hit return]" is displayed; following the indicated carriage return, the program returns to QUERY mode. Otherwise, one can scroll through the queue of matching requests, displaying each in sequence with additional GOs. When GO is selected while the last request is displayed, the program prints "end of selected change requests [hit return]"; again, CNGBOCHS returns to QUERY mode following the carriage return.

While viewing a given request in the queue, users can tab through the form and enter or modify values within any data field for which they have write access. Within table fields, one can search for character strings by pressing <ctrl>-F (on some machines the alt key also works) and entering the search string after the prompt "Enter string to find: ". If a match is found, the cursor is moved to the matching row, and the displayed portion of the table field scrolled if necessary. If no match is found, the program responds with "Could not find value [hit return]".

EXAMPLE.

system .... in query mode
user ........ has defined a query such as the example given in 3.3.1
user ........ R3 function key <GO>
system .... selects all change requests that match the specified query, stores these requests in a match queue, and displays the first request in this queue
user ........ R3, R3, R3, ... (scrolls through the queue)

NOTE: One can list the contents of this queue to an ASCII file using the LIST option (see 3.3.3), save updates to one or more of the matching requests using the SAVE option (3.3.4), and print selected requests using the WRITE option (3.3.5).

3.3.3 LIST: Writing Selected Change Requests to an ASCII File

When the LIST option is selected, the research group, CR id number, and CR summary fields of all change requests in the queue are written to an ASCII file whose name is specified by the user. (Note that there is no function key associated with this option.)

EXAMPLE

user ........ has used the GO option to generate a non-empty match queue, as exemplified in 3.3.2
system ..... displaying a change request from the match queue
user ........ R11 function key: enter list <cr> <LIST>
system ..... File name to write:
user ........ queue_list
system .... writes the research group, CR id number, and CR summary fields of all change requests in the match queue to the ASCII file queue_list

3.3.4 SAVE (R5): Saving Modified or New Requests to CNGREQ

When the SAVE option is selected, CNGBOCHS writes the change request currently displayed to the CNGREQ database. This request may already exist in CNGREQ (but has been modified during the current CNGBOCHS session) or it may be a new request. If it exists in CNGREQ, then the updated version will overwrite the older one; if the modifications cause the request to change status, all relevant parties will be informed automatically via email. (If one attempts to save an existing request that has not been modified, the system will respond with "No changes were made [hit return]".)

EXAMPLE (saving a modified request)

user ........ has modified the change request currently displayed, which already exists in CNGREQ
user ........ R5 function key <SAVE>
system .... checks the modified request for obvious errors (e.g., data fields left blank that require input) and prompts the user for corrections if necessary
system .... Entering save routine, please wait (writes the displayed change request to CNGREQ, overwriting the older version)

If the change request to be saved is new (see 3.3.6), the procedure is only slightly different than that described above.

EXAMPLE (saving a new request)

user ........ has finished entering a new change request
user ........ R5 function key <SAVE>
system ..... checks the modified request for obvious errors (e.g., data fields left blank that require input) and prompts the user for corrections if necessary
system ..... assigns the new request the next available id number, files the request in CNGREQ, and sends email notification of this submittal to the relevant task leader.
system ..... Request added, generate another one? (y/n)
user ........ responds appropriately

3.3.5 WRITE (R6): Printing Selected Change Requests

When the WRITE option is selected, the request currently displayed is written to an Interleaf-formatted file which will be placed in the present working directory. This file will also be copied to your personal Interleaf desktop directory if the unix environment variable CNGBOCHS_INTERLEAF_DIR is appropriately defined in your .login file; the new file will overwrite a previously existing file of the same name.
EXAMPLE - (where CNGBOCHS_INTERLEAF_DIR is set to s05:/us/johnson/desktop)

- In query mode
- R5 function key <NEW>
- A new change request form is displayed, the menu of run-time options is updated, and current values for the requester name, email address, and phone number are entered automatically if the login account you are using is known to CNGBOCHS. You now have edit access to these three fields as well as to research group, CR description, and db/swr version.

- Modifies or fills in values for the fields requester name, email address, and phone number. Enters values for the field research group, table field CR description (a complete description of the change request), and field db/swr version, if appropriate.

NOTE: This new request can be filed in CNGREQ using the SAVE option (3.3.4).

3.3.6 NEW (R5): Entering a New Change Request

This option allows the user to create a new change request and file it in the CNGREQ database. (Alternatively, new requests can be submitted using the standard email system, as described in 2.1). If this option is selected, a new change request form is displayed, the values of all data fields known to the system are entered automatically (e.g., requester name, email address, and phone number), and a new menu of run-time options appears: QUIT, SAVE, and QUERY. The user then defines the new request by moving through this form and entering values for the appropriate fields (in general, these are the research group, CR description, and db/swr version fields).

EXAMPLE

- R6 function key <WRITE>
- File name to write:
- Writes the change request to filename and places this file in the present working directory
- Copies to s05:/us/johnson/desktop (y/n)?
- Copies filename to the Interleaf desktop directory; the file will appear as a document icon on the desktop within several minutes.

3.3.7 CLEAR (R4): Clearing the Displayed Change Request Form

When the CLEAR option is selected, all field values currently specified on the displayed change request form are cleared; the cursor remains in the position it occupied when CLEAR was selected.
EXAMPLE

system ... in query mode
user .... moves through the form and enters values for several data fields;
tabs to the table field **CR description**.
user .... R4 function key <CLEAR>
system .... all previously specified data-field values are cleared;
the cursor remains in the table field **CR description**

3.3.8 **HELP (R2): Getting On-line Help**

When the HELP option is selected, the menu of query-mode run-time options is
replaced with the **Ingres** help menu, and the cursor is placed at the end of this line.
The **Ingres** help options **WhatToDo**, **Keys**, **Field**, **Help**, and **End**, are summarized
below; the function keys that activate these options are given parenthetically.
(Currently, HELP (or Keys; see below) cannot be accessed using the indicated
function key (R2); to select these options, move the cursor to the menu bar (R11),
then enter the appropriate character string: "h <cr>" or "k <cr>".)

**WhatToDo (R1) ...** displays a summary of each option available on the
**CNGBOCHS** query-mode menu

**Keys (R2) ...............** displays a table of the mapping between the function and control keys,
and their current definition for the **CNGBOCHS** query-mode menu

**Field (R3) .............** displays information about what values are allowed in the field
where the cursor is currently positioned

**Help (F2) .............** displays the information provided in this list

**End (F3) .............** returns you to the **CNGBOCHS** query-mode menu

EXAMPLE

system .... in query mode
user .... R11 function key; enter help <cr> <Help>
system .... displays **Ingres** help menu
user .... F2 function key <Help>
system .... displays the information provided in the list given above
user .... F3 function key <End>
system .... returns to query mode

3.3.9 **QUIT (R1): Exiting CNGBOCHS**

Selecting the QUIT option allows you to exit **CNGBOCHS** at any time. If the change
request currently displayed has been modified, you must confirm the exit, as illustrated
in the following example.
3.4 Use of CNGBOCHS by Local Research Groups

Specific combinations of the run-time options described above are frequently used by task leaders, technical reviewers, programmers, and verifiers associated with local research groups, e.g. GEMBOCHS and EQ3/6. In this section, we provide generic examples of these combinations that track the evolution of mythical change request GEMBOCHS 123.

3.4.1 Task Leaders

The GEMBOCHS task leader has received email notification that a new change request has been received; it has been set to “review” status and assigned the CR id number 123. After reviewing this request, the task leader uses CNGBOCHS to enter some initial comments and solicit an additional technical review. If the requester’s login name is unknown to CNGREQ, the task leader must also transfer the requester’s full name and phone number from the request into the appropriate CNGREQ fields.

Example

TL ...... rlogin s60.es.lnl.gov
TL ...... cngbochs
system .... in query mode with cursor in the field research group
TL ...... gembochs
TL ...... tabs to the field CR id number
TL ...... 123
TL ...... R3 function key <GO>
system .... displays gembochs CR 123
TL ...... (If the requester name field is blank, reads the change request, then moves cursor to the requester name and phone number fields and makes the appropriate entries.) Tabs cursor to the task leader’s CR summary field and enters a brief summary of the change request. Tabs to the task leader’s comments table field and enters some initial comments regarding the request. Tabs to the attachments field and enters “yes” or “none” together with a brief description if necessary. If applicable, tabs to the db/swr version field and enters the appropriate version number(s). Tabs to the assigned reviewers table field and enters the desired technical reviewer (typing “? <cr>” produces a list of available reviewers).

TL ...... R5 function key <SAVE>
system ... Entering save routine, please wait
Sends an email note to the selected reviewer(s); this note includes a description of the change request, the task leader's initial comments, and instructions on how to return the review comments via email.

Several days later, CNGBOCHS has informed the task leader that the solicited technical review has been returned via email (see 2.2.1 for procedural details) and filed in the CNGREQ database. This email note includes the original description of the change request, the task leader's initial comments, and the technical reviewer's remarks. The task leader now uses CNGBOCHS to classify the request, describe an appropriate resolution strategy, assign programmers to implement this plan, and assign a verifier to ensure its accurate implementation.

**EXAMPLE**

```plaintext
TL ....... rlogin s60.es.llnl.gov
TL ....... cngbochs
system ... in query mode with cursor in the field research group
TL ....... gembochs
TL ....... tabs to the field CR id number
TL ....... 123
TL ....... R3 function key <GO>
system ... displays gembochs CR 123
TL ....... Tabs cursor to the task leader's CR classification field and enters an "X" in the appropriate category (e.g., "new data"). Tabs to the task leader's comments table field and enters the resolution strategy in sufficient detail. Tabs to the assigned programmers table field and enters the selected programmers (typing "? <cr>" produces a list of available programmers). Tabs to the assigned verifier field and enters the desired verifier (typing "? <cr>" produces a list of available verifiers).

TL ....... R5 function key <SAVE>
``` system ... Entering save routine, please wait
```plaintext
system ... Updates the status of gembochs CR 123 to “assigned”; sends an email note to the assigned programmers; this note includes a description of the change request and the task leader's comments.

3.4.2 Technical Reviewers

The assigned technical reviewer for GEMBOCHS CR 123 has received email notification of this assignment, and prefers to submit the review via CNGBOCHS, instead of responding through email (see 2.1).

**EXAMPLE**

```plaintext
reviewer .. rlogin s60.es.llnl.gov
reviewer .. cngbochs
system ... in query mode with cursor in the field research group
reviewer .. gembochs
reviewer .. tabs to the field CR id number
```
reviewer .. 123
reviewer .. R3 function key <GO>
system ... displays gembochs CR 123
reviewer .. Tabs cursor to the technical reviewers' comments table field and arrows to the row that has the first two letters of the appropriate login name in the first column; enters review comments.
reviewer .. R5 function key <SAVE>
system ... Entering save routine, please wait
system ... Sends an email note to the relevant task leader; this note includes a description of the change request, the task leader's comments, and the technical reviewer's remarks.

3.4.3 Programmers

Each assigned programmer has received email notification that the status of GEMBOCHS CR 123 has been upgraded to "assigned"; i.e., it has been reviewed and a resolution strategy proposed, which the programmers are responsible for implementing. After completing their work, each programmer uses CNGBOCHS to report the status of the assignment and any relevant comments.

EXAMPLE

pgmr .... rlogin s60.es.llnl.gov
pgmr .... cngbochs
system ... in query mode with cursor in the field research group
pgmr .... gembochs
pgmr .... tabs to the field CR id number
pgmr .... 123
pgmr .... R3 function key <GO>
system ... displays gembochs CR 123
pgmr .... Tabs cursor to the programmers' status table field and arrows to the row that has the first two letters of the appropriate login name in the first column. Tabs to the column finished if the assignment has been completed successfully (or the column problem found if the assignment could not be completed) and enters "X". Tabs to the programmers' comments table field, arrows to the appropriate row, and enters any relevant comments.
pgmr .... R5 function key <SAVE>
system ... Entering save routine, please wait
system ... If all assigned programmers have set the programmers' status table field to "finished", the change request status is set to "verification", and email notification is forwarded to the verifier. If any one of the assigned programmers set the programmer's status table field to "problem found", the change request status reverts to "review", and an email message is sent to the task leader.

3.4.4 Verifiers

The assigned verifier has received email notification that the status of GEMBOCHS CR 123 has been upgraded to "verification"; i.e., it has been reviewed by the task leader and the resulting resolution strategy successfully implemented by the assigned
programmers. After reviewing the modifications incorporated by the programmers, the verifier uses CNGBOCHS to report the status of the review and any relevant comments.

EXAMPLE

verifier ... rlogin s60.es.llnl.gov
verifier ... cngbochs
system ... in query mode with cursor in the field research group
verifier ... gembochs
verifier ... tabs to the field CR id number
verifier ... 123
verifier ... R3 function key <GO>
system ... displays gembochs CR 123
verifier ... Tabs cursor to the verifier's status table field; arrows to the column finished if the assignment has been completed successfully (or the column problem found if the assignment could not be completed) and enters "X". Tabs to the verifier's comments table field and enters any relevant comments.
verifier ... R5 function key <SAVE>
system ... Entering save routine, please wait
system ... If the verifier.status field has been set to "finished", the change request status is set to "completion", and email notification is forwarded to the task leader and change requester; if this field has been set to "problem found", the change request status reverts to "assigned", and an email message is sent to the programmers.

5.0 Concluding Remarks

The CNGBOCHS software package provides an easy-to-use, secure system for documenting the evolution of change requests submitted to GEMBOCHS, EQ3/6, or other research groups from their initial submittal to their final resolution. The package is based on the widespread availability and use of the standard electronic mail system (email), which has been successfully interfaced with a dedicated Ingres relational database and the Interleaf desktop publishing package. This integrated Ingres–Email–Interleaf system permits research groups and their associates to use email for all correspondence associated with the submittal and resolution of change requests, and maintains the complete history of each such request in a relational database (CNGREQ) that can be queried and updated using a form-driven interface (CNGBOCHS) and selectively printed using a built-in interface to Interleaf.
## Appendix A: CNGBOCHS Change-Request Form

### CNGBOCHS version: version #

#### CHANGE REQUEST (CR) SUMMARY

- **research group**: program, requester
- **CR id number**: program
- **date received**: program
- **date completed**: program
- **requester name**: program, requester, task leader
- **email address**: program, requester, task leader
- **phone number**: program, requester, task leader

<table>
<thead>
<tr>
<th>CR description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initially filled by program or requester</td>
</tr>
</tbody>
</table>

**attachments**: requester, task leader

#### TASK LEADER'S REVIEW

- **task leader**: program
- **CR summary**: TASK LEADER
- **db/swr version**: TASK LEADER

<table>
<thead>
<tr>
<th>CR classif'n</th>
<th>invalid: _</th>
<th>error: _</th>
<th>new data: _</th>
<th>other: _</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>CR status</th>
<th>review: _</th>
<th>assigned: _</th>
<th>verify: _</th>
<th>completion: _</th>
</tr>
</thead>
</table>

**task leader's comments**

**TASK LEADER**

### CR species

<table>
<thead>
<tr>
<th>TASK LEADER</th>
<th>TASK LEADER</th>
</tr>
</thead>
</table>

### assigned reviewers

<table>
<thead>
<tr>
<th>TASK LEADER</th>
</tr>
</thead>
</table>

### assigned programmers

<table>
<thead>
<tr>
<th>TASK LEADER</th>
</tr>
</thead>
</table>

### assigned verifier

<table>
<thead>
<tr>
<th>TASK LEADER</th>
</tr>
</thead>
</table>

QUIT(R1) HELP(R2) GO(R3) CLEAR(R4) NEW(R5)
QUIT(R1) HELP(R2) GO(R3) QUERY(R4) SAVE(R5) WRITE(R6) LIST

-18-
## Appendix A: CNGBOCHS Change-Request Form (Continued)

### TECHNICAL REVIEWS

<table>
<thead>
<tr>
<th>name</th>
<th>date notified</th>
<th>date returned</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>id</th>
<th>technical reviewers’ comments</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TASK LEADER, REVIEWER (using login name), or program</td>
</tr>
</tbody>
</table>

### PROGRAMMERS’ STATUS

<table>
<thead>
<tr>
<th>name</th>
<th>date notified</th>
<th>received</th>
<th>finished</th>
<th>prob</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>id</th>
<th>programmers’ comments</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TASK LEADER, or PROGRAMMER (using login name)</td>
</tr>
</tbody>
</table>

### VERIFIER

<table>
<thead>
<tr>
<th>name</th>
<th>date notified</th>
<th>received</th>
<th>finished</th>
<th>prob</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>id</th>
<th>verifier’s comments</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TASK LEADER or VERIFIER (using login name)</td>
</tr>
</tbody>
</table>

QUIT(R1)  HELP(R2)  GO(R3)  CLEAR(R4)  NEW(R5)
QUIT(R1)  HELP(R2)  GO(R3)  QUERY(R4)  SAVE(R5)  WRITE(R6)  LIST
Appendix B: CNGBOCHS Interleaf Document

CNGBOCHS Version: 3.0

CHANGE REQUEST (CR) SUMMARY

Research Group:
CR id number:
Date Received:
Date Completed:

Requester Name:
Email Address:
Phone Number:

CR description:

Attachments:

TASK LEADER'S REVIEW

Task Leader:
CR Summary:
Db|Swr Version:
CR Classif'n:
CR Status:

Task Leader's Comments:

CR Species: Formulas

Assigned Reviewers: Assigned Programmers: Assigned Verifier

-20-
Appendix B: CNGBOCHS Interleaf Document (continued)

**TECHNICAL REVIEWS**

<table>
<thead>
<tr>
<th>Name</th>
<th>Date Notified</th>
<th>Date Returned</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Comments

**PROGRAMMERS' STATUS**

<table>
<thead>
<tr>
<th>Name</th>
<th>Date Notified</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Comments

**VERIFIER'S STATUS**

<table>
<thead>
<tr>
<th>Name</th>
<th>Date Notified</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Comments
END

4/18/94

FILED

DATE