2

Sta # 10 OCT 11 1994	ENGINEERING DATA TRANSMITTAL	Page 1 of 1 1. EDT No 141247
--------------------------------	-------------------------------------	--

2. To: (Receiving Organization) Distribution	3. From: (Originating Organization) Information and Scientific Systems	4. Related EDT No.: N/A
5. Proj./Prog./Dept./Div.: Liquid Effluent Monitoring Information System (LEMIS)	6. Cog. Engr.: R. T. Adams	7. Purchase Order No.: N/A
8. Originator Remarks: For Release		9. Equip./Component No.: N/A
11. Receiver Remarks:		10. System/Bldg./Facility: N/A
		12. Major Assm. Dwg. No.: N/A
		13. Permit/Permit Application No.: N/A
		14. Required Response Date: N/A

15. DATA TRANSMITTED					(F)	(G)	(H)	(I)
(A) Item No.	(B) Document/Drawing No.	(C) Sheet No.	(D) Rev. No.	(E) Title or Description of Data Transmitted	Impact Level	Reason for Transmittal	Originator Disposition	Receiver Disposition
1	WHC-SD-WM-TP-165		0	Liquid Effluent Monitoring Information System (LEMIS) Test Plans Release 1.2	NA	2		

16. KEY			
Impact Level (F)	Reason for Transmittal (G)		Disposition (H) & (I)
1, 2, 3, or 4 (see MRP 5.43)	1. Approval 2. Release 3. Information	4. Review 5. Post-Review 6. Dist. (Receipt Acknow. Required)	1. Approved 2. Approved w/comment 3. Disapproved w/comment 4. Reviewed no/comment 5. Reviewed w/comment 6. Receipt acknowledged

17. SIGNATURE/DISTRIBUTION (See Impact Level for required signatures)										(G)	(H)
Reason	Disp.	(J) Name	(K) Signature	(L) Date	(M) MSIN	(J) Name	(K) Signature	(L) Date	(M) MSIN	Reason	Disp.
2		Cog. Eng. R. T. Adams	<i>R. T. Adams</i>	8/16/94	B1-18						
2		Cog. Mgr. D. A. Guettler	<i>D. A. Guettler</i>	8/25/94	B1-18						

18. Signature of EDT Originator <i>R. T. Adams</i> 8/16/94	19. Authorized Representative Date for Receiving Organization <i>L. L. Lang</i> 9/11/94	20. Cognizant/Project Engineer's Manager <i>D. A. Guettler</i> 8/25/94	21. DOE APPROVAL (if required) Ltr. No. ☐ Approved ☐ Approved w/comments ☐ Disapproved w/comments
---	--	---	--

7400-172-2 (07/91) GEF097

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

BD-7400-172-1 (07/91)

RELEASE AUTHORIZATION

**Document
Number:**

WHC-SD-WM-TP-165 Rev. 0

**Document
Title:**

Liquid Effluent Monitoring Information System
(LEMIS) Test Plans Release 1.2

Release Date:

August 30, 1994

* * * * *

This document was reviewed following the
procedures described in WHC-CM-3-4 and is:

APPROVED FOR PUBLIC RELEASE

* * * * *

WHC Information Release Administration Specialist:

Christine Willingham

C. Willingham
(Signature)

August 30, 1994
(Date)

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

SUPPORTING DOCUMENT

1. Total Pages 376

<p>2. Title</p> <p>Liquid Effluent Monitoring Information System Test Plans Release 1.2</p>	<p>3. Number</p> <p>WHC-SD-WM-TP-165</p>	<p>4. Rev No.</p> <p>0</p>
<p>5. Key Words</p> <p>project DMR liquid effluent</p>	<p>6. Author</p> <p>Name: R. T. Adams</p> <p><i>R. T. Adams</i> Signature</p> <p>Organization/Charge Code 62230/R40630</p>	
<p>7. Abstract</p> <p>The Liquid Effluent Monitoring Information System (LEMIS) is being developed as the organized information repository facility in support of the liquid effluent monitoring requirements of the Tri-Party Agreement. It is necessary to provide an automated repository into which the results from liquid effluent sampling will be placed. This repository must provide for effective retention, review, and retrieval of selected sample data by authorized persons and organizations.</p> <p>This System Architecture document is the aggregation of the DMR P+ methodology project management deliverables. Together they represent a description of the project and its plan through four Releases, corresponding to the definition and prioritization of requirements defined by the user.</p>		
<p>8. PURPOSE AND USE OF DOCUMENT - This document was prepared for use within the U.S. Department of Energy and its contractors. It is to be used only to perform, direct, or integrate work under U.S. Department of Energy contracts. This document is not approved for public release until reviewed.</p> <p>PATENT STATUS - This document is not to be transmitted in advance of patent clearance, is made available in confidence solely for use in performance of work under contracts with the U.S. Department of Energy. This document is not to be published nor its contents otherwise disseminated or used for purposes other than specified above before patent approval for such release or use has been secured, upon request, from the Patent Counsel, U.S. Department of Energy Field Office, Richland, WA.</p> <p>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, nor any of their contractors, subcontractors or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or any third party's use or the results of such use 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 or its contractors or subcontractors. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.</p>		<p>10. RELEASE STAMP</p> <div data-bbox="1024 1266 1437 1500"> <p>OFFICIAL RELEASE</p> <p>BY WHC</p> <p>DATE</p> <p>OCT 11 1994</p> <p>Ita #10</p> </div>
<p>9. Impact Level N/A</p>		

APPROVED FOR
PUBLIC RELEASE
CW
8-20-94

MASTER

878

. This page intentionally left blank.

CONTENTS

1.0	LEMIS		1
1.1	LE0014U1	MAINTAIN QUALIFIER CODE TABLE	1
1.2	LE4100U1	ENTER CONSTITUENT RESULTS	13
2.0	GENERIC		33
2.1	GE1310R1	PERSONS IN SYSTEM REPORT	33
2.2	GE1310U1	PEOPLE RELATED TO ORGS	33
2.3	GE1320R1	ORGANIZATIONS IN THE SYSTEM	51
2.4	GE2110Q1	TRANSACTION PICKLIST	51
2.5	GE2110U1	TRANSACTION MAINTENANCE	52
2.6	GE2120U1	MAINTAIN APPLICATION RECORD	94
2.7	GE2210U1	MAINTAIN ROLE	116
2.8	GE2510R1	TRANSACTION LIST REPORT	141
2.9	GE2520R1	ROLE LIST REPORT	141
2.10	GE2530R1	TRANSACTIONS AUTHORIZED	141
2.11	GE2540R1	USERS ASSIGNED TO A ROLE	141
2.12	GE2610U1	MAINTAIN DATA OWNERSHIP	142
2.13	GE3110U1	FUNCTION KEY HELP MAINTENANCE	156
2.14	GE3220U1	MAINTAIN SYSTEM COMMENTS	182
2.15	GE3241R1	OUTSTANDING CHANGE REQUESTS	230
2.16	GE3310U1	MAINTAIN GENERIC CODES	231
2.17	GE3721R1	SYSTEM ACCESS REPORT	241
2.18	GE3742R1	SYSTEM USE STATISTICS REPORT	241
2.19	GE3820R1	MESSAGES REPORT	241
2.20	GE3910U1	MAINTAIN PRINTER LIST	242
2.21	GE3920U1	MAINTAIN REPORT PARAMETERS	260
2.22	GE3930Q1	GENERATE AND PRINT REPORT	288
2.23	GE3940U1	REPORT FILE MAINTENANCE	307
2.24	GE6011R1	ONLINE DOCUMENTATION SUMMARY REPORT	316
2.25	GE6011U1	MAINTAIN ONLINE DOCUMENTATION	317
2.26	GE6013U1	DOCUMENT TYPE MAINT	348
2.27	GE6015U1	MAINTAIN DOCUMENT START POINT	360
3.0	LIBRARY		371
3.1	LI5051R1	CONSTITUENT/METHOD REPORT	371
3.2	LI5053R1	ANALYTICAL METHODS LISTING	371
3.3	LI5055R1	METHOD/CONSTITUENTS REPORT	371
3.4	LI5056R1	LABORATORY/METHOD REPORT	371

This page intentionally left blank.

LIQUID EFFLUENT MONITORING INFORMATION SYSTEM (LEMIS)
TEST PLANS RELEASE 1.2

1.0 LEMIS

1.1 LE0014U1 MAINTAIN QUALIFIER CODE TABLE

LEMIS Module Test Checklist Report

Name: MAINTAIN QUALIFIER CODE TABLE
Short Name: LE0014U1
Purpose: Maintain the qualifier code table
Initial Test Date: 12/16/92 LC
Verification Date: 12/16/92

	Disp	----- Security -----			
Help	Fkeys	Ins	Del	Query	Upd
()	()	()	()	()	()

Block: TYPE_QUALIFIER_CODE
Title: Qualifier Type

Help	Scroll	Text	Navigate	Ins	Del	Upd	Qry
()	()	()	()	()	()	()	()

Data Element	Datatype	Dis Len	Help	LOV	Val/ Range	Not Null	Ins
QUALIFIER	CHAR	6	()	()	()	()	()
DESCRIPTION	CHAR	30	()	()	()	()	()

Module Description

The transaction displays the table code and the code description. The date added and update person, where present, are not displayed. These fields are set by the software. The operator may add new entries, update existing entries, and may delete any entry as long as it is not being used elsewhere in the database.

Module Help

This module supports the maintenance and retrieval of qualifier codes. This table maintains a list of valid codes that describe the types of qualifiers that may occur.

To store a new qualifier code:

1. Use the down arrow key to move the cursor to a blank line.

If all of the spaces are filled and more qualifier codes are to be added, press the down arrow key to the bottom entry and then continue to press the down arrow key until a blank entry appears. The already entered codes will scroll upward, leaving space for an additional code to be entered.

2. Enter the appropriate qualifier code.
3. Enter a 1-30 character description of the code.
4. To add more codes, press the down arrow key to the next blank space and repeat the entry process.
5. Press the COMMIT key (F10) to store the selected codes in the database.

To update an existing code:

1. Press the down arrow key until the cursor displays the entry to be updated.
2. If the code is to be changed, enter the replacement code over the existing code. CAUTION: Changing a qualifier code will cause the replacement value to be entered everywhere in the database where the original value was used.
3. If the description is to be changed, TAB to the description field and make the change.
4. Press the COMMIT key (F10) to update the selected codes in the database.

To delete an existing code:

1. Press the down arrow key until the cursor is located at the code to be deleted.
2. Press the DELETE key. NOTE: If the code is being used by other data in the database, it cannot be deleted. An error message will be displayed.
3. If no error occurs, press the COMMIT key (F10) to delete the code from the database.

Module Test Plan

TEST PLAN

Module Test Plan

Testing of the transaction is performed at five levels:

1. Developer testing.
2. Secondary testing.
3. System testing.
4. End user testing.
5. Production certification.

As each applicable section of this test plan is completed, the designated individual responsible for the section will sign the plan in the appropriate places. Each check off item will be completed. The status of the transaction will be updated in the data dictionary as the transaction progresses through the development and test steps.

Developer Testing

As each individual transaction is generated, the developer performs tests against the transaction as defined in the detail test plan. Upon successful completion of this testing, the transaction is made available for secondary testing.

- () 1 - Transaction under development, untested.
 - () Transaction added to system and security files.
 - () Specific test plan/test results/test case added.
 - () Test plan narrative script loaded.
- () 2 - Development complete, initial unit testing in progress.
 - () Help added.
 - () LOV tested.
- () 3 - Initial unit testing complete.

Secondary Testing

Secondary testing is performed by a person external to the development team. Tests are performed using a standardized test checklist. Initial testing is performed to ensure that the transaction meets the functional requirements of the transaction and satisfies the test plan narrative that follows.

Test status conditions are:

- () 4 - Secondary testing in progress.
- () 5 - Secondary testing complete, problems found.
Tester _____ Date Complete _____
- () 6 - Problem resolution in progress.
- () 7 - Problem resolution complete.
Developer Date Complete 12/16/92
- () 8 - Secondary testing complete, no problems found.

When problems are found:

Module Test Plan

1. The tester passes the checklist to the project manager for review and action.
2. The project manager assigns the checklist to the developer of the transaction for correction of the problems.
3. The developer corrects the described problems, records the corrective actions taken and the number of hours of corrective effort used, tests the transaction for correct operation, and returns the checklist to the project manager.
4. The project manager returns the checklist to the tester for retesting.

The above sequence of events repeats until the tester signs off the transaction as passing the test requirements.

System Testing

System testing is performed by a second independent tester. This tester repeats the test plan scenario, the test cases, and any other tests desired under end-user operating conditions.

- () 9 - System testing in progress.
- () 10 - System testing complete, problems found.
Tester _____ Date Complete _____
- () 11 - Problem resolution in progress.
- () 12 - Problem resolution complete.
Developer _____ Date Complete _____
- () 13 - System testing complete, no problems found.

When problems are found, the same corrective action and retest event sequence as defined in secondary testing above are performed.

End User Testing

The transaction is tested by end users. When appropriate, problem resolution takes place as in previous test segments.

Production Status

- () 19 - The Project Manager reviews all test results and determines that the transaction has successfully completed testing.
 - () Reviews tests results.
 - () Reviews problem reports to ensure all corrections done.
 - () Updates NOTES text to reflect testing results.
 - Project Manager _____ Date Complete _____
- () 20 - The transaction is placed into the production environment.
 - () Software files in production directories.
 - () DMR 550 updated to add transaction.
 - () DMR 900 updated to add transaction.
 - () Transaction added to production usage menu.

Module Test Plan

System Architect A

Date Complete 12/16/92

Module Test Case

- (✓) Attempt to store duplicate records in the database. Duplicate records are those where the unique identifier or any unique field contains duplicate values over 2 or more rows.
 - (✓) Duplicate row insertion attempts result in error message COM00102 [Duplicate value, enter new value or F1 for help]
- (✓) Display transaction help. Verify that the narrative adequately describes the purpose and functions of the transaction in end-user terms.
 - (✓) Transaction help describes:
 - (✓) Purpose of the transaction
 - (✓) Functions performed by the transaction
 - (✓) Where multiple page, what happens on each page
- (✓) Add a new record by using the insert key.
 - (✓) The insert key clears all blocks of the entry screen.
 - (✓) A new record can be successfully added and committed.
- (✓) Add a new record by scrolling past the last entry using the down arrow key.
 - (✓) Synchronization of all blocks occurs while scrolling to the end of the string of retrieved records.
 - (✓) All fields of the transaction blank when the end of the string of retrieved records is reached.
 - (✓) A new data record can be successfully added.
- (✓) Query a single record.
 - (✓) A single record is retrieved.
 - (✓) Pressing ~~SHIFT~~/F12 clears the record and permits entering a second query.
 - (✓) The second query successfully retrieves the desired data.
- (✓) Query multiple records using wild cards.
 - (✓) Pressing F12 clears the screen and makes the transaction ready for a query.
 - (✓) A query composed of a trailing % wildcard retrieves the desired data. Verify by running the same query through SQL*Plus.
 - (✓) A query using a prefix % wildcard retrieves the correct data.

- Verify by running the same query under SQL*Plus.
- (✓) A query using both prefix and suffix % wildcards retrieves the correct data. Verify by running the same query through
- (✓) Change each field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - (✓) Change each field individually, committing after each change.
 - () Change each field, as per the field test plan, for both valid and invalid values.
- (✓) Delete an existing record.
 - (✓) The selected record is deleted from the database. Verify by using SQL*Plus to attempt to retrieve the record.
- () Display the -JN journal record rows for the table to verify that the records changed and deleted above are entered into the journal table correctly.
 - () The _JN journal table for the affected table should contain rows for the initial addition of the record to the database, each change made to the database, and the deletion of the record from the database.
- () Use SQL*Plus to verify all data fields are stored correctly, including ID_ keys, date-updated, owner role, and any other autogenerated fields.
 - () All data fields are correctly stored in the database.

Block: TYPE_QUALIFIER_CODE
Title: Qualifier Type

Block Description

Block Test Case

Block Test Results Expected

Field Name: QUALIFIER

Field Description

Block: TYPE_QUALIFIER_CODE
Title: Qualifier Type

Field Name: QUALIFIER

Field Test Case

- () TAB through the field. Backtab to the field, enter data. After entering data in the next field, backtab to the field, and backspace the data from the field. The database should contain a null value for the field.
- () Enter data in the field. After entering data in the next field, backtab to the field, and space through the data. The database should contain a null value for the field.
- (/) Begin field with a space.
- (/) Enter data beyond length of field.
 - () If autoskip is in effect, cursor will skip to the next field.
 - (/) If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (/) Set keyboard on lower case and enter data. Data should be converted to upper case.
- (/) Clear an existing field and attempt to update the record.

not null field

Field Test Results Expected

- (/) Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.
 - () Field help describes:
 - (/) Purpose of the field
 - (/) LOV reference if available
 - (/) Valid range of values where present
 - (/) Required indicator when field is NOT NULL
 - () Default value when appropriate
- (/) Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
- (/) Change the field, committing after each change.

Block: TYPE_QUALIFIER_CODE
Title: Qualifier Type

Field Name: QUALIFIER

Field Test Results Expected

- () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
- N/A () Store a series of records using the SHIFT/~~FT~~^{FL} [duplicate field] key to carry forward field values from the previously stored record.
- () Display the -JN journal record rows for the table to verify that the records changed and deleted above are entered into the journal table correctly.
- () The -JN journal table for the affected table should contain rows for the initial addition of the record to the database, each change made to the database, and the deletion of the record from the database.
- () Use SQL*Plus to verify all data fields are processed correctly.
- () Bypass the field by using the TAB key.
 - () No error should occur.
 - () The database contains a null value for the field.
- 11/28 () Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
- () Delete an existing value. The field should be cleared and stored as a null value.
- () Any alphanumeric characters may be entered into the field.
- () Attempt to enter data as lower case. An error should result.
- () Delete an existing value.
 - () For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.
- () Attempt to bypass the field. An error message should tell you that the field data must be entered.
- () Enter data into the field, backspace the data out of the field and

Block: TYPE_QUALIFIER_CODE
Title: Qualifier Type

Field Name: QUALIFIER

Field Test Results Expected

attempt to TAB to the next field. An error should result.

Block: TYPE_QUALIFIER_CODE
Title: Qualifier Type

Field Name: DESCRIPTION

Field Description

Field Test Case

- () TAB through the field. Backtab to the field, enter data. After entering data in the next field, backtab to the field, and backspace the data from the field. The database should contain a null value for the field.
- () Enter data in the field. After entering data in the next field, backtab to the field, and space through the data. The database should contain a null value for the field.
- (/) Begin field with a space.
- (/) Enter data beyond length of field.
 - () If autoskip is in effect, cursor will skip to the next field.
 - (/) If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (/) Set keyboard on lower case and enter data. Data should be converted to upper case.
- (/) Clear an existing field and attempt to update the record.

not null field

Field Test Results Expected

- (/) Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.
 - () Field help describes:
 - (/) Purpose of the field
 - (/) LOV reference if available
 - (/) Valid range of values where present
 - (/) Required indicator when field is NOT NULL
 - (/) Default value when appropriate
- (/) Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and

Block: TYPE_QUALIFIER_CODE
Title: Qualifier Type

Field Name: DESCRIPTION

Field Test Results Expected

- after the changes were made.
- (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - (✓) Store a series of records using the SHIFT/P7 [duplicate field] key to carry forward field values from the previously stored record.
 - () Display the -JN journal record rows for the table to verify that the records changed and deleted above are entered into the journal table correctly.
 - () The _JN journal table for the affected table should contain rows for the initial addition of the record to the database, each change made to the database, and the deletion of the record from the database.
 - () Use SQL*Plus to verify all data fields are processed correctly.
 - () All data fields are correctly stored in the database.
 - () Bypass the field by using the TAB key.
 - () No error should occur.
 - () The database contains a null value for the field.
 - () Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
 - () Delete an existing value. The field should be cleared and stored as a null value.
 - (✓) Any alphanumeric characters may be entered into the field.
 - (✓) Attempt to enter data as lower case. ~~An error should result.~~
 - (✓) Delete an existing value.
 - (✓) For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.
 - (✓) Attempt to bypass the field. An error message should tell you

Not null field

Block: TYPE_QUALIFIER_CODE
Title: Qualifier Type

Field Name: DESCRIPTION

Field Test Results Expected

that the field data must be entered.

- () Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should result.

1.2 LE4100U1 ENTER CONSTITUENT RESULTS

LEMIS Module Test Checklist Report

Name: ENTER CONSTITUENT RESULTS
Short Name: LE4100U1
Purpose: Store and support constituent results informa
Initial Test Date: 1/26 JC
Verification Date: Yue JC

Module Description

This module is used to store and maintain constituent results information. The form is two pages long and consists of three blocks.

To enter constituent results data:

- 1) Enter the HEIS number and press <shift-F12>. The associated laboratory will appear.
- 2) Press <Page Down> to create Results of Sample Analysis Records in the following block and to move your cursor to the DATE EXTRACTED field in that block. The REQ NUM, ANALYSIS CLASS, and METHOD fields are query only and cannot be changed.
- 3) Enter the Date Extracted and Date Analysis. Both of these fields are optional.
- 4) If desired, you may press <F7> to add any associated text for each request line number. If you choose this option, the TEXT? field will change from "N" to "Y" after you have committed the text.
- 5) Press the <Page Down> key to move to page 2 of the form where you will find the associated constituents for the analysis class and method that you chose on page 1. Your cursor will be in the

Module Description

Concentration Value field. Cas Id and Constituent Id are query only fields and may not be modified.

- 6) Enter the Concentration Value and press <Tab> to move to the next field.
- 7) The Unit of Measure field will contain a default value. You may leave it as it is or change it. A LOV (list of values) is available by pressing <F9>. Press tab to move to the next field.
- 8) The Qualifier field is defaulted to "UJ". You may leave it as it is or change it by entering a valid value or pressing <F9> for the list of values.
- 9) You may add associated text for this each record by pressing <F7>. If you chose this option and commit your text, the Txt? field will change from "N" to "Y".
- 10) Use <Down arrow> to move through the list of constituents. As you move down through the constituent list, the Qualifier field will change from blank to "UJ". This is to signify that you have addressed this record so that if you leave this function and return at a later date, you will know by looking at the Qualifier column which records you have already updated.
- 11) Press <F10> to commit your changes.

Module Help

This transaction is used to add or maintain sampling results. ~~This module is used to store and maintain constituent results information including lab results, containers used, and constituents information.~~ The form is two pages long and consists of three blocks. To enter constituent results data:

- 1) Enter the HEIS number and press <shift-F12>. The associated laboratory will appear.
- 2) Press <Page Down> to create Results of Sample Analysis Records in the following block and to move your cursor to the DATE EXTRACTED field in that block. The REQ NUM, ANALYSIS CLASS, and METHOD fields are query only and cannot be changed.
- 3) Enter the Date Extracted and Date Analysis. Both of these fields are optional.
- 4) If desired, you may press <F7> to add any associated text for each request line number. If you choose this option, the TEXT? field will change from "N" to "Y" after you have committed the text.
- 5) Press the <Page Down> key to move to page 2 of the form where you will find the associated constituents for the analysis class and method that you chose on page 1. Your cursor will be in the Concentration Value field. Cas Id and Constituent Id are query

Module Help

- only fields and may not be modified.
- 6) Enter the Concentration Value and press <Tab> to move to the next field.
 - 7) The Unit of Measure field will contain a default value. You may leave it as it is or change it. A LOV (list of values) is available by pressing <F9>. Press tab to move to the next field.
 - 8) The Qualifier field is defaulted to "UJ". You may leave it as it is or change it by entering a valid value or pressing <F9> for the list of values.
 - 9) You may add associated text for this each record by pressing <F7>. If you chose this option and commit your text, the Txt? field will change from "N" to "Y".
 - 10) Use <Down arrow> to move through the list of constituents. As you move down through the constituent list, the Qualifier field will change from blank to "UJ". This is to signify that you have addressed this record so that if you leave this function and return at a later date, you will know by looking at the Qualifier column which records you have already updated.
 - 11) Press <F10> to commit your changes.

UNIT TEST CHECKLIST

(✓) Module Function Key

- (✓) Show Version : Shift-F10 :
- (✓) Comment/Bug : Shift-F3 :
- (✓) Fast Access : F5 :
- (✓) Back : F4 or Esc :
- (✓) Main Menu : Shift-F4 :
- (✓) Key Template : Ctrl-F1 :
- (✓) Context Prnt : F11 :
- (✓) Query Print : Ctrl-F11 :
- (✓) Oracle Print : Shift-F11 :

(✓) Block Title: Sample
Block Table: SAMPLE

(✓) Help : F1 :	(✓) Key Help : Shift-F1 :
(✓) Online Doc : F2 :	(✓) Dis Journal : Shift-F2 :
(✓) Related Data : F3 :	(✓) Edit Text : F7 :
(✓) Spec Func 1 : Shift-F5 :	(✓) Spec Func 2 : Ctrl-F5 :
(✓) Home/End : Home/End :	(✓) Prv/Nxt Block: PgDn/PgUp:
(✓) Scroll Up/Dn : Shift-Up/Dn:	(✓) Prv/Nxt Field: Tab/Sh-Tb:
(✓) Prv/Nxt Rec : Up/Down :	(✓) Commit Form : F10 :
(✓) Clear Field : F8 :	(✓) Clear Record : Shift-F8 :
(✓) Clear Form : Shift-F8 :	(✓) Exit System : Ctrl-F4 :

Ctrl-F8

(✓) Block Title: Sample
Block Table: SAMPLE

(✓) Enter Query : F12 :	(✓) Count Query : Ctrl-F12 :
(✓) Correct Query Fields? :	(✓) Uppercase and LOVs? :
(✓) Execute Qry : Shift-F12 :	(✓) Correct Query Order? :

(✓) Block Title: Results of Sample Analysis
Block Table: RESULTS

(✓) Help : F1 :	(✓) Key Help : Shift-F1 :
(✓) Online Doc : F2 :	(✓) Dis Journal : Shift-F2 :
(✓) Related Data : F3 :	(✓) Edit Text : F7 :
(✓) Spec Func 1 : Shift-F5 :	(✓) Spec Func 2 : Ctrl-F5 :

(✓) Home/End : Home/End :	(✓) Prv/Nxt Block: PgDn/PgUp:
(✓) Scroll Up/Dn : Shift-Up/Dn:	(✓) Prv/Nxt Field: Tab/Sh-Tb:
(✓) Prv/Nxt Rec : Up/Down :	(✓) Commit Form : F10 :

(✓) Clear Field : F8 :	(✓) Clear Record : Shift-F8 :
(✓) Clear Form : Shift-F8 :	(✓) Exit System : Ctrl-F4 :

(✓) Enter Query : F12 :	(✓) Count Query : Ctrl-F12 :
(✓) Correct Query Fields? :	(✓) Uppercase and LOVs? :
(✓) Execute Qry : Shift-F12 :	(✓) Correct Query Order? :

() Block Title: Constituents Found In Results
Block Table: CONSTITUENT_RESULTS

(✓) Help : F1 :	(✓) Key Help : Shift-F1 :
(✓) Online Doc : F2 :	(✓) Dis Journal : Shift-F2 :
(✓) Related Data : F3 :	(✓) Edit Text : F7 :
(✓) Spec Func 1 : Shift-F5 :	(✓) Spec Func 2 : Ctrl-F5 :

(✓) Home/End : Home/End :	(✓) Prv/Nxt Block: PgDn/PgUp:
(✓) Scroll Up/Dn : Shift-Up/Dn:	(✓) Prv/Nxt Field: Tab/Sh-Tb:
(✓) Prv/Nxt Rec : Up/Down :	(✓) Commit Form : F10 :

(✓) Clear Field : F8 :	(✓) Clear Record : Shift-F8 :
(✓) Clear Form : Shift-F8 :	(✓) Exit System : Ctrl-F4 :

(✓) Enter Query : F12 :	(✓) Count Query : Ctrl-F12 :
(✓) Correct Query Fields? :	(✓) Uppercase and LOVs? :
(✓) Execute Qry : Shift-F12 :	(✓) Correct Query Order? :

TEST PLAN

Testing of the transaction is performed at four levels:

1. Developer testing.

2. Secondary testing.
3. End user testing.
4. Production release & verification.

As each applicable section of this test plan is completed, the designated individual responsible for the section will sign the plan in the appropriate places. Each check off item will be completed. The status of the transaction will be updated in the data dictionary as the transaction progresses through the development and test steps.

Developer Testing

As each individual transaction is generated, the developer performs tests against the transaction to insure that it meets specifications and works properly. Upon successful completion of this testing, the transaction is made available for secondary testing.

Test status conditions are:

- (✓) 1 - Transaction under development, untested.
 - (✓) Transaction added to menu and security.
 - (✓) Transaction moved to development directory (addmod).
- (✓) 2 - Development complete, initial unit testing in progress.
 - (✓) Help added
 - (✓) Unit Test checklist completed.
 - (✓) Specific test plan added (if needed).
 - (✓) Any bugs found in Unit test fixed.
 - () Updated test booklet printed.
- (✓) 3 - Initial unit testing complete.

Developer T. Adams Date Complete 1/26/93

Secondary Testing

Secondary testing is performed by a person external to the development team. Tests are performed using a standardized test checklist. Initial testing is performed to ensure that the transaction meets the functional requirements of the transaction and satisfies the test plan narrative that follows.

- (✓) 4 - Secondary testing in progress.
- () 5 - Secondary testing complete, problems found. Resolution in progress.
- () 6 - Unable to resolve problem at this time.
- () 7 - Problem resolution complete.
- (✓) 8 - Secondary testing complete, no problems found.

Tester Joanne Campbell Date Complete 1/26/93

When problems are found:

1. The tester passes the checklist to the project manager for review and action.
2. The project manager assigns the checklist to the developer of

- the transaction for correction of the problems.
3. The developer corrects the described problems, records the corrective actions taken and the number of hours of corrective effort used, tests the transaction for correct operation, and returns the checklist to the project manager.
 4. The project manager returns the checklist to the tester for retesting.

The above sequence of events repeats until the tester signs off the transaction as passing the test requirements.

End User Testing

The transaction is tested by end users for a predetermined time period. This testing is done in the development environment. If no problems are encountered, or all problems are resolved at the end of this time period, the module is ready for Project Manager review.

- () 9 - End User testing in progress.
- () 10 - Problem found, resolution in progress.
- () 11 - Problem resolution complete.
- Developer _____ Date Complete _____
- () 12 - Resolution verified.
- Tester _____ Date Complete _____

Production Status

- () 13 - Ready for Project Manager review.
 - () The Project Manager reviews all test results and determines that the transaction has completed testing.
- () 14 - The transaction is ready to be placed into production.
 - () Software files moved to production directory.
 - () Transaction added to production menu.
 - () Transaction access given to production roles.
 - () Validation test performed in production.
 - () Production documentation up to date.
- () 15 - The transaction is in production

System Architect A Date Complete 1/26/93

Module Test Plan

- () Display transaction help. Verify that the narrative adequately describes the purpose and functions of the transaction in end-user terms.
- () Transaction help describes:
 - () Purpose of the transaction
 - () Functions performed by the transaction

- () Where multiple block, what happens in each block.

Block Title: Sample
 Block Table: SAMPLE
 Block Usage: Query Only

Block Help

This block shows the list of samples that have been defined for this location in this SAF. Each sample may be set to a different lab, and have a different purpose (ie Primary, Blank, Dup). To see the analysis to be performed for a sample, move the cursor to the sample and press <PageDown>.

Block Test Plan

- (✓) Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms:
- () Block help describes:
 - (✓) Purpose of the block
 - (✓) Options available in the block
 - (✓) The relationship of the block to the transaction or to the principal parent block of the transaction
 - () Navigation to other blocks of the transaction
 - () Where block synchronization is used, ensure that the block data is in sync with the parent block.
 - (.) Query a single record:
 - (✓) A single record is retrieved.
 - () Pressing the F12 key clears the record and permits entering a second query.
 - () The second query successfully retrieves the desired data.
 - () Query multiple records using wild cards:
 - (✓) Pressing F12 clears the block and makes the block ready for a query.
 - (✓) A query composed of a trailing % wildcard retrieves the desired data.
 - (✓) A query using a prefix % wildcard retrieves the correct data.
 - () A query using both prefix and suffix % wildcards retrieves the correct data.
 - () A query using the underscore () position wildcard retrieves the correct data.

() Verify that the data queries in the block are ordered logically.

Field Prompt: Lab
Field Usage: Queryable Only
Field Name: IDENT_LAB

Field Help

This field uniquely identifies the data item to the database. An error will occur if you attempt to enter a identification value already present in the database.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

() Enter query mode and verify that this field is accessible.

() Enter query mode and enter a valid value in this field.
() Only records matching the entered value should be displayed.

Field Prompt: HEIS Number
Field Usage: Non Updateable
Field Name: IDENT_HEIS_NUM

Field Help

Enter the unique HEIS number identifier used to identify the sample.

This data field:

- Must contain a value, may not be left blank.
- May contain any combination of characters.

Field Test Plan

() Enter query mode and verify that this field is accessible.

() Enter query mode and enter a valid value in this field.
() Only records matching the entered value should be displayed.

NA() Enter a duplicate value. An error should occur.

NA() From LOV, select a value already present in the database. An error should occur.

Block Title: Results of Sample Analysis
 Block Table: RESULTS
 Block Usage: Update Only

Block Help

This table work in conjunction with others to provide a comprehensive picture of sample analysis results. There may be more than one result information set that occurs when samples are split across laboratories or when laboratory returns are staggered.

Block Test Plan

(✓) Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms:

(✓) Block help describes:

- () Purpose of the block
- () Options available in the block
- () The relationship of the block to the transaction or to the principal parent block of the transaction
- () Navigation to other blocks of the transaction

() Where block synchronization is used, ensure that the block data is in sync with the parent block.

(✓) Query a single record:

- (✓) A single record is retrieved.
- () Pressing the F12 key clears the record and permits entering a second query.
- () The second query successfully retrieves the desired data.

(✓) Query multiple records using wild cards:

- (✓) Pressing F12 clears the block and makes the block ready for a query.
- (✓) A query composed of a trailing % wildcard retrieves the desired data.
- () A query using a prefix % wildcard retrieves the correct data.
- (✓) A query using both prefix and suffix % wildcards retrieves the correct data.
- () A query using the underscore (_) position wildcard retrieves the correct data.

?
NA (✓) Verify that the data queries in the block are ordered logically.

(✓) Update a record:

- N/A
N/A
N/A
- () The change count should increment.
 - () The date updated should become the current date.
 - () An error should occur if the operator owner identifier does not match the data owner identifier.

N/A } Delete a record from the block:

- () If the record has dependent records, the delete should fail unless the help text indicates that a cascading delete is permitted.
- () Use the clear rcd key (CTRL/F8).
 - () For a single record block, the block should clear.
 - () For a multi-record block, the record pointed to by the cursor should clear.
- () Use the delete key.
 - () For a single record block, the block should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should appear in the block.
 - () For a multiple record block, the entry should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should move up to fill the cleared space.

Field Prompt: Extracted
Field Usage: Updateable
Field Name: DATE_EXTRACTED

Field Help

Field Test Plan

- (✓) Verify that the field only allows valid dates.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Skip the field using the TAB key.
- (✓) Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
- (✓) Delete an existing value. The field should be cleared and stored:
 - (✓) Delete the value using the space bar.
 - (✓) Delete the value using the clear field key (F8).

Field Prompt: Analysis
Field Usage: Updateable
Field Name: DATE_ANALYSIS

Field Help

Field Test Plan

- (✓) Verify that the field only allows valid dates.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Skip the field using the TAB key.
- (✓) Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
- (✓) Delete an existing value. The field should be cleared and stored:
 - (✓) Delete the value using the space bar.
 - (✓) Delete the value using the clear field key (F8).

Field Prompt: Text?
 Field Usage: Queryable Only
 Field Name: FLAG_TEXT

Field Help

 This field is a flag that identifies the presence of additional textual information about the data item. If "N" is displayed, no text is currently present. If "Y" is displayed, text of one or more types is present. To enter or display text, press F7. A popup screen will display the default text type for the data item. If this default text type is the one you want, you may either read or add text. If you want to use another available text type for the data item, press LOV (F9) to display and select the desired text type. Once you have entered any text into the database, the indicator flag for the data item will be set to "Y".

This data field:

- Provides a default value. To accept the default value, press the ENTER key or TAB key.
- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

-
- () Enter query mode and verify that this field is accessible.
 - () Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.
 - () Test LOV:
 - () Select the first entry in the LOV.
 - () Select the last entry in the LOV.
 - () Use the FIND option and select an entry.
 - () Press the TEXT key (F7). The text popup should be displayed.
 - () With the cursor at the text type, press LOV (F9). Where available, a list of other text types available for the entry will be displayed. Select one text type for testing.
 - () Press the ENTER key to move the cursor to the text block.
 - () Enter two lines of text. The cursor should automatically move to the next line when one line is full.
 - () Press the COMMIT key (F10). Press ENTER in response to prompt

until the popup disappears and the primary screen reappears.

() The text flag should be set to 'Y'.

() Select and update text that already exists.

Block Title: Constituents Found In Results
 Block Table: CONSTITUENT_RESULTS
 Block Usage: Update Only

Block Help

This table contains the analytical results for one constituent for a specific sample as determined by the analytical laboratory that performed the analysis.

Block Test Plan

- (/) Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms:
 - (/) Block help describes:
 - (/) Purpose of the block
 - (/) Options available in the block
 - (/) The relationship of the block to the transaction or to the principal parent block of the transaction
 - () Navigation to other blocks of the transaction
- () Where block synchronization is used, ensure that the block data is in sync with the parent block.
- (/) Query a single record:
 - (/) A single record is retrieved.
 - (/) Pressing the F12 key clears the record and permits entering a second query.
 - (/) The second query successfully retrieves the desired data.
- (/) Query multiple records using wild cards:
 - (/) Pressing F12 clears the block and makes the block ready for a query.
 - (/) A query composed of a trailing % wildcard retrieves the desired data.
 - (/) A query using a prefix % wildcard retrieves the correct data.
 - (/) A query using both prefix and suffix % wildcards retrieves the correct data.
 - ? () A query using the underscore (_) position wildcard retrieves the correct data.
- (/) Verify that the data queries in the block are ordered logically.
- () Update a record:
 - () The change count should increment.

- () The date updated should become the current date.
 () An error should occur if the operator owner identifier does not match the data owner identifier.

() Delete a record from the block:

- () If the record has dependent records, the delete should fail unless the help text indicates that a cascading delete is permitted.
 () Use the clear rcd key (CTRL/F8).
 () For a single record block, the block should clear.
 () For a multi-record block, the record pointed to by the cursor should clear.
 () Use the delete key.
 () For a single record block, the block should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should appear in the block.
 () For a multiple record block, the entry should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should move up to fill the cleared space.

Field Prompt: Conc Value
 Field Usage: Updateable
 Field Name: CONC_VALUE

Field Help

Field Test Plan

- (✓) Attempt to store alphabetic characters in the field. An error should result.
 (✓) Begin the field entry with a space. An error should result.
 (✓) Attempt to enter a longer data value than allowed.
 (✓) Enter query mode and verify that this field is accessible.
 (✓) Enter query mode and enter a valid value in this field.
 (✓) Only records matching the entered value should be displayed.
 (✓) Attempt to store decimal data. Data to the limit of the decimal places provided by the field definition will be allowed.
 (✓) Help for the field should identify the number of decimal places allowed.
 (✓) Store a data value without decimal. The value should be zero to the right of the decimal place.
 (✓) Store a single digit decimal value.

- (✓) Store a decimal value to the maximum number of allowed decimal positions.
- MP (✓) Store a decimal value greater than the maximum number of allowed decimal positions. An error should occur.
- (✓) Skip the field using the TAB key.
- (✓) Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
- (✓) Delete an existing value. The field should be cleared and stored:
 - (✓) Delete the value using the space bar.
 - (✓) Delete the value using the clear field key (F8).

Field Prompt: Unit
 Field Usage: Updateable
 Field Name: CONC_UNIT_MEASURE

Field Help

Field Test Plan

- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (✓) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.
- NA () Accept the default.
- (✓) Modify each default value with a valid alternative.
- (✓) Modify each default value with an invalid alternative *via*
- (✓) Skip the field using the TAB key.
- (✓) Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
- (✓) Delete an existing value. The field should be cleared and stored:
 - (✓) Delete the value using the space bar.
 - (✓) Delete the value using the clear field key (F8).
- (✓) Test LOV:
 - (✓) Select the first entry in the LOV.
 - (✓) Select the last entry in the LOV.
 - () Use the FIND option and select an entry.

Field Prompt: Qual
Field Usage: Updateable
Field Name: QUALIFIER

Field Help

This field contains the data validation qualifier codes.

This data field:

- Is optional and may be left empty.
- May contain any character combination.
- All input is converted to upper case.

Field Test Plan

-
- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
 - (✓) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
 - (✓) Enter query mode and verify that this field is accessible.
 - (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
 - (✓) Accept the default.
 - (✓) Modify each default value with a valid alternative.
 - (✓) Modify each default value with an invalid alternative.
 - (✓) Skip the field using the TAB key.
 - (✓) Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
 - (✓) Delete an existing value. The field should be cleared and stored:
 - () Delete the value using the space bar.
 - () Delete the value using the clear field key (F8).
 - (✓) Test LOV:
 - (✓) Select the first entry in the LOV.
 - (✓) Select the last entry in the LOV.
 - (✓) Use the FIND option and select an entry.

Field Prompt: Text?
 Field Usage: Queryable Only
 Field Name: FLAG_TEXT

Field Help

This field is a flag that identifies the presence of additional textual information about the data item. If "N" is displayed, no text is currently present. If "Y" is displayed, text of one or more types is present. To enter or display text, press F7. A popup screen will display the default text type for the data item. If this default text type is the one you want, you may either read or add text. If you want to use another available text type for the data item, press LOV (F9) to display and select the desired text type. Once you have entered any text into the database, the indicator flag for the data item will be set to "Y".

This data field:

- Provides a default value. To accept the default value, press the ENTER key or TAB key.
- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

- () Enter query mode and verify that this field is accessible.
- () Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.
- () Test LOV:
 - () Select the first entry in the LOV.
 - () Select the last entry in the LOV.
 - () Use the FIND option and select an entry.
- () Press the TEXT key (F7). The text popup should be displayed.
- () With the cursor at the text type, press LOV (F9). Where available, a list of other text types available for the entry will be displayed. Select one text type for testing.
- () Press the ENTER key to move the cursor to the text block.
- () Enter two lines of text. The cursor should automatically move to the next line when one line is full.
- () Press the COMMIT key (F10). Press ENTER in response to prompt

until the popup disappears and the primary screen reappears.

() The text flag should be set to 'Y'.

() Select and update text that already exists.

2.0 GENERIC

2.1 GE1310R1 PERSONS IN SYSTEM REPORT

Initial Test Date: 5/4/94 *LC* Verification Date: 5/4/94 *LC*

- (*/*) Verify that the report is accessible from the menu.
- (*/*) Verify that the report is run correctly when requested.
- (*/*) Verify that the report format is correct.
- (*/*) Verify that the correct data is listed in the report.

2.2 GE310U1 PEOPLE RELATED TO ORGS

LEMIS Module Test Checklist Report

Name: GE1310U1 - PEOPLE RELATED TO O
 Short Name: GE1310U1
 Purpose: Maintain the relationship between persons and

Initial Test Date: 2/22/93 *LC*

Verification Date: 2/26/93 *LC*

Module Description

Module Help

This transaction is used to associate people to people or organizations, and organizations to people or organizations.

To associate people to organizations:

- 1) Move to the Organization block and enter query mode by pressing <F12>. If "Enter a query; ... " appears at the bottom of the screen, the form is already in query mode.
- 2) Enter the identifier or title of the desired organization.
- 3) Press <PageDown> to move to the "Related People" block.
- 4) Navigate to an open record using the down arrow key, or by pressing <Insert>.
- 5) Enter the oracle ID or employee # of the person to associate. List of Values is available.
- 6) Repeat steps 4 & 5 until all of the desired people are entered.
- 7) Press <F10> to commit the associations.

To disassociate people:

Steps 1)->3) are the same.

Module Help

- 4) Use the arrow keys to move the cursor to the person to disassociate.
- 5) Press the <Delete> key.
- 6) Press <F10> to commit the disassociation.

UNIT TEST CHECKLIST

(✓) Module Function Key

- (✓) Show Version : Shift-F10 :
- (✓) Comment/Bug : Shift-F3 :
- (✓) Fast Access : F5 :
- (✓) Back : F4 or Esc :
- (✓) Main Menu : Shift-F4 :
- (✓) Key Template : Ctrl-F1 :
- (✓) Context Prnt : F11 :
- (✓) Query Print : Ctrl-F11 :
- (✓) Oracle Print : Shift-F11 :

(✓) Block Title: Organization
Block Table: ORGANIZATION

(✓) Help : F1 :	(✓) Key Help : Shift-F1 :
(✓) Online Doc : F2 :	(✓) Dis Journal : Shift-F2 :
(✓) Related Data : F3 :	(✓) Edit Text : F7 :
(✓) Spec Func 1 : Shift-F5 :	(✓) Spec Func 2 : Ctrl-F5 :
(✓) Home/End : Home/End :	(✓) Prv/Nxt Block: PgDn/PgUp:
(✓) Scroll Up/Dn : Shft-Up/Dn:	(✓) Prv/Nxt Field: Tab/Sh-Tb:
(✓) Prv/Nxt Rec : Up/Down :	(✓) Commit Form : F10 :
(✓) Clear Field : F8 :	(✓) Clear Record : Shift-F8 :
(✓) Clear Form : Shift-F8 :	(✓) Exit System : Ctrl-F4 :
(✓) Enter Query : F12 :	(✓) Count Query : Ctrl-F12 :
(✓) Correct Query Fields? :	(✓) Uppercase and LOVs? :
(✓) Execute Qry : Shift-F12:	(✓) Correct Query Order? :

() Block Title: Related People
Block Table: FUNC_UNIT_NEST

(☒) Help : F1 :	(☒) Key Help : Shift-F1 :
(☒) Online Doc : F2 :	(☒) Dis Journal : Shift-F2 :
(☒) Related Data : F3 :	(☒) Edit Text : F7 :
(☒) Spec Func 1 : Shift-F5 :	(☒) Spec Func 2 : Ctrl-F5 :
(☒) Home/End : Home/End :	(☒) Prv/Nxt Block: PgDn/PgUp:
(☒) Scroll Up/Dn : Shft-Up/Dn:	(☒) Prv/Nxt Field: Tab/Sh-Tb:
(☒) Prv/Nxt Rec : Up/Down :	(☒) Commit Form : F10 :
(☒) Clear Field : F8 :	(☒) Clear Record : Shift-F8 :
(☒) Clear Form : Shift-F8 :	(☐) Exit System : Ctrl-F4 :
(☒) Enter Query : F12 :	(☒) Count Query : Ctrl-F12 :
(☒) Correct Query Fields? :	(☒) Uppercase and LOVs? :
(☒) Execute Qry : Shift-F12:	(☐) Correct Query Order? :
(☒) Insert Rec : Insert :	(☒) Add record at end :
(☒) Mandatory checks OK? :	(☒) Validation checks OK? :
(☒) List of Values OK? :	(☒) Duplicate Fld: Shift-F6 :
(☒) Duplicate Rec: F6 :	(☒) Duplicate checks OK? :
(☒) Optional fields marked? :	(☐) Correct fields uppercase:
(☒) Delete Record: Delete :	

TEST PLAN

Testing of the transaction is performed at four levels:

1. Developer testing.
2. Secondary testing.
3. End user testing.
4. Production release & verification.

As each applicable section of this test plan is completed, the designated individual responsible for the section will sign the plan in the appropriate places. Each check off item will be completed. The status of the transaction will be updated in the data dictionary as the transaction progresses through the development and test steps.

Developer Testing

As each individual transaction is generated, the developer performs tests against the transaction to insure that it meets specifications and works properly. Upon successful completion of this testing, the transaction is made available for secondary testing.

Test status conditions are:

- (X) 1 - Transaction under development, untested.
 - (X) Transaction added to menu and security.
 - (X) Transaction moved to development directory (addmod).
 - (X) 2 - Development complete, initial unit testing in progress.
 - (X) Help added
 - (X) Unit Test checklist completed.
 - (X) Specific test plan added (if needed).
 - (X) Any bugs found in Unit test fixed.
 - (X) Updated test booklet printed.
 - (X) 3 - Initial unit testing complete.
- Developer H Date Complete 2/19/92

Secondary Testing

Secondary testing is performed by a person external to the development team. Tests are performed using a standardized test checklist. Initial testing is performed to ensure that the transaction meets the functional requirements of the transaction and satisfies the test plan narrative that follows.

- (X) 4 - Secondary testing in progress.
 - (X) 5 - Secondary testing complete, problems found. Resolution in progress.
 - () 6 - Unable to resolve problem at this time.
 - () 7 - Problem resolution complete.
 - (X) 8 - Secondary testing complete, no problems found.
- Tester HC Date Complete 2/24/93

When problems are found:

1. The tester passes the checklist to the project manager for review and action.
2. The project manager assigns the checklist to the developer of the transaction for correction of the problems.
3. The developer corrects the described problems, records the corrective actions taken and the number of hours of corrective effort used, tests the transaction for correct operation, and returns the checklist to the project manager.
4. The project manager returns the checklist to the tester for retesting.

The above sequence of events repeats until the tester signs off the transaction as passing the test requirements.

End User Testing

The transaction is tested by end users for a predetermined time period. This testing is done in the development environment. If no problems are encountered, or all problems are resolved at the end of this time period, the module is ready for Project Manager review.

- () 9 - End User testing in progress.
- () 10 - Problem found, resolution in progress.
- () 11 - Problem resolution complete.
Developer _____ Date Complete _____
- () 12 - Resolution verified.
Tester _____ Date Complete _____

Production Status

- (✓) 13 - Ready for Project Manager/System Architect review.
() The Project Manager or System Architect reviews all test results and determines that the transaction has completed testing.
Architect/Manager AA Date Complete 2/26/93
- (✓) 14 - The transaction is ready to be placed into production.
(✓) Software files moved to production directory.
(✓) Transaction added to production menu.
(✓) Transaction access given to production roles.
(✓) Validation test performed in production.
() Production documentation up to date.
- (✓) 15 - The transaction has been put into production
By AA Date Done 2/26/93

Module Test Plan

- (✓) Display transaction help. Verify that the narrative adequately describes the purpose and functions of the transaction in end-user terms.
- (✓) Transaction help describes:
 - (✓) Purpose of the transaction
 - (✓) Functions performed by the transaction
 - (✓) Where multiple block, what happens in each block.

Block Title: Organization
Block Table: ORGANIZATION
Block Usage: Query Only

Block Help

This table contains information about any organizations that are relevant to this system. Any organization that needs to be recorded somewhere in the system must be entered into this table.

Block Test Plan

- (/) Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms:
 - (/) Block help describes:
 - (/) Purpose of the block
 - (/) Options available in the block
 - (/) The relationship of the block to the transaction or to the principal parent block of the transaction
 - (/) Navigation to other blocks of the transaction
- () Where block synchronization is used, ensure that the block data is in sync with the parent block.
- (/) Query a single record:
 - (/) A single record is retrieved.
 - (/) Pressing the F12 key clears the record and permits entering a second query.
 - () The second query successfully retrieves the desired data.
- () Query multiple records using wild cards:
 - (/) Pressing F12 clears the block and makes the block ready for a query.
 - (/) A query composed of a trailing % wildcard retrieves the desired data.
 - (/) A query using a prefix % wildcard retrieves the correct data.
 - (/) A query using both prefix and suffix % wildcards retrieves the correct data.
 - (/) A query using the underscore (_) position wildcard retrieves the correct data.
- (/) Verify that the data queries in the block are ordered logically.

Field Prompt: Ident
 Field Usage: Non Updateable
 Field Name: IDENT_ORGAN

Field Help

This field uniquely identifies the data item to the database. An error will occur if you attempt to enter a identification value already present in the database. The format is: XXX/XXX (alphanumeric).

Field Help

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

- (X) Enter query mode and verify that this field is accessible.
- (X) Enter query mode and enter a valid value in this field.
 - (X) Only records matching the entered value should be displayed.
- (X) Enter a duplicate value. An error should occur.
- (X) From LOV, select a value already present in the database. An error should occur.

*885
help does not
state field is
non updated*

Field Prompt: Title
Field Usage: Queryable Only
Field Name: TITLE

Field Help

Enter a 1-40 title for the data item. The title should be as descriptive as possible. It will be placed in the database in all caps form.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.

Field Test Plan

- (X) Enter query mode and verify that this field is accessible.
- (X) Enter query mode and enter a valid value in this field.
 - (X) Only records matching the entered value should be displayed.

Field Prompt: Text?

Field Usage: Queryable Only

Field Name: FLAG_TEXT

Field Help

This field is a flag that identifies the presence of additional textual information about the data item. If "N" is displayed, no text is currently present. If "Y" is displayed, text of one or more types is present. To enter or display text, press F7. A popup screen will display the default text type for the data item. If this default text type is the one you want, you may either read or add text. If you want to use another available text type for the data item, press LOV (F9) to display and select the desired text type. Once you have entered any text into the database, the indicator flag for the data item will be set to "Y".

This data field:

- Provides a default value. To accept the default value, press the ENTER key or TAB key.
- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

- () Enter query mode and verify that this field is accessible.
- () Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.
- () Press the TEXT key (F7). The text popup should be displayed.
- () With the cursor at the text type, press LOV (F9). Where available, a list of other text types available for the entry will be displayed. Select one text type for testing.
- () Press the ENTER key to move the cursor to the text block.
- () Enter two lines of text. The cursor should automatically move to the next line when one line is full.
- () Press the COMMIT key (F10). Press ENTER in response to prompt until the popup disappears and the primary screen reappears.
- () The text flag should be set to 'Y'.
- () Select and update text that already exists.

Block Title: Related People
 Block Table: FUNC UNIT_NEST
 Block Usage: Undefined

Block Help

The table provides the ability to associate several types of relationships between the parent tables may be available. Use the LOV on the RELATIONSHIP field to display this list of possible relationships.

Block Test Plan

- (X) Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms:
 - (X) Block help describes:
 - (X) Purpose of the block
 - (X) Options available in the block
 - (X) The relationship of the block to the transaction or to the principal parent block of the transaction
 - (X) Navigation to other blocks of the transaction
- (X) Where block synchronization is used, ensure that the block data is in sync with the parent block.
- (X) Query a single record:
 - (X) A single record is retrieved.
 - (X) Pressing the F12 key clears the record and permits entering a second query.
 - (X) The second query successfully retrieves the desired data.
- (X) Query multiple records using wild cards:
 - (X) Pressing F12 clears the block and makes the block ready for a query.
 - (X) A query composed of a trailing % wildcard retrieves the desired data.
 - (X) A query using a prefix % wildcard retrieves the correct data.
 - (X) A query using both prefix and suffix % wildcards retrieves the correct data.
 - (X) A query using the underscore (_) position wildcard retrieves the correct data.
- (X) Verify that the data queries in the block are ordered logically.
- (X) Store a series of records using the F6 [duplicate record] key to

carry forward field values from the previously stored record.

- (/) Attempting to store the duplicated record without change should result in an error.
- () Change key fields and store the record. It should store successfully.

Field Prompt: First Name
Field Usage: Queryable Only
Field Name: NAME_FIRST

Field Help

Enter the first name of the person.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

- (/) Enter query mode and verify that this field is accessible.
- (/) Enter query mode and enter a valid value in this field.
 - (/) Only records matching the entered value should be displayed.
- () Enter a duplicate value. An error should occur.
- (/) From LOV, select a value already present in the database. An error should occur.

Field Prompt: Init
Field Usage: Queryable Only
Field Name: INITIALS

Field Help

Enter the middle initial(s) of the person. Use 'X' if this person has no middle initials.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

- (/) Enter query mode and verify that this field is accessible.
- (/) Enter query mode and enter a valid value in this field.
 - (/) Only records matching the entered value should be displayed.
- () Enter a duplicate value. An error should occur.
- () From LOV, select a value already present in the database. An error should occur.

Field Prompt: Last Name
Field Usage: Queryable Only
Field Name: NAME_LAST

Field Help

Enter the last name of the person.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

- (☒) Enter query mode and verify that this field is accessible.
- (☒) Enter query mode and enter a valid value in this field.
 - (☒) Only records matching the entered value should be displayed.
- (☒) Enter a duplicate value. An error should occur.
- (☒) From LOV, select a value already present in the database. An error should occur.

Field Prompt: Ora ID/Empl#
 Field Usage: Queryable Only
 Field Name: IDENT_PERSON

Field Help

 This field contains a person's oracle ID or employee number. If this person is to be using the database it must contain the person's oracle ID. This is usually OPS\$ followed by their employee number.
 EXAMPLE: OPS\$E83681

If the person will not be using the database, then their employee number may be entered.

 This data field:

- Is optional and may be left empty.
- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

887 OK

Field Test Plan

-
- (-) Enter query mode and verify that this field is accessible.
 - () Enter query mode and enter a valid value in this field.
 () Only records matching the entered value should be displayed.
 - () Enter a duplicate value. An error should occur.
 - () From LOV, select a value already present in the database. An error should occur.

Field Prompt: Rel
Field Usage: Non Updateable
Field Name: RELATIONSHIP

Field Help

This field contains a three-character code that defines the nature of the relationship between two data items in the database. Use LOV to display a list of the valid values for this relationship.

This data field:

- Provides a default value. To accept the default value, press the ENTER key or TAB key.
- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

-
- (X) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
 - (X) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (X) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
 - () Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - (X) Attempt to change data in the field. An error should result.
 - (X) Use the spacebar to attempt to change data.
 - (X) Use the clear field key (F8) to attempt to change data.
 - (X) Attempt to delete data in the field. An error should result.
 - (X) Use the spacebar to attempt to delete data.
 - (X) Use the clear field key (F8) to attempt to delete data.
 - (X) Enter query mode and verify that this field is accessible.
 - (X) Enter query mode and enter a valid value in this field.
 - (X) Only records matching the entered value should be displayed.
 - () Attempt to delete an existing data field value. An error should result.
 - () Use the space bar to attempt to delete the field data.
 - () Use clear field (F8) to attempt to delete the field data.

386
Default does not appear
and cannot be chosen

Duplicate

- () Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- () Test LOV:
 - () Select the first entry in the LOV.
 - () Select the last entry in the LOV.
 - () Use the FIND option and select an entry.
- () Enter a duplicate value. An error should occur.
- () From LOV, select a value already present in the database. An error should occur.
- () Attempt to enter an invalid relationship. An error should occur.

24-FEB-93

Page 1 of 1

Programmer's Bug Report
WHC-SD-WM-TP-165 REV 0

Application: GENERIC Application Maintenance

Transaction: GE1310U1 Associate Persons with Organizations

=====
Comment ID : 885 Priority: LOW Date: 22-FEB-93
=====

Status : HANDLED Date: 24-FEB-93 Class:
Type : PROBLEM
Assigned To: KEN ATKINS
Entered By : LORRAINE CAMPBELL
System ID : 92000010388 Oracle ROWID: 00000128.0003.000B
Last Error : -0
Last Msg : FRM-40350 Query caused no records to be retrieved.
Table : ORGANIZATION
Block : ORGAN
Field : ORGAN.IDENT_ORGAN

DESCRIPTION

This is a non-updatable field. Why doesn't help state this?

CORRECTIVE ACTION

2/24/93 - Added module level field help that describes this better.

fe *OK*
2/24 *4/13*

Programmer's Bug Report
WHC-SD-WM-TP-165 REV 0

Application: GENERIC Application Maintenance

Transaction: GE1310U1 Associate Persons with Organizations

=====

Comment ID : 886 Priority: LOW Date: 23-FEB-93

=====

Status : HANDLED Date: 24-FEB-93 Class:

Type : PROBLEM

Assigned To: KEN ATKINS

Entered By : LORRAINE CAMPBELL

System ID : 92000010388

Oracle ROWID: 00000121.0006.000B

Last Error : -0

Last Msg : FRM-40202 Field must be entered.

Table : FUNC_UNIT_NEST

Block : FUNEST

Field : FUNEST.RELATIONSHIP

DESCRIPTION

Neither the Enter key nor the Tab key will accept the default. The default does not appear in the field.

CORRECTIVE ACTION

2/24/93 - Added default to field.

OK
RC 2/26/93

24-FEB-93

Page 1 of 1

Programmer's Bug Report
WHC-SD-WM-TP-165 REV 0

Application: GENERIC Application Maintenance

Transaction: GE1310U1 Associate Persons with Organizations

=====
Comment ID : 887 Priority: LOW Date: 23-FEB-93
=====

Status : HANDLED Date: 24-FEB-93 Class:
Type : PROBLEM
Assigned To: KEN ATKINS
Entered By : LORRAINE CAMPBELL
System ID : 92000010388 Oracle ROWID: 00000121.0006.000B
Last Error : -0
Last Msg : FRM-40200 Field is protected against update.
Table : FUNC UNIT_NEST
Block : FUNEST
Field : FUNEST.DSP_IDENT_PERSON

DESCRIPTION

help states field is optional and can be left empty, and also that a
blank is not allowed.

CORRECTIVE ACTION

2/24/93 - Fixed help.

*OK
fe 2/26/93*

2.3 GE1320R1 ORGANIZATIONS IN THE SYSTEM

Initial Test Date: 8/5/94 Verification Date: 8/5/94

- (✓) Verify that the report is accessible from the menu.
- (✓) Verify that the report is run correctly when requested.
- (✓) Verify that the report format is correct.
- (✓) Verify that the correct data is listed in the report.

2.4 GE2110Q1 TRANSACTION PICKLIST

Testing for this module was performed during the testing of
GE2110U1 TRANSACTION MAINTENANCE.

2.5 GE2110U1 TRANSACTION MAINTENANCE

LEMIS Module Test Checklist Report

Name: GE2110U1 - TRANSACTION MAINTEN
 Short Name: GE2110U1
 Purpose: To maintain a list of transactions available

Initial Test Date: 12/13/92 RC 12/21/92 RJA

Verification Date: 1/25/93 RJA

	Disp	----- Security -----			
Help	Fkeys	Ins	Del	Query	Upd
()	()	()	()	()	()

Block: APPLICATION
 Title: Application

Help	Scroll	Text	Navigate	Ins	Del	Upd	Qry
()	()	()	()	()	()	()	()

Data Element	Datatype	Dis Len	Help	LOV	Val/ Range	Not Null	Ins
IDENT APPLICATION	CHAR	2	()	()	()	()	()
NAME APPLICATION	CHAR	50	()	()	()	()	()

Block: TRANSACTION_LIST
Title: Transaction

Help	Scroll	Text	Navigate	Ins	Del	Upd	Qry
()	()	()	()	()	()	()	()
Data Element	Datatype	Dis Len	Help	LOV	Val/ Range	Not Null	Ins
NAME_TRANSACTION	CHAR	8	()	()	()	()	()
DESCR_TRANSACTION	CHAR	38	()	()	()	()	()
TYPE_TRANSACTION	CHAR	8	()	()	()	()	()
LEVEL_DATA_VALID	NUMBER	1	()	()	()	()	()

Block: TRANSACTION_LIST
Title: Transaction

Help	Scroll	Text	Navigate	Ins	Del	Upd	Qry
()	()	()	()	()	()	()	()
Data Element	Datatype	Dis Len	Help	LOV	Val/ Range	Not Null	Ins
DESCR_TRANSACTION	CHAR	38	()	()	()	()	()
TYPE_TRANSACTION	CHAR	8	()	()	()	()	()

Block: TRANSACTION_NEST
Title: Related Transactions

Help	Scroll	Text	Navigate	Ins	Del	Upd	Qry
()	()	()	()	()	()	()	()
Data Element	Datatype	Dis Len	Help	LOV	Val/ Range	Not Null	Ins
MENU_CODE	CHAR	1	()	()	()	()	()
MENU_SEQUENCE	NUMBER	3	()	()	()	()	()
NAME_TRANSACTION_CLD	CHAR	8	()	()	()	()	()

Block: TYPE_DATA_VALID_CODE
Title: Data Validation Levels

Help	Scroll	Text	Navigate	Ins	Del	Upd	Qry
()	()	()	()	()	()	()	()
Data Element	Datatype	Dis Len	Help	LOV	Val/ Range	Not Null	Ins

Module Description

Module Help

This transaction is used to maintain a list of all transactions in. This screen supports the update and retrieval functions for the. This screen supports the update and retrieval functions for the the system and to define the menu structure for the system. This information is used to display the menus and for system security. xxxxxxxxxxxxxxxxxxxxxxxx upon which the liquid effluent sampling requirement definitions upon which the liquid effluent sampling

The first block is a query only block that is used to select the activities are based. This screen is used to define the xxxxxxxxxxxx activities are based. This screen is used to define the TPA and its application. The second block is the list of transactions for the selected application. The third block is a list of related milestones and submilestones that relate to liquid effluent monitor-transactions for the selected transaction (Used for menus).

ing. The status of a requirement may also be updated from this To create a new transaction:

screen.

1. Make sure the correct application is displayed in the first block. If it is not press <F12> (enter query) and type the two

character application identifier (LOV is available). Then press <Shift-F12> to execute the query.

To store a new xxxxxxxxxxxx:

2. Press <PageDown> to move the the transaction block.
3. If there are transactions displayed in this block, press

To store a new requirement:

<Insert> to open a new record.

4. Type the name of the transaction to be created and press <Enter>.

1. Enter the xxxxxxxxxxxx identifier. This identifier may be up to THIS NAME MUST BE THE SAME AS THE SQL*FORMS NAME FOR THE TRANSACTION TO WORK. This 8 character identifier also needs to be nn characters in length. The identifier

1. Enter the requirement identifier. This identifier may be up to unique. If the "COM-00102" message appears, this transaction has already been created.

20 characters in length. If the requirement is the TPA, "TPA"

5. Enter a brief description of the transaction and press <Enter>.

This is the text that will appear on the menu for this

is adequate as the identifier. TPA milestones and submilestones transaction.

must be unique or an error will occur. The field is required.

should be identified as "TPA MILESTONE 17.nn". The identifier

6. Enter the type of transaction (FORM,MENU,REPORT etc). A list of

Module Help

2. TAB and enter the title of the xxxxxxxxxxxx. The title may be must be unique or an error will occur. The field is required. values is available.
from one to 40 characters in length and is required.
 2. TAB and enter the title of the requirement. The title may be
 7. The "Lev" field is used to allow transactions to be shared accross
 3. TAB to the REQ TYPE field. Use LOV to select the appropriate from one to 40 characters in length and is required.
applications. This should usually be left as "0".
type code:
 3. TAB to the REQ TYPE field. Use LOV to select the appropriate
 8. Press <Shift-F5> to enter the Menu Help Text. A popup display
 - a. Press the down arrow key until the cursor points to the type code:
of two lines will appear on the screen. Enter a brief description desired code.
 - a. Press the down arrow key until the cursor points to the of what functions this transaction performs. This text will be
 - b. Press the ENTER key to select the desired code onto the desired code.
displayed at the bottom of the menu when the cursor is on this screen.
 - b. Press the ENTER key to select the desired code onto the transaction.
 - c. The desired code and its description will be displayed.
screen.
 9. Press <F10> to save the new transaction.
 4. TAB to the REV field. This is the revision identifier of the
 - c. The desired code and its description will be displayed.
-
- xxxxxxxxxxxx. The field is required. A default value of zero is
4. TAB to the REV field. This is the revision identifier of the To add a transaction to a menu:
displayed. To accept the default, press the ENTER key. To requirement. The field is required. A default value of zero is
- enter a different revision level, key the revision number displayed. To accept the default, press the ENTER key. To
1. Query the desired application in the "Application" block.
over the default and press ENTER. (A number is required.)
enter a different revision level, key the revision number
 2. Place the cursor on the desired MENU transaction. One way to
over the default and press ENTER. (A number is required.)
make this easier is to query only the menu transactions in the
The Text? flag indicates whether free-form text is associated with the
transaction block. This is done by pressing <F12> and entering
requirement. If the flag is set to 'N', no text is present. To add
The Text? flag indicates whether free-form text is associated with the
"MENU" into the "Type" field, then pressing <Shift-F12>.

Module Help

text, press the TEXT key (F7). A popup screen will appear that will requirement. If the flag is set to 'N', no text is present. To add 3. Press <PageDown> to go to the "Related Transactions" block.

permit you to enter as much text as desired. Several different types text, press the TEXT key (F7). A popup screen will appear that will 4. Use the down arrow key to move to an empty record, or press of text may be available. Once the text has been stored, the text permit you to enter as much text as desired. Several different types <Insert> to open an empty record.

flag will be set to 'Y'. Additional text processing assistance can be of text may be available. Once the text has been stored, the text 5. Enter a one character menu code. This can be a number, or an obtained by pressing the HELP key (F1) while the text popup window is flag will be set to 'Y'. Additional text processing assistance can be upper or lower case letter. Press <Enter> being displayed.

obtained by pressing CTRL/K while the text popup window is being 6. Enter the sequence this item should appear on the menu. This must

displayed.

be a number. The sequence of the other items on this menu may 5. Press the COMMIT key (F10) to store the data in the database.

be reentered if needed.

5. Press the COMMIT key (F10) to store the data in the database.

7. Enter the transaction code. This is the eight character code To update an existing xxxxxxxxxxxx:

for the desired transaction. List of Values is available here.

To update an existing requirement:

8. Press <F10> to save the new information.

1. Press the ENTER QY key (F12).

2. Enter the identifier of the xxxxxxxxxxxx to be updated. If you

1. Press the ENTER QY key (F12).

only know part of the identifier, enter that part followed by a

2. Enter the identifier of the requirement to be updated.

To create a Menu:

percent sign (%). This will cause those xxxxxxxxxxxx that begin

3. Press the RUN QY key (Shift/F12). The requirement will be

with the partial identifier to be queried.

displayed on the screen.

Creating a menu is the same as creating any other transaction

3. Press the RUN QY key (Shift/F12). The xxxxxxxxxxxx will be

4. Press the TAB key until the cursor is at the field to be

except:

displayed on the screen; if more than one xxxxxxxxxxxx was

Module Help

modified.

5. Enter the new data in the field and press ENTER.
queried, the first one matching the query will be displayed. To
- a) The "Type" must be MENU.
locate the desired xxxxxxxxxxxx:
6. If more than one field is to be updated, repeat the above
- b) The transaction code does not have to match any program name.
a. Press the down arrow key until the desired xxxxxxxxxxxx is
process for each field.
Some standardization should be used in coming up with the
displayed.
7. Press the COMMIT key (F10) to store the changes in the database.
menu transaction codes.
4. Press the TAB key until the cursor is at the field to be

modified.

If the text associated with the requirement is to be modified:

5. Enter the new data in the field and press ENTER.
6. If more than one field is to be updated, repeat the above
1. Press the TEXT key (F7). The default description text will be
process for each field.
displayed. If you wish to modify another text type:
a. Press LOV (F9) to display the other text options.
7. Press the COMMIT key (F10) to store the changes in the database.

b. Use the down arrow key to move the cursor to the desired

If the text associated with the xxxxxxxxxxxx is to be modified:
text type.

- c. Press ENTER to select the desired text type. The text for
1. Press the TEXT key (F7). The default description text will be
that text type will be displayed.
displayed. If you wish to modify another text type:
2. Press the PAGE DOWN key to move the cursor to the text block.
a. Press LOV (F9) to display the other text options.
3. Use the up and down arrow keys to move the cursor to the line to
b. Use the down arrow key to move the cursor to the desired
be modified. Change the text as required. If special editing
text type.
of text is necessary, use the KEY HELP (Shift/F1) to display
c. Press ENTER to select the desired text type. The text for
the editing keys available.
that text type will be displayed.
4. Press the COMMIT key (F10) to update the database.
2. Press the PAGE DOWN key to move the cursor to the text block.
3. Use the up and down arrow keys to move the cursor to the line to
be modified. Change the text as required. If special editing

Module Help

Requirements may not be deleted if subordinate requirements or of text is necessary, use the KEY_HELP (Shift/F1) to display projects are present. To delete a requirement: the editing keys available.

4. Press the COMMIT key (F10) to update the database.
 1. Press the ENTER QY key (F12).
 2. Enter the identifier of the requirement to be updated.
 3. Press the RUN QY key (Shift/F12). The requirement will be xxxxxxxxxxxx may not be deleted if subordinate xxxxxxxxxxxx or displayed on the screen. xxxxxxxx are present. To delete a xxxxxxxx:
 4. Press the DELETE grey key. The screen should clear.
 5. Press the COMMIT key (F10) to update the database.
 1. Press the ENTER QY key (F12).
 2. Enter the identifier of the xxxxxxxxxxxx to be deleted.
-
3. Press the RUN QY key (Shift/F12). The xxxxxxxxxxxx will be Each requirement may be assigned to a higher level requirement. For example, a TPA milestone is assigned to the TPA; a submilestone is displayed on the screen.
 4. Press the DELETE grey key. The screen should clear. assigned to a milestone.
 5. Press the COMMIT key (F10) to update the database.

To associate a requirement with its parent:

Each xxxxxxxxxxxx may be assigned to a xxxxxxxxxxxxxxxxxxxxxxxxxx. For example,

1. Press the PAGE DOWN key to move the cursor to the REQ ID field of the Parent Requirement block.
2. Enter the identifier of the parent requirement, if known. If To associate a xxxxxxxxxxxx with its xxxxxx: not known, press LOV (F9) to display a list of available parent requirements:
1. Press the PAGE DOWN key to move the cursor to the xxxxxx field
 - a. Press the down arrow key until the cursor points to the of the xxxxxxxxxxxxxxxxxx block. desired parent requirement.
2. Enter the identifier of the xxxxxxxxxxxxxxxxxxxxxx, if known. If
 - b. Press the ENTER key to select the parent requirement.

Module Help

not known, press LOV (F9) to display a list of available
The parent requirement and its information will be displayed
xxxxxxxxxxxx:

on the screen.

- a. Press the down arrow key until the cursor points to the
3. Press the COMMIT key (F10) to store the association of the
desired xxxxxxxxxxxxxxxxx.
requirement to its parent in the database.
- b. Press the ENTER key to select the xxxxxxxxxxxxxxxxx.

The xxxxxxxxxxxxxxxxx and its information will be displayed

To remove the association between a requirement and its parent:
on the screen.

3. Press the COMMIT key (F10) to store the association of the
1. Press the PAGE DOWN key to move the cursor to the REQ ID field
xxxxxxxxxxxx to its xxxxxx in the database.
of the Parent Requirement block.
2. If the requirement identifier displayed is the one whose
To remove the association between a xxxxxxxxxxx and its xxxxxx:
relationship is to be deleted, press the DELETE key.

3. Press the COMMIT key (F10) to complete the update of the
1. Press the PAGE DOWN key to move the cursor to the xxxxxx field
database.
of the xxxxxxxxxxxxxxxxx block.

2. If the xxxxxxxxxxx identifier displayed is the one whose

relationship is to be deleted, press the DELETE key.

3. Press the COMMIT key (F10) to complete the update of the
database.

The status of each requirement is tracked in the Requirement Status

block. Here you may identify the different types of status conditions

and the dates on which the condition occurs. Unlimited text may be
The status of each xxxxxxxxxxx is tracked in the xxxxxxxxxxx Status
added to clarify the status condition. To add a new status condition:
block. Here you may identify the different types of status conditions

and the dates on which the condition occurs. Unlimited text may be

1. Press the PAGE DOWN key to move the cursor to the first
added to clarify the status condition. To add a new status condition:
status line.

2. Press the down arrow key, if required, to move the cursor to
1. Press the PAGE DOWN key to move the cursor to the first
the first blank line or press the INSERT key.
status line.

Module Help

3. Press LOV to display the available valid status condition codes.
2. Press the down arrow key, if required, to move the cursor to
4. Use the up and down arrow keys to select the desired status the first blank line.
condition.
3. Press LOV to display the available valid status condition codes.
5. Press the COMMIT (F10) key to move the status to the screen.
4. Use the up and down arrow keys to select the desired status
6. To add more statuses, press the down arrow key to the next condition.
blank space and repeat the selection process.
5. Press the COMMIT (F10) key to move the status to the screen.
6. To add more statuses, press the down arrow key to the next
If all of the spaces are filled and more statuses are to be added,
blank space and repeat the selection process.
press the down arrow key and the already entered records will scroll
upward, leaving space for an additional record to be added.
If all of the spaces are filled and more statuses are to be added,
press the down arrow key and the already entered records will scroll
7. Press the COMMIT (F10) key to store the selected status
upward, leaving space for an additional record to be added.
conditions in the database as associated with the requirement.
7. Press the COMMIT (F10) key to store the selected status
To store text expanding the description of a status condition:
conditions in the database as associated with the xxxxxxxxxxxx.
1. Use the up and down arrow keys to move the cursor to the status
To store text expanding the description of a status condition:
condition line where text is to be stored.
2. Press the TEXT key (F7). The text popup window will be
1. Use the up and down arrow keys to move the cursor to the status
displayed with cursor at the text type code. If a text type
condition line where text is to be stored.
other than that displayed is desired:
2. Press the TEXT key (F7). The text popup window will be
 - a. Press LOV (F9) to display the other text options.
displayed with cursor at the text type code. If a text type
 - b. Use the down arrow key to move the cursor to the desired
other than that displayed is desired:
text type.
 - a. Press LOV (F9) to display the other text options.
 - c. Press ENTER to select the desired text type. The text for
 - b. Use the down arrow key to move the cursor to the desired
that text type, if any, will be displayed.

Module Help

- text type.
3. Press the PAGE DOWN key to move the cursor to the text block.
 - c. Press ENTER to select the desired text type. The text for
 4. If no text is present, enter the desired text. If text is that text type, if any, will be displayed.
- present:
3. Press the PAGE DOWN key to move the cursor to the text block.
 - a. Use the up and down arrow keys to move the cursor to the
 4. If no text is present, enter the desired text. If text is line to be modified. Change the text as required. If present:
 - special editing of text is necessary, use the KEY HELP
 - a. Use the up and down arrow keys to move the cursor to the (Shift/F1) to display the editing keys available. line to be modified. Change the text as required. If
 - b. To add new text at the end, press the down arrow key until special editing of text is necessary, use the KEY HELP a blank line appears. Enter the text. (Shift/F1) to display the editing keys available.
 5. Press the COMMIT key (F10) to store the text in the database.
 - b. To add new text at the end, press the down arrow key until
 6. Press the ENTER key until the popup window disappears and the a blank line appears. Enter the text.
- requirement screen reappears.
5. Press the COMMIT key (F10) to store the text in the database.
 7. If more text is to be added to other status condition entries,
 6. Press the ENTER key until the popup window disappears and the use the up and down arrow keys to move the cursor to the desired requirement screen reappears.
 - entry and repeat the above steps.
 7. If more text is to be added to other status condition entries,
-
- use the up and down arrow keys to move the cursor to the desired entry and repeat the above steps.
-

Module Test Plan

TEST PLAN

Testing of the transaction is performed at five levels:

1. Developer testing.
2. Secondary testing.
3. System testing.
4. End user testing.
5. Production certification.

Module Test Plan

As each applicable section of this test plan is completed, the designated individual responsible for the section will sign the plan in the appropriate places. Each check off item will be completed. The status of the transaction will be updated in the data dictionary as the transaction progresses through the development and test steps.

Developer Testing

As each individual transaction is generated, the developer performs tests against the transaction as defined in the detail test plan. Upon successful completion of this testing, the transaction is made available for secondary testing.

- () 1 - Transaction under development, untested.
 - () Transaction added to system and security files.
 - () Specific test plan/test results/test case added.
 - () Test plan narrative script loaded.
- () 2 - Development complete, initial unit testing in progress.
 - () Help added
 - () LOV tested
- () 3 - Initial unit testing complete.

Secondary Testing

Secondary testing is performed by a person external to the development team. Tests are performed using a standardized test checklist. Initial testing is performed to ensure that the transaction meets the functional requirements of the transaction and satisfies the test plan narrative that follows.

- () 4 - Secondary testing in progress.
- () 5 - Secondary testing complete, problems found.
 - Tester _____ Date Complete _____
- () 6 - Problem resolution in progress.
- () 7 - Problem resolution complete.
 - Developer _____ Date Complete _____
- (/) 8 - Secondary testing complete, no problems found.

When problems are found:

1. The tester passes the checklist to the project manager for review and action.
2. The project manager assigns the checklist to the developer of the transaction for correction of the problems.
3. The developer corrects the described problems, records the corrective actions taken and the number of hours of corrective effort used, tests the transaction for correct operation, and returns the checklist to the project manager.
4. The project manager returns the checklist to the tester for

Module Test Plan

retesting.

The above sequence of events repeats until the tester signs off the transaction as passing the test requirements.

System Testing

System testing is performed by a second independent tester. This tester repeats the test plan scenario, the test cases, and any other tests desired under end-user operating conditions.

Test status conditions are:

- () 9 - System testing in progress.
- () 10 - System testing complete, problems found.
Tester _____ Date Complete _____
- () 11 - Problem resolution in progress.
- () 12 - Problem resolution complete.
Developer _____ Date Complete _____
- () 13 - System testing complete, no problems found.

When problems are found, the same corrective action and retest event sequence as defined in secondary testing above are performed.

End User Testing

The transaction is tested by end users. When appropriate, problem resolution takes place as in previous test segments.

Production Status

() 19 - The Project Manager reviews all test results and determines that the transaction has successfully completed testing.

- () Reviews tests results.
- () Reviews problem reports to ensure all corrections done.
- () Updates NOTES text to reflect testing results.

Project Manager ByA Date Complete _____

() 20 - The transaction is placed into the production environment.

- () Software files in production directories.
- () DMR 550 updated to add transaction.
- () DMR 900 updated to add transaction.
- () Transaction added to production usage menu.

System Architect A Date Complete 1/25/93

Module Test Case

- Filter only*
- (✓) Display transaction help. Verify that the narrative adequately

describes the purpose and functions of the transaction in end-user terms.

- (✓) Transaction help describes:
 - (✓) Purpose of the transaction
 - (✓) Functions performed by the transaction
 - (✓) Where multiple page, what happens on each page

() Add a new record by using the insert key.

- () The insert key clears all blocks of the entry screen. When the block supports multiple rows, all fields of the row pointed to by the cursor are cleared.
- () A new record can be successfully added and committed.

() Add a new record by scrolling past the last entry using the down arrow key.

- () Synchronization of all blocks occurs while scrolling to the end of the string of retrieved records.
- () All fields of the transaction blank when the end of the string of retrieved records is reached.
- () A new data record can be successfully added.

(✓) Query a single record.

- () A single record is retrieved.
- (✓) Pressing F12 clears the record and permits entering a second query.
- () The second query successfully retrieves the desired data.

(✓) Query multiple records using wild cards.

- (✓) Pressing F12 clears the screen and makes the transaction ready for a query.
- (✓) A query composed of a trailing % wildcard retrieves the desired data. Verify by running the same query through SQL*Plus.
- (✓) A query using a prefix % wildcard retrieves the correct data. Verify by running the same query under SQL*Plus.
- (✓) A query using both prefix and suffix % wildcards retrieves the correct data. Verify by running the same query through

() Delete an existing record.

- () The selected record is deleted from the database. Verify by using SQL*Plus to attempt to retrieve the record.

Block: APPLICATION
Title: Application

Query only

Block Description

Block Test Case

Block Test Results Expected

- (/) Display block help. Verify that the narrative adequately
- () Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms.
- () Block help describes:
- () Block help describes:
 - () Purpose of the block
 - () Purpose of the block
 - () Options available in the block
 - () Options available in the block
 - () The relationship of the block to the transaction or to the principal parent block of the transaction
 - () The relationship of the block to the transaction or to the principal parent block of the transaction
 - () Navigation to other blocks of the transaction
 - () Navigation to other blocks of the transaction
- n/a () Store a series of record using the F6 [duplicate record] key
- () Store a series of record using the F6 [duplicate record] key to carry forward field values from the previously stored record. to carry forward field values from the previously stored record.
- () Attempting to store the duplicated record without change
- () Attempting to store the duplicated record without change should result in an error.
- () Change key fields and store the record. It should store successfully.
- () Change key fields and store the record. It should store successfully.
- n/a () Delete a record from the block. If the record has dependent
- () Delete a record from the block. If the record has dependent records, the delete should fail unless the help text indicates that a cascading delete is permitted.

Block Test Results Expected

that a cascading delete is permitted.

- () Where block synchronization is used, ensure that the block data
- () Where block synchronization is used, ensure that the block data is in sync with the parent block.
- () Where block synchronization is used, ensure that the block data is in sync with the parent block.

- (/) Query a single record.
- () Query a single record.

- (/) A single record is retrieved.
- () A single record is retrieved.
- (/) Pressing the F12 key clears the record and permits entering
- () Pressing the F12 key clears the record and permits entering a second query.
- (/) The second query successfully retrieves the desired data.
- () The second query successfully retrieves the desired data.

- (/) Query multiple records using wild cards.
- () Query multiple records using wild cards.

- (/) Pressing F12 clears the block and makes the block ready for
- () Pressing F12 clears the block and makes the block ready for a query.
- (/) A query composed of a trailing % wildcard retrieves the
- () A query composed of a trailing % wildcard retrieves the desired data.
- (/) A query using a prefix % wildcard retrieves the correct
- () A query using a prefix % wildcard retrieves the correct data.
- (/) A query using both prefix and suffix % wildcards retrieves
- () A query using both prefix and suffix % wildcards retrieves the correct data.
- () A query using both prefix and suffix % wildcards retrieves the correct data.

Field Name: IDENT_APPLICATION

ordered logically.
ordered logically.

Field Name: IDENT_APPLICATION

Query Log

Field Description

This field contains a standardized unique identifier that provides a single format across all significant tables in the database. The number value is generated by a sequence generator IDENT_SEQUENCE. This generator creates an 11-digit number, the first two of which are the fiscal year. The generator is reset every fiscal year and the year value incremented. As each new significant table row is created, the sequence generator supplies the unique identifier and increments by one. Under this approach no two significant table rows in the database have the same identifier value.

Field Test Case

- NA* () Begin field with a space.
- () Enter data beyond length of field.
 - () If autoskip is in effect, cursor will skip to the next field.
 - () If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- NA* () Set keyboard on lower case and enter data. Data should be converted to upper case.
- () Clear an existing field and attempt to update the record.

Field Test Results Expected

- ☒ Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.
 - () Field help describes:
 - () Purpose of the field
 - () LOV reference if available
 - () Valid range of values where present
 - () Required indicator when field is NOT NULL
 - () Default value when appropriate
- NA* () Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.

Field Name: IDENT_APPLICATION

Field Test Results Expected

- n/a*
- () Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Store a series of records using the SHIFT/F6 [duplicate field] key to carry forward field values from the previously stored record.
 - () Use SQL*Plus to verify all data fields are processed correctly.
 - () All data fields are correctly stored in the database.
 - () Any alphanumeric characters may be entered into the field.
 - () Delete an existing value.
 - () For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.
 - () Attempt to bypass the field. An error message should tell you that the field data must be entered.
 - () Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should result.
 - () Attempt to enter data as lower case. An error should result.

Field Name: NAME APPLICATION

Query only

Field Description

Field Test Case

- () Begin field with a space.
- () Enter data beyond length of field.
 - () If autoskip is in effect, cursor will skip to the next field.
 - () If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- () Set keyboard on lower case and enter data. Data should be converted to upper case.
- () Clear an existing field and attempt to update the record.

Field Test Results Expected

- (1) Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.
 - () Field help describes:
 - () Purpose of the field
 - () LOV reference if available
 - () Valid range of values where present
 - () Required indicator when field is NOT NULL
 - () Default value when appropriate
- () Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - () Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
- () Store a series of records using the SHIFT/F6 [duplicate field] key to carry forward field values from the previously stored record.
- () Use SQL*Plus to verify all data fields are processed correctly.

Field Name: NAME_APPLICATION

Field Test Results Expected

- n/A {
- () All data fields are correctly stored in the database.
 - () Any alphanumeric characters may be entered into the field.
 - () Delete an existing value.
 - () For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.
 - () Attempt to bypass the field. An error message should tell you that the field data must be entered.
 - () Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should result.

Block: TRANSACTION LIST
 Title: Transaction

Block Description

Block Test Case

Block Test Results Expected

- (✓) Display block help. Verify that the narrative adequately
- () Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms.
- () Block help describes:
- () Block help describes:
 - () Purpose of the block
 - () Purpose of the block
 - () Options available in the block
 - () Options available in the block
 - () The relationship of the block to the transaction or to the principal parent block of the transaction
 - () The relationship of the block to the transaction or to the principal parent block of the transaction
 - () Navigation to other blocks of the transaction
 - () Navigation to other blocks of the transaction
- n/a () Store a series of record using the F6 [duplicate record] key
- () Store a series of record using the F6 [duplicate record] key to carry forward field values from the previously stored record.
- () Attempting to store the duplicated record without change
- (✓) Attempting to store the duplicated record without change should result in an error.
- () Change key fields and store the record. It should store successfully.
- () Change key fields and store the record. It should store successfully.
- (✓) Delete a record from the block. If the record has dependent
- () Delete a record from the block. If the record has dependent

Block: TRANSACTION_LIST
Title: Transaction

Block Test Results Expected

records, the delete should fail unless the help text indicates
records, the delete should fail unless the help text indicates
that a cascading delete is permitted.
that a cascading delete is permitted.

- (✓) Where block synchronization is used, ensure that the block data
- () Where block synchronization is used, ensure that the block data
is in sync with the parent block.
is in sync with the parent block.

- (✓) Query a single record.
- () Query a single record.

- (✓) A single record is retrieved.
- () A single record is retrieved.
- () Pressing the F12 key clears the record and permits entering
- (✓) Pressing the F12 key clears the record and permits entering
a second query.
a second query.
- (✓) The second query successfully retrieves the desired data.
- () The second query successfully retrieves the desired data.

- (✓) Query multiple records using wild cards.
- () Query multiple records using wild cards.

- (✓) Pressing F12 clears the block and makes the block ready for
- () Pressing F12 clears the block and makes the block ready for
a query.
a query.
- (✓) A query composed of a trailing % wildcard retrieves the
- () A query composed of a trailing % wildcard retrieves the
desired data.
desired data.
- (✓) A query using a prefix % wildcard retrieves the correct
- () A query using a prefix % wildcard retrieves the correct
data.
data.
- (✓) A query using both prefix and suffix % wildcards retrieves
- () A query using both prefix and suffix % wildcards retrieves
the correct data.
the correct data.

Block: TRANSACTION_LIST
Title: Transaction

Block Test Results Expected

- (✓) Verify that the data queried in the block are ordered logically.
- () Verify that the data queried in the block are ordered logically.

Field Name: NAME_TRANSACTION

Field Description

This field is the name of the transaction (form, report, etc).
It is a mandatory 8 characters in length and should be formatted in the following way:

AASFUOTn

where AA = Application Name (ie GE, LE)
S = Subsystem number
F = Function number
U = Unit Process number
T = transaction type: (U)pdate, (Q)uery, (R)eport etc.

Field Test Case

- (✓) Begin field with a space.
- (✓) Enter data beyond length of field.
 - () If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Set keyboard on lower case and enter data. Data should be converted to upper case.

n/a () Clear an existing field and attempt to update the record.

Field Test Results Expected

- () Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.

fixed
help - Field
Blank Both
allowable / non allowable

Block: TRANSACTION_LIST
Title: Transaction

Field Name: NAME_TRANSACTION

Field Test Results Expected

- () Field help describes:
 - () Purpose of the field
 - () LOV reference if available
 - () Valid range of values where present
 - () Required indicator when field is NOT NULL
 - () Default value when appropriate
- () Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - () Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
- () Store a series of records using the SHIFT/F6 [duplicate field] key to carry forward field values from the previously stored record.
- () Use SQL*Plus to verify all data fields are processed correctly.
 - () All data fields are correctly stored in the database.
- () Any alphanumeric characters may be entered into the field.
- () Delete an existing value.
 - () For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.
- () Attempt to bypass the field. An error message should tell you that the field data must be entered.
- () Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should result.
- () Attempt to enter data as lower case. An error should result.

Block: TRANSACTION_LIST
Title: Transaction

Field Name: DESCR_TRANSACTION

Field Description

Field Test Case

- (✓) Begin field with a space.
- (✓) Enter data beyond length of field.
 - () If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- 1/A () Set keyboard on lower case and enter data. Data should be converted to upper case.
- (✓) Clear an existing field and attempt to update the record.

Field Test Results Expected

- (✓) Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.
 - () Field help describes:
 - () Purpose of the field
 - () LOV reference if available
 - () Valid range of values where present
 - () Required indicator when field is NOT NULL
 - () Default value when appropriate
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
- (✓) Store a series of records using the SHIFT/F6 [duplicate field] key

Block: TRANSACTION_LIST
Title: Transaction

Field Name: DESCR_TRANSACTION

Field Test Results Expected

to carry forward field values from the previously stored record.

() Use SQL*Plus to verify all data fields are processed correctly.

() All data fields are correctly stored in the database.

(✓) Any alphanumeric characters may be entered into the field.

(✓) Delete an existing value.

(✓) For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.

(✓) Attempt to bypass the field. An error message should tell you that the field data must be entered.

(✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should result.

Block: TRANSACTION_LIST
Title: Transaction

Field Name: TYPE_TRANSACTION

Field Description

The type of transaction. Used to classify the transactions for reporting purposes.

Field Test Case

- ☒ () Begin field with a space.
- ☒ Enter data beyond length of field.
 - ☐ If autoskip is in effect, cursor will skip to the next field.
 - ☒ If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- ☒ Set keyboard on lower case and enter data. Data should be converted to upper case.
- ☒ Clear an existing field and attempt to update the record.

Field Test Results Expected

- ☒ Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.
 - ☐ Field help describes:
 - ☐ Purpose of the field
 - ☐ LOV reference if available
 - ☐ Valid range of values where present
 - ☐ Required indicator when field is NOT NULL
 - ☐ Default value when appropriate
- ☒ Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - ☒ Change the field, committing after each change.
 - ☐ Change the field where a range of valid values exists:
 - ☐ Value below valid range should display message COM00101 or COM00103 (if no LOV).

Block: TRANSACTION_LIST
Title: Transaction

Field Name: TYPE_TRANSACTION

Field Test Results Expected

- ☒ Store a series of records using the SHIFT/F6 [duplicate field] key to carry forward field values from the previously stored record.
- ☐ Use SQL*Plus to verify all data fields are processed correctly.
- ☐ All data fields are correctly stored in the database.
- n/a* ☐ Any alphanumeric characters may be entered into the field.
- ☒ Delete an existing value.
 - ☐ For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.
- ☒ Attempt to bypass the field. An error message should tell you that the field data must be entered.
- ☐ Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should result.
- ☒ Attempt to enter data as lower case. ~~An error should result.~~

Block: TRANSACTION_LIST
Title: Transaction

Field Name: LEVEL_DATA_VALID

Field Description

Field Test Case

(✓) Clear an existing field and attempt to update the record.

Field Test Results Expected

(✓) Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.

() Field help describes:
 () Purpose of the field
 () LOV reference
 () Default value

(✓) Test LOV.

(✓) Select a value from LOV.

(✓) Attempt to enter a value not in LOV. An error should result.

(✓) Delete an existing value.

(✓) For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.

(✓) Attempt to bypass the field. An error message should tell you that the field data must be entered.

(✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should result.

Block: TRANSACTION_LIST
Title: Transaction

Block Description

Block Test Case

Block Test Results Expected

Field Name: DESCR_TRANSACTION

Field Description

Field Test Case

- (✓) Begin field with a space.
- (✓) Enter data beyond length of field.
 - () If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Set keyboard on lower case and enter data. Data should be converted to upper case. *NOT ON THIS FIELD.*
- (✓) Clear an existing field and attempt to update the record.

Field Test Results Expected

- (✓) Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.
 - (✓) Field help describes:
 - (✓) Purpose of the field
 - (✓) LOV reference if available
 - (✓) Valid range of values where present
 - (✓) Required indicator when field is NOT NULL
 - () Default value when appropriate
- () Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the

Block: TRANSACTION_LIST
Title: Transaction

Field Name: DESCR_TRANSACTION

Field Test Results Expected

changes were made; display the record using SQL*Plus before and after the changes were made.

- ☐ Change the field, committing after each change.
- ☐ Change the field where a range of valid values exists:
 - ☐ Value below valid range should display message COM00101 or COM00103 (if no LOV).
- ☒ Store a series of records using the SHIFT/F6 [duplicate field] key to carry forward field values from the previously stored record.
- ☐ Use SQL*Plus to verify all data fields are processed correctly.
 - ☐ All data fields are correctly stored in the database.
- ☒ Any alphanumeric characters may be entered into the field.
- ☒ Delete an existing value.
 - ☐ For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.
- ☒ Attempt to bypass the field. An error message should tell you that the field data must be entered.
- ☒ Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should result.

Block: TRANSACTION_LIST
Title: Transaction

Field Name: TYPE_TRANSACTION

Field Description

The type of transaction. Used to classify the transactions for reporting purposes.

Field Test Case

- (✓) Begin field with a space.
- (✓) Enter data beyond length of field.
 - () If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Set keyboard on lower case and enter data. Data should be converted to upper case.
- (✓) Clear an existing field and attempt to update the record.

Field Test Results Expected

- (✓) Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.
 - (✓) Field help describes:
 - (✓) Purpose of the field
 - (✓) LOV reference if available
 - () Valid range of values where present
 - (✓) Required indicator when field is NOT NULL
 - (✓) Default value when appropriate
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).

Block: TRANSACTION_LIST
Title: Transaction

Field Name: TYPE_TRANSACTION

Field Test Results Expected

- (✓) Store a series of records using the SHIFT/F6 [duplicate field] key to carry forward field values from the previously stored record.
- () Use SQL*Plus to verify all data fields are processed correctly.
- () All data fields are correctly stored in the database.
- (✓) Any alphanumeric characters may be entered into the field. (60V)
- (✓) Delete an existing value.
- () For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.
- (✓) Attempt to bypass the field. An error message should tell you that the field data must be entered.
- (✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should result.
- (✓) Attempt to enter data as lower case. ~~An error should result.~~

converts to upper

Block: TRANSACTION_NEST
Title: Related Transactions

Block Description

Block Test Case

Block Test Results Expected

- (✓) Display block help. Verify that the narrative adequately
- (✓) Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms. describes the purpose and functions of the block in end-user terms.
- (✓) Block help describes:
- (✓) Block help describes:
 - () Purpose of the block
 - () Purpose of the block
 - () Options available in the block
 - () Options available in the block
 - () The relationship of the block to the transaction or to
 - () The relationship of the block to the transaction or to the principal parent block of the transaction
 - () the principal parent block of the transaction
 - () Navigation to other blocks of the transaction
 - () Navigation to other blocks of the transaction
- (✓) Store a series of record using the F6 [duplicate record] key
- (✓) Store a series of record using the F6 [duplicate record] key to carry forward field values from the previously stored record. to carry forward field values from the previously stored record.
- (✓) Attempting to store the duplicated record without change
- (✓) Attempting to store the duplicated record without change should result in an error. should result in an error.
- (✓) Change key fields and store the record. It should store
- (✓) Change key fields and store the record. It should store successfully. successfully.
- (✓) Delete a record from the block. If the record has dependent
- (✓) Delete a record from the block. If the record has dependent

Block: TRANSACTION_NEST
 Title: Related Transactions

Block Test Results Expected

- records, the delete should fail unless the help text indicates
 records, the delete should fail unless the help text indicates
 that a cascading delete is permitted.
 that a cascading delete is permitted.
- () Where block synchronization is used, ensure that the block data
 () Where block synchronization is used, ensure that the block data
 is in sync with the parent block.
 is in sync with the parent block.
- (✓) Query a single record.
 () Query a single record.
- (✓) A single record is retrieved.
 () A single record is retrieved.
 (✓) Pressing the F12 key clears the record and permits entering
 () Pressing the F12 key clears the record and permits entering
 a second query.
 a second query.
 (✓) The second query successfully retrieves the desired data.
 () The second query successfully retrieves the desired data.
- (✓) Query multiple records using wild cards.
 () Query multiple records using wild cards.
- () Pressing F12 clears the block and makes the block ready for
 (✓) Pressing F12 clears the block and makes the block ready for
 a query.
 a query.
 () A query composed of a trailing % wildcard retrieves the
 (✓) A query composed of a trailing % wildcard retrieves the
 desired data.
 desired data.
 () A query using a prefix % wildcard retrieves the correct
 (✓) A query using a prefix % wildcard retrieves the correct
 data.
 data.
 () A query using both prefix and suffix % wildcards retrieves
 (✓) A query using both prefix and suffix % wildcards retrieves
 the correct data.
 the correct data.

Block: TRANSACTION_NEST
Title: Related Transactions

Block Test Results Expected

- (✓) Verify that the data queried in the block are ordered logically.
- (✓) Verify that the data queried in the block are ordered logically.

Field Name: MENU_CODE

Field Description

Field Test Case

- (✓) TAB through the field. Backtab to the field, enter data. After entering data in the next field, backtab to the field, and backspace the data from the field. The database should contain a null value for the field.
- (✓) Enter data in the field. After entering data in the next field, backtab to the field, and space through the data. The database should contain a null value for the field.
- (✓) Begin field with a space.
- (✓) Enter data beyond length of field.
 - () If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Set keyboard on lower case and enter data. Data should be converted to upper case.

FIELD MUST BE ENTERED

(NOT FOR THIS FIELD)

Field Test Results Expected

- (✓) Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.
 - (✓) Field help describes:
 - (✓) Purpose of the field
 - (✓) LOV reference if available
 - (✓) Valid range of values where present

(747) NOT OPTIONAL FIELD

Block: TRANSACTION_NEST
Title: Related Transactions

Field Name: MENU_CODE

Field Test Results Expected

- ☐ Required indicator when field is NOT NULL
- ☐ Default value when appropriate
- ☒ Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - ☐ Change the field, committing after each change.
 - ☐ Change the field where a range of valid values exists:
 - ☐ Value below valid range should display message COM00101 or COM00103 (if no LOV).
- ☒ Store a series of records using the SHIFT/F6 [duplicate field] key to carry forward field values from the previously stored record.
- ☒ Use SQL*Plus to verify all data fields are processed correctly.
 - ☒ All data fields are correctly stored in the database.
- ☒ Bypass the field by using the TAB key.
 - ☐ No error should occur.
 - ☐ The database contains a null value for the field.
- ☒ Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
- ☒ Delete an existing value. The field should be cleared and stored as a null value.
- ☐ Any alphanumeric characters may be entered into the field.

Block: TRANSACTION_NEST
Title: Related Transactions

Field Name: MENU_SEQUENCE

Field Description

Field Test Case

- (✓) Clear an existing field and attempt to update the record.
- (✓) Attempt to enter alpha data.
- (✓) Attempt to enter longer value than allowed..
- (✓) Attempt to enter decimal value. An error will occur if no provision has been made for decimal data in the element definition of the table.
- (✓) Attempt to enter negative value. No error should occur. *RANGE 1-999*

Field Test Results Expected

- (✓) Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.
 - (✓) Field help describes:
 - (✓) Purpose of the field
 - (✓) LOV reference if available
 - (✓) Valid range of values where present
 - (✓) Required indicator when field is NOT NULL
 - (✓) Default value when appropriate
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
- (✓) Store a series of records using the SHIFT/F6 [duplicate field] key to carry forward field values from the previously stored record.

(748)

FIELD
CAN NOT
BE NEGATIVE

Block: TRANSACTION_NEST
Title: Related Transactions

Field Name: MENU_SEQUENCE

Field Test Results Expected

- () Use SQL*Plus to verify all data fields are processed correctly.
- (✓) All data fields are correctly stored in the database.
- (✓) Delete an existing value.
- () For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.
- (✓) Attempt to bypass the field. An error message should tell you that the field data must be entered.
- (✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should result.
- (✓) Numeric data (0-9), comma (,), plus (+), minus (-), and period (.) are permitted.
- (✓) Begin field entry with a space. ~~An error should occur.~~

Block: TRANSACTION_NEST
Title: Related Transactions

Field Name: NAME_TRANSACTION_CLD

Field Description

Field Test Case

- () Begin field with a space.
- (✓) Enter data beyond length of field.
 - () If autoskip is in effect, cursor will skip to the next field.
 - () If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Set keyboard on lower case and enter data. Data should be converted to upper case.
- (✓) Clear an existing field and attempt to update the record.

Field Test Results Expected

- (✓) Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.
 - () Field help describes:
 - () Purpose of the field
 - () LOV reference if available
 - () Valid range of values where present
 - () Required indicator when field is NOT NULL
 - () Default value when appropriate
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - () Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
- (✓) Store a series of records using the SHIFT/F6 [duplicate field] key

Block: TRANSACTION_NEST
Title: Related Transactions

Field Name: NAME_TRANSACTION_CLD

Field Test Results Expected

- to carry forward field values from the previously stored record.
- () Use SQL*Plus to verify all data fields are processed correctly.
 - () All data fields are correctly stored in the database.
 - () Any alphanumeric characters may be entered into the field.
 - () Delete an existing value.
 - () For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.
 - () Attempt to bypass the field.. An error message should tell you that the field data must be entered.
 - () Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should result.
 - ~~() Attempt to enter data as lower case. An error should result.~~

Block: TYPE_DATA_VALID_CODE
Title: Data Validation Levels

Block Description

Block Test Case

Block Test Results Expected

Programmer's Bug Report

Application: GENERIC Application Maintenance

Transaction: GE2110U1 Transaction Maintenance

=====

Comment ID : 746 Priority: LOW Date: 18-DEC-92

=====

Status : FIXED Date: 28-DEC-92 Class:

Type : PROBLEM

Assigned To: KEN ATKINS

Entered By : LORRAINE CAMPBELL

System ID : Oracle ROWID: 00000160.0009.000B

Last Error : -0

Last Msg : FRM-41800 List of values not available for this field

Table : TRANSACTION_LIST

Block : TRNLIS

Field : TRNLIS.NAME_TRANSACTION

DESCRIPTION

help states field blank both allowable and nonallowable

CORRECTIVE ACTION

fixed

Programmer's Bug Report

Application: GENERIC Application Maintenance

Transaction: GE2110U1 Transaction Maintenance

=====

Comment ID : 747 Priority: LOW Date: 21-DEC-92

=====

Status : FIXED Date: 28-DEC-92 Class:

Type : PROBLEM

Assigned To: KEN ATKINS

Entered By : TODD ADAMS (D)

System ID : Oracle ROWID: 0000011E.000D.000F

Last Error : -0

Last Msg : FRM-40400 Transaction complete -- 1 records posted an

Table : TRANSACTION_NEST

Block : TRNNST

Field : TRNNST.MENU_CODE

DESCRIPTION

Field help states that this field is optional. It is not.

CORRECTIVE ACTION

12/28/92 - Help fixed.

2.6 GE2120U1 MAINTAIN APPLICATION RECORD

850 OK
851 OK
852 OK

LEMIS Module Test Checklist Report

Name: GE2120U1 - MAINTAIN APPLICATION
 Short Name: GE2120U1
 Purpose: To maintain the application header record.
 Initial Test Date: 7/2 LC
 Verification Date: 2/3 LC

Module Description

Module Help

Use this module to add, update or delete application header information such as acronym and application name.

UNIT TEST CHECKLIST

(✓) Module	Function Key
(✓) Show Version	: Shift-F10 :
(✓) Comment/Bug	: Shift-F3 :
(✓) Fast Access	: F5 :
(✓) Back	: F4 or Esc :
(✓) Main Menu	: Shift-F4 :
(✓) Key Template	: Ctrl-F1 :
(✓) Context Prnt	: F11 :
(✓) Query Print	: Ctrl-F11 :
(✓) Oracle Print	: Shift-F11 :

(✓) Block Title: Application
 Block Table: APPLICATION

(✓) Block Title: Application
Block Table: APPLICATION

(✓) Help : F1 :	(✓) Key Help : Shift-F1 :
(✓) Online Doc : F2 :	(✓) Dis Journal : Shift-F2 :
(✓) Related Data : F3 :	(✓) Edit Text : F7 :
(✓) Spec Func 1 : Shift-F5 :	(✓) Spec Func 2 : Ctrl-F5 :

(✓) Home/End : Home/End :	(✓) Prv/Nxt Block: PgDn/PgUp:
(✓) Scroll Up/Dn : Shift-Up/Dn:	(✓) Prv/Nxt Field: Tab/Sh-Tb:
(✓) Prv/Nxt Rec : Up/Down :	(✓) Commit Form : F10 :

(✓) Clear Field : F8 :	(✓) Clear Record : Shift-F8 :
(✓) Clear Form : Shift-F8 :	(✓) Exit System : Ctrl-F4 :

(✓) Enter Query : F12 :	(✓) Count Query : Ctrl-F12 :
(✓) Correct Query Fields? :	(✓) Uppercase and LOVs? :
(✓) Execute Qry : Shift-F12:	(✓) Correct Query Order? :

(✓) Insert Rec : Insert :	(✓) Add record at end :
(✓) Mandatory checks OK? :	(✓) Validation checks OK? :
(✓) List of Values OK? :	(✓) Duplicate Fld: Shift-F6 :
(✓) Duplicate Rec: F6 :	(✓) Duplicate checks OK? :
(✓) Optional fields marked? :	(✓) Correct fields uppercase:

(✓) Delete Record: Delete :	

(✓) Block Title: Persons associated with Application
Block Table: APPLICATION_PERSON_J

(✓) Help : F1 :	(✓) Key Help : Shift-F1 :
(✓) Online Doc : F2 :	(✓) Dis Journal : Shift-F2 :
(✓) Related Data : F3 :	(✓) Edit Text : F7 :
(✓) Spec Func 1 : Shift-F5 :	(✓) Spec Func 2 : Ctrl-F5 :

(✓) Home/End : Home/End :	(✓) Prv/Nxt Block: PgDn/PgUp:
(✓) Scroll Up/Dn : Shift-Up/Dn:	(✓) Prv/Nxt Field: Tab/Sh-Tb:
(✓) Prv/Nxt Rec : Up/Down :	(✓) Commit Form : F10 :

(✓) Clear Field : F8 :	(✓) Clear Record : Shift-F8 :
(✓) Clear Form : Shift-F8 :	(✓) Exit System : Ctrl-F4 :

(✓) Enter Query : F12 :	(✓) Count Query : Ctrl-F12 :
(✓) Correct Query Fields? :	(✓) Uppercase and LOVs? :
(✓) Execute Qry : Shift-F12:	(✓) Correct Query Order? :

(✓) Insert Rec : Insert :	(✓) Add record at end :
(✓) Mandatory checks OK? :	(✓) Validation checks OK? :
(✓) List of Values OK? :	(✓) Duplicate Fld: Shift-F6 :
(✓) Duplicate Rec: F6 :	(✓) Duplicate checks OK? :
(✓) Optional fields marked? :	(✓) Correct fields uppercase:

(✓) Delete Record: Delete :	

TEST PLAN

Testing of the transaction is performed at four levels:

1. Developer testing.
2. Secondary testing.
3. End user testing.
4. Production release & verification.

As each applicable section of this test plan is completed, the designated individual responsible for the section will sign the plan in the appropriate places. Each check off item will be completed. The status of the transaction will be updated in the data dictionary as the transaction progresses through the development and test steps.

Developer Testing

As each individual transaction is generated, the developer performs tests against the transaction to insure that it meets specifications and works properly. Upon successful completion of this testing, the transaction is made available for secondary testing.

Test status conditions are:

- (✓) 1 ✓ Transaction under development, untested.
- (✓) 1 ✓ Transaction added to menu and security.
- (✓) 1 ✓ Transaction moved to development directory (addmod).
- (✓) 2 - Development complete, initial unit testing in progress.
- (✓) 2 ✓ Help added
- (✓) 2 ✓ Unit Test checklist completed.
- (✓) 2 ✓ Specific test plan added (if needed).
- (✓) 2 ✓ Updated test booklet printed.
- (✓) 3 - Initial unit testing complete.

Developer RJA Date Complete 1/27/93

Secondary Testing

Secondary testing is performed by a person external to the development team. Tests are performed using a standardized test checklist. Initial testing is performed to ensure that the transaction meets the functional requirements of the transaction and satisfies the test plan narrative that follows.

- (✓) 4 - Secondary testing in progress.
- (✓) 5 - Secondary testing complete, problems found. Resolution in progress.
- (✓) 6 - Unable to resolve problem at this time.
- (✓) 7 - Problem resolution complete.
- (✓) 8 - Secondary testing complete, no problems found.

Tester *L* Date Complete 2/3/93

When problems are found:

1. The tester passes the checklist to the project manager for review and action.
2. The project manager assigns the checklist to the developer of the transaction for correction of the problems.
3. The developer corrects the described problems, records the corrective actions taken and the number of hours of corrective effort used, tests the transaction for correct operation, and returns the checklist to the project manager.
4. The project manager returns the checklist to the tester for retesting.

The above sequence of events repeats until the tester signs off the transaction as passing the test requirements.

End User Testing

The transaction is tested by end users for a predetermined time period. This testing is done in the development environment. If no problems are encountered, or all problems are resolved at the end of this time period, the module is ready for Project Manager review.

- () 9 - End User testing in progress.
- () 10 - Problem found, resolution in progress.
- () 11 - Problem resolution complete.
Developer _____ Date Complete _____
- () 12 - Resolution verified.
Tester _____ Date Complete _____

Production Status

- () 13 - Ready for Project Manager review.
 - () The Project Manager reviews all test results and determines that the transaction has completed testing.
- () 14 - The transaction is ready to be placed into production.
 - () Software files moved to production directory.
 - () Transaction added to production menu.
 - () Transaction access given to production roles.
 - () Validation test performed in production.
 - () Production documentation up to date.
- () 15 - The transaction is in production

System Architect *A* Date Complete 2/3/93

Module Test Plan

(✓) Display transaction help. Verify that the narrative adequately describes the purpose and functions of the transaction in end-user terms.

(✓) Transaction help describes:

- (✓) Purpose of the transaction
- (✓) Functions performed by the transaction
- (✓) Where multiple block, what happens in each block.

TEST PLAN

Testing of the transaction is performed at five levels:

1. Developer testing.
2. Secondary testing.
3. System testing.
4. End user testing.
5. Production certification.

As each applicable section of this test plan is completed, the designated individual responsible for the section will sign the plan in the appropriate places. Each check off item will be completed. The status of the transaction will be updated in the data dictionary as the transaction progresses through the development and test steps.

Developer Testing

As each individual transaction is generated, the developer performs tests against the transaction as defined in the detail test plan. Upon successful completion of this testing, the transaction is made available for secondary testing.

- () 1 - Transaction under development, untested.
 - () Transaction added to system and security files.
 - () Specific test plan/test results/test case added.
 - () Test plan narrative script loaded.
- () 2 - Development complete, initial unit testing in progress.
 - () Help added
 - () LOV tested
- () 3 - Initial unit testing complete.

Developer _____ Date Complete _____

Secondary Testing

Secondary testing is performed by a person external to the development team. Tests are performed using a standardized test checklist. Initial testing is performed to ensure that the transaction meets the functional requirements of the transaction and satisfies the test plan narrative that follows.

- ()-4 - Secondary testing in progress.
- () 5 - Secondary testing complete, problems found.
- () 6 - Problem resolution in progress.
- () 7 - Problem resolution complete.
- () 8 - Secondary testing complete, no problems found.

Tester _____ Date Complete _____

When problems are found:

1. The tester passes the checklist to the project manager for review and action.
2. The project manager assigns the checklist to the developer of the transaction for correction of the problems.
3. The developer corrects the described problems, records the corrective actions taken and the number of hours of corrective effort used, tests the transaction for correct operation, and returns the checklist to the project manager.
4. The project manager returns the checklist to the tester for retesting.

The above sequence of events repeats until the tester signs off the transaction as passing the test requirements.

System Testing

System testing is performed by a second independent tester. This tester repeats the test plan scenario, the test cases, and any other tests desired under end-user operating conditions.

Test status conditions are:

- () 9 - System testing in progress.
- () 10 - System testing complete, problems found.
Tester _____ Date Complete _____
- () 11 - Problem resolution in progress.
- () 12 - Problem resolution complete.
Developer _____ Date Complete _____
- () 13 - System testing complete, no problems found.

When problems are found, the same corrective action and retest event sequence as defined in secondary testing above are performed.

End User Testing

The transaction is tested by end users. When appropriate, problem resolution takes place as in previous test segments.

Production Status

- () 19 - The Project Manager reviews all test results and determines that the transaction has successfully completed testing.
- () Reviews tests results.

- () Reviews problem reports to ensure all corrections done.
- () Updates NOTES text to reflect testing results.
- Project Manager _____ Date Complete _____
- () 20 - The transaction is placed into the production environment.
- () Software files in production directories.
- () DMR 550 updated to add transaction.
- () DMR 900 updated to add transaction.
- () Transaction added to production usage menu.
- System Architect CA Date Complete 2/3/93

Block Title: Application
 Block Table: APPLICATION
 Block Usage: Full Function

Block Help

 This table is a list of all of the applications in the current database. It is used to determine which users have access to an application, and to group the transactions into applications. It is also used to display the title of the application in the application startup screen.

Block Test Plan

- (/) Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms:
 - (/) Block help describes:
 - (/) Purpose of the block
 - (/) Options available in the block
 - (/) The relationship of the block to the transaction or to the principal parent block of the transaction
 - (/) Navigation to other blocks of the transaction
- (/) Where block synchronization is used, ensure that the block data is in sync with the parent block.
- (/) Query a single record:
 - (/) A single record is retrieved.
 - (/) Pressing the F12 key clears the record and permits entering a second query.
 - () The second query successfully retrieves the desired data.
- (/) Query multiple records using wild cards:
 - (/) Pressing F12 clears the block and makes the block ready for a query.

- (✓) A query composed of a trailing % wildcard retrieves the desired data.
- (✓) A query using a prefix % wildcard retrieves the correct data.
- (✓) A query using both prefix and suffix % wildcards retrieves the correct data.
- (✓) A query using the underscore (_) position wildcard retrieves the correct data.
- (✓) Verify that the data queries in the block are ordered logically.
- (✓) Store a series of records using the F6 [duplicate record] key to carry forward field values from the previously stored record.
 - (✓) Attempting to store the duplicated record without change should result in an error.
 - () Change key fields and store the record. It should store successfully.
- (✓) Update a record:
 - (✓) The change count should increment.
 - () The date updated should become the current date.
 - () An error should occur if the operator owner identifier does not match the data owner identifier.
- (✓) Delete a record from the block:
 - () If the record has dependent records, the delete should fail unless the help text indicates that a cascading delete is permitted.
 - () Use the clear rcd key (CTRL/F8).
 - () For a single record block, the block should clear.
 - () For a multi-record block, the record pointed to by the cursor should clear.
 - (✓) Use the delete key.
 - () For a single record block, the block should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should appear in the block.
 - (✓) For a multiple record block, the entry should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should move up to fill the cleared space.

Field Prompt: ID
 Field Usage: Non Updateable
 Field Name: IDENT_APPLICATION

Field Help

 This field is a two character code used to identify the application.

Field Help

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

-
- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
 - (✓) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
 - (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - (✓) Attempt to change data in the field. An error should result.
 - (✓) Use the spacebar to attempt to change data.
 - (✓) Use the clear field key (F8) to attempt to change data.
 - (✓) Attempt to delete data in the field. An error should result.
 - (✓) Use the spacebar to attempt to delete data.
 - (✓) Use the clear field key (F8) to attempt to delete data.
 - (✓) Enter query mode and verify that this field is accessible.
 - (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
 - (✓) Attempt to delete an existing data field value. An error should result.
 - (✓) Use the space bar to attempt to delete the field data.
 - (✓) Use clear field (F8) to attempt to delete the field data.
 - (✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
 - (✓) Enter a duplicate value. An error should occur.
 - (✓) From LOV, select a value already present in the database. An error should occur.

Field Prompt: Acronym
 Field Usage: Updateable
 Field Name: APPLICATION_ACRONYM

Field Help

This is a 1-10 character acronym used to represent the application name.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (✓) Enter data beyond the length of the field:
 - (✓) If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Attempt to delete an existing data field value. An error should result.
 - (✓) Use the space bar to attempt to delete the field data.

- (✓) Use clear field (F8) to attempt to delete the field data.
- (✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.

Field Prompt: Application Name
 Field Usage: Updateable
 Field Name: NAME_APPLICATION

Field Help

Enter the 1-40 character name of the application.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.

Field Test Plan

- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (✓) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Attempt to delete an existing data field value. An error should result.
 - (✓) Use the space bar to attempt to delete the field data.
 - (✓) Use clear field (F8) to attempt to delete the field data.

- () Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.

Field Prompt: Val Lev
 Field Usage: Updateable
 Field Name: LEVEL_DATA_VALID

Field Help

 This field contains the current data validation level for the data record. The field permits separating data records by the status of their validation. The field value will default to a value predefined for the data record and may be modified by the user at the time the data record is stored or at any time when the data is updated.

This data field:

- Is required
- Must contain a valid value
- May contain a number ranging from 0 through 9
- Has a default assigned. To accept the default, press the ENTER key or TAB key.
- Must contain a valid value. A blank is not allowed.

Field Test Plan

-
- (✓) Attempt to store alphabetic characters in the field. An error should result.
 - (✓) Begin the field entry with a space. An error should result.
 - (✓) Attempt to enter a longer data value than allowed.
 - (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - (✓) Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
 - (✓) Enter query mode and verify that this field is accessible.
 - (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.

- (✓) Attempt to enter a decimal value. An error should occur.
- (✓) Accept the default.
- () Modify each default value with a valid alternative.
- () Modify each default value with an invalid alternative.
- 01/8 (✓) Attempt to delete an existing data field value. An error should result.
 - () Use the space bar to attempt to delete the field data.
 - () Use clear field (F8) to attempt to delete the field data.
- (✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- (✓) Test LOV:
 - (✓) Select the first entry in the LOV.
 - (✓) Select the last entry in the LOV.
 - (✓) Use the FIND option and select an entry.
- (✓) Attempt to store an invalid code. An error should result.

Field Prompt: Text?

Field Usage: Queryable Only

Field Name: FLAG_TEXT

Field Help

This data field is a flag that indicates whether supporting text is present for the data record. When set to 'N', no supporting text is present. When the user adds supporting text, the flag is automatically set to 'Y'. The user may not modify the flag manually.

This data field:

- May contain characters 'Y' or 'N'
- Provides a default value of 'N'
- Must contain a valid value. A blank is not allowed
- All input is converted to upper case
- Contents are automatically generated by the system and cannot be updated by the user.

Field Test Plan

- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Press the TEXT key (F7). The text popup should be displayed.
- (✓) With the cursor at the text type, press LOV (F9). Where available, a list of other text types available for the entry will be displayed. Select one text type for testing.
- (✓) Press the ENTER key to move the cursor to the text block.
- (✓) Enter two lines of text. The cursor should automatically move to the next line when one line is full.
- (✓) Press the COMMIT key (F10). Press ENTER in response to prompt until the popup disappears and the primary screen reappears.
- (✓) The text flag should be set to 'Y'.
- (✓) Select and update text that already exists.

F10
F10
ENTER

Block Title: Persons associated with Application
 Block Table: APPLICATION_PERSON_J
 Block Usage: Undefined

Block Help

Block Test Plan

- (✓) Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms:
 - (✓) Block help describes:
 - (✓) Purpose of the block
 - (✓) Options available in the block
 - (✓) The relationship of the block to the transaction or to the principal parent block of the transaction
 - (✓) Navigation to other blocks of the transaction
- (✓) Where block synchronization is used, ensure that the block data is in sync with the parent block.
- (✓) Query a single record:
 - (✓) A single record is retrieved.
 - (✓) Pressing the F12 key clears the record and permits entering a second query.
 - (✓) The second query successfully retrieves the desired data.
- (✓) Query multiple records using wild cards:
 - (✓) Pressing F12 clears the block and makes the block ready for a query.
 - (✓) A query composed of a trailing % wildcard retrieves the desired data.
 - (✓) A query using a prefix % wildcard retrieves the correct data.
 - (✓) A query using both prefix and suffix % wildcards retrieves the correct data.
 - (✓) A query using the underscore (_) position wildcard retrieves the correct data.
- () Verify that the data queries in the block are ordered logically.
- () Store a series of records using the F6 [duplicate record] key to carry forward field values from the previously stored record.
 - (✓) Attempting to store the duplicated record without change should result in an error.

Not Sost
 Low
 Querying
 Field
 850
 851
 Wrong
 OK

N/A ()- Change key fields and store the record. It should store successfully.

Field Prompt: First Name
Field Usage: Queryable Only
Field Name: NAME_FIRST

Field Help

Enter the first name of the person.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

- (X) Enter query mode and verify that this field is accessible.
- (X) Enter query mode and enter a valid value in this field.
 - (X) Only records matching the entered value should be displayed.
- (X) Enter a duplicate value. An error should occur.
- (X) From LOV, select a value already present in the database. An error should occur.

Field Prompt: Initials
Field Usage: Queryable Only
Field Name: INITIALS

Field Help

Enter the middle initial(s) of the person. Use 'X' if this person has no middle initials.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- ✓) Enter a duplicate value. An error should occur.
- ✓) From LOV, select a value already present in the database. An error should occur.

Field Prompt: Last Name
Field Usage: Queryable Only
Field Name: NAME_LAST

Field Help

Enter the last name of the person.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Enter a duplicate value. An error should occur.
- (✓) From LOV, select a value already present in the database. An error should occur.

Field Prompt: Relationship
 Field Usage: Non Updateable
 Field Name: RELATIONSHIP

Field Help

This data field:

- Provides a default value. To accept the default value, press the ENTER key or TAB key.
- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

- (/ Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (/ Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (/ If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- () Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (/ Attempt to change data in the field. An error should result.
 - (/ Use the spacebar to attempt to change data.
 - (/ Use the clear field key (F8) to attempt to change data.
- (/ Attempt to delete data in the field. An error should result.
 - (/ Use the spacebar to attempt to delete data.
 - (/ Use the clear field key (F8) to attempt to delete data.
- (/ Enter query mode and verify that this field is accessible.
- (/ Enter query mode and enter a valid value in this field.
 - (/ Only records matching the entered value should be displayed.
- (/ Accept the default.
- () Modify each default value with a valid alternative.
- () Modify each default value with an invalid alternative.
- (/ Attempt to delete an existing data field value. An error should result.
 - (/ Use the space bar to attempt to delete the field data.

7
() Use clear field (F8) to attempt to delete the field data.

NA
() Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.

() Enter a duplicate value. An error should occur.

NA
() From LOV, select a value already present in the database. An error should occur.

7
() Attempt to enter an invalid relationship. An error should occur.

2.7 GE2210U1 MAINTAIN ROLE

LEMIS Module Test Checklist Report

Name: GE2210U1 - MAINTAIN ROLE
 Short Name: GE2210U1
 Purpose: To maintain the Role, including transactions
 Initial Test Date: JE 2/2/93
 Verification Date: JE 2/3/93

Module Description

Module Help

UNIT TEST CHECKLIST

(✓) Module	Function Key
(✓)	Show Version : Shift-F10 :
(✓)	Comment/Bug : Shift-F3 :
(✓)	Fast Access : F5 :
(✓)	Back : F4 or Esc :
(✓)	Main Menu : Shift-F4 :
(✓)	Key Template : Ctrl-F1 :
(✓)	Context Prnt : F11 :
(✓)	Query Print : Ctrl-F11 :
(✓)	Oracle Print : Shift-F11 :
- (✓) Block Title: Application
 Block Table: APPLICATION

(✓) Block Title: Application
Block Table: APPLICATION

(✓) Help : F1 :	(✓) Key Help : Shift-F1 :
(✓) Online Doc : F2 :	(✓) Dis Journal : Shift-F2 :
(✓) Related Data : F3 :	(✓) Edit Text : F7 :
(✓) Spec Func 1 : Shift-F5 :	(✓) Spec Func 2 : Ctrl-F5 :
<hr/>	
(✓) Home/End : Home/End :	(✓) Prv/Nxt Block: PgDn/PgUp:
(✓) Scroll Up/Dn : Shift-Up/Dn:	(✓) Prv/Nxt Field: Tab/Sh-Tb:
(✓) Prv/Nxt Rec : Up/Down :	(✓) Commit Form : F10 :
<hr/>	
(✓) Clear Field : F8 :	(✓) Clear Record : Shift-F8 :
(✓) Clear Form : Shift-F8 :	(✓) Exit System : Ctrl-F4 :
<hr/>	
(✓) Enter Query : F12 :	(✓) Count Query : Ctrl-F12 :
(✓) Correct Query Fields? :	(✓) Uppercase and LOVs? :
(✓) Execute Qry : Shift-F12:	(✓) Correct Query Order? :

(✓) Block Title: Role
Block Table: ROLE

(✓) Help : F1 :	(✓) Key Help : Shift-F1 :
(✓) Online Doc : F2 :	(✓) Dis Journal : Shift-F2 :
(✓) Related Data : F3 :	(✓) Edit Text : F7 :
(✓) Spec Func 1 : Shift-F5 :	(✓) Spec Func 2 : Ctrl-F5 :
<hr/>	
(✓) Home/End : Home/End :	(✓) Prv/Nxt Block: PgDn/PgUp:
(✓) Scroll Up/Dn : Shift-Up/Dn:	(✓) Prv/Nxt Field: Tab/Sh-Tb:
(✓) Prv/Nxt Rec : Up/Down :	(✓) Commit Form : F10 :
<hr/>	
(✓) Clear Field : F8 :	(✓) Clear Record : Shift-F8 :
(✓) Clear Form : Shift-F8 :	(✓) Exit System : Ctrl-F4 :
<hr/>	
(✓) Enter Query : F12 :	(✓) Count Query : Ctrl-F12 :
(✓) Correct Query Fields? :	(✓) Uppercase and LOVs? :
(✓) Execute Qry : Shift-F12:	(✓) Correct Query Order? :
<hr/>	
(✓) Insert Rec : Insert :	(✓) Add record at end :
(✓) Mandatory checks OK? :	(✓) Validation checks OK? :
(✓) List of Values OK? :	(✓) Duplicate Fld: Shift-F6 :
(✓) Duplicate Rec: F6 :	(✓) Duplicate checks OK? :
(✓) Optional fields marked? :	(✓) Correct fields uppercase:
<hr/>	
(✓) Delete Record: Delete :	

(✓) Block Title: Transactions Authorized
Block Table: TRANSACTION_ACCESS

(✓) Help : F1 :	(✓) Key Help : Shift-F1 :
(✓) Online Doc : F2 :	(✓) Dis Journal : Shift-F2 :
(✓) Related Data : F3 :	(✓) Edit Text : F7 :
(✓) Spec Func 1 : Shift-F5 :	(✓) Spec Func 2 : Ctrl-F5 :
(✓) Home/End : Home/End :	(✓) Prv/Nxt Block: PgDn/PgUp:
() Scroll Up/Dn : Shft-Up/Dn:	(✓) Prv/Nxt Field: Tab/Sh-Tb:
(✓) Prv/Nxt Rec : Up/Down :	(✓) Commit Form : F10 :
(✓) Clear Field : F8 :	(✓) Clear Record : Shift-F8 :
(✓) Clear Form : Shift-F8 :	(✓) Exit System : Ctrl-F4 :
(✓) Enter Query : F12 :	(✓) Count Query : Ctrl-F12 :
(✓) Correct Query Fields? :	(✓) Uppercase and LOVs? :
(✓) Execute Qry : Shift-F12:	(✓) Correct Query Order? :
(✓) Insert Rec : Insert :	(✓) Add record at end :
(✓) Mandatory checks OK? :	(✓) Validation checks OK? :
(✓) List of Values OK? :	(✓) Duplicate Fld: Shift-F6 :
(✓) Duplicate Rec: F6 :	(✓) Duplicate checks OK? :
(✓) Optional fields marked? :	(✓) Correct fields uppercase:
(✓) Delete Record: Delete :	

TEST PLAN

Testing of the transaction is performed at four levels:

1. Developer testing.
2. Secondary testing.
3. End user testing.
4. Production release & verification.

As each applicable section of this test plan is completed, the designated individual responsible for the section will sign the plan in the appropriate places. Each check off item will be completed. The status of the transaction will be updated in the data dictionary as the transaction progresses through the development and test steps.

Developer Testing

As each individual transaction is generated, the developer performs tests against the transaction to insure that it meets specifications and works properly. Upon successful completion of this testing, the transaction is made available for secondary testing.

Test status conditions are:

- (✓) 1 - Transaction under development, untested.
 - () 1 - Transaction added to menu and security.
 - (✓) 2 - Transaction moved to development directory (addmod).
 - (✓) 2 - Development complete, initial unit testing in progress.
 - (✓) Help added
 - (✓) Unit Test checklist completed.
 - (✓) Specific test plan added (if needed).
 - (✓) Updated test booklet printed.
 - (✓) 3 - Initial unit testing complete.
- Developer ROA Date Complete 1/27/93

Secondary Testing

Secondary testing is performed by a person external to the development team. Tests are performed using a standardized test checklist. Initial testing is performed to ensure that the transaction meets the functional requirements of the transaction and satisfies the test plan narrative that follows.

- (✓) 4 - Secondary testing in progress.
 - (✓) 5 - Secondary testing complete, problems found. Resolution in progress.
 - () 6 - Unable to resolve problem at this time.
 - () 7 - Problem resolution complete.
 - (✓) 8 - Secondary testing complete, no problems found.
- Tester AC Date Complete 2/3/93

When problems are found:

1. The tester passes the checklist to the project manager for review and action.
2. The project manager assigns the checklist to the developer of the transaction for correction of the problems.
3. The developer corrects the described problems, records the corrective actions taken and the number of hours of corrective effort used, tests the transaction for correct operation, and returns the checklist to the project manager.
4. The project manager returns the checklist to the tester for retesting.

The above sequence of events repeats until the tester signs off the transaction as passing the test requirements.

End User Testing

The transaction is tested by end users for a predetermined time period. This testing is done in the development environment. If no problems are encountered, or all problems are resolved at the end of this time period, the module is ready for Project Manager review.

- () 9 - End User testing in progress.

- () 10 - Problem found, resolution in progress.
- () 11 - Problem resolution complete.
Developer _____ Date Complete _____
- () 12 - Resolution verified.
Tester _____ Date Complete _____

Production Status

- () 13 - Ready for Project Manager review.
() The Project Manager reviews all test results and determines that the transaction has completed testing.
- () 14 - The transaction is ready to be placed into production.
() Software files moved to production directory.
() Transaction added to production menu.
() Transaction access given to production roles.
() Validation test performed in production.
() Production documentation up to date.
- () 15 - The transaction is in production

System Architect A Date Complete 2/3/93

Module Test Plan

- (/) Display transaction help. Verify that the narrative adequately describes the purpose and functions of the transaction in end-user terms.
- (/) Transaction help describes:
 - (/) Purpose of the transaction
 - () Functions performed by the transaction
 - (/) Where multiple block, what happens in each block.

Block Title: Application
Block Table: APPLICATION
Block Usage: Query Only

Block Help

This table is a list of all of the applications in the current database. It is used to determine which users have access to an application, and to group the transactions into applications. It is also used to display the title of the application in the application startup screen.

Block Test Plan

Block Test Plan

-
- (/) Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms:
 - (/) Block help describes:
 - (/) Purpose of the block
 - (/) Options available in the block
 - (/) The relationship of the block to the transaction or to the principal parent block of the transaction
 - (/) Navigation to other blocks of the transaction
 - () Where block synchronization is used, ensure that the block data is in sync with the parent block.
 - (/) Query a single record:
 - (/) A single record is retrieved.
 - (/) Pressing the F12 key clears the record and permits entering a second query.
 - (/) The second query successfully retrieves the desired data.
 - (/) Query multiple records using wild cards:
 - (/) Pressing F12 clears the block and makes the block ready for a query.
 - (/) A query composed of a trailing % wildcard retrieves the desired data.
 - (/) A query using a prefix % wildcard retrieves the correct data.
 - (/) A query using both prefix and suffix % wildcards retrieves the correct data.
 - (/) A query using the underscore (_) position wildcard retrieves the correct data.
 - (/) Verify that the data queries in the block are ordered logically.

Field Prompt: ID

Field Usage: Non Updateable

Field Name: IDENT_APPLICATION

Field Help

This field is a two character code used to identify the application.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.

Field Help

- All input is converted to upper case.

Field Test Plan

- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.

NA () Enter a duplicate value. An error should occur.

NA () From LOV, select a value already present in the database. An error should occur.

Field Prompt: Name

Field Usage: Queryable Only

Field Name: NAME_APPLICATION

Field Help

Enter the 1-40 character name of the application.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.

Field Test Plan

- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.

Block Title: Role
 Block Table: ROLE
 Block Usage: Full Function

Block Help

This table contains a list of the roles defined by the system.
 All transaction access in the system is determined by means of these 'Roles'.

This table includes an identifier for the role, as well as a more descriptive title.

Block Test Plan

- (/) Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms:

(/) Block help describes:

- (/) Purpose of the block
- (/) Options available in the block
- (/) The relationship of the block to the transaction or to the principal parent block of the transaction
- () Navigation to other blocks of the transaction

- N/A) Where block synchronization is used, ensure that the block data is in sync with the parent block.

- (/) Query a single record:

- (/) A single record is retrieved.
- (/) Pressing the F12 key clears the record and permits entering a second query.
- (/) The second query successfully retrieves the desired data.

- (/) Query multiple records using wild cards:

- (/) Pressing F12 clears the block and makes the block ready for a query.
- (/) A query composed of a trailing % wildcard retrieves the desired data.
- (/) A query using a prefix % wildcard retrieves the correct data.
- (/) A query using both prefix and suffix % wildcards retrieves the correct data.
- (/) A query using the underscore (_) position wildcard retrieves the correct data.

- (/) Verify that the data queries in the block are ordered logically.

*Sorting
845 not
Functioning*

- (✓) Store a series of records using the F6 [duplicate record] key to carry forward field values from the previously stored record.
 - (✓) Attempting to store the duplicated record without change should result in an error.
 - (✓) Change key fields and store the record. It should store successfully.
- (✓) Update a record:
 - (✓) The change count should increment.
 - () The date updated should become the current date.
 - () An error should occur if the operator owner identifier does not match the data owner identifier.
- (✓) Delete a record from the block:
 - () If the record has dependent records, the delete should fail unless the help text indicates that a cascading delete is permitted.
 - (✓) Use the clear rcd key (CTRL/F8).
 - () For a single record block, the block should clear.
 - (✓) For a multi-record block, the record pointed to by the cursor should clear.
 - (✓) Use the delete key.
 - () For a single record block, the block should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should appear in the block.
 - (✓) For a multiple record block, the entry should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should move up to fill the cleared space.

Field Prompt: Role
 Field Usage: Updateable
 Field Name: IDENT_ROLE

Field Help

This field is a 12 character unique text used to identify the role. It is stored in the database as the owner of the data. It should describe the role if possible.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

- (✓) Verify that the field accepts character or numeric data within

Field Test Plan

the limits imposed by validation criteria.

- (✓) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - N/A () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Attempt to delete an existing data field value. An error should result.
 - (✓) Use the space bar to attempt to delete the field data.
 - (✓) Use clear field (F8) to attempt to delete the field data.
- (✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- (✓) Enter a duplicate value. An error should occur.
- N/A () From LOV, select a value already present in the database. An error should occur.

Field Prompt: Title
 Field Usage: Updateable
 Field Name: TITLE_ROLE

Field Help

Enter a 1-40 title for the data item. The title should be used by the software to more efficiently support the database in all caps form.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (✓) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - () Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Attempt to delete an existing data field value. An error should result.
 - (✓) Use the space bar to attempt to delete the field data.

- (~~/~~) Use clear field (F8) to attempt to delete the field data.
- (~~/~~) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.

Field Prompt: Min Val
 Field Usage: Updateable
 Field Name: MIN_VAL_ACCESS

Field Help

This data field:

- Provides a default value. To accept the default value, press the ENTER key or TAB key.
- Must contain a valid value. A blank is not allowed.
- Must contain a numeric value (0-9), a comma (,), plus (+), minus (-), or period (.).

Field Test Plan

- (✓) Attempt to store alphabetic characters in the field. An error should result.
- (✓) Begin the field entry with a space. An error should result.
- N/A () Attempt to enter a longer data value than allowed.
- () Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - (✓) Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - (✓) Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - (✓) Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Attempt to enter a decimal value. An error should occur.
- (✓) Accept the default.
- (✓) Modify each default value with a valid alternative.

OK
field help not descriptive enough

- (✓) Modify each default value with an invalid alternative.
- (✓) Attempt to delete an existing data field value. An error should result.
 - (✓) Use the space bar to attempt to delete the field data.
 - (✓) Use clear field (F8) to attempt to delete the field data.
- (✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- (✓) Test LOV:
 - (✓) Select the first entry in the LOV.
 - (✓) Select the last entry in the LOV.
 - (✓) Use the FIND option and select an entry.

Field Prompt: Val Lev
 Field Usage: Updateable
 Field Name: LEVEL_DATA_VALID

Field Help

This data field:

- Provides a default value. To accept the default value, press the ENTER key or TAB key.
- Must contain a valid value. A blank is not allowed.
- Must contain a numeric value (0-9), a comma (,), plus (+), minus (-), or period (.)

Field Test Plan

- (✓) Attempt to store alphabetic characters in the field. An error should result.
- (✓) Begin the field entry with a space. An error should result.
- (✓) Attempt to enter a longer data value than allowed.
- () Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - () Change the field, committing after each change.
 - (✓) Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Attempt to enter a decimal value. An error should occur.
- (✓) Accept the default.
- (✓) Modify each default value with a valid alternative.

OK LC
 Descriptive
 847 help

- (/) Modify each default value with an invalid alternative.
- (/) Attempt to delete an existing data field value. An error should.. result.
 - (/) Use the space bar to attempt to delete the field data.
 - (/) Use clear field (F8) to attempt to delete the field data.
- (/) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- (/) Test LOV:
 - (/) Select the first entry in the LOV.
 - (/) Select the last entry in the LOV.
 - (/) Use the FIND option and select an entry.
- (/) Attempt to store an invalid code. An error should result.

Field Prompt:
 Field Usage: Queryable Only
 Field Name: FLAG_TEXT

Field Help

 This field is a flag that identifies the presence of additional textual information about the data item. If "N" is displayed, no text is currently present. If "Y" is displayed, text of one or more types is present. To enter or display text, press F7. A popup screen will display the default text type for the data item. If this default text type is the one you want, you may either read or add text. If you want to use another available text type for the data item, press LOV (F9) to display and select the desired text type. Once you have entered any text into the database, the indicator flag for the data item will be set to "Y".

This data field:

- Provides a default value. To accept the default value, press the ENTER key or TAB key.
- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

-
- () Enter query mode and verify that this field is accessible.
 - () Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.
 - (✓) Press the TEXT key (F7). The text popup should be displayed.
 - (✓) With the cursor at the text type, press LOV (F9). Where available, a list of other text types available for the entry will be displayed. Select one text type for testing.
 - (✓) Press the ENTER key to move the cursor to the text block.
 - (✓) Enter two lines of text. The cursor should automatically move to the next line when one line is full.
 - () Press the COMMIT key (F10). Press ENTER in response to prompt until the popup disappears and the primary screen reappears.
 - (✓) The text flag should be set to 'Y'.
 - (✓) Select and update text that already exists.

not a problem
PC/8
F10 twice and Return to MSF
OK

Block Title: Transactions Authorized
 Block Table: TRANSACTION_ACCESS
 Block Usage: Full Function

Block Help

This table contains a list of the transactions that each role has access to. It also contains the level of access for each transaction.

Block Test Plan

- (✓) Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms:
 - (✓) Block help describes:
 - (✓) Purpose of the block
 - (✓) Options available in the block
 - (✓) The relationship of the block to the transaction or to the principal parent block of the transaction
 - (✓) Navigation to other blocks of the transaction
- (✓) Where block synchronization is used, ensure that the block data is in sync with the parent block.
- (✓) Query a single record:
 - (✓) A single record is retrieved.
 - (✓) Pressing the F12 key clears the record and permits entering a second query.
 - (✓) The second query successfully retrieves the desired data.
- (✓) Query multiple records using wild cards:
 - (✓) Pressing F12 clears the block and makes the block ready for a query.
 - (✓) A query composed of a trailing % wildcard retrieves the desired data.
 - (✓) A query using a prefix % wildcard retrieves the correct data.
 - (✓) A query using both prefix and suffix % wildcards retrieves the correct data.
 - (✓) A query using the underscore (_) position wildcard retrieves the correct data.
- (✓) Verify that the data queries in the block are ordered logically.
- (✓) Store a series of records using the F6 [duplicate record] key to carry forward field values from the previously stored record.

- (✓) Attempting to store the duplicated record without change should result in an error.
- (✓) () Change key fields and store the record. It should store successfully.
- (✓) Update a record:
 - (✓) The change count should increment.
 - () The date updated should become the current date.
 - () An error should occur if the operator owner identifier does not match the data owner identifier.
- (✓) Delete a record from the block:
 - () If the record has dependent records, the delete should fail unless the help text indicates that a cascading delete is permitted.
 - (✓) Use the clear rcd key (CTRL/F8).
 - () For a single record block, the block should clear.
 - () For a multi-record block, the record pointed to by the cursor should clear.
 - (✓) Use the delete key.
 - () For a single record block, the block should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should appear in the block.
 - (✓) For a multiple record block, the entry should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should move up to fill the cleared space.

Field Prompt: Transact
 Field Usage: Non Updateable
 Field Name: NAME_TRANSACTION

Field Help

This field should contain the name of a valid transaction. Use LOV to see a current list of valid transactions. If you need to add a transaction to the system, you must first add the transaction using the MAINTAIN TRANSACTION transaction (GE2110U1).

This data field:

- Is optional and may be left empty.
- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

-
- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.

Field Test Plan

-
- (/ Enter data beyond the length of the field:
 - (/ If autoskip is in effect, cursor will skip to the next field.
 - (/ If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
 - (/ Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - (/ Attempt to change data in the field. An error should result.
 - (/ Use the spacebar to attempt to change data.
 - (/ Use the clear field key (F8) to attempt to change data.
 - (/ Attempt to delete data in the field. An error should result.
 - (/ Use the spacebar to attempt to delete data.
 - (/ Use the clear field key (F8) to attempt to delete data.
 - (/ Enter query mode and verify that this field is accessible.
 - (/ Enter query mode and enter a valid value in this field.
 - (/ Only records matching the entered value should be displayed.
 - (/ Attempt to delete an existing data field value. An error should result.
 - (/ Use the space bar to attempt to delete the field data.
 - (/ Use clear field (F8) to attempt to delete the field data.
 - (/ Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
 - () Test LOV:
 - () Select the first entry in the LOV.
 - () Select the last entry in the LOV.
 - () Use the FIND option and select an entry.
 - (/ Enter a duplicate value. An error should occur.
 - (/ From LOV, select a value already present in the database. An error should occur.

Picklist not LOV

Field Prompt: Description
Field Usage: Queryable Only
Field Name: DESCR_TRANSACTION

Field Help

A brief description of the transaction. This field will be used as the transaction name in the menus, and on any reports. The description may be entered in mixed case.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.

Field Test Plan

- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Enter a duplicate value. An error should occur.
- (✓) From LOV, select a value already present in the database. An error should occur.

```
Field Prompt: QUAD
Field Usage: Updateable
Field Name: CODE ACCESS
```

Field Help

The transaction access code consists of four Y/N flags that determine the access each role is allowed for each transaction. The four flags are in the following order: Q U A D

Where Q = Query
 U = Update
 A = Add
 D = Delete

Example:

```
User MAY (Y) query using this transaction.
|+---User MAY NOT (N) Update using this transaction.
||+---User MAY (Y) add using this transaction.
|||+---User MAY NOT (N) delete using this transaction.
||||
YNYN
```

This data field:

- Is optional and may be left empty.
- Provides a default value. To accept the default value, press the ENTER key or TAB key.
- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

nit

- () Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (X) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (X) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (X) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (X) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (X) Change the field, committing after each change.

- (/) Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (/) Enter query mode and verify that this field is accessible.
- (/) Enter query mode and enter a valid value in this field.
 - (/) Only records matching the entered value should be displayed.
- (/) Accept the default.
- (/) Modify each default value with a valid alternative.
- () Modify each default value with an invalid alternative.
- (/) Attempt to delete an existing data field value. An error should result.
 - (/) Use the space bar to attempt to delete the field data.
 - (/) Use clear field (F8) to attempt to delete the field data.
- (/) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- (/) Test LOV:
 - (/) Select the first entry in the LOV.
 - (/) Select the last entry in the LOV.
 - (/) Use the FIND option and select an entry.

Field Prompt: Expires
Field Usage: Updateable
Field Name: DATE_EXPIRES

Field Help

This field contains the date this role's access to the system should expire. CURRENTLY NOT IMPLEMENTED!!

Enter the date value using the format dd-MON-yy
where:

- dd - is the numeric day of the month
- MON - is the three character definition of month
(JAN, FEB, MAR, APR, MAY, JUN, JUL
AUG, SEP, OCT, NOV, DEC)
- yy - is the numeric value of the year within the century.

The system will display an error if you enter the data in any other format.

This data field:

- Is optional and may be left empty.
- Date data must be entered in the format mm/dd/yy
where:
 - mm = the month of the year in numeric form 01-12
 - dd = the day of the month; days prior to 10 should have a leading zero, e.g., May 1 should be shown as 01
either upper or lower case, e.g., September = SEP
 - yy = the last two digits of the calendar year, e.g., 1992 is entered as 92.

Field Test Plan

- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- () Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - () Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.

*ok
Conflicting
849 formats
in 12/19*

- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Skip the field using the TAB key.
- (✓) Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
- (✓) Delete an existing value. The field should be cleared and stored:
 - (✓) Delete the value using the space bar.
 - (✓) Delete the value using the clear field key (F8).

2.8 GE2510R1 TRANSACTION LIST REPORTInitial Test Date: LC 7/1/93 Verification Date: 8/5/94

- (X) Verify that the report is accessible from the menu.
- (X) Verify that the report is run correctly when requested.
- (X) Verify that the report format is correct.
- (X) Verify that the correct data is listed in the report.

2.9 GE2520R1 ROLE LIST REPORTInitial Test Date: LC 5/9/94 Verification Date: 5/9/94

- (X) Verify that the report is accessible from the menu.
- (X) Verify that the report is run correctly when requested.
- (X) Verify that the report format is correct.
- (X) Verify that the correct data is listed in the report.

2.10 GE2530R1 TRANSACTIONS AUTHORIZEDInitial Test Date: LC 5/9/94 Verification Date: 5/9/94

- (X) Verify that the report is accessible from the menu.
- (X) Verify that the report is run correctly when requested.
- (X) Verify that the report format is correct.
- (X) Verify that the correct data is listed in the report.

2.11 GE2540R1 USERS ASSIGNED TO A ROLEInitial Test Date: LC 5/9/94 Verification Date: 5/9/94

- (X) Verify that the report is accessible from the menu.
- (X) Verify that the report is run correctly when requested.
- (X) Verify that the report format is correct.
- (X) Verify that the correct data is listed in the report.

2.12 GE2610U1 MAINTAIN DATA OWNERSHIP

LEMIS Module Test Checklist Report

Name: GE2610U1 - MAINTAIN DATA OWNER
 Short Name: GE2610U1
 Purpose: maintain data ownership

Initial Test Date: 12/10/92 *AC*

Verification Date: 2/12/93 *AC*

	Disp	----- Security -----			
Help	Fkeys	Ins	Del	Query	Upd
()	()	()	()	()	()

Block: APPLICATION
 Title: Application

Help	Scroll	Text	Navigate	Ins	Del	Upd	Qry
()	()	()	()	()	()	()	()

Data Element	Datatype	Dis Len	Help	LOV	Val/ Range	Not Null	Ins
IDENT_APPLICATION	CHAR	2	()	()	()	()	()
APPLICATION_ACRONYM	CHAR	10	()	()	()	()	()
NAME_APPLICATION	CHAR	40	()	()	()	()	()

Block: DATA_OWNER
 Title: Data Ownership

Help	Scroll	Text	Navigate	Ins	Del	Upd	Qry
()	()	()	()	()	()	()	()

Data Element	Datatype	Dis Len	Help	LOV	Val/ Range	Not Null	Ins
IDENT_OWNER	CHAR	12	()	()	()	()	()
DESCR_OWNER	CHAR	30	()	()	()	()	()
LEVEL_DATA_VALID	NUMBER	1	()	()	()	()	()
FLAG_TEXT	CHAR	1	()	()	()	()	()

Module Description

Module Help

Module Test Plan

TEST PLAN

Testing of the transaction is performed at five levels:

1. Developer testing.
2. Secondary testing.
3. System testing.
4. End user testing.
5. Production certification.

As each applicable section of this test plan is completed, the designated individual responsible for the section will sign the plan in the appropriate places. Each check off item will be completed. The status of the transaction will be updated in the data dictionary as the transaction progresses through the development and test steps.

Developer Testing

As each individual transaction is generated, the developer performs tests against the transaction as defined in the detail test plan. Upon successful completion of this testing, the transaction is made available for secondary testing.

Test status conditions are:

- () 1 - Transaction under development, untested.
- () Transaction added to system and security files.

Module Test Plan

- () Specific test plan/test results/test case added.
- () Test plan narrative script loaded.
- () 2 - Development complete, initial unit testing in progress.
- () Help added
- () LOV tested
- () 3 - Initial unit testing complete.

Developer 12/16/92 Date Complete 12/16/92

Secondary Testing

Secondary testing is performed by a person external to the development team. Tests are performed using a standardized test checklist. Initial testing is performed to ensure that the transaction meets the functional requirements of the transaction and satisfies the test plan narrative that follows.

- () 4 - Secondary testing in progress.
- () 5 - Secondary testing complete, problems found.
- () 6 - Problem resolution in progress.
- () 7 - Problem resolution complete.
- () 8 - Secondary testing complete, no problems found.

Tester jc Date Complete 2/12/93

When problems are found:

1. The tester passes the checklist to the project manager for review and action.
2. The project manager assigns the checklist to the developer of the transaction for correction of the problems.
3. The developer corrects the described problems, records the corrective actions taken and the number of hours of corrective effort used, tests the transaction for correct operation, and returns the checklist to the project manager.
4. The project manager returns the checklist to the tester for retesting.

The above sequence of events repeats until the tester signs off the transaction as passing the test requirements.

System Testing

System testing is performed by a second independent tester. This tester repeats the test plan scenario, the test cases, and any other tests desired under end-user operating conditions.

- () 9 - System testing in progress.
- () 10 - System testing complete, problems found.
- Tester _____ Date Complete _____
- () 11 - Problem resolution in progress.

Module Test Plan

- () 12 - Problem resolution complete.
 Developer _____ Date Complete _____
- () 13 - System testing complete, no problems found.

When problems are found, the same corrective action and retest event sequence as defined in secondary testing above are performed.

End User Testing

The transaction is tested by end users. When appropriate, problem resolution takes place as in previous test segments.

Production Status

() 19 - The Project Manager reviews all test results and determines that the transaction has successfully completed testing.

- () Reviews tests results.
 () Reviews problem reports to ensure all corrections done.
 () Updates NOTES text to reflect testing results.
 Project Manager R. J. A. Date Complete _____

() 20 - The transaction is placed into the production environment.

- () Software files in production directories.
 () DMR 550 updated to add transaction.
 () DMR 900 updated to add transaction.
 () Transaction added to production usage menu.
 System Architect [Signature] Date Complete 2/12/93

Module Test Case

- for Data Ownership

- ✓ Attempt to store duplicate records in the database. Duplicate records are those where the unique identifier or any unique field contains duplicate values over 2 or more rows.
- (✓) Duplicate row insertion attempts result in error message COM00102 [Duplicate value, enter new value or F1 for help]
- () Display transaction help. Verify that the narrative adequately describes the purpose and functions of the transaction in end-user terms.
- 729/11 ✓ () Transaction help describes:
 () Purpose of the transaction
 () Functions performed by the transaction
 () Where multiple page, what happens on each page
- ✓ Add a new record by using the insert key.
- (✓) The insert key clears all blocks of the entry screen.

NO transaction help

- (/) A new record can be successfully added and committed.
- (/) Add a new record by scrolling past the last entry using the down arrow key.
 - (/) Synchronization of all blocks occurs while scrolling to the end of the string of retrieved records.
 - (/) All fields of the transaction blank when the end of the string of retrieved records is reached.
 - (/) A new data record can be successfully added.
- (/) Query a single record.
 - (/) A single record is retrieved.
 - (/) Pressing ~~SHIFT~~/F12 clears the record and permits entering a second query.
 - (/) The second query successfully retrieves the desired data.
- (/) Query multiple records using wild cards.
 - (/) Pressing F12 clears the screen and makes the transaction ready for a query.
 - (/) A query composed of a trailing % wildcard retrieves the desired data. Verify by running the same query through
 - (/) A query using a prefix % wildcard retrieves the correct data. Verify by running the same query under SQL*Plus.
 - (/) A query using both prefix and suffix % wildcards retrieves the correct data. Verify by running the same query through SQL*Plus.
- (/) Change each field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - (/) Change each field individually, committing after each change.
 - (/) Change each field, as per the field test plan, for both valid and invalid values.
- (/) Delete an existing record.
 - (/) The selected record is deleted from the database. Verify by using SQL*Plus to attempt to retrieve the record.
- (/) Display the -JN journal record rows for the table to verify that the records changed and deleted above are entered into the journal table correctly.
 - (/) The _JN journal table for the affected table should contain rows for the initial addition of the record to the database, each change made to the database, and the deletion of the record from the database.

() Use SQL*Plus to verify all data fields are stored correctly, including ID_ keys, date-updated, owner role, and any other autogenerated fields.

() All data fields are correctly stored in the database.

Block: DATA_OWNER
Title: Data Ownership

Block Description

Block Test Case

Block Test Results Expected

Field Name: IDENT_OWNER

Field Description

Field Test Case

(✓) Begin field with a space.

(✓) Enter data beyond length of field.

() If autoskip is in effect, cursor will skip to the next field.

() If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.

(✓) Set keyboard on lower case and enter data. Data should be converted to upper case.

(✓) Clear an existing field and attempt to update the record.

Field Test Results Expected

(✓) Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.

() Field help describes:

(✓) Purpose of the field

() LOV reference if available

Block: DATA_OWNER
Title: Data Ownership

Field Name: IDENT_OWNER

Field Test Results Expected

- ☒ Valid range of values where present
- ☒ Required indicator when field is NOT NULL
- ☐ Default value when appropriate
- ☒ Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
- ☒ Change the field, committing after each change.
- ☐ Change the field where a range of valid values exists:
 - ☐ Value below valid range should display message COM00101 or COM00103 (if no LOV).
- N/A ☐ Store a series of records using the SHIFT/F6 [duplicate field] key to carry forward field values from the previously stored record.
- ☐ Use SQL*Plus to verify all data fields are processed correctly.
- ☐ Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - ☐ Change the field, committing after each change.
 - ☐ Change the field where a range of valid values exists:
 - ☐ Value below valid range should display message COM00101 or COM00103 (if no LOV).
- ☐ Store a series of records using the SHIFT/F7 [duplicate field] key to carry forward field values from the previously stored record.
- ☐ Display the -JN journal record rows for the table to verify that the records changed and deleted above are entered into the journal table correctly.
 - ☐ The _JN journal table for the affected table should contain rows for the initial addition of the record to the database, each change made to the database, and the deletion of the record from the database.
- ☐ Use SQL*Plus to verify all data fields are processed correctly.

Block: DATA_OWNER
Title: Data Ownership

Field Name: IDENT_OWNER

Field Test Results Expected

- ☐ () All data fields are correctly stored in the database.
- ☒ (✓) Any alphanumeric characters may be entered into the field.
- ☒ (✓) Delete an existing value.
- ☒ (✓) For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.
- ☒ (✓) Attempt to bypass the field. An error message should tell you that the field data must be entered.
- ☒ (✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should result.
- ☒ (✓) Attempt to enter data as lower case. ~~An error should result.~~ ^c

Block: DATA_OWNER
Title: Data Ownership

Field Name: DESCR_OWNER

Field Description

Field Test Case

- (/) Begin field with a space.
- (/) Enter data beyond length of field.
 - () If autoskip is in effect, cursor will skip to the next field.
 - (/) If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (/) Set keyboard on lower case and enter data. Data should be converted to upper case.
- (/) Clear an existing field and attempt to update the record.

Field Test Results Expected

- (/) Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.
 - (/) Field help describes:
 - (/) Purpose of the field
 - (/) LOV reference if available
 - (/) Valid range of values where present
 - (/) Required indicator when field is NOT NULL
 - (/) Default value when appropriate
- (/) Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - (/) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
- (/) Store a series of records using the SHIFT/F6 [duplicate field] key

Block: DATA_OWNER
Title: Data Ownership

Field Name: DESCR_OWNER

Field Test Results Expected

- to carry forward field values from the previously stored record.
- () Use SQL*Plus to verify all data fields are processed correctly.
- () Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
- () Change the field, committing after each change.
- () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
- () Store a series of records using the SHIFT/F7 [duplicate field] key to carry forward field values from the previously stored record.
- () Display the -JN journal record rows for the table to verify that the records changed and deleted above are entered into the journal table correctly.
 - () The _JN journal table for the affected table should contain rows for the initial addition of the record to the database, each change made to the database, and the deletion of the record from the database.
- () Use SQL*Plus to verify all data fields are processed correctly.
- () All data fields are correctly stored in the database.
- () Any alphanumeric characters may be entered into the field.
- () Delete an existing value.
 - () For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.
- () Attempt to bypass the field. An error message should tell you that the field data must be entered.
- () Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should result.

Block: DATA_OWNER
Title: Data Ownership

Field Name: DESCR_OWNER

Field Test Results Expected

(.) Attempt to enter data as lower case. ~~An error should result.~~

Block: DATA_OWNER
Title: Data Ownership

Field Name: LEVEL_DATA_VALID

Field Description

Field Test Case

- (/ Accept the default.
- (/ Override the default with a valid entry value.
- n/a () Override the default with an invalid data value.
- (/ Clear an existing field and attempt to update the record.
- (/ Attempt to enter alpha data.
- (/ Attempt to enter longer value than allowed.
- (/ Attempt to enter decimal value. An error will occur if no provision has been made for decimal data in the element definition of the table.
- (/ Attempt to enter negative value. No error should occur.

Field Test Results Expected

- (/ Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.
 - (/ Field help describes:
 - (/ Purpose of the field
 - (/ LOV reference if available
 - (/ Valid range of values where present
 - (/ Required indicator when field is NOT NULL
 - (/ Default value when appropriate
- (/ Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - (/ Change the field, committing after each change.

Block: DATA_OWNER
Title: Data Ownership

Field Name: LEVEL_DATA_VALID

Field Test Results Expected

- () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
- (/ Store a series of records using the SHIFT/F6 [duplicate field] key to carry forward field values from the previously stored record.
- () Use SQL*Plus to verify all data fields are processed correctly.
- () Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - () Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
- () Store a series of records using the SHIFT/F7 [duplicate field] key to carry forward field values from the previously stored record.
- () Display the -JN journal record rows for the table to verify that the records changed and deleted above are entered into the journal table correctly.
 - () The _JN journal table for the affected table should contain rows for the initial addition of the record to the database, each change made to the database, and the deletion of the record from the database.
- () Use SQL*Plus to verify all data fields are processed correctly.
 - () All data fields are correctly stored in the database.
- (/ Modify each default value with a valid alternative.
- (/ Modify each default value with an invalid alternative.
- (/ Delete an existing value.
 - (/ For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.

Application: GENERIC Application Maintenance

Transaction: GE2610U1 Maintain Data Ownership

```
=====
Comment ID : 657      Priority: LOW      Date: 30-NOV-92
=====
Status      : FIXED      Date: 12-FEB-93      Class:
Type        : PROBLEM
Assigned To : KEN ATKINS
Entered By  : TODD ADAMS (D)
System ID   : 4459      Oracle ROWID: 00000A47.0000.000B
Last Error  : -0
Last Msg    : FRM-40202 Field must be entered.
Table       : DATA OWNER
Block       : DATOWN
Field       : DATOWN.IDENT_OWNER
=====
```

DESCRIPTION

No transaction level help.

CORRECTIVE ACTION

2/12/93 - Transaction help added.

2.13 GE3110U1 FUNCTION KEY HELP MAINTENANCE

LEMIS Module Test Checklist Report

Name: GE3110U1 - FUNCTION KEY HELP M
 Short Name: GE3110U1
 Purpose: To maintain the Function Key Help information
 Initial Test Date: 12/18/92 *RC*
 Verification Date: 2/12/93

	Disp	----- Security -----			
Help	Fkeys	Ins	Del	Query	Upd
()	()	()	()	()	()

Block: KEY_HELP
 Title: Function Key Help

Help	Scroll	Text	Navigate	Ins	Del	Upd	Qry
()	()	()	()	()	()	()	()

Data Element	Datatype	Dis Len	Help	LOV	Val/ Range	Not Null	Ins
TYPE_TERMINAL	CHAR	10	()	()	()	()	()
NAME_TRANSACTION	CHAR	8	()	()	()	()	()
FOR_BLOCK	CHAR	20	()	()	()	()	()
KEYSTROKE	CHAR	15	()	()	()	()	()
TYPE_PROCEDURE	CHAR	1	()	()	()	()	()
KEY_PROCEDURE	CHAR	19	()	()	()	()	()
DESC_FUNCTION	CHAR	58	()	()	()	()	()

Block: TRANSACTION_LIST
 Title: Transaction

Help ()	Scroll ()	Text ()	Navigate ()	Ins ()	Del ()	Upd ()	Qry ()
Data Element	Datatype	Dis Len	Help	LOV	Val/ Range	Not Null	Ins
DESCR_TRANSACTION	CHAR	1	()	()	()	()	()

Module Description

Module Help

This transaction is used to add and maintain function key help information for the generic system's function key help facility.

Module Test Plan

TEST PLAN

Testing of the transaction is performed at five levels:

1. Developer testing.
2. Secondary testing.
3. System testing.
4. End user testing.
5. Production certification.

As each applicable section of this test plan is completed, the designated individual responsible for the section will sign the plan in the appropriate places. Each check off item will be completed. The status of the transaction will be updated in the data dictionary as the transaction progresses through the development and test steps.

Developer Testing

As each individual transaction is generated, the developer performs tests against the transaction as defined in the detail test plan. Upon successful completion of this testing, the transaction is made available for secondary testing.

- () 1 - Transaction under development, untested.
- () Transaction added to system and security files.
- () Specific test plan/test results/test case added.

Module Test Plan

- () Test plan narrative script loaded.
- () 2 - Development complete, initial unit testing in progress.
- () Help added
- () LOV tested
- () 3 - Initial unit testing complete.
- Developer AK Date Complete 12/14/92

Secondary Testing

Secondary testing is performed by a person external to the development team. Tests are performed using a standardized test checklist. Initial testing is performed to ensure that the transaction meets the functional requirements of the transaction and satisfies the test plan narrative that follows.

- () 4 - Secondary testing in progress.
- () 5 - Secondary testing complete, problems found.
- () 6 - Problem resolution in progress.
- () 7 - Problem resolution complete.
- () 8 - Secondary testing complete, no problems found.
- Tester 2/12/93 Date Complete AK

When problems are found:

1. The tester passes the checklist to the project manager for review and action.
2. The project manager assigns the checklist to the developer of the transaction for correction of the problems.
3. The developer corrects the described problems, records the corrective actions taken and the number of hours of corrective effort used, tests the transaction for correct operation, and returns the checklist to the project manager.
4. The project manager returns the checklist to the tester for retesting.

The above sequence of events repeats until the tester signs off the transaction as passing the test requirements.

System Testing

System testing is performed by a second independent tester. This tester repeats the test plan scenario, the test cases, and any other tests desired under end-user operating conditions.

Test status conditions are:

- () 9 - System testing in progress.
- () 10 - System testing complete, problems found.
- Tester _____ Date Complete _____
- () 11 - Problem resolution in progress.

Module Test Plan

- () 12 - Problem resolution complete.
 Developer _____ Date Complete _____
- () 13 - System testing complete, no problems found.

When problems are found, the same corrective action and retest event sequence as defined in secondary testing above are performed.

End User Testing

The transaction is tested by end users. When appropriate, problem resolution takes place as in previous test segments.

Production Status

- () 19 - The Project Manager reviews all test results and determines that the transaction has successfully completed testing.
- () Reviews tests results.
 - () Reviews problem reports to ensure all corrections done.
 - () Updates NOTES text to reflect testing results.
- Project Manager ROB Date Complete _____
- () 20 - The transaction is placed into the production environment.
- () Software files in production directories.
 - () DMR 550 updated to add transaction.
 - () DMR 900 updated to add transaction.
 - () Transaction added to production usage menu.
- System Architect A Date Complete 2/12/93

Module Test Case

- (/ Attempt to store duplicate records in the database. Duplicate records are those where the unique identifier or any unique field contains duplicate values over 2 or more rows.
- (/ Duplicate row insertion attempts result in error message COM00102 [Duplicate value, enter new value or F1 for help]
- (/ Display transaction help. Verify that the narrative adequately describes the purpose and functions of the transaction in end-user terms.
- (/ Transaction help describes:
- (/ Purpose of the transaction
 - () Functions performed by the transaction
 - () Where multiple page, what happens on each page
- (/ Add a new record by using the insert key.
- (/ The insert key clears all blocks of the entry screen.

- (/) A new record can be successfully added and committed.
- (/) Add a new record by scrolling past the last entry using the down arrow key.
 - (/) Synchronization of all blocks occurs while scrolling to the end of the string of retrieved records.
 - (/) All fields of the transaction blank when the end of the string of retrieved records is reached.
 - (/) A new data record can be successfully added.
- (/) Query a single record.
 - (/) A single record is retrieved.
 - (/) Pressing ~~SHIFT~~/F12 clears the record and permits entering a second query.
 - (/) The second query successfully retrieves the desired data.
- () Query multiple records using wild cards.
 - (/) Pressing F12 clears the screen and makes the transaction ready for a query.
 - (/) A query composed of a trailing % wildcard retrieves the desired data. Verify by running the same query through SQL*Plus.
 - (/) A query using a prefix % wildcard retrieves the correct data. Verify by running the same query under SQL*Plus.
 - (/) A query using both prefix and suffix % wildcards retrieves the correct data. Verify by running the same query through
- (/) Change each field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - (/) Change each field individually, committing after each change.
 - () Change each field, as per the field test plan, for both valid and invalid values.
- (/) Delete an existing record.
 - (/) The selected record is deleted from the database. Verify by using SQL*Plus to attempt to retrieve the record.
- () Display the -JN journal record rows for the table to verify that the records changed and deleted above are entered into the journal table correctly.
 - () The _JN journal table for the affected table should contain rows for the initial addition of the record to the database, each change made to the database, and the deletion of the record from the database.

() Use SQL*Plus to verify all data fields are stored correctly, including ID_ keys, date-updated, owner role, and any other autogenerated fields.

() All data fields are correctly stored in the database.

Block: KEY_HELP
Title: Function Key Help

Block Description

Block Test Case

Block Test Results Expected

Field Name: TYPE_TERMINAL

Field Description

The type of terminal that the user is using. This is needed to display a keystroke that will match what the user sees on their keys.

Field Test Case

n/x () Begin field with a space.

- () Enter data beyond length of field.
 - () If autoskip is in effect, cursor will skip to the next field.
 - () If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.

✓ () Set keyboard on lower case and enter data. Data should be converted to upper case.

✓ () Clear an existing field and attempt to update the record.

Field Test Results Expected

✓ () Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.

() Field help describes:

Block: KEY_HELP
Title: Function Key Help

Field Name: TYPE_TERMINAL

Field Test Results Expected

- ☒ Purpose of the field
- ☐ LOV reference if available
- ☒ Valid range of values where present
- ☒ Required indicator when field is NOT NULL
- ☒ Default value when appropriate
- ☒ Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - ☒ Change the field, committing after each change.
 - ☐ Change the field where a range of valid values exists:
 - ☐ Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - ☐ Value above valid range should display message COM00101 or *no*
- ☒ Store a series of records using the SHIFT/*fl* [duplicate field] key to carry forward field values from the previously stored record.
- ☐ Display the -JN journal record rows for the table to verify that the records changed and deleted above are entered into the journal table correctly.
 - ☐ The -JN journal table for the affected table should contain rows for the initial addition of the record to the database, each change made to the database, and the deletion of the record from the database.
- ☐ Use SQL*Plus to verify all data fields are processed correctly.
 - ☐ All data fields are correctly stored in the database.
- ☒ Any alphanumeric characters may be entered into the field.
- attempt to TAB to the next field. An error should result.
- ☒ Attempt to enter data as lower case. ~~An error should result.~~
- ☒ Delete an existing value.

Block: KEY_HELP
Title: Function Key Help

Field Name: TYPE_TERMINAL

Field Test Results Expected

- (✓) For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.
- (✓) Attempt to bypass the field. An error message should tell you that the field data must be entered.
- () Enter data into the field, backspace the data out of the field and

Block: KEY_HELP
 Title: Function Key Help

Field Name: NAME_TRANSACTION

Field Description

This is the 8 character name of the transaction. Each transaction can have its own key help entries. If 'ALL' is entered here this entry will be displayed for all transactions.

Field Test Case

- () TAB through the field. Backtab to the field, enter data. After entering data in the next field, backtab to the field, and backspace the data from the field. The database should contain a null value for the field.
- () Enter data in the field. After entering data in the next field, backtab to the field, and space through the data. The database should contain a null value for the field.

*Not Null
field*

11A () Begin field with a space.

- (✓) Enter data beyond length of field.
 - () If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Set keyboard on lower case and enter data. Data should be converted to upper case.
- (✓) Clear an existing field and attempt to update the record.

Field Test Results Expected

- (✓) Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.

- 735
 x [signature]
- (✓) Field help describes:
 - (✓) Purpose of the field
 - () LOV reference if available
 - (✓) Valid range of values where present
 - (✓) Required indicator when field is NOT NULL
 - () Default value when appropriate

*help - optional Blank
 - no Blank allowed*

Block: KEY_HELP
Title: Function Key Help

Field Name: NAME_TRANSACTION

Field Test Results Expected

- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or ^{FL}
- (✓) Store a series of records using the SHIFT/~~F7~~ [duplicate field] key to carry forward field values from the previously stored record.
- () Display the -JN journal record rows for the table to verify that the records changed and deleted above are entered into the journal table correctly.
 - () The -JN journal table for the affected table should contain rows for the initial addition of the record to the database, each change made to the database, and the deletion of the record from the database.
- () Use SQL*Plus to verify all data fields are processed correctly.
 - () All data fields are correctly stored in the database.
- () Bypass the field by using the TAB key.
 - () No error should occur.
 - () The database contains a null value for the field.
- () Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
- () Delete an existing value. The field should be cleared and stored as a null value.
- (✓) Any alphanumeric characters may be entered into the field.

not null fields

attempt to TAB to the next field. An error should result.

Block: KEY_HELP
Title: Function Key Help

Field Name: NAME_TRANSACTION

Field Test Results Expected

- (✓) Attempt to enter data as lower case. ~~An error should result.~~
- (✓) Delete an existing value.
 - (✓) For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.
- (✓) Attempt to bypass the field. An error message should tell you that the field data must be entered.
- (✓) Enter data into the field, backspace the data out of the field and

Block: KEY_HELP
Title: Function Key Help

Field Name: FOR_BLOCK

Field Description

Field Test Case

- (✓) Begin field with a space.
- () Enter data beyond length of field.
 - () If autoskip is in effect, cursor will skip to the next field.
 - () If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Set keyboard on lower case and enter data. Data should be converted to upper case.
- (✓) Clear an existing field and attempt to update the record.

Field does not overlay characters

Field Test Results Expected

- (✓) Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.
 - () Field help describes:
 - (✓) Purpose of the field
 - (✓) LOV reference if available
 - (✓) Valid range of values where present
 - () Required indicator when field is NOT NULL
 - () Default value when appropriate
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or
- (✓) Store a series of records using the SHIFT/^{F6}E7 [duplicate field] key

Block: KEY_HELP
Title: Function Key Help

Field Name: FOR_BLOCK

Field Test Results Expected

- to carry forward field values from the previously stored record.
- () Display the -JN journal record rows for the table to verify that the records changed and deleted above are entered into the journal table correctly.
- () The _JN journal table for the affected table should contain rows for the initial addition of the record to the database, each change made to the database, and the deletion of the record from the database.
- () Use SQL*Plus to verify all data fields are processed correctly.
- () All data fields are correctly stored in the database.
- (/ Any alphanumeric characters may be entered into the field.
- (/ Attempt to enter data as lower case. ~~An error should result.~~
- (/ Delete an existing value.
 - (/ For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.
- (/ Attempt to bypass the field. An error message should tell you that the field data must be entered.
- (/ Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should result.

Block: KEY_HELP
Title: Function Key Help

Field Name: KEYSTROKE

Field Description

Enter the keystroke that this entry represents. It should be entered just as it appears on the user's keyboard.

Field Test Case

- () Begin field with a space.
- () Enter data beyond length of field.
 - () If autoskip is in effect, cursor will skip to the next field.
 - () If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- n/a () Set keyboard on lower case and enter data. Data should be converted to upper case.
- () Clear an existing field and attempt to update the record.

Field Test Results Expected

- () Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.
 - () Field help describes:
 - () Purpose of the field
 - () LOV reference if available
 - () Valid range of values where present
 - () Required indicator when field is NOT NULL
 - () Default value when appropriate
- () Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - () Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or

Block: KEY_HELP
Title: Function Key Help

Field Name: KEYSTROKE

Field Test Results Expected

- () Store a series of records using the SHIFT/^{fl}F7 [duplicate field] key to carry forward field values from the previously stored record.
- () Display the -JN journal record rows for the table to verify that the records changed and deleted above are entered into the journal table correctly.
 - () The _JN journal table for the affected table should contain rows for the initial addition of the record to the database, each change made to the database, and the deletion of the record from the database.
- () Use SQL*Plus to verify all data fields are processed correctly.
 - () All data fields are correctly stored in the database.
- (✓) Any alphanumeric characters may be entered into the field.
- (✓) attempt to TAB to the next field. An error should result.
- (✓) Delete an existing value.
 - (✓) For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.
- (✓) Attempt to bypass the field. An error message should tell you that the field data must be entered.
- (✓) Enter data into the field, backspace the data out of the field and

Block: KEY_HELP
Title: Function Key Help

Field Name: TYPE_PROCEDURE

Field Description

This is the type of procedure. If 'K' is entered the SQL*FORMS procedure specified in NAME_PROCEDURE will be executed. If 'P' is entered then the specified custom procedure will be called.

****NOTE**** All procedures used in the 'P' entries must be hard coded in the show_key_help procedure.

Field Test Case

() TAB through the field. Backtab to the field, enter data. After entering data in the next field, backtab to the field, and backspace the data from the field. The database should contain a null value for the field.

11/18 () Enter data in the field. After entering data in the next field, backtab to the field, and space through the data. The database should contain a null value for the field.

936 () Accept the default. - no Default - no hint text forced

11/18 () Override the default with a valid entry value.

() Override the default with an invalid data value.

() Begin field with a space.

() Enter data beyond length of field.

() If autoskip is in effect, cursor will skip to the next field.

() If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.

() Set keyboard on lower case and enter data. Data should be converted to upper case.

Field Test Results Expected

() Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.

Block: KEY_HELP
Title: Function Key Help

Field Name: TYPE_PROCEDURE

Field Test Results Expected

- (/) Field help describes:
 - (/) Purpose of the field
 - (/) LOV reference if available
 - (/) Valid range of values where present
 - (/) Required indicator when field is NOT NULL
 - (/) Default value when appropriate
- (/) Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - (/) Change the field, committing after each change.
 - (/) Change the field where a range of valid values exists:
 - (/) Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - (/) Value above valid range should display message COM00101 or
- (/) Store a series of records using the SHIFT/*File* [duplicate field] key to carry forward field values from the previously stored record.
- (/) Display the -JN journal record rows for the table to verify that the records changed and deleted above are entered into the journal table correctly.
 - (/) The -JN journal table for the affected table should contain rows for the initial addition of the record to the database, each change made to the database, and the deletion of the record from the database.
- (/) Use SQL*Plus to verify all data fields are processed correctly.
- (/) Bypass the field by using the TAB key.
 - (/) No error should occur.
 - (/) The database contains a null value for the field.
- NA* (/) Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
- (/) Delete an existing value. The field should be cleared and stored as a null value.

Block: KEY_HELP
Title: Function Key Help

Field Name: TYPE_PROCEDURE

Field Test Results Expected

- ☐ () Modify each default value with a valid alternative.
- ☐ () Modify each default value with an invalid alternative.
- ☐ () Any alphanumeric characters may be entered into the field.
- ☒ (X) Attempt to enter data as lower case. ~~An error should result.~~

Block: KEY_HELP
 Title: Function Key Help

Field Name: KEY_PROCEDURE

Field Description

This is the name of the procedure to be executed if the user presses <return> from the function key pop-up. It can be a standard forms procedure (such as NEXT_BLOCK) or a form level procedure (see procedure type)

Field Test Case

- (✓) TAB through the field. Backtab to the field, enter data. After entering data in the next field, backtab to the field, and backspace the data from the field. The database should contain a null value for the field.
- (✓) Enter data in the field. After entering data in the next field, backtab to the field, and space through the data. The database should contain a null value for the field.
- (✓) Begin field with a space.
- (✓) Enter data beyond length of field.
 - () If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Set keyboard on lower case and enter data. Data should be converted to upper case.

Field Test Results Expected

- (✓) Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.
 - () Field help describes:
 - () Purpose of the field
 - () LOV reference if available
 - () Valid range of values where present
 - () Required indicator when field is NOT NULL
 - () Default value when appropriate
- (✓) Change the field in an existing record. Verify that the change

Block: KEY_HELP
Title: Function Key Help

Field Name: KEY_PROCEDURE

Field Test Results Expected

occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.

- ☒ Change the field, committing after each change.
- ☐ Change the field where a range of valid values exists:
 - ☐ Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - ☐ Value above valid range should display message COM00101 or
- ☒ Store a series of records using the SHIFT/^{file}F7 [duplicate field] key to carry forward field values from the previously stored record.
- ☐ Display the -JN journal record rows for the table to verify that the records changed and deleted above are entered into the journal table correctly.
- ☐ The _JN journal table for the affected table should contain rows for the initial addition of the record to the database, each change made to the database, and the deletion of the record from the database.
- ☐ Use SQL*Plus to verify all data fields are processed correctly.
- ☒ Bypass the field by using the TAB key.
 - ☒ No error should occur.
 - ☒ The database contains a null value for the field.
- ☒ Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
- ☒ Delete an existing value. The field should be cleared and stored as a null value.
- ☒ Any alphanumeric characters may be entered into the field.
- ☒ Attempt to enter data as lower case. ~~An error should result.~~

Block: KEY_HELP
 Title: Function Key Help

Field Name: DESC_FUNCTION

Field Description

This is the description of the function to be performed. It will appear in the pop-up display of function key information.

Field Test Case

- (/) Begin field with a space.
- (/) Enter data beyond length of field.
 - () If autoskip is in effect, cursor will skip to the next field.
 - (/) If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (/) Set keyboard on lower case and enter data. Data should be ~~converted to upper case.~~ *field can be LC charac.*
- (/) Clear an existing field and attempt to update the record.

Field Test Results Expected

- (/) Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.
 - () Field help describes:
 - () Purpose of the field
 - () LOV reference if available
 - () Valid range of values where present
 - () Required indicator when field is NOT NULL
 - () Default value when appropriate
- (/) Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - (/) Change the field, committing after each change.
 - (/) Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or

Block: KEY_HELP
Title: Function Key Help

Field Name: DESC_FUNCTION

Field Test Results Expected

- (✓) Store a series of records using the SHIFT/F7 ^{Fl} [duplicate field] key to carry forward field values from the previously stored record.
- () Display the -JN journal record rows for the table to verify that the records changed and deleted above are entered into the journal table correctly.
 - () The _JN journal table for the affected table should contain rows for the initial addition of the record to the database, each change made to the database, and the deletion of the record from the database.
- () Use SQL*Plus to verify all data fields are processed correctly.
 - () All data fields are correctly stored in the database.
- (✓) Any alphanumeric characters may be entered into the field.
- (✓) Delete an existing value.
 - () For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.
- (✓) Attempt to bypass the field. An error message should tell you that the field data must be entered.
- (✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should result.

Block: TRANSACTION_LIST
Title: Transaction

Block Description

Block Test Case

Block Test Results Expected

Field Name: DESCR_TRANSACTION

Field Description

Field Test Case

- () Begin field with a space.
- () Enter data beyond length of field.
 - () If autoskip is in effect, cursor will skip to the next field.
 - () If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- () Set keyboard on lower case and enter data. Data should be converted to upper case.
- () Clear an existing field and attempt to update the record.

Field Test Results Expected

- () Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.
 - () Field help describes:
 - () Purpose of the field
 - () LOV reference if available
 - () Valid range of values where present
 - () Required indicator when field is NOT NULL
 - () Default value when appropriate
- () Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and

Block: TRANSACTION_LIST
Title: Transaction

Field Name: DESCR_TRANSACTION

Field Test Results Expected

after the changes were made.

- () Change the field, committing after each change.
- () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
- () Store a series of records using the SHIFT/F7^{fl} [duplicate field] key to carry forward field values from the previously stored record.
- () Display the -JN journal record rows for the table to verify that the records changed and deleted above are entered into the journal table correctly.
 - () The _JN journal table for the affected table should contain rows for the initial addition of the record to the database, each change made to the database, and the deletion of the record from the database.
- () Use SQL*Plus to verify all data fields are processed correctly.
 - () All data fields are correctly stored in the database.
- () Any alphanumeric characters may be entered into the field.
- () Delete an existing value.
 - () For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.
- () Attempt to bypass the field. An error message should tell you that the field data must be entered.
- () Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should result.

12-FEB-93

WHC-SD-WM-TP-165 REV 0
Programmer's Bug Report

Page 1 of 1

Application: GENERIC Application Maintenance

Transaction: GE3110U1 Function Key Help Maintenance

=====
Comment ID : 735 Priority: LOW Date: 18-DEC-92
=====

Status : FIXED Date: 12-FEB-93 Class:
Type : PROBLEM
Assigned To: KEN ATKINS
Entered By : LORRAINE CAMPBELL
System ID : Oracle ROWID: 00000C78.000A.000B
Last Error : -0
Last Msg : FRM-40202 Field must be entered.
Table : KEY_HELP
Block : KEYHLP
Field : KEYHLP.NAME_TRANSACTION

DESCRIPTION

help states field optional and can be left blank. It also states that
a blank is not allowed.

CORRECTIVE ACTION

2/12/93 - Help has been fixed.

Handwritten signature

Application: GENERIC Application Maintenance

Transaction: GE3110U1 Function Key Help Maintenance

=====

Comment ID : 736 Priority: LOW Date: 18-DEC-92

=====

Status : FIXED Date: 12-FEB-93 Class:

Type : PROBLEM

Assigned To: KEN ATKINS

Entered By: LORRAINE CAMPBELL

System ID : Oracle ROWID: 00000C78.000A.000B

Last Error : -0

Last Msg : FRM-40204 Cursor is at beginning of field value.

Table : KEY_HELP

Block : KEYHLP

Field : KEYHLP.TYPE_PROCEDURE

DESCRIPTION

help states that there is a default but cannot accept it with an enter
nor tab, field is left blank. There is no hint text for this field

CORRECTIVE ACTION

2/12/93 - Fixed.

2.14 GE3220U1 MAINTAIN SYSTEM COMMENTS

LEMIS Module Test Checklist Report

Name: GE3220U1 - MAINTAIN SYSTEM COM
Short Name: GE3220U1
Purpose: To query, modify, and Delete system comments.
Initial Test Date: 2/22 JC
Verification Date: 2/26 JC 3/3 JC

Module Description

This transaction is used to display, (and possibly delete or modify) system comments that have been entered.

Module Help

This transaction is used to display, assign, and dispose of system comments.

To assign new comments to a programmer:

- 1) Press <F12> to enter query mode.
- 2) Press <Enter> or <Tab> until the cursor is in the status field.
- 3) Type "NEW" into the status field and press <Shift F12> to query the data. This will list all of the NEW problems/comments.
- 4) Read the text of the comment and verify that the following fields were entered correctly, and fix them if necessary:
 - a) Type: This should only be set to PROBLEM if this is really a bug and not a change request or comment.
 - b) Priority: Verify that the priority makes sense.
 - c) Transaction: Verify that the transaction is correct for this comment. Sometimes the users will enter the comment from the wrong transaction or from the menu.

Module Help

- 5) Move the cursor to the "Assigned" field and press <F9>. This will display the Person Directory.
- 6) Select the appropriate person to assign this problem to. Press <F10> to return the name to the System Comments form.
- 7) Press <F10> to commit the change. The status of the comment will automatically be set to "ASSIGNED".

To dispose a resolved problem/comment:

- 1) Press <F12> to enter query mode, and query the desired comment, using the comment ID or transaction name for the selection.
- 2) Move the cursor to the "Disposition" field and press <F9> to select the desired disposition. The status of the comment will be set to "HANDLED".

A corrective action should be entered when the comment is handled. This should be done for all disposition codes except for NOTPROB and REPEAT.

- 3) Press <PageDown> to move the cursor into the "Text Type" block.
- 4) Press <Insert> to add a new text record to this comment.
- 5) Enter "COR" for corrective action, and press <PageDown> to add the text for the corrective action.
- 6) Enter a brief description of the bug fix or other method of handling the problem/comment. Make sure and enter the date this was done.
- 7) Press <F10> to save the text.
- 8) Press <F10> to commit the new text record.

To delete a comment:

- 1) Press <F12> to enter query mode, and query the desired comment, using the comment ID or transaction name for the selection.
- 2) Press <PageDown> to move to the text record block.
- 3) Press <Delete> to delete the comment description record.
- 4) Press <F10> to commit this deletion.
- 5) Press <PageUp> to move back to the comment / bugs block.
- 6) Press <Delete> to delete the comment / bug.
- 7) Press <F10> to commit your deletion.

UNIT TEST CHECKLIST

() Module	Function Key
() Show Version	: Shift-F10 :
() Comment/Bug	: Shift-F3 :
() Fast Access	: F5 :
() Back	: F4 or Esc :
() Main Menu	: Shift-F4 :
() Key Template	: Ctrl-F1 :
() Context Prnt	: F11 :
() Query Print	: Ctrl-F11 :
() Oracle Print	: Shift-F11 :

() Block Title: Comments / Bugs
 Block Table: SYSTEM_COMMENT

(✓) Help : F1	:	(✓) Key Help : Shift-F1
(✓) Online Doc : F2	:	(✓) Dis Journal : Shift-F2
(✓) Related Data : F3	:	(✓) Edit Text : F7
(✓) Spec Func 1 : Shift-F5	:	(✓) Spec Func 2 : Ctrl-F5
<hr/>		
(✓) Home/End : Home/End	:	(✓) Prv/Nxt Block: PgDn/PgUp:
(✓) Scroll Up/Dn : Shift-Up/Dn	:	(✓) Prv/Nxt Field: Tab/Sh-Tb:
(✓) Prv/Nxt Rec : Up/Down	:	(✓) Commit Form : F10
<hr/>		
(✓) Clear Field : F8	:	(✓) Clear Record : Shift-F8
(✓) Clear Form : Shift-F8	:	(✓) Exit System : Ctrl-F4
<hr/>		
(✓) Enter Query : F12	:	(✓) Count Query : Ctrl-F12
(✓) Correct Query Fields?	:	(✓) Uppercase and LOVs?
(✓) Execute Qry : Shift-F12	:	(✓) Correct Query Order?
<hr/>		
(✓) Insert Rec : Insert	:	(✓) Add record at end
(✓) Mandatory checks OK?	:	(✓) Validation checks OK?
(✓) List of Values OK?	:	(✓) Duplicate Fld: Shift-F6
(✓) Duplicate Rec: F6	:	(✓) Duplicate checks OK?
(✓) Optional fields marked?	:	(✓) Correct fields uppercase:
<hr/>		
(✓) Delete Record: Delete	:	

() Block Title: Related Text
 Block Table: GENERIC_TEXT

(✓) Help : F1	:	(✓) Key Help : Shift-F1
(✓) Online Doc : F2	:	(✓) Dis Journal : Shift-F2
(✓) Related Data : F3	:	(✓) Edit Text : F7
(✓) Spec Func 1 : Shift-F5	:	(✓) Spec Func 2 : Ctrl-F5
<hr/>		
(✓) Home/End : Home/End	:	(✓) Prv/Nxt Block: PgDn/PgUp:
(✓) Scroll Up/Dn : Shift-Up/Dn	:	(✓) Prv/Nxt Field: Tab/Sh-Tb:
(✓) Prv/Nxt Rec : Up/Down	:	(✓) Commit Form : F10
<hr/>		
(✓) Clear Field : F8	:	(✓) Clear Record : Shift-F8
(✓) Clear Form : Shift-F8	:	(✓) Exit System : Ctrl-F4
<hr/>		
(✓) Enter Query : F12	:	(✓) Count Query : Ctrl-F12
(✓) Correct Query Fields?	:	(✓) Uppercase and LOVs?
(✓) Execute Qry : Shift-F12	:	(✓) Correct Query Order?
<hr/>		
(✓) Insert Rec : Insert	:	(✓) Add record at end
(✓) Mandatory checks OK?	:	(✓) Validation checks OK?
(✓) List of Values OK?	:	(✓) Duplicate Fld: Shift-F6
(✓) Duplicate Rec: F6	:	(✓) Duplicate checks OK?
(✓) Optional fields marked?	:	(✓) Correct fields uppercase:
<hr/>		
(✓) Delete Record: Delete	:	

TEST PLAN

Testing of the transaction is performed at four levels:

1. Developer testing.
2. Secondary testing.
3. End user testing.
4. Production release & verification.

As each applicable section of this test plan is completed, the designated individual responsible for the section will sign the plan in the appropriate places. Each check off item will be completed. The status of the transaction will be updated in the data dictionary as the transaction progresses through the development and test steps.

Developer Testing

As each individual transaction is generated, the developer performs tests against the transaction to insure that it meets specifications and works properly. Upon successful completion of this testing, the transaction is made available for secondary testing.

Test status conditions are:

- (☒) 1 - Transaction under development, untested.
 - (☒) Transaction added to menu and security.
 - (☒) Transaction moved to development directory (addmod).
- (☒) 2 - Development complete, initial unit testing in progress.
 - (☒) Help added
 - (☒) Unit Test checklist completed.
 - (☒) Specific test plan added (if needed).
 - (☒) Any bugs found in Unit test fixed.
 - (☒) Updated test booklet printed.
- (☒) 3 - Initial unit testing complete.
 Developer *[Signature]* Date Complete 2/19/93

Secondary Testing

Secondary testing is performed by a person external to the development team. Tests are performed using a standardized test checklist. Initial testing is performed to ensure that the transaction meets the functional requirements of the transaction and satisfies the test plan narrative that follows.

- (☒) 4 - Secondary testing in progress.
- (☒) 5 - Secondary testing complete, problems found. Resolution in progress.
- (☐) 6 - Unable to resolve problem at this time.
- (☒) - Problem resolution complete.

- (☒) 8 - Secondary testing complete, no problems found.
 - Tester RC Date Complete 2/26/93

When problems are found:

1. The tester passes the checklist to the project manager for review and action.
2. The project manager assigns the checklist to the developer of the transaction for correction of the problems.
3. The developer corrects the described problems, records the corrective actions taken and the number of hours of corrective effort used, tests the transaction for correct operation, and returns the checklist to the project manager.
4. The project manager returns the checklist to the tester for retesting.

The above sequence of events repeats until the tester signs off the transaction as passing the test requirements.

End User Testing

The transaction is tested by end users for a predetermined time period. This testing is done in the development environment. If no problems are encountered, or all problems are resolved at the end of this time period, the module is ready for Project Manager review.

- () 9 - End User testing in progress.
 () 10 - Problem found, resolution in progress.
 () 11 - Problem resolution complete.
 Developer _____ Date Complete _____
 () 12 - Resolution verified.
 Tester _____ Date Complete _____

Production Status

- (☒) 13 - Ready for Project Manager/System Architect review.
 () The Project Manager or System Architect reviews all test results and determines that the transaction has completed testing.
 Architect/Manager RA Date Complete 2/26/93
- (☒) 14 - The transaction is ready to be placed into production.
 (☒) Software files moved to production directory.
 (☒) Transaction added to production menu.
 (☒) Transaction access given to production roles.
 (☒) Validation test performed in production.
 () Production documentation up to date.
- (☒) 15 - The transaction has been put into production

By RA Date Done 3/10/93

- (✓) Display transaction help. Verify that the narrative adequately describes the purpose and functions of the transaction in end-user terms.
 - (✓) Transaction help describes:
 - (✓) Purpose of the transaction
 - (✓) Functions performed by the transaction
 - (✓) Where multiple block, what happens in each block.
- (✓) Test the ability to delete a comment. See Module Help for procedure.

Block Title: Comments / Bugs
Block Table: SYSTEM COMMENT
Block Usage: Full Function

Block Help

You may store comments and bug reports into this table. This information will be reported to the application manager and if needed to the system development team.

Block Test Plan

- (✓) Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms:
 - (✓) Block help describes:
 - (✓) Purpose of the block
 - (✓) Options available in the block
 - (✓) The relationship of the block to the transaction or to the principal parent block of the transaction
 - (✓) Navigation to other blocks of the transaction
- () Where block synchronization is used, ensure that the block data is in sync with the parent block.
- (✓) Query a single record:
 - (✓) A single record is retrieved.
 - (✓) Pressing the F12 key clears the record and permits entering a second query.
 - (✓) The second query successfully retrieves the desired data.
- (✓) Query multiple records using wild cards:

- (/) Pressing F12 clears the block and makes the block ready for a query.
- (/) A query composed of a trailing % wildcard retrieves the desired data.
- (/) A query using a prefix % wildcard retrieves the correct data.
- (/) A query using both prefix and suffix % wildcards retrieves the correct data.
- (/) A query using the underscore (_) position wildcard retrieves the correct data.
- (/) Verify that the data queries in the block are ordered logically.
- (/) Store a series of records using the F6 [duplicate record] key to carry forward field values from the previously stored record.
- () Attempting to store the duplicated record without change should result in an error.
 - () Change key fields and store the record. It should store successfully.
- (/) Update a record:
 - (/) The change count should increment.
 - () The date updated should become the current date.
 - () An error should occur if the operator owner identifier does not match the data owner identifier.
- (/) Delete a record from the block:
 - () If the record has dependent records, the delete should fail unless the help text indicates that a cascading delete is permitted.
 - (/) Use the clear rcd key (CTRL/F8).
 - () For a single record block, the block should clear.
 - () For a multi-record block, the record pointed to by the cursor should clear.
 - (/) Use the delete key.
 - () For a single record block, the block should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should appear in the block.
 - (/) For a multiple record block, the entry should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should move up to fill the cleared space.

Field Prompt: Disposition
 Field Usage: Updateable
 Field Name: COMMENT_DISPOSITION

Field Help

This code specifies the final disposition of this comment.

Field Help

List of Values is available.

NOTE: When a disposition is entered, the description must also be entered.

Field Test Plan

-
- (/) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
 - (/) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (/) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
 - (/) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - (/) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (/) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
 - (/) Enter query mode and verify that this field is accessible.
 - (/) Enter query mode and enter a valid value in this field.
 - (/) Only records matching the entered value should be displayed.
 - (/) Skip the field using the TAB key.
 - (/) Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
 - (/) Delete an existing value. The field should be cleared and stored:
 - (/) Delete the value using the space bar.
 - (/) Delete the value using the clear field key (F8).

- (X) Test LOV:
 - () Select the first entry in the LOV.
 - (X) Select the last entry in the LOV.
 - () Use the FIND option and select an entry.

no testing for
Assigned
Date ?

Field Prompt: Description
 Field Usage: Updateable
 Field Name: DESCRIPTION

Field Help

A brief description of this comment/problem. Used to make it easier to identify specific comments. Enter "." to default this field to the first 60 characters of the description of the bug.

Field Test Plan

- (X) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (X) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (X) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (X) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (X) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (X) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (X) Enter query mode and verify that this field is accessible.
- (X) Enter query mode and enter a valid value in this field.
 - (X) Only records matching the entered value should be displayed.
- (X) Skip the field using the TAB key.
- (X) Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
- (X) Delete an existing value. The field should be cleared and stored:

Handwritten notes:
 OK
 880
 NO Hint put
 FI bring up Hint to
 FI again brings up
 HLP to

- (✓) Delete the value using the space bar.
- (✓) Delete the value using the clear field key (F8).

Field Prompt: Status
 Field Usage: Updateable
 Field Name: COMMENT_STATUS

Field Help

 This field determines the status of this comment/problem. Use List of Values to see valid stati.

This data field:

- Provides a default value. To accept the default value, press the ENTER key or TAB key.
- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

-
- (/) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
 - (/) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (/) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
 - (/) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - (/) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (/) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
 - (/) Enter query mode and verify that this field is accessible.
 - (/) Enter query mode and enter a valid value in this field.
 - (/) Only records matching the entered value should be displayed.
 - () Accept the default.

881
some or 880
JKR

- (X) Modify each default value with a valid alternative.
- () Modify each default value with an invalid alternative.
- (X) Attempt to delete an existing data field value. An error should result.
 - () Use the space bar to attempt to delete the field data.
 - () Use clear field (F8) to attempt to delete the field data.
- (X) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- (X) Test LOV:
 - (X) Select the first entry in the LOV.
 - (X) Select the last entry in the LOV.
 - (X) Use the FIND option and select an entry.

ck
879 error msg.
wrong

Field Prompt: Priority
 Field Usage: Updateable
 Field Name: PRIORITY

Field Help

Enter the priority for this bug or comment. Use LOV to see valid values. Use the priority to indicate how important this comment is for you to continue using the system.

This data field:

- Is optional and may be left empty.
- Provides a default value. To accept the default value, press the ENTER key or TAB key.
- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

- (/ Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (/ Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (/ If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (/ Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (/ Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - () Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (/ Enter query mode and verify that this field is accessible.
- (/ Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.

- (X) Accept the default.
- (X) Modify each default value with a valid alternative.
- (X) Modify each default value with an invalid alternative.
- (X) Attempt to delete an existing data field value. An error should result.
 - (X) Use the space bar to attempt to delete the field data.
 - (X) Use clear field (F8) to attempt to delete the field data.
- (X) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- (X) Test LOV:
 - (X) Select the first entry in the LOV.
 - (X) Select the last entry in the LOV.
 - (X) Use the FIND option and select an entry.

Field Prompt: Type
 Field Usage: Updateable
 Field Name: TYPE_COMMENT

*Not
 accessible*

Field Help

Enter COMMENT to indicate that this is only a comment, or suggestion.
 Enter BUG if you are reporting a problem.

This data field:

- Provides a default value. To accept the default value, press the ENTER key or TAB key.
- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

- () Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- () Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - () If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- () Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- () Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - () Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- () Enter query mode and verify that this field is accessible.
- () Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.
- () Accept the default.

- (☒) Modify each default value with a valid alternative.
- () Modify each default value with an invalid alternative.
- () Attempt to delete an existing data field value. An error should result.
 - () Use the space bar to attempt to delete the field data.
 - () Use clear field (F8) to attempt to delete the field data.
- () Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- () Test LOV:
 - () Select the first entry in the LOV.
 - () Select the last entry in the LOV.
 - () Use the FIND option and select an entry.

*no testing for
CMT ID
BY
DATE ?*

Field Prompt: Application:
 Field Usage: Updateable
 Field Name: IDENT_APPLICATION

Field Help

 This field contains the two character application ID.

This data field:

- Provides a default value. To accept the default value, press the ENTER key or TAB key.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

-
- () Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
 - (X) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (X) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
 - (X) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - (X) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (X) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
 - (X) Enter query mode and verify that this field is accessible.
 - (X) Enter query mode and enter a valid value in this field.
 - (X) Only records matching the entered value should be displayed.
 - () Accept the default.
 - (X) Modify each default value with a valid alternative.

- ☒ Modify each default value with an invalid alternative.
- ☒ Attempt to delete an existing data field value. An error should result.
 - ☒ Use the space bar to attempt to delete the field data.
 - ☒ Use clear field (F8) to attempt to delete the field data.
- ☒ Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- ☒ Test LOV:
 - ☒ Select the first entry in the LOV.
 - ☒ Select the last entry in the LOV.
 - ☒ Use the FIND option and select an entry.

Field Prompt: Transaction
 Field Usage: Updateable
 Field Name: NAME_TRANSACTION

Field Help

 This is the name of the transaction that the user was in when the system comment key was pressed.

This data field:

- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

-
- (/) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
 - (/) Enter data beyond the length of the field:
 - (/) If autoskip is in effect, cursor will skip to the next field.
 - (/) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
 - (/) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - (/) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (/) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
 - (/) Enter query mode and verify that this field is accessible.
 - (/) Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.
 - (/) Attempt to delete an existing data field value. An error should result.
 - (/) Use the space bar to attempt to delete the field data.
 - (/) Use clear field (F8) to attempt to delete the field data.

- (✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- (✓) Test LOV:
 - (✓) Select the first entry in the LOV.
 - (✓) Select the last entry in the LOV.
 - (✓) Use the FIND option and select an entry.

Field Prompt: Description
 Field Usage: Queryable Only ✓
 Field Name: DESCR_TRANSACTION

Field Help

 A brief description of the transaction. This field will be used as the transaction name in the menus, and on any reports. The description may be entered in mixed case.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.

Field Test Plan

- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.

Field Prompt: Stat Date
 Field Usage: Queryable Only
 Field Name: DATE_STATUS

no field available

Field Help

 This is the date the status of this comment last changed.
 It is automatically updated when the status changes.

Field Test Plan

-
- () Enter query mode and verify that this field is accessible.
 - () Enter query mode and enter a valid value in this field.
 () Only records matching the entered value should be displayed.
 - () Accept the default.
 - () Modify each default value with a valid alternative.
 - () Modify each default value with an invalid alternative.

Field Prompt: Reference #
 Field Usage: Queryable Only
 Field Name: IDENT_COMMENT

no field available

Field Help

-
- Contents are automatically generated by the system.

 This data field:

- Must contain a valid value. A blank is not allowed.
- Must contain a numeric value (0-9), a comma (,), plus (+), minus (-), or period (.).

Field Test Plan

-
- () Enter query mode and verify that this field is accessible.
 - () Enter query mode and enter a valid value in this field.
 () Only records matching the entered value should be displayed.
 - () Attempt to enter a decimal value. An error should occur.
 - () Enter a duplicate value. An error should occur.
 - () From LOV, select a value already present in the database. An error should occur.

Field Prompt: Date
 Field Usage: Queryable Only
 Field Name: DATE_COMMENT

no field

Field Help

Enter the date value using the format dd-MON-yy

where:

- dd - is the numeric day of the month
- MON - is the three character definition of month
 (JAN, FEB, MAR, APR, MAY, JUN, JUL
 AUG, SEP, OCT, NOV, DEC)
- yy - is the numeric value of the year within the century.

The system will display an error if you enter the data in any other format.

This data field:

- Must contain a valid value. A blank is not allowed.
- Date data must be entered in the format dd-MMM-yy
 where:
 - dd = the day of the month; days prior to 10 should have a leading zero, e.g., May 1 should be shown as 01
 - MMM = the first three characters of the month of the year in either upper or lower case, e.g., September = SEP
 - yy = the last two digits of the calendar year, e.g., 1992 is entered as 92.
- Contents are automatically generated by the system.
- Date data must be entered in the format mm/dd/yy
 mm = the month of the year in numeric form 01-12

Field Test Plan

- () Enter query mode and verify that this field is accessible.
- () Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.

Field Prompt: Classification
 Field Usage: Updateable
 Field Name: CLASSIFICATION

no field

Field Help

This data field:

- Is optional and may be left empty.
- Provides a default value. To accept the default value, press the ENTER key or TAB key.
- May contain any character combination.
- All input is converted to upper case.

Field Test Plan

- () Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- () Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - () If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- () Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- () Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - () Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- () Enter query mode and verify that this field is accessible.
- () Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.
- () Accept the default.
- () Modify each default value with a valid alternative.

- () Modify each default value with an invalid alternative.
- () Skip the field using the TAB key.
- () Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
- () Delete an existing value. The field should be cleared and stored:
 - () Delete the value using the space bar.
 - () Delete the value using the clear field key (F8).

Field Prompt: Flag Print
 Field Usage: Updateable
 Field Name: FLAG_PRINT

no field

Field Help

This data field:

- Is optional and may be left empty.
- May contain any character combination.
- All input is converted to upper case.

Field Test Plan

- () Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- () Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - () If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- () Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- () Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - () Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- () Enter query mode and verify that this field is accessible.
- () Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.
- () Skip the field using the TAB key.
- () Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.

() Delete an existing value. The field should be cleared and stored:

- () Delete the value using the space bar.
- () Delete the value using the clear field key (F8).

Field Prompt: LEMIS ID
Field Usage: Non Updateable
Field Name: ID_VAL

no field

Field Help

The system ID value for the current row when the user pressed the system comment key. May be useful for debugging.

This data field:

- Is optional and may be left empty.
- Must contain a numeric value (0-9), a comma (,), plus (+), minus (-), or period (.).

Field Test Plan

-
- () Attempt to store alphabetic characters in the field. An error should result.
 - () Begin the field entry with a space. An error should result.
 - () Attempt to enter a longer data value than allowed.
 - () Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - () Attempt to change data in the field. An error should result.
 - () Use the spacebar to attempt to change data.
 - () Use the clear field key (F8) to attempt to change data.
 - () Attempt to delete data in the field. An error should result.
 - () Use the spacebar to attempt to delete data.
 - () Use the clear field key (F8) to attempt to delete data.
 - () Enter query mode and verify that this field is accessible.
 - () Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.
 - () Attempt to enter a decimal value. An error should occur.

Field Prompt: ROWID
Field Usage: Non Updateable
Field Name: ROWID_VAL

no
field

Field Help

The Oracle ROWID value for the current row when the user pressed the system comment key. May be useful for debugging purposes.

This data field:

- Is optional and may be left empty.
- May contain any character combination.
- All input is converted to upper case.

Field Test Plan

- () Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- () Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - () If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- () Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- () Attempt to change data in the field. An error should result.
 - () Use the spacebar to attempt to change data.
 - () Use the clear field key (F8) to attempt to change data.
- () Attempt to delete data in the field. An error should result.
 - () Use the spacebar to attempt to delete data.
 - () Use the clear field key (F8) to attempt to delete data.
- () Enter query mode and verify that this field is accessible.
- () Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.

Field Prompt: From Block
 Field Usage: Non Updateable
 Field Name: FROM_BLOCK

Field Help

 The block the cursor was in when the user pressed system comment.

This data field:

- Is optional and may be left empty.
- May contain any character combination.
- All input is converted to upper case.

Field Test Plan

(☒) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.

(☒) Enter data beyond the length of the field:

- () If autoskip is in effect, cursor will skip to the next field.
- () If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.

(☒) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.

(☒) Attempt to change data in the field. An error should result.

- () Use the spacebar to attempt to change data.
- () Use the clear field key (F8) to attempt to change data.

(☒) Attempt to delete data in the field. An error should result.

- () Use the spacebar to attempt to delete data.
- () Use the clear field key (F8) to attempt to delete data.

(☒) Enter query mode and verify that this field is accessible.

(☒) Enter query mode and enter a valid value in this field.

- (☒) Only records matching the entered value should be displayed.

Field Prompt: Last
Field Usage: Non Updateable
Field Name: LAST_ERR_TYPE

mg

Field Help

The last ERROR_TYPE before the user pressed the system comment key.

This data field:

- Is optional and may be left empty.
- May contain any character combination.
- All input is converted to upper case.

Field Test Plan

-
- () Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
 - () Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - () If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
 - () Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - () Attempt to change data in the field. An error should result.
 - () Use the spacebar to attempt to change data.
 - () Use the clear field key (F8) to attempt to change data.
 - () Attempt to delete data in the field. An error should result.
 - () Use the spacebar to attempt to delete data.
 - () Use the clear field key (F8) to attempt to delete data.
 - () Enter query mode and verify that this field is accessible.
 - () Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.

Field Prompt: Error
 Field Usage: Non Updateable
 Field Name: LAST_ERR_CODE

NO field

Field Help

The last ERROR_CODE before the user pressed the system comment key.

This data field:

- Is optional and may be left empty.
- Must contain a numeric value (0-9), a comma (,), plus (+), minus (-), or period (.).

Field Test Plan

- () Attempt to store alphabetic characters in the field. An error should result.
- () Begin the field entry with a space. An error should result.
- () Attempt to enter a longer data value than allowed.
- () Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- () Attempt to change data in the field. An error should result.
 - () Use the spacebar to attempt to change data.
 - () Use the clear field key (F8) to attempt to change data.
- () Attempt to delete data in the field. An error should result.
 - () Use the spacebar to attempt to delete data.
 - () Use the clear field key (F8) to attempt to delete data.
- () Enter query mode and verify that this field is accessible.
- () Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.
- () Attempt to enter a decimal value. An error should occur.

Field Prompt:
Field Usage: Non Updateable
Field Name: LAST_ERR_TEXT

no field

Field Help

The last ERROR_TEXT before the user pressed the system comment key.

This data field:

- Is optional and may be left empty.
- May contain any character combination.

Field Test Plan

-
- () Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
 - () Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - () If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
 - () Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - () Attempt to change data in the field. An error should result.
 - () Use the spacebar to attempt to change data.
 - () Use the clear field key (F8) to attempt to change data.
 - () Attempt to delete data in the field. An error should result.
 - () Use the spacebar to attempt to delete data.
 - () Use the clear field key (F8) to attempt to delete data.
 - () Enter query mode and verify that this field is accessible.
 - () Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.

Field Prompt: Last
Field Usage: Non Updateable
Field Name: LAST_MSG_TYPE

NO Field

Field Help

The last MESSAGE_TYPE before the user pressed the system comment key.

This data field:

- Is optional and may be left empty.
- May contain any character combination.
- All input is converted to upper case.

Field Test Plan

-
- () Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
 - () Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - () If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
 - () Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - () Attempt to change data in the field. An error should result.
 - () Use the spacebar to attempt to change data.
 - () Use the clear field key (F8) to attempt to change data.
 - () Attempt to delete data in the field. An error should result.
 - () Use the spacebar to attempt to delete data.
 - () Use the clear field key (F8) to attempt to delete data.
 - () Enter query mode and verify that this field is accessible.
 - () Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.

Field Prompt:
Field Usage: Non Updateable
Field Name: LAST_MSG_CODE

NO

Field Help

The last MESSAGE_CODE before the user pressed the system comment key.

This data field:

- Is optional and may be left empty.
- Must contain a numeric value (0-9), a comma (,), plus (+), minus (-), or period (.).

Field Test Plan

-
- () Attempt to store alphabetic characters in the field. An error should result.
 - () Begin the field entry with a space. An error should result.
 - () Attempt to enter a longer data value than allowed.
 - () Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - () Attempt to change data in the field. An error should result.
 - () Use the spacebar to attempt to change data.
 - () Use the clear field key (F8) to attempt to change data.
 - () Attempt to delete data in the field. An error should result.
 - () Use the spacebar to attempt to delete data.
 - () Use the clear field key (F8) to attempt to delete data.
 - () Enter query mode and verify that this field is accessible.
 - () Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.
 - () Attempt to enter a decimal value. An error should occur.

Field Prompt: Message
 Field Usage: Non Updateable
 Field Name: LAST_MSG_TEXT

NO field

Field Help

 The last MESSAGE_TEXT before the user pressed the system comment key.

This data field:

- Is optional and may be left empty.
- May contain any character combination.

Field Test Plan

-
- () Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
 - () Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - () If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
 - () Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - () Attempt to change data in the field. An error should result.
 - () Use the spacebar to attempt to change data.
 - () Use the clear field key (F8) to attempt to change data.
 - () Attempt to delete data in the field. An error should result.
 - () Use the spacebar to attempt to delete data.
 - () Use the clear field key (F8) to attempt to delete data.
 - () Enter query mode and verify that this field is accessible.
 - () Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.

Field Prompt: From Field
 Field Usage: Non Updateable
 Field Name: FROM_FIELD

Field Help

 The field the cursor was in when the user pressed system comment.

This data field:

- Is optional and may be left empty.
- May contain any character combination.
- All input is converted to upper case.

Field Test Plan

-
- () ^{NA} Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
 - () ^{NA} Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - () If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
 - () ^{NA} Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - () ^{NA} Attempt to change data in the field. An error should result.
 - () Use the spacebar to attempt to change data.
 - () Use the clear field key (F8) to attempt to change data.
 - () ^{NA} Attempt to delete data in the field. An error should result.
 - () Use the spacebar to attempt to delete data.
 - () Use the clear field key (F8) to attempt to delete data.
 - () ^{NA} Enter query mode and verify that this field is accessible.
 - () ^{NA} Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.
- NA/NA

Field Prompt: From Table
 Field Usage: Non Updateable
 Field Name: FROM_TABLE

Field Help

The base table of the block the cursor was in when the user pressed system comment.

This data field:

- Is optional and may be left empty.
- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

- 11/15 Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- 11/15 Enter data beyond the length of the field:
- () If autoskip is in effect, cursor will skip to the next field.
 - () If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- () Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- 11/15 Attempt to change data in the field. An error should result.
- () Use the spacebar to attempt to change data.
 - () Use the clear field key (F8) to attempt to change data.
- 11/15 Attempt to delete data in the field. An error should result.
- () Use the spacebar to attempt to delete data.
 - () Use the clear field key (F8) to attempt to delete data.
- () Enter query mode and verify that this field is accessible.
- () Enter query mode and enter a valid value in this field.
- () Only records matching the entered value should be displayed.

OK

882

Blank allowed
cannot be left blank

Block Title: Related Text
Block Table: GENERIC TEXT
Block Usage: Full Function

no field

Block Help

This table is used to store textual data that is related to the current data table.

Block Test Plan

- () Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms:
 - () Block help describes:
 - () Purpose of the block
 - () Options available in the block
 - () The relationship of the block to the transaction or to the principal parent block of the transaction
 - () Navigation to other blocks of the transaction
- () Where block synchronization is used, ensure that the block data is in sync with the parent block.
- () Query a single record:
 - () A single record is retrieved.
 - () Pressing the F12 key clears the record and permits entering a second query.
 - () The second query successfully retrieves the desired data.
- () Query multiple records using wild cards:
 - () Pressing F12 clears the block and makes the block ready for a query.
 - () A query composed of a trailing % wildcard retrieves the desired data.
 - () A query using a prefix % wildcard retrieves the correct data.
 - () A query using both prefix and suffix % wildcards retrieves the correct data.
 - () A query using the underscore (_) position wildcard retrieves the correct data.
- () Verify that the data queries in the block are ordered logically.
- () Store a series of records using the F6 [duplicate record] key to carry forward field values from the previously stored record.

- () Attempting to store the duplicated record without change should result in an error.
- () Change key fields and store the record. It should store successfully.
- () Update a record:
 - () The change count should increment.
 - () The date updated should become the current date.
 - () An error should occur if the operator owner identifier does not match the data owner identifier.
- () Delete a record from the block:
 - () If the record has dependent records, the delete should fail unless the help text indicates that a cascading delete is permitted.
 - () Use the clear rcd key (CTRL/F8).
 - () For a single record block, the block should clear.
 - () For a multi-record block, the record pointed to by the cursor should clear.
 - () Use the delete key.
 - () For a single record block, the block should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should appear in the block.
 - () For a multiple record block, the entry should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should move up to fill the cleared space.
- () Enter text for one text type (press <PageDown> from Text Type)
Make sure the text is saved to the database.
- () Edit the text for one text type.

Field Prompt: Type Text
 Field Usage: Non Updateable
 Field Name: TYPE_TEXT

Field Help

 This field identifies the type of text that is associated with the

This data field:

- Provides a default value. To accept the default value, press the ENTER key or TAB key.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

-
- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.

Field Test Plan

- (☒) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (☒) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (☒) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (☒) Attempt to change data in the field. An error should result.
 - () Use the spacebar to attempt to change data.
 - (☒) Use the clear field key (F8) to attempt to change data.
- (☒) Attempt to delete data in the field. An error should result.
 - () Use the spacebar to attempt to delete data.
 - (☒) Use the clear field key (F8) to attempt to delete data.
- () Enter query mode and verify that this field is accessible.
- () Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.
- () Accept the default.
- () Modify each default value with a valid alternative.
- () Modify each default value with an invalid alternative.
- () Attempt to delete an existing data field value. An error should result.
 - () Use the space bar to attempt to delete the field data.
 - () Use clear field (F8) to attempt to delete the field data.
- (☒) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- () Test LOV:
 - (☒) Select the first entry in the LOV.
 - (☒) Select the last entry in the LOV.
 - (☒) Use the FIND option and select an entry.
- () Enter a duplicate value. An error should occur.
- () From LOV, select a value already present in the database. An error should occur.

J
DP3
query Doesn't work

Dupl, cat

N/A
N/A

Field Prompt: Text
 Field Usage: Undefined
 Field Name: TEXT

Field Help

 This data field:

Field Test Plan

-
- () Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - () Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - () Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
 - () Skip the field using the TAB key.
 - () Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
 - () Delete an existing value. The field should be cleared and stored:
 - () Delete the value using the space bar.
 - () Delete the value using the clear field key (F8).

896
no description

OK

695

884

no help function

tab does not work

Application: GENERIC Application Maintenance

Transaction: GE3220U1 System Comments

```
=====
Comment ID : 879      Priority: LOW      Date: 22-FEB-93
=====
Status      : HANDLED      Date: 24-FEB-93      Class:
Type        : PROBLEM
Assigned To : KEN ATKINS
Entered By  : LORRAINE CAMPBELL
System ID   : 8159      Oracle ROWID: 00000127.0005.000D
Last Error  : -0
Last Msg    : FRM-40202 Field must be entered.
Table       : SYSTEM_COMMENT
Block       : SYSCOM
Field       : SYSCOM.COMMENT_STATUS
=====
```

DESCRIPTION

Hint Text error message is wrong. Should state field must be entered.

CORRECTIVE ACTION

2/24/93 - Made field mandatory.

OK
2/26/93

Application: GENERIC Application Maintenance

Transaction: GE3220U1 System Comments

=====

Comment ID : 880 Priority: LOW Date: 22-FEB-93

=====

Status : HANDLED Date: 24-FEB-93 Class:

Type : PROBLEM

Assigned To: KEN ATKINS

Entered By : LORRAINE CAMPBELL

System ID : 8159 Oracle ROWID: 00000127.0005.000D

Last Error : -0

Last Msg : FRM-40102 Record must be entered or deleted first.

Table : SYSTEM COMMENT

Block : SYSCOM

Field : SYSCOM.DESCRPTION

DESCRIPTION

There is no hint text for this field. Pressing F1 brings up hint text. Pressing F1 again brings up help.

CORRECTIVE ACTION

2/24/93 - Set auto-hint on.

OK
JC 2/26/93

Application: GENERIC Application Maintenance

Transaction: GE3220U1 System Comments

=====

Comment ID : 881 Priority: LOW Date: 22-FEB-93

=====

Status : HANDLED Date: 24-FEB-93 Class:

Type : PROBLEM

Assigned To: KEN ATKINS

Entered By : LORRAINE CAMPBELL

System ID : 8159 Oracle ROWID: 00000127.0005.000D

Last Error : -0

Last Msg : FRM-40407 Transaction complete -- posted records comm

Table : SYSTEM_COMMENT

Block : SYSCOM

Field : SYSCOM.COMMENT_STATUS

DESCRIPTION

Pressing F1 brings hint text, Pressing F1 again brings help.

CORRECTIVE ACTION

2/24/93 - Set auto-hint on.

K
sc 2/24/93

Application: GENERIC Application Maintenance

Transaction: GE3220U1 System Comments

=====

Comment ID : 882 Priority: LOW Date: 22-FEB-93

=====

Status : HANDLED Date: 24-FEB-93 Class:

Type : PROBLEM

Assigned To: KEN ATKINS

Entered By : LORRAINE CAMPBELL

System ID : 2999 Oracle ROWID: 00000008.0008.000D

Last Error : -0

Last Msg : FRM-40353 Query cancelled.

Table : SYSTEM COMMENT

Block : SYSCOM

Field : SYSCOM.FROM_TABLE

DESCRIPTION

Help states that field is optional and also states that field cannot be left empty.

Fields "From Field", "Block," and "Table" are all non-updatable fields but help does not state this for any of them, rather it states that they are optional and will accept any character, upper cased.

CORRECTIVE ACTION

2/24/93 - Help updated.

Application: GENERIC Application Maintenance

Transaction: GE3220U1 System Comments

```
=====
Comment ID : 883 Priority: LOW Date: 22-FEB-93
=====
Status : HANDLED Date: 24-FEB-93 Class:
Type : PROBLEM
Assigned To: KEN ATKINS
Entered By : LORRAINE CAMPBELL
System ID : 8159 Oracle ROWID: 000000F8.0009.000D
Last Error : -0
Last Msg : FRM-40100 At first record.
Table : GENERIC TEXT
Block : COM_TEXT
Field : COM_TEXT.TYPE_TEXT
=====
```

DESCRIPTION

If query is suppose to work for this field, it doesn't.

CORRECTIVE ACTION

2/24/93 - This block is NOT queryable.

*2/26/93 OK
Jc*

Application: GENERIC Application Maintenance

Transaction: GE3220U1 System Comments

Comment ID : 895 Priority: LOW Date: 24-FEB-93

Status : HANDLED Date: 24-FEB-93 Class: OPERATIONAL
Type : COMMENT

Assigned To: KEN ATKINS

Entered By : KEN ATKINS

System ID : Oracle ROWID:

Last Error : -

Last Msg : -

Table : GENERIC_TEXT

Block : GENTXT

Field : TEXT

DESCRIPTION

No help available from this field.

CORRECTIVE ACTION

2/24/93 - Help cannot be accessed from an edit text popup. The test book was incorrect.

OK
fe 2/26/93

Application: GENERIC Application Maintenance

Transaction: GE3220U1 System Comments

=====

Comment ID : 904 Priority: LOW Date: 26-FEB-93

=====

Status : HANDLED Date: 26-FEB-93 Class: OPERATIONAL

Type : COMMENT

Assigned To : KEN ATKINS

Entered By : KEN ATKINS

System ID : Oracle ROWID:

Last Error : -

Last Msg : -

Table :

Block :

Field : TYPE_TEXT

DESCRIPTION

No descriptive help for this field.

CORRECTIVE ACTION

2/26/93 - Added descriptive help.

OK *[Signature]*
3/3

2.15 GE3241R1 OUTSTANDING CHANGE REQUESTS

Initial Test Date: 8/5/94 Verification Date: 8/5/94

- (☒) Verify that the report is accessible from the menu.
- (☒) Verify that the report is run correctly when requested.
- (☒) Verify that the report format is correct.
- (☒) Verify that the correct data is listed in the report.

2.16 GE3310U1 MAINTAIN GENERIC CODES

LEMIS Module Test Checklist Report

Name: MAINTAIN GENERIC CODES
Short Name: GE3310U1
Purpose: To maintain the Generic Code Table
Initial Test Date: 2/19/93 *fe*
Verification Date: 2/19/93 *fe*

Module Description

This transaction is used to maintain the generic code table, VALID_VALUES.

Module Help

This transaction is used to maintain the generic code table (VALID_VALUES). This table is used to store codes that need to have a brief description displayed on a transaction, but do not warrant their own code table.

To add codes to the generic code table:

- 1) Navigate to an open record by using the arrow keys or pressing <Insert>
- 2) Enter the Domain. The domain consists of the application, table, and field separated by "." See the field help for a full description.
- 3) Enter a 1-6 character code value.
- 4) Enter a brief description of this code.
- 5) Press <F10> to save the code.

Repeat steps 1)-5) for each code to add.

Module Help

To use this code table in a transaction use the following where clause:

WHERE DOMAIN = 'appl.table.field'

where appl = the application ident (ie LE)
 table = the table name (ie SAMPLE AUTHORIZATION)
 field = the field name (ie TYPE_SAMP)

example: WHERE DOMAIN = 'LE.SAMPLE_AUTHORIZATION.TYPE_SAMP'

UNIT TEST CHECKLIST

() Module Function Key

- (✓) Show Version : Shift-F10 :
- (✓) Comment/Bug : Shift-F3 :
- (✓) Fast Access : F5 :
- (✓) Back : F4 or Esc :
- (✓) Main Menu : Shift-F4 :
- (✓) Key Template : Ctrl-F1 :
- (✓) Context Prnt : F11 :
- (✓) Query Print : Ctrl-F11 :
- (✓) Oracle Print : Shift-F11 :

(✓) Block Title: Generic Code Table Values Block Table: VALID_VALUES

(✓) Help : F1 :	(✓) Key Help : Shift-F1 :
(✓) Online Doc : F2 :	(✓) Dis Journal : Shift-F2 :
(✓) Related Data : F3 :	(✓) Edit Text : F7 :
(✓) Spec Func 1 : Shift-F5 :	(✓) Spec Func 2 : Ctrl-F5 :
(✓) Home/End : Home/End :	(✓) Prv/Nxt Block: PgDn/PgUp:
(✓) Scroll Up/Dn : Shft-Up/Dn:	(✓) Prv/Nxt Field: Tab/Sh-Tb:
(✓) Prv/Nxt Rec : Up/Down :	(✓) Commit Form : F10 :
(✓) Clear Field : F8 :	(✓) Clear Record : Shift-F8 :
(✓) Clear Form : Shift-F8 :	(✓) Exit System : Ctrl-F4 :
(✓) Enter Query : F12 :	(✓) Count Query : Ctrl-F12 :
(✓) Correct Query Fields? :	(✓) Uppercase and LOVs? :
(✓) Execute Qry : Shift-F12:	(✓) Correct Query Order? :
(✓) Insert Rec : Insert :	(✓) Add record at end :
(✓) Mandatory checks OK? :	(✓) Validation checks OK? :
(✓) List of Values OK? :	(✓) Duplicate Fld: Shift-F6 :
(✓) Duplicate Rec: F6 :	(✓) Duplicate checks OK? :
(✓) Optional fields marked? :	(✓) Correct fields uppercase:

() Block Title: Generic Code Table Values
 Block Table: VALID_VALUES

() Delete Record: Delete :

TEST PLAN

Testing of the transaction is performed at four levels:

1. Developer testing.
2. Secondary testing.
3. End user testing.
4. Production release & verification.

As each applicable section of this test plan is completed, the designated individual responsible for the section will sign the plan in the appropriate places. Each check off item will be completed. The status of the transaction will be updated in the data dictionary as the transaction progresses through the development and test steps.

Developer Testing

As each individual transaction is generated, the developer performs tests against the transaction to insure that it meets specifications and works properly. Upon successful completion of this testing, the transaction is made available for secondary testing.

Test status conditions are:

- () 1 - Transaction under development, untested.
 - () Transaction added to menu and security.
 - () Transaction moved to development directory (addmod).
- () 2 - Development complete, initial unit testing in progress.
 - () Help added
 - () Unit Test checklist completed.
 - () Specific test plan added (if needed).
 - () Any bugs found in Unit test fixed.
 - () Updated test booklet printed.
- () 3 - Initial unit testing complete.

Developer [Signature] Date Complete 2/18/93

Secondary Testing

Secondary testing is performed by a person external to the development team. Tests are performed using a standardized test checklist. Initial testing is performed to ensure that the transaction meets the functional requirements of the transaction and satisfies the test plan narrative that follows.

- (☒) 4 - Secondary testing in progress.
 () 5 - Secondary testing complete, problems found. Resolution in progress.
 () 6 - Unable to resolve problem at this time.
 () 7 - Problem resolution complete.
 (☒) 8 - Secondary testing complete, no problems found.
 Tester LC Date Complete 2/19/93

When problems are found:

1. The tester passes the checklist to the project manager for review and action.
2. The project manager assigns the checklist to the developer of the transaction for correction of the problems.
3. The developer corrects the described problems, records the corrective actions taken and the number of hours of corrective effort used, tests the transaction for correct operation, and returns the checklist to the project manager.
4. The project manager returns the checklist to the tester for retesting.

The above sequence of events repeats until the tester signs off the transaction as passing the test requirements.

End User Testing

The transaction is tested by end users for a predetermined time period. This testing is done in the development environment. If no problems are encountered, or all problems are resolved at the end of this time period, the module is ready for Project Manager review.

- () 9 - End User testing in progress.
 () 10 - Problem found, resolution in progress.
 () 11 - Problem resolution complete.
 Developer _____ Date Complete _____
 (☒) 12 - Resolution verified.
 Tester _____ Date Complete _____

Production Status

- (☒) 13 - Ready for Project Manager/System Architect review.
 () The Project Manager or System Architect reviews all test results and determines that the transaction has completed testing.
 Architect/Manager CA Date Complete 2/14/93
 () 14 - The transaction is ready to be placed into production.
 (☒) Software files moved to production directory.
 (☒) Transaction added to production menu.
 (☒) Transaction access given to production roles.
 (☒) Validation test performed in production.
 () Production documentation up to date.
 () 15 - The transaction has been put into production

By *A* Date Done 2/17/73 Module Test Plan

(✓) Display transaction help. Verify that the narrative adequately describes the purpose and functions of the transaction in end-user terms.

- (✓) Transaction help describes:
- (✓) Purpose of the transaction
 - (✓) Functions performed by the transaction
 - (✓) Where multiple block, what happens in each block.

Block Test Plan

- (✓) Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms:
 - () Block help describes:
 - (✓) Purpose of the block
 - (✓) Options available in the block
 - (✓) The relationship of the block to the transaction or to the principal parent block of the transaction
 - () Navigation to other blocks of the transaction
- (✓) Where block synchronization is used, ensure that the block data is in sync with the parent block.
- (✓) Query a single record:
 - (✓) A single record is retrieved.
 - (✓) Pressing the F12 key clears the record and permits entering a second query.
 - (✓) The second query successfully retrieves the desired data.
- (✓) Query multiple records using wild cards:
 - (✓) Pressing F12 clears the block and makes the block ready for a query.
 - (✓) A query composed of a trailing % wildcard retrieves the desired data.
 - (✓) A query using a prefix % wildcard retrieves the correct data.
 - (✓) A query using both prefix and suffix % wildcards retrieves the correct data.
 - (✓) A query using the underscore (_) position wildcard retrieves the correct data.
- (✓) Verify that the data queries in the block are ordered logically.
- (✓) Store a series of records using the F6 [duplicate record] key to carry forward field values from the previously stored record.
 - (✓) Attempting to store the duplicated record without change should result in an error.
 - (✓) Change key fields and store the record. It should store successfully.
- (✓) Update a record:
 - (✓) The change count should increment.
 - () The date updated should become the current date.

2.
() An error should occur if the operator owner identifier does not match the data owner identifier.

(/ Delete a record from the block:

() If the record has dependent records, the delete should fail unless the help text indicates that a cascading delete is permitted.

(/ Use the clear rcd key (CTRL/F8).

(/ For a single record block, the block should clear.

(/ For a multi-record block, the record pointed to by the cursor should clear.

(/ Use the delete key.

() For a single record block, the block should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should appear in the block.

(/ For a multiple record block, the entry should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should move up to fill the cleared space.

Field Prompt: Domain

Field Usage: Updateable

Field Name: TYPE_DOMAIN

Field Help

This column contains the index that is used to find the correct codes for any table and column that uses this code table. The format of the value in this column should be>:

APPL.TABLE.COLUMN

Where APPL = The two character application identifier.

TABLE = The name of the table that the column is in.

COLUMN = The name of the column that uses this code.

Example: LE.SAMPLE_AUTHORIZATION.TYPE_SAMPLE

Field Test Plan

(/ Verify that the field accepts character or numeric data within the limits imposed by validation criteria.

(/ Enter data beyond the length of the field:

() If autoskip is in effect, cursor will skip to the next field.

(/ If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.

- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Attempt to delete an existing data field value. An error should result.
 - (✓) Use the space bar to attempt to delete the field data.
 - (✓) Use clear field (F8) to attempt to delete the field data.
- (✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- (✓) Enter a duplicate value. An error should occur.
- () From LOV, select a value already present in the database. An error should occur.

N/A

Field Prompt: Type Code
 Field Usage: Updateable
 Field Name: TYPE_CODE

Field Help

This is the generic code value.

Field Test Plan

- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (✓) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Attempt to delete an existing data field value. An error should result.
 - (✓) Use the space bar to attempt to delete the field data.
 - (✓) Use clear field (F8) to attempt to delete the field data.
- (✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- () Enter a duplicate value. An error should occur.

- 27/8
- () From LOV, select a value already present in the database. An error should occur.

Field Prompt: Description
 Field Usage: Updateable
 Field Name: DESCRIPTION

Field Help

This is a brief description of the generic code.

Field Test Plan

- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (✓) Enter data beyond the length of the field:
- () If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
- (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
- (✓) Only records matching the entered value should be displayed.
- (✓) Attempt to delete an existing data field value. An error should result.
- (✓) Use the space bar to attempt to delete the field data.
 - (✓) Use clear field (F8) to attempt to delete the field data.
- (✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.

2.17 GE3721R1 SYSTEM ACCESS REPORTInitial Test Date: LC 8/8/94 Verification Date: 8/8/94

- (X) Verify that the report is accessible from the menu.
- (X) Verify that the report is run correctly when requested.
- (X) Verify that the report format is correct.
- (X) Verify that the correct data is listed in the report.

2.18 GE3742R1 SYSTEM USE STATISTICS REPORTInitial Test Date: LC 8/8/94 Verification Date: 8/8/94

- (X) Verify that the report is accessible from the menu.
- (X) Verify that the report is run correctly when requested.
- (X) Verify that the report format is correct.
- (X) Verify that the correct data is listed in the report.

2.19 GE3820R1 MESSAGES REPORTInitial Test Date: LC 8/8/94 Verification Date: 8/8/94

- (X) Verify that the report is accessible from the menu.
- (X) Verify that the report is run correctly when requested.
- (X) Verify that the report format is correct.
- (X) Verify that the correct data is listed in the report.

2.20 GE3910U1 MAINTAIN PRINTER LIST

LEMIS Module Test Checklist Report

Name: GE3910U1 - MAINTAIN PRINTER LI
Short Name: GE3910U1
Purpose: To maintain a list of valid printers and info
Initial Test Date: 1/26/93 RJA
Verification Date: 1/26/93

Module Description

This transaction is used to maintain a list of valid printers, and to store the SQL*Reportwriter DESTYPE values for for print orientation and size options:

80 character portrait
132 character portrait
132 character landscape
Max character landscape (The maximum usable for each printer)

Module Help

This transaction is used to maintain a list of valid printers, and to store the SQL*Reportwriter DESTYPE values for for print orientation and size options:

80 character portrait
132 character portrait
132 character landscape
Max character landscape (The maximum usable for each printer)

To add a new printer:

- 1) Move to an open record by pressing <Insert> or pressing the down

Module Help

- arrow key until the cursor is at a blank record.
- 2) Enter the printer ID. This MUST match the unix printer ID given to that printer (the id used with the -P option in lpr) If it does not match no output will go to the printer. It MUST also be the same case as the unix id (ie if the unix ID is mo904 then this value cannot be M0904)

If the new printer is a network printer, then the system manager must add it to the system's list of valid network printers before the system can print to it.

- 3) Enter a brief description of the location and type of printer.
- 4) Enter the SQL*Reportwriter DESTYPE values for this printer. For a list of valid values and their meanings contact the Oracle DBA.

There are for output formats defined for each printer:

Port 80 = 80 character portrait
 Port 132 = 132 character portrait
 Land 132 = 132 character landscape
 Land Max = Maximum usable characters in landscape for this printer.

There is no validation for these fields, so make sure they are entered correctly or the reports will not work.

If any of these fields are left blank, this indicates that this printer cannot print in that format. If a report is defined with that format, then this printer will not be valid for that report.

For more information about the DESTYPE field, look in the SQL*Reportwriter manuals or contact the oracle DBA.

UNIT TEST CHECKLIST

- (✓) Module Function Key
-
- (✓) Show Version : Shift-F10 :
 (✓) Comment/Bug : Shift-F3 :
 (✓) Fast Access : F5 :
 (✓) Back : F4 or Esc :
 (✓) Main Menu : Shift-F4 :
 (✓) Key Template : Ctrl-F1 :
 (✓) Context Prnt : F11 :
 (✓) Query Print : Ctrl-F11 :
 (✓) Oracle Print : Shift-F11 :

- () Block Title: Valid Printers
 Block Table: PRINTER_LIST

() Block Title: Valid Printers
Block Table: PRINTER_LIST

(☒) Help : F1 :	(☒) Key Help : Shift-F1 :
(☒) Online Doc : F2 :	(☒) Dis Journal : Shift-F2 :
(☒) Related Data : F3 :	(☒) Edit Text : F7 :
(☒) Spec Func 1 : Shift-F5 :	(☒) Spec Func 2 : Ctrl-F5 :

(☒) Home/End : Home/End :	(☒) Prv/Nxt Block: PgDn/PgUp:
(☒) Scroll Up/Dn : Shift-Up/Dn:	(☒) Prv/Nxt Field: Tab/Sh-Tb:
(☒) Prv/Nxt Rec : Up/Down :	(☒) Commit Form : F10 :

(☒) Clear Field : F8 :	(☒) Clear Record : Shift-F8 :
(☒) Clear Form : Shift-F8 :	(☒) Exit System : Ctrl-F4 :

(☒) Enter Query : F12 :	(☒) Count Query : Ctrl-F12 :
(☒) Correct Query Fields? :	(☒) Uppercase and LOVs? :
(☒) Execute Qry : Shift-F12:	(☒) Correct Query Order? :

(☒) Insert Rec : Insert :	(☒) Add record at end :
(☒) Mandatory checks OK? :	(☒) Validation checks OK? :
(☒) List of Values OK? :	(☒) Duplicate Fld: Shift-F6 :
(☒) Duplicate Rec: F6 :	(☒) Duplicate checks OK? :
(☒) Optional fields marked? :	(☒) Correct fields uppercase:

(☒) Delete Record: Delete :	

TEST PLAN

Testing of the transaction is performed at four levels:

1. Developer testing.
2. Secondary testing.
3. End user testing.
4. Production release & verification.

As each applicable section of this test plan is completed, the designated individual responsible for the section will sign the plan in the appropriate places. Each check off item will be completed. The status of the transaction will be updated in the data dictionary as the transaction progresses through the development and test steps.

Developer Testing

As each individual transaction is generated, the developer performs tests against the transaction to insure that it meets specifications and works properly. Upon successful completion of this testing, the transaction is made available for secondary testing.

Test status conditions are:

- (✓) 1 - Transaction under development, untested.
 (✓) Transaction added to menu and security.
 (✓) Transaction moved to development directory (addmod).
 (✓) 2 - Development complete, initial unit testing in progress.
 (✓) Help added
 (✓) Unit Test checklist completed.
 (✓) Specific test plan added (if needed).
 (✓) Any bugs found in Unit test fixed.
 (✓) Updated test booklet printed.
 (✓) 3 - Initial unit testing complete.
 Developer Date Complete 1/20/93

Secondary Testing

Secondary testing is performed by a person external to the development team. Tests are performed using a standardized test checklist. Initial testing is performed to ensure that the transaction meets the functional requirements of the transaction and satisfies the test plan narrative that follows.

- (✓) 4 - Secondary testing in progress.
 (✓) 5 - Secondary testing complete, problems found. Resolution in progress.
 (✓) 6 - Unable to resolve problem at this time.
 (✓) 7 - Problem resolution complete.
 (✓) 8 - Secondary testing complete, no problems found.
 Tester Date Complete

When problems are found:

1. The tester passes the checklist to the project manager for review and action.
2. The project manager assigns the checklist to the developer of the transaction for correction of the problems.
3. The developer corrects the described problems, records the corrective actions taken and the number of hours of corrective effort used, tests the transaction for correct operation, and returns the checklist to the project manager.
4. The project manager returns the checklist to the tester for retesting.

The above sequence of events repeats until the tester signs off the transaction as passing the test requirements.

End User Testing

The transaction is tested by end users for a predetermined time period. This testing is done in the development environment. If no problems are encountered, or all problems are resolved at the end of this time period, the module is ready for Project Manager review.

- () 9 - End User testing in progress.
- () 10 - Problem found, resolution in progress.
- () 11 - Problem resolution complete.
Developer _____ Date Complete _____
- () 12 - Resolution verified.
Tester _____ Date Complete _____

Production Status

- (☒) 13 - Ready for Project Manager review.
() The Project Manager reviews all test results and determines that the transaction has completed testing.
- () 14 - The transaction is ready to be placed into production.
(☒) Software files moved to production directory.
(☒) Transaction added to production menu.
(☒) Transaction access given to production roles.
(☒) Validation test performed in production.
() Production documentation up to date.
- () 15 - The transaction is in production

System Architect AA Date Complete 1/26/93

Module Test Plan

- (☒) Display transaction help. Verify that the narrative adequately describes the purpose and functions of the transaction in end-user terms.
- (☒) Transaction help describes:
 - (☒) Purpose of the transaction
 - (☒) Functions performed by the transaction
 - (☒) Where multiple block, what happens in each block.

Block Title: Valid Printers
Block Table: PRINTER_LIST
Block Usage: Full Function

Block Help

This table is used to store a list of valid printers for the system and the valid SQL*Reportwriter DESTYPE values for four printer orientations and character sizes. See the module help for a complete description.

Block Test Plan

Block Test Plan

- (✓) Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms:
 - (✓) Block help describes:
 - (✓) Purpose of the block
 - (✓) Options available in the block
 - (✓) The relationship of the block to the transaction or to the principal parent block of the transaction
 - (✓) Navigation to other blocks of the transaction
- () Where block synchronization is used, ensure that the block data is in sync with the parent block.
- (✓) Query a single record:
 - (✓) A single record is retrieved.
 - (✓) Pressing the F12 key clears the record and permits entering a second query.
 - (✓) The second query successfully retrieves the desired data.
- () Query multiple records using wild cards:
 - (✓) Pressing F12 clears the block and makes the block ready for a query.
 - (✓) A query composed of a trailing % wildcard retrieves the desired data.
 - (✓) A query using a prefix % wildcard retrieves the correct data.
 - (✓) A query using both prefix and suffix % wildcards retrieves the correct data.
 - (✓) A query using the underscore (_) position wildcard retrieves the correct data.
- (✓) Verify that the data queries in the block are ordered logically.
- (✓) Store a series of records using the F6 [duplicate record] key to carry forward field values from the previously stored record.
 - (✓) Attempting to store the duplicated record without change should result in an error.
 - (✓) Change key fields and store the record. It should store successfully.
- () Update a record:
 - () The change count should increment.
 - () The date updated should become the current date.
 - () An error should occur if the operator owner identifier does

not match the data owner identifier.

- (✓) Delete a record from the block:
 - (✓) If the record has dependent records, the delete should fail unless the help text indicates that a cascading delete is permitted.
 - (✓) Use the clear rcd key (CTRL/F8).
 - () For a single record block, the block should clear.
 - () For a multi-record block, the record pointed to by the cursor should clear.
 - (✓) Use the delete key.
 - (✓) For a single record block, the block should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should appear in the block.
 - (✓) For a multiple record block, the entry should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should move up to fill the cleared space.

Field Prompt: Printer ID
 Field Usage: Updateable
 Field Name: IDENT_PRINTER

Field Help

 This field is the printer ID. This field MUST match the unix printer ID given to this printer (the ID used with the -P option in lpr). If it does not match no output will go to the printer. It MUST also be in the same case as the unix id (ie if the unix id is 'mo904' then this value cannot be 'M0904')

Field Test Plan

-
- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
 - (✓) Enter data beyond the length of the field:
 - (✓) If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
 - (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - (✓) Change the field where a range of valid values exists:

- () Value below valid range should display message COM00101 or COM00103 (if no LOV).
- () Value above valid range should display message COM00101 or COM00103 (if no LOV).
- () Minimum value within the range should be accepted.
- () Maximum value within the range should be accepted.
- () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Attempt to delete an existing data field value. An error should result.
 - (✓) Use the space bar to attempt to delete the field data.
 - (✓) Use clear field (F8) to attempt to delete the field data.
- (✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- (✓) Enter a duplicate value. An error should occur.
- (✓) From LOV, select a value already present in the database. An error should occur.

Field Prompt: Description
 Field Usage: Updateable
 Field Name: DESCRIPTION

Field Help

A brief description of the location. It should include both the location of the printer and the type of printer it is.

Example: HP1111 IN 703 BASEMENT

Field Test Plan

- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (✓) Enter data beyond the length of the field:
 - (✓) If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - () Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- () Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.
- (✓) Attempt to delete an existing data field value. An error should result.
 - (✓) Use the space bar to attempt to delete the field data.
 - (✓) Use clear field (F8) to attempt to delete the field data.
- (✓) Enter data into the field, backspace the data out of the field and

attempt to TAB to the next field. An error should occur.

Field Prompt: Port 80
 Field Usage: Updateable
 Field Name: DRIVER_P80

Field Help

The SQL*Reportwriter DESTYPE value for this printer to print in 80 column portrait format. For a list of valid values, contact the Oracle DBA.

There is no validation for this field, so make sure that it is entered correctly.

If this field is left blank, then a report defined with this format may not use this printer.

Field Test Plan

- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (✓) Enter data beyond the length of the field:
 - (✓) If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - () Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.
- (✓) Accept the default.

- 2/A*
- () Modify each default value with a valid alternative.
 - N/A* () Modify each default value with an invalid alternative.
 - (✓) Skip the field using the TAB key.
 - (✓) Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
 - (✓) Delete an existing value. The field should be cleared and stored:
 - (✓) Delete the value using the space bar.
 - (✓) Delete the value using the clear field key (F8).

Field Prompt: Port 132
 Field Usage: Updateable
 Field Name: DRIVER_P132

Field Help

The SQL*Reportwriter DESTYPE value for this printer to print in 132 column portrait format. For a list of valid values, contact the Oracle DBA.

There is no validation for this field, so make sure that it is entered correctly.

If this field is left blank, then a report defined with this format may not use this printer.

Field Test Plan

- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (✓) Enter data beyond the length of the field:
 - (✓) If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - () Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Accept the default.

- ☒ Modify each default value with a valid alternative.
- ☒ Modify each default value with an invalid alternative.
- ☒ Skip the field using the TAB key.
- ☒ Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
- ☒ Delete an existing value. The field should be cleared and stored:
 - ☒ Delete the value using the space bar.
 - ☒ Delete the value using the clear field key (F8).

Field Prompt: Land 132
 Field Usage: Updateable
 Field Name: DRIVER_L132

Field Help

The SQL*Reportwriter DESTYPE value for this printer to print in 132 column landscape format. For a list of valid values, contact the Oracle DBA.

There is no validation for this field, so make sure that it is entered correctly.

If this field is left blank, then a report defined with this format may not use this printer.

Field Test Plan

- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (✓) Enter data beyond the length of the field:
 - (✓) If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Accept the default.

2/2/78
Modify each default value with a valid alternative.

2/2/78
Modify each default value with an invalid alternative.

(✓) Skip the field using the TAB key.

(✓) Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.

(✓) Delete an existing value. The field should be cleared and stored:

(✓) Delete the value using the space bar.

(✓) Delete the value using the clear field key (F8).

Field Prompt: Land Max
 Field Usage: Updateable
 Field Name: DRIVER_LMAX

Field Help

The SQL*Reportwriter DESTYPE value for this printer to print in values, contact the Oracle DBA.
 maximum columns feasible in landscape format. For a list of valid

There is no validation for this field, so make sure that it is entered correctly.

If this field is left blank, then a report defined with this format may not use this printer.

Field Test Plan

- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (✓) Enter data beyond the length of the field:
 - (✓) If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.
- (✓) Accept the default.

- (✓) ~~N/A~~ Modify each default value with a valid alternative.
- (✓) ~~N/A~~ Modify each default value with an invalid alternative.
- (✓) Skip the field using the TAB key.
- (✓) Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
- (✓) Delete an existing value. The field should be cleared and stored:
 - (✓) Delete the value using the space bar.
 - (✓) Delete the value using the clear field key (F8).

2.21 GE3920U1 MAINTAIN REPORT PARAMETERS

LEMIS Module Test Checklist Report

Name: GE3920U1-MAINTAIN REPORT PARAM
 Short Name: GE3920U1
 Purpose: To maintain a list of report parameters for e
 Initial Test Date: 4/26/82 AE 1/26/83 AE
 Verification Date: 4/12/83 fc

Module Description

 This transaction is used to enter and maintain a list of parameters and information needed to display and validate those parameters for a REPORT transaction. This information is used by GE3930Q1 to display and validate the list of parameters for each report.

Module Help

 This transaction is used to enter and maintain a list of parameters and information needed to display and validate those parameters for a REPORT transaction. This information is used by GE3930Q1 to display and validate the list of parameters for each report.

To add parameters to a report:

- 1) Query the report in the top block (the 'Report') block.
- 2) Press <PageDown> to move into the 'Report Parameters' block.
- 3) Move to an open record by pressing the down arrow key until the cursor is at an open record or by pressing the <Insert> key.
 A sequence number should automatically be generated.
- 4) Press <Enter> to accept the sequence number, or enter another sequence number and press <Enter>. The sequence number will determine the order the parameters will appear on the screen.
- 5) Enter the parameter name. This MUST match the SQL*Reportwriter

Module Help

- parameter name for the parameter value to be correctly passed into reportwriter.
- 6) Enter the prompt that is to be displayed to the user for this parameter and press <Enter>.
 - 7) Enter the datatype for this parameter and press <Enter>. The datatype will determine what type of validation will occur. (ie if the datatype is NUMBER then the parameter will not accept non-numeric input, if the datatype is DATE, the input must be in a valid date format) List of values is available.
 - 8) Enter the maximum length of the parameter and press <Enter>. If the user enters a value that is longer than this length, they will receive an error.
 - 9) If this parameter is mandatory, enter 'Y'. If it is optional enter 'N' and press <Enter>. If the parameter is set to mandatory, the user will be forced to enter a value in this parameter.
 - 10) Enter a default value for this parameter and press <Enter>. If a default value is entered, the parameter will start with this value displayed. The user may then override the default with a valid alternative.

When you leave the 'Default' field, the screen will move to the left and three more fields will be displayed for this parameter. These fields are optional and may be skipped by using another key to leave the 'Default' field (down arrow, F10 etc). <Page Down> will move directly to these fields, and <Page Up> will move back to the first screen's fields.

- 11) Enter the validation conditions (if applicable). GE3930Q1 supports limited database validation using these three fields. If nothing is entered, no database validation will occur.
 - a) Enter the Validation Table and press <Enter>. This field determines which table will be used for validation.
 - b) Enter the validation Column and press <Enter>. This field determines which column in the table will be used for validation.
 - c) Enter the validation Where Clause and press <Enter>. This field specifies a where clause that will be used in the validation SQL statement.

If values are entered into these fields the value entered for this parameter will be validated using the following SQL construct:

```
SELECT 'X' FROM <Validation Table>
WHERE <Validation Column> = <Parameter Value>
<Validation Where>
```

Example: If <Validation Table> = 'APPLICATION'
 <Validation Column> = 'IDENT_APPLICATION'
 <Validation Where> = 'LEVEL_DATA_VALID = 9'
 <Parameter Value> = 'GE'

Module Help

The SQL statement would be:

```
SELECT 'X' FROM APPLICATION
WHERE IDENT_APPLICATION = 'GE'
AND LEVEL_DATA_VALID = 9
```

UNIT TEST CHECKLIST

(✓) Module Function Key

(✓) Show Version : Shift-F10 :
 (✓) Comment/Bug : Shift-F3 :
 (✓) Fast Access : F5 :
 (✓) Back : F4 or Esc :
 (✓) Main Menu : Shift-F4 :
 (✓) Key Template : Ctrl-F1 :
 (✓) Context Prnt : F11 :
 (✓) Query Print : Ctrl-F11 :
 (✓) Oracle Print : Shift-F11 :

(✓) Block Title: Report
Block Table: TRANSACTION_LIST

(✓) Help : F1 :	(✓) Key Help : Shift-F1 :
(✓) Online Doc : F2 :	(✓) Dis Journal : Shift-F2 :
(✓) Related Data : F3 :	(✓) Edit Text : F7 :
(✓) Spec Func 1 : Shift-F5 :	(✓) Spec Func 2 : Ctrl-F5 :

(✓) Home/End : Home/End :	(✓) Prv/Nxt Block: PgDn/PgUp:
(✓) Scroll Up/Dn : Shft-Up/Dn:	(✓) Prv/Nxt Field: Tab/Sh-Tb:
(✓) Prv/Nxt Rec : Up/Down :	(✓) Commit Form : F10 :

(✓) Clear Field : F8 :	(✓) Clear Record : Shift-F8 :
(✓) Clear Form : Shift-F8 :	(✓) Exit System : Ctrl-F4 :

(✓) Enter Query : F12 :	(✓) Count Query : Ctrl-F12 :
(✓) Correct Query Fields? :	(✓) Uppercase and LOVs? :
(✓) Execute Qry : Shift-F12:	(✓) Correct Query Order? :

() Block Title: Report Parameters
Block Table: REPORT_PARAMETERS

(☒) Help	: F1	:	(☒) Key Help	: Shift-F1	:
(☒) Online Doc	: F2	:	(☒) Dis Journal	: Shift-F2	:
() Related Data	: F3	:	(☒) Edit Text	: F7	:
(☒) Spec Func 1	: Shift-F5	:	(☒) Spec Func 2	: Ctrl-F5	:

(☒) Home/End	: Home/End	:	(☒) Prv/Nxt Block	: PgDn/PgUp	:
(☒) Scroll Up/Dn	: Shift-Up/Dn	:	(☒) Prv/Nxt Field	: Tab/Sh-Tb	:
(☒) Prv/Nxt Rec	: Up/Down	:	(☒) Commit Form	: F10	:

(☒) Clear Field	: F8	:	(☒) Clear Record	: Shift-F8	:
(☒) Clear Form	: Shift-F8	:	(☒) Exit System	: Ctrl-F4	:

(☒) Enter Query	: F12	:	(☒) Count Query	: Ctrl-F12	:
(☒) Correct Query Fields?	:	:	(☒) Uppercase and LOVs?	:	:
(☒) Execute Qry	: Shift-F12	:	(☒) Correct Query Order?	:	:

(☒) Insert Rec	: Insert	:	(☒) Add record at end	:	:
(☒) Mandatory checks OK?	:	:	(☒) Validation checks OK?	:	:
(☒) List of Values OK?	:	:	(☒) Duplicate Fld	: Shift-F6	:
(☒) Duplicate Rec	: F6	:	(☒) Duplicate checks OK?	:	:
(☒) Optional fields marked?	:	:	(☒) Correct fields uppercase	:	:

(☒) Delete Record	: Delete	:			

TEST PLAN

Testing of the transaction is performed at four levels:

1. Developer testing.
2. Secondary testing.
3. End user testing.
4. Production release & verification.

As each applicable section of this test plan is completed, the designated individual responsible for the section will sign the plan in the appropriate places. Each check off item will be completed. The status of the transaction will be updated in the data dictionary as the transaction progresses through the development and test steps.

Developer Testing

As each individual transaction is generated, the developer performs tests against the transaction to insure that it meets specifications and works properly. Upon successful completion of this testing, the transaction is made available for secondary testing.

Test status conditions are:

- (✓) 1 = Transaction under development, untested.
 (✓) Transaction added to menu and security.
 (✓) Transaction moved to development directory (addmod).
 (✓) 2 - Development complete, initial unit testing in progress.
 (✓) Help added
 (✓) Unit Test checklist completed.
 (✓) Specific test plan added (if needed).
 (✓) Any bugs found in Unit test fixed.
 (✓) Updated test booklet printed.
 (✓) 3 - Initial unit testing complete.
 Developer A Date Complete 1/21/93

Secondary Testing

Secondary testing is performed by a person external to the development team. Tests are performed using a standardized test checklist. Initial testing is performed to ensure that the transaction meets the functional requirements of the transaction and satisfies the test plan narrative that follows.

- (✓) 4 - Secondary testing in progress.
 (✓) 5 - Secondary testing complete, problems found. Resolution in progress.
 () 6 - Unable to resolve problem at this time.
 (✓) 7 - Problem resolution complete.
 () 8 - Secondary testing complete, no problems found.
 Tester A Date Complete 1/26/93

When problems are found:

1. The tester passes the checklist to the project manager for review and action.
2. The project manager assigns the checklist to the developer of the transaction for correction of the problems.
3. The developer corrects the described problems, records the corrective actions taken and the number of hours of corrective effort used, tests the transaction for correct operation, and returns the checklist to the project manager.
4. The project manager returns the checklist to the tester for retesting.

The above sequence of events repeats until the tester signs off the transaction as passing the test requirements.

End User Testing

The transaction is tested by end users for a predetermined time period. This testing is done in the development environment. If no problems are encountered, or all problems are resolved at the end of this time period, the module is ready for Project Manager review.

- () 9 - End User testing in progress.
- () 10 - Problem found, resolution in progress.
- () 11 - Problem resolution complete.
Developer _____ Date Complete _____
- () 12 - Resolution verified.
Tester _____ Date Complete _____

Production Status

- (✓) 13 - Ready for Project Manager review.
() The Project Manager reviews all test results and determines that the transaction has completed testing.
- (✓) 14 - The transaction is ready to be placed into production.
(✓) Software files moved to production directory.
(✓) Transaction added to production menu.
(✓) Transaction access given to production roles.
(✓) Validation test performed in production.
(✓) Production documentation up to date.
- () 15 - The transaction is in production

System Architect A Date Complete 2/12/93

Module Test Plan

- (✓) Display transaction help. Verify that the narrative adequately describes the purpose and functions of the transaction in end-user terms.
- (✓) Transaction help describes:
 - (✓) Purpose of the transaction
 - (✓) Functions performed by the transaction
 - (✓) Where multiple block, what happens in each block.

Block Title: Report
Block Table: TRANSACTION_LIST
Block Usage: Query Only

Block Help

This table contains a list of all of the transactions in the system. It is used in processing the transaction security for the system; and to display the titles of the menus and transactions for the menus.

Block Test Plan

- (✓) Display block help. Verify that the narrative adequately

Block Test Plan

describes the purpose and functions of the block in end-user terms:

(✓) Block help describes:

- (✓) Purpose of the block
- (✓) Options available in the block
- (✓) The relationship of the block to the transaction or to the principal parent block of the transaction
- (✓) Navigation to other blocks of the transaction

() Where block synchronization is used, ensure that the block data is in sync with the parent block.

(✓) Query a single record:

- (✓) A single record is retrieved.
- (✓) Pressing the F12 key clears the record and permits entering a second query.
- (✓) The second query successfully retrieves the desired data.

() Query multiple records using wild cards:

- (✓) Pressing F12 clears the block and makes the block ready for a query.
- (✓) A query composed of a trailing % wildcard retrieves the desired data.
- (✓) A query using a prefix % wildcard retrieves the correct data.
- (✓) A query using both prefix and suffix % wildcards retrieves the correct data.
- (✓) A query using the underscore (_) position wildcard retrieves the correct data.

(✓) Verify that the data queries in the block are ordered logically.

Field Prompt: Appl

Field Usage: Non Updateable

Field Name: IDENT_APPLICATION

Field Help

This field contains the two character application identifier.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

-
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 (✓) Only records matching the entered value should be displayed.
- nix) Test LOV:
- () Select the first entry in the LOV.
 - () Select the last entry in the LOV.
 - () Use the FIND option and select an entry.

Field Prompt: Transaction
 Field Usage: Non Updateable
 Field Name: NAME_TRANSACTION

Field Help

This field is the name of the transaction (form, report, etc).
 It is a mandatory 8 characters in length and should be formatted
 in the following way:

AASFUOTn

where AA = Application Name (ie GE, LE)
 S = Subsystem number
 F = Function number
 U = Unit Process number
 T = transaction type: (U)pdate, (Q)uery, (R)eport etc.

It should ALWAYS be the same as the executable file name.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

-
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 (✓) Only records matching the entered value should be displayed.
- (✓) Enter a duplicate value. An error should occur.
- () From LOV, select a value already present in the database. An
 nix error should occur.

Field Prompt:
Field Usage: Non Updateable
Field Name: DESCR_TRANSACTION

Field Help

A brief description of the transaction. This field will be used as the transaction name in the menus, and on any reports. The description may be entered in mixed case.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.

Field Test Plan

-
- (☒) Enter query mode and verify that this field is accessible.
 - () Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.

N/A

Block Title: Report Parameters
 Block Table: REPORT PARAMETERS
 Block Usage: Full Function

Block Help

This table is used to define report parameters for use in the generic parameter collection screen.

Block Test Plan

- () Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms:
 - () Block help describes:
 - () Purpose of the block
 - () Options available in the block
 - () The relationship of the block to the transaction or to the principal parent block of the transaction
 - () Navigation to other blocks of the transaction
 - () Where block synchronization is used, ensure that the block data is in sync with the parent block.
- () Query a single record:
 - () A single record is retrieved.
 - () Pressing the F12 key clears the record and permits entering a second query.
 - () The second query successfully retrieves the desired data.
- () Query multiple records using wild cards:
 - () Pressing F12 clears the block and makes the block ready for a query.
 - () A query composed of a trailing % wildcard retrieves the desired data.
 - () A query using a prefix % wildcard retrieves the correct data.
 - () A query using both prefix and suffix % wildcards retrieves the correct data.
 - () A query using the underscore (_) position wildcard retrieves the correct data.
- () Verify that the data queries in the block are ordered logically.
- () Store a series of records using the F6 [duplicate record] key to carry forward field values from the previously stored record.

840
 help screen for seg-
 appears FI
 fixed A

- (✓) Attempting to store the duplicated record without change should result in an error.
- (✓) Change key fields and store the record. It should store successfully.
- (✓) Update a record:
 - () The change count should increment.
 - () The date updated should become the current date.
 - N/A () An error should occur if the operator owner identifier does not match the data owner identifier.
- (✓) Delete a record from the block:
 - () If the record has dependent records, the delete should fail unless the help text indicates that a cascading delete is permitted.
 - () Use the clear rcd key (CTRL/F8).
 - () For a single record block, the block should clear.
 - () For a multi-record block, the record pointed to by the cursor should clear.
 - (X) Use the delete key.
 - () For a single record block, the block should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should appear in the block.
 - () For a multiple record block, the entry should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should move up to fill the cleared space.
- (✓) Verify that the navigation between page 1 and page 2 works by pressing Next Page from every Page 1 field and Prev Page from every Page 2 field.
- (✓) Verify that new sequence numbers are generated.
 - (✓) When <Insert> is pressed to make a new parameter
 - (✓) When down arrow is used to go to an open record.
 - N/A (✓) The first time a parameter is added to a report.

Field Prompt: Seq
 Field Usage: Updateable
 Field Name: SEQ_PARAM

Field Help

The sequence the parameter will be displayed on the parameter collection screen.

Field Test Plan

- () Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (✓) Enter data beyond the length of the field:

Field Test Plan

- (✓) If autoskip is in effect, cursor will skip to the next field.
- (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- () Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
- (✓) Change the field, committing after each change.
- () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
- (✓) Only records matching the entered value should be displayed.
- (✓) Attempt to enter a decimal value. An error should occur.
- (✓) Attempt to delete an existing data field value. An error should result.
 - (✓) Use the space bar to attempt to delete the field data.
 - (✓) Use clear field (F8) to attempt to delete the field data.
- (✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- (✓) Enter a duplicate value. An error should occur.
- () From LOV, select a value already present in the database. An error should occur.

SAME BUG
AS ON
PAGE 10

Fixed A

#844-866
ERROR:
ACCEPTED
CHAR &
DECIMAL
DATA.
A

Field Prompt: Parameter
 Field Usage: Updateable
 Field Name: PARAM_NAME

Field Help

The name of the SQL*Reportwriter parameter. Must match the name used in Reportwriter for the parameter to be passed correctly to Reportwriter.

Field Test Plan

- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (✓) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Attempt to delete an existing data field value. An error should result.
 - (✓) Use the space bar to attempt to delete the field data.
 - (✓) Use clear field (F8) to attempt to delete the field data.
- (✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.

(✓) Enter a duplicate value. An error should occur.

(✓) From LOV, select a value already present in the database. An error should occur.

N/A

Field Prompt: Prompt
 Field Usage: Updateable
 Field Name: PROMPT

Field Help

The prompt that will be displayed to the user for this parameter.

Field Test Plan

- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (✓) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - () If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Attempt to delete an existing data field value. An error should result.
 - (✓) Use the space bar to attempt to delete the field data.
 - (✓) Use clear field (F8) to attempt to delete the field data.
- (✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.

Field Prompt: Datatype
 Field Usage: Updateable
 Field Name: DATATYPE

Field Help

The datatype for this parameter. This will determine what type of validation will occur. (ie if the datatype is NUMBER then the parameter will not accept non-numeric input, and if it is DATE the input must be a valid date format)

List of Values is available.

Field Test Plan

- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (✓) Enter data beyond the length of the field:
 - (✓) If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Accept the default.
- (✓) Modify each default value with a valid alternative.
- (✓) Modify each default value with an invalid alternative.

- (✓) Attempt to delete an existing data field value. An error should result.
 - (✓) Use the space bar to attempt to delete the field data.
 - (✓) Use clear field (F8) to attempt to delete the field data.
- (✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- (✓) Test LOV:
 - (✓) Select the first entry in the LOV.
 - (✓) Select the last entry in the LOV.
 - (✓) Use the FIND option and select an entry.

Field Prompt: Len
 Field Usage: Updateable
 Field Name: PARAM_LENGTH

Field Help

The maximum length allowed for this parameter.

If the user enters a value that is longer than this length, they will see an error.

Field Test Plan

- (✓) Attempt to store alphabetic characters in the field. An error should result.
- (N) Begin the field entry with a space. An error should result.
- (✓) Attempt to enter a longer data value than allowed.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (X) Attempt to enter a decimal value. An error should occur.
- (✓) Attempt to delete an existing data field value. An error should result.
 - (✓) Use the space bar to attempt to delete the field data.
 - (✓) Use clear field (F8) to attempt to delete the field data.
- (✓) Enter data into the field, backspace the data out of the field and

attempt to TAB to the next field. An error should occur.

Field Prompt: Mand?
 Field Usage: Updateable
 Field Name: FLAG_MANDATORY

Field Help

If = 'Y' then this parameter is mandatory for this report.

Field Test Plan

- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (✓) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Accept the default.
- (✓) Modify each default value with a valid alternative.
- (✓) Modify each default value with an invalid alternative.
- (✓) Attempt to delete an existing data field value. An error should result.
 - (✓) Use the space bar to attempt to delete the field data.
 - (✓) Use clear field (F8) to attempt to delete the field data.

- (✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- (✓) Test LOV:
 - (✓) Select the first entry in the LOV.
 - (✓) Select the last entry in the LOV.
 - () Use the FIND option and select an entry.

Field Prompt: Default
 Field Usage: Updateable
 Field Name: DEFAULT_VALUE

Field Help

The default value to be displayed for this parameter.

Field Test Plan

- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (✓) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Skip the field using the TAB key.
- (✓) Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
- (✓) Delete an existing value. The field should be cleared and stored:
 - (✓) Delete the value using the space bar.
 - (✓) Delete the value using the clear field key (F8).

Field Prompt: Validation Table

Field Usage: Updateable

Field Name: VAL_TABLE

Field Help

The name of the table that will be used for validation of the parameter.

Field Test Plan

- (/) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (/) Enter data beyond the length of the field:
 - (/) If autoskip is in effect, cursor will skip to the next field.
 - (/) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (/) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (/) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (/) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (/) Enter query mode and verify that this field is accessible.
- (/) Enter query mode and enter a valid value in this field.
 - (/) Only records matching the entered value should be displayed.
- (/) Skip the field using the TAB key.
- (/) Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
- (/) Delete an existing value. The field should be cleared and stored:
 - (/) Delete the value using the space bar.

(~) Delete the value using the clear field key (F8).

Field Prompt: Validation Column
 Field Usage: Updateable
 Field Name: VAL_COLUMN

Field Help

The name of the column in VAL_TABLE that will be used to validate this parameter.

See the transaction help for a full description of this functionality.

Field Test Plan

- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (✓) Enter data beyond the length of the field:
 - (✓) If autoskip is in effect, cursor will skip to the next field.
 - () If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Skip the field using the TAB key.
- (✓) Enter data into the field, backspace the data out of the field and TAB to the next field. The database should contain a null value.
- (✓) Delete an existing value. The field should be cleared and stored:

- (✓) Delete the value using the space bar.
- (✓) Delete the value using the clear field key (F8).

Application: GENERIC Application Maintenance

Transaction: GE3920U1 Maintain report Parameters

=====

Comment ID :	840	Priority: LOW	Date: 26-JAN-93
--------------	-----	---------------	-----------------

=====

Status :	FIXED	Date: 11-FEB-93	Class:
Type :	PROBLEM		
Assigned To:	KEN ATKINS		
Entered By :	LORRAINE CAMPBELL		
System ID :		Oracle ROWID: 00000017.0000.000B	
Last Error :	-0		
Last Msg :	FRM-40353 Query cancelled.		
Table :	REPORT_PARAMETERS		
Block :	REPPAR		
Field :	REPPAR.SEQ_PARAM		

DESCRIPTION

weird screen appears the first time F1 is pressed. Press esc and F1 again and regular help appears. press F1 again and the screen help screen for seq_peram appears again.

CORRECTIVE ACTION

2/11/92 - Fixed - set automatic hint on.

Of Je

Application: GENERIC Application Maintenance

Transaction: GE3920U1 Maintain report Parameters

=====

Comment ID : 866 Priority: LOW Date: 11-FEB-93

=====

Status : FIXED Date: 11-FEB-93 Class: OPERATIONAL

Type : COMMENT

Assigned To: KEN ATKINS

Entered By : KEN ATKINS

System ID : Oracle ROWID:

Last Error : -

Last Msg : -

Table :

Block :

Field :

DESCRIPTION

Accepted character and decimal data.

CORRECTIVE ACTION

2/11/93 - Changed type to INT.

2.22 GE3930Q1 GENERATE AND PRINT REPORT

LEMIS Module Test Checklist Report

Name: GE3930Q1-GENERATE AND PRINT RE
Short Name: GE3930Q1
Purpose: To generate and print a report

Initial Test Date: Y2K LC

Verification Date: 2/12/93 LC

Module Description

This module is used to generate and print reports. It includes the capability to accept report parameters (defined using GE3920U1). It allows the user to select BATCH or ONLINE, send the report to a selected PRINTER, to a FILE, or view the report on the screen.

Module Help

This transaction is used to collect parameters for reports, to generate the reports and to specify the report destination.

- 1) First all of the parameters for the report must be entered. Some of the parameters may have default values, if so they may be overridden. Some of the parameters may be mandatory. If so then the cursor will not leave the field until a value is entered. The parameters may also have a format (DATE, NUMBER, CHARACTER) defined. If so then the parameter must be entered in the correct format.
- 2) When all of the parameters have been entered, press <F10> or <Enter> from the last parameter, and ALL parameters will be validated for consistency with the database. If any of the parameters are not valid, the cursor will move to the parameter and an error message will be displayed.
- 3) If all of the parameters are valid, the "Report Control Screen"

Module Help

will appear. This screen is used to determine when the report will be run and where the output will go.

4) Enter one of:

- O = Online. Causes the report to be run immediately.
- B = Batch. Requests the report at a later time.

The selection of this flag will determine which remaining fields need to be entered.

5) If "Batch" (B) was selected above the "Time" field will appear for input. Enter a time in the following format:

HHMM

where HH = the hour in 24 hour format. (ie 13 for 1 pm)
and MM = the minute (ie 1520 = 3:20)

and press <Enter>.

This will determine when the report will run. If possible enter a value after 1700 (5:00pm) or before 0600 (6:00 am) to reduce the daytime load on the system.

6) The Report destination field will appear next. Enter one of:

- P = Sends the output directly to the specified printer.
- F = Sends the output to a named file.
- E = Views the output directly on the screen.

The 'E' option will only appear if "Online" was selected above.

The selection of this flag will determine which additional fields will be required.

7) If "Printer" was selected as the destination, the following three fields will appear:

Number of copies - NOT IMPLEMENTED YET. LEAVE AS 1.
Destination - Enter the name of the printer that this report is to be printed on.
This must be one of the following:

- 1) A printer attached to the system.
- 2) A network printer that has been defined for this system.
- 3) A local printer attached to LPT1

To select from a valid list of printers, use LOV (F9).

Any selection made that begins with LPT1 will be printed to

Module Help

This field is used to specify delivery information if the a printer attached to the LPT1 port of your PC.
output is to be send to the requestor via mail.

Delivery Information - NOT IMPLEMENTED YET. LEAVE BLANK.

- 8) If "File" was selected as the destination,
Enter the filename for the output file from this report.
If nothing is entered, a filename consisting of the user id,
report name, and time will be automatically generated.

Any character/number combination may be used. The filename
will automatically be given the extension '.srep' so that
the file will appear in the list for the "Manage Report
Output Files" screen. To access this screen after the
report has run, press Shift-F5.

- 9) When all entry has been completed, press <Enter> until the
cursor is visible after the "->" at the bottom of the screen.

- 10) Press <F10> to submit or generate (and print) the report.

If the report is online, a "Working . . ." message will
appear. When you are asked to acknowledge the message
the report is finished.

If the report is BATCH, the screen will clear and you will
see a message specifying the job number.

UNIT TEST CHECKLIST

() Module Function Key

- () Show Version : Shift-F10 :
() Comment/Bug : Shift-F3 :
() Fast Access : F5 :
() Back : F4 or Esc :
() Main Menu : Shift-F4 :
() Key Template : Ctrl-F1 :
() Context Prnt : F11 :
() Query Print : Ctrl-F11 :
() Oracle Print : Shift-F11 :

() Block Title: Report Parameters
Block Table: REPORT_PARAMETERS

() Block Title: Report Parameters
Block Table: REPORT_PARAMETERS

☒ Help	: F1	:	☒ Key Help	: Shift-F1 :
☒ Online Doc	: F2	:	☒ Dis Journal	: Shift-F2 :
☒ Related Data	: F3	:	☒ Edit Text	: F7 :
☒ Spec Func 1	: Shift-F5	:	☒ Spec Func 2	: Ctrl-F5 :
<hr/>				
☒ Home/End	: Home/End	:	☒ Prv/Nxt Block	: PgDn/PgUp:
☒ Scroll Up/Dn	: Shft-Up/Dn	:	☒ Prv/Nxt Field	: Tab/Sh-Tb:
☒ Prv/Nxt Rec	: Up/Down	:	☒ Commit Form	: F10 :
<hr/>				
☒ Clear Field	: F8	:	☒ Clear Record	: Shift-F8 :
☒ Clear Form	: Shift-F8	:	☒ Exit System	: Ctrl-F4 :
<hr/>				
☒ Enter Query	: F12	:	☒ Count Query	: Ctrl-F12 :
☒ Correct Query Fields?	:	:	☒ Uppercase and LOVs?	:
☒ Execute Qry	: Shift-F12	:	☒ Correct Query Order?	:

TEST PLAN

Testing of the transaction is performed at four levels:

1. Developer testing.
2. Secondary testing.
3. End user testing.
4. Production release & verification.

As each applicable section of this test plan is completed, the designated individual responsible for the section will sign the plan in the appropriate places. Each check off item will be completed. The status of the transaction will be updated in the data dictionary as the transaction progresses through the development and test steps.

Developer Testing

As each individual transaction is generated, the developer performs tests against the transaction to insure that it meets specifications and works properly. Upon successful completion of this testing, the transaction is made available for secondary testing.

Test status conditions are:

- (☒) 1 - Transaction under development, untested.
 - (☒) Transaction added to menu and security.
 - (☒) Transaction moved to development directory (addmod).
- (☒) 2 - Development complete, initial unit testing in progress.
 - (☒) Help added
 - (☒) Unit Test checklist completed.
 - (☒) Specific test plan added (if needed).

- (☒) Any bugs found in Unit test fixed.
 (☒) Updated test booklet printed.
 (☒) 3 - Initial unit testing complete.
 Developer AB Date Complete 1/19/93

Secondary Testing

Secondary testing is performed by a person external to the development team. Tests are performed using a standardized test checklist. Initial testing is performed to ensure that the transaction meets the functional requirements of the transaction and satisfies the test plan narrative that follows.

- (☒) 4 - Secondary testing in progress.
 (☒) 5 - Secondary testing complete, problems found. Resolution in progress.
 (☒) 6 - Unable to resolve problem at this time.
 (☒) 7 - Problem resolution complete.
 () 8 - Secondary testing complete, no problems found.
 Tester JE Date Complete 2/12/93

When problems are found:

1. The tester passes the checklist to the project manager for review and action.
2. The project manager assigns the checklist to the developer of the transaction for correction of the problems.
3. The developer corrects the described problems, records the corrective actions taken and the number of hours of corrective effort used, tests the transaction for correct operation, and returns the checklist to the project manager.
4. The project manager returns the checklist to the tester for retesting.

The above sequence of events repeats until the tester signs off the transaction as passing the test requirements.

End User Testing

The transaction is tested by end users for a predetermined time period. This testing is done in the development environment. If no problems are encountered, or all problems are resolved at the end of this time period, the module is ready for Project Manager review.

- () 9 - End User testing in progress.
 () 10 - Problem found, resolution in progress.
 () 11 - Problem resolution complete.
 Developer _____ Date Complete _____
 () 12 - Resolution verified.
 Tester _____ Date Complete _____

Production Status

- () 13 - Ready for Project Manager review.
 - () The Project Manager reviews all test results and determines that the transaction has completed testing.
- () 14 - The transaction is ready to be placed into production.
 - (✓) Software files moved to production directory.
 - (✓) Transaction added to production menu.
 - (✓) Transaction access given to production roles.
 - (✓) Validation test performed in production.
 - () Production documentation up to date.
- () 15 - The transaction is in production

System Architect ADate Complete 2/12/93

Module Test Plan

- (✓) Display transaction help. Verify that the narrative adequately describes the purpose and functions of the transaction in end-user terms.

- (✓) Transaction help describes:

- () Purpose of the transaction
- () Functions performed by the transaction
- () Where multiple block, what happens in each block.

Block Title: Report Parameters
 Block Table: REPORT PARAMETERS
 Block Usage: Query Only

Block Help

 This table is used to define report parameters for use in the generic parameter collection screen.

Block Test Plan

- (✓) Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms:

- (✓) Block help describes:

- () Purpose of the block
- () Options available in the block
- () The relationship of the block to the transaction or to the principal parent block of the transaction

834 OK
 typographic
 (4)
 fixed A

Block Test Plan

-
- () Navigation to other blocks of the transaction
 - () Where ~~block synchronization is used~~, ensure that the block data is in sync with the parent block.
 - () Query a single record
 - () A single record ~~is retrieved~~.
 - () Pressing the F12 key clears the record and permits entering a second query.
 - () The second query successfully retrieves the desired data.
 - () Query multiple records using wild cards:
 - () Pressing F12 clears the block and makes the block ready for a query.
 - () A query composed of a trailing % wildcard retrieves the desired data.
 - () A query using a prefix % wildcard retrieves the correct data.
 - () A query using both prefix and suffix % wildcards retrieves the correct data.
 - () A query using the underscore (_) position wildcard retrieves the correct data.
 - () Verify that the data queries in the block are ordered logically.
 - **NOTE** May need to add dummy parameter to a report in order to perform all of these tests.
 - (✓) Enter a value longer than the allowed width. Should get error.
 - (✓) Enter a invalid date into a field that should be a date. Should get an error.
 - (✓) Enter a character value into a numeric parameter. Should get an error.
 - (✓) Attempt to skip a mandatory field without entering anything. Should get an error.
 - (✓) Attempt to generate the report <F10> with nothing in a mandatory field. Should get an error.
 - (✓) Attempt to generate a report with an invalid value in one of the parameters. (see the report parameter screen for current report to see how validation is done)
 - (✓) Verify that Output file management screen appears when Shift-F5 is pressed.
 - (✓) Verify that Help about this specific report is displayed when Ctrl-F5 is pressed.

Block Title: Report Control Screen
 Block Table: None
 Block Usage: Control

835 *grep not from
 appears
 Lost!!
 Nov 867
 fixed A
 OK

Block Test Plan

-
- (✓) Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms
 - () Block help describes:
 - (✓) Purpose of the block
 - (✓) Options available in the block
 - (✓) The relationship of the block to the transaction.
 - (✓) Navigation to other blocks of the transaction.
 - (✓) Verify that Output file management screen appears when Shft-F5 is pressed.
 - (✓) Press <esc> or <F4> from any field within the block. The cursor should move to the first field, and the rest of the fields should be cleared.
 - () Attempt the following combinations of entry and verify that the report is printed appropriately.

Generation Type	Destination Type	Destination
(✓) Online	Printer	Any network printer
(✓) Online	Printer	Printer connected to LPT1
(✓) Online	File	Any filename
(✓) Online	Edit	
() Batch (now)	Printer	Any network printer
(✓) Batch (now)	File	default filename
() Batch (in 10 min)	Printer	Any network printer

Handwritten notes:
 - 836
 738
 fixed

Field Prompt: Generate Report Online or Batch (O, B)
 Field Usage: Control

Field Test Plan

-
- (✓) Enter an invalid value. An error should result.
 - (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - (✓) Accept the default.
 - (✓) Modify the default value with a valid alternative.
 - (✓) Modify the default value with an invalid alternative.
 - (✓) Use the space bar and F8 to attempt to delete the data. An error should result when <Tab> or <Enter> is pressed to leave the field.

Field Prompt: Time

Field Test Plan

-
- **NOTE**** This field will only appear if 'B' is entered in the previous field.
- (✓) Enter a character value. An error should result.
 - (✓) Validate that only valid times in HHMM format are allowed.
 - (✓) Display column help. Verify that the narrative adequately

describes the column's purpose and functions in end-user terms.

Field Prompt: Send Report to Printer, File, or Edit (P,F,E)
or Send Report to Printer or File (P,F)

Field Test Plan

-
- **NOTE**** If 'Batch' is selected above, only P and F are valid
If 'Online' is selected, P, F, and E are valid.
- (X) Enter an invalid value. An error should result.
 - (X) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - (X) Accept the default.
 - (X) Modify the default value with a valid alternative.
 - (X) Modify the default value with an invalid alternative.
 - (X) Use the space bar and F8 to attempt to delete the data. An error should result when <Tab> or <Enter> is pressed to leave the field

Field Prompt: Filename

Field Test Plan

-
- **NOTE**** This field will only appear if 'File' is selected
- (X) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - (X) Enter a combination of numeric and character data and press. *838* *Ised*
<Enter> to accept the data.
 - (X) Clear the field and press <Enter> to accept the data. A filename should be automatically generated.

Field Prompt: Number of Copies

Field Test Plan

-
- **NOTE**** This field will only appear if 'Printer' is accepted.
- (X) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - (X) Attempt to store alphabetic characters in the field. An error should result.

Field Prompt: Destination

Field Test Plan

-
- **NOTE**** This field will only appear if 'Printer' is accepted.
- (X) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - (X) Test LOV:
 - (X) Select the first entry in the LOV.
 - (X) Select the last entry in the LOV.
 - (X) Use the FIND option and select an entry.
 - (X) Enter an invalid value. An error should result.

- (/) Modify the default value with a valid alternative.
- (/) Modify the default value with an invalid alternative.
- () Use the space bar and F8 to attempt to delete the data. An error should result when <Tab> or <Enter> is pressed to leave the field

839

fixed A

Field Prompt: Delivery Information

Field Test Plan

 NOTE This field will only appear if 'Printer' is accepted.

- (/) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.

Field Prompt: Answer questions until cursor is here ->

Field Test Plan

- (/) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (/) Press <F10> to generate the report.

Field Prompt: Prompt

Field Usage: Non Updateable

Field Name: PROMPT

Field Help

NP

Field Test Plan

- () Enter query mode and verify that this field is accessible.
- () Enter query mode and enter a valid value in this field.
- () Only records matching the entered value should be displayed.

Application: GENERIC Application Maintenance

Transaction: GE3930Q1 Generate and Print Report

=====
Comment ID : 834 Priority: LOW Date: 26-JAN-93
=====

Status : FIXED Date: 11-FEB-93 Class:
Type : PROBLEM

Assigned To: KEN ATKINS

Entered By : LORRAINE CAMPBELL

System ID : Oracle ROWID: 00000017.0000.000B

Last Error : -0

Last Msg : FRM-40353 Query cancelled.

Table : REPORT_PARAMETERS

Block : REPPAR

Field : REPPAR.PARAM_VALUE

DESCRIPTION

item (7) the is spelled thje

CORRECTIVE ACTION

2/11/93 - Fixed

OK

Application: HMID Hazardous Material Inventory Database

Transaction: GE3930Q1 Generate and Print Report

=====

Comment ID : 832 Priority: LOW Date: 21-JAN-93

=====

Status : FIXED Date: 04-FEB-93 Class:

Type : PROBLEM

Assigned To: KEN ATKINS

Entered By : HMID ADMINISTRATOR

System ID : Oracle ROWID:

Last Error : -0

Last Msg : FRM-40400 Transaction complete -- 2 records posted an

Table :

Block : SWIP890

Field : SWIP890.RC_GENERATE

DESCRIPTION

I just noticed that this screen has the LEMIS name across the
top..... Should it be "GENERIC"?

CORRECTIVE ACTION

02/04/93 - Fixed

Application: GENERIC Application Maintenance

Transaction: GE3930Q1 Generate and Print Report

=====

Comment ID : 867 Priority: LOW Date: 11-FEB-93

=====

Status : FIXED Date: 11-FEB-93 Class: OPERATIONAL

Type : COMMENT

Assigned To : KEN ATKINS

Entered By : KEN ATKINS

System ID : Oracle ROWID:

Last Error : -

Last Msg : -

Table :

Block :

Field :

DESCRIPTION

When Shift-F5 pressed from parameter block, *.srep not found error occurred.

CORRECTIVE ACTION

2/11/93 - Fixed so that control returns immediately back to GE3930Q1 when there are no reports. Cannot get rid of *.srep not found message (unix will not let me) so do REDISPLAY if there were no reports. This causes the screen to FLASH if there are no reports. Hopefully this will be acceptable to the users.*FEOK*

Application: GENERIC Application Maintenance

saction: GE3930Q1 Generate and Print Report

=====

Comment ID : 838 Priority: LOW Date: 26-JAN-93

=====

Status : FIXED Date: 11-FEB-93 Class:

Type : PROBLEM

Assigned To: KEN ATKINS

Entered By : LORRAINE CAMPBELL

System ID : Oracle ROWID: 00000017.0013.000B

Last Error : -0

Last Msg : FRM-40401 No changes to commit.

Table : REPORT_PARAMETERS

Block : REPPAR

Field : REPPAR.PARAM_VALUE

DESCRIPTION

filename would not let user enter his/her own filename, instead a big
long naming scheme is automatically given.

CORRECTIVE ACTION

2/11/93 - Fixed.

Application: GENERIC Application Maintenance

Transaction: GE3930Q1 Generate and Print Report

=====

Comment ID : 839 Priority: LOW Date: 26-JAN-93

=====

Status : FIXED Date: 11-FEB-93 Class:

Type : PROBLEM

Assigned To: KEN ATKINS

Entered By : LORRAINE CAMPBELL

System ID : Oracle ROWID:

Last Error : -0

Last Msg : FRM-41800 List of values not available for this field

Table :

Block : SWIP890

Field : SWIP890.RC_PR_WHERE

DESCRIPTION

If this field cannot be left blank then we have a problem, if a blank
is ok...No problem

CORRECTIVE ACTION

2/11/93 - Made mandatory.*jc*
OK

Date: 11-FEB-93

Page: 1

OFFICE OF SAMPLE MANAGEMENT
SAMPLING AUTHORIZATION FORM
SAF #: 92-283

LIQUID EFFLUENT

PROJECT NUMBER : 92-283

PROJECT TITLE : 400 AREA SECONDARY COOLING WATER

HASM PROJ COORD: M. HENDRIX

CUSTOMER NAME : D. SPEER

PHONE #: 3-1110

MSIN: R1-48

ORGANIZATION : 81221

SAMPLING LOCATN: FFTF STREAM - FAST FLUX TEST FACILITY STREAM

SAMPLING DATE : 09-SEP-92

PROTOCOLS : RCRA

DATA TURN REQS : REGULAR

SAMPLE MATRIX : WATER

LABORATORY

TMA SEATTLE , WA

WESTON PORTLAND , OR

COMMENTS:

- 1) All Radchem aliquots will be sent to the WESTON lab for analysis at Ecotek and will require individual HEIS sample numbers.
- 2) All Coliform aliquots will be sent to the HEHF lab and will require individual HEIS Sample numbers. All BOD aliquots will be sent to the TMA Laboratory.
- 3) All analyses will be priority T.A.T. except Radchem which will be regular T.A.T.
- 4) Revision 1 of SAF is a change of number of samples taken.

Application: GENERIC Application Maintenance

Transaction: GE3930Q1 Generate and Print Report

=====

Comment ID :	790	Priority: LOW	Date: 12-JAN-93
--------------	-----	---------------	-----------------

=====

Status :	ASSIGNED	Date: 18-JAN-93	Class:
Type :	PROBLEM		
Assigned To:	KEN ATKINS		
Entered By :	LARRY TOWNER		
System ID :		Oracle ROWID:	
Last Error :	-0		
Last Msg :	FRM-40350 Query caused no records to be retrieved.		
Table :			
Block :	SWIP890		
Field :	SWIP890.RC_PR_DELIVER		

DESCRIPTION

Delivery information has no help. The prompt is less than helpful.

CORRECTIVE ACTION

2/11/93 - Help has been added.

Application: GENERIC Application Maintenance

Transaction: GE3930Q1 Generate and Print Report

=====

Comment ID : 790 Priority: LOW Date: 12-JAN-93

=====

Status : FIXED Date: 11-FEB-93 Class:

Type : PROBLEM

Assigned To: KEN ATKINS

Entered By : LARRY TOWNER

System ID : Oracle ROWID:

Last Error : -0

Last Msg : FRM-40350 Query caused no records to be retrieved.

Table :

Block : SWIP890

Field : SWIP890.RC_PR_DELIVER

DESCRIPTION

Delivery information has no help. The prompt is less than helpful.

CORRECTIVE ACTION

2/11/93 - Help has been added.

Application: GENERIC Application Maintenance

Transaction: GE3930Q1 Generate and Print Report

=====

Comment ID : 836 Priority: LOW Date: 26-JAN-93

=====

Status : FIXED Date: 11-FEB-93 Class:

Type : PROBLEM

Assigned To: KEN ATKINS

Entered By : LORRAINE CAMPBELL

System ID : Oracle ROWID:

Last Error : -0

Last Msg : -0

Table :

Block : SWIP890

Field : SWIP890.RC_GENERATE

DESCRIPTION

when choosing an lpt1 port (mine prints to ken's office) the system
bombed me out to a p prompt with "press enter to return to an SQL..."
WHY!

CORRECTIVE ACTION

2/11/93 - Fixed. Had to clear the screen and ask user to press return
when print is done. Using NO_SCREEN on the host command seemed to
interfere with pcprint. To get this to work you may have to use a
dedicated printer instead of the printer sharing box.

jc ol

2.23 GE3940U1 REPORT FILE MAINTENANCE

LEMIS Module Test Checklist Report

Name: GE3940U1-REPORT FILE MAINTENAN
Short Name: GE3940U1
Purpose: To maintain report output files.

Initial Test Date: 1/26/93 RJA

Verification Date: 1/26/93 RJA

Module Description

This transaction is used to maintain report output files. With it the user can Edit, Rename, Print, or Delete report files.

Module Help

This transaction is used to Edit, Rename, Print, or Delete previously created report output files.

When you enter this screen you will see a list of all of your report output files (all files ending in '.srep' which is automatically appended to your report output files), their creation date and time and their size.

To perform a function on a file:

- 1) Use the arrow keys to move the cursor to the desired file.
- 2) Select one of the following functions:

P = Print the selected file
E = Edit the selected file
R = Rename the selected file
D = Delete the selected file

Module Help

- 3) Press <F10> to perform the selected function.
- 4) You will be prompted for appropriate information for each action:
- P (Print) - You will be prompted for the PRINT SITE SELECTION
Enter the printer name or use LOV <F9> to see valid values.
 - R (Rename)- You will be prompted for the new file name. Enter the desired file name. Keep the '.srep' extension if you want to continue managing the file with this screen.
 - D (Delete)- You will be asked to confirm the deletion by entering 'N' and pressing <F10> again.
 - E (Edit) - No other parameters. See 'Using Report Browser' below for help with report browser.
- 5) You will see a message indicating the the selected action was perfomed.

USING REPORT BROWSER

If (E) 'Edit' was selected above, you will enter a report browser to view the selected report. This browser will show you the report and allow you to move around in the report. The commands are:

- d = Move down one screen.
- u = Move up one screen.
- l = Move the report to the left.
- r = Move the report to the right.
- t = Move to the top of the report.
- b = Move to the bottom of the report.
- / = Search for a text string. After typing '/', type the string to search for and press <Enter>. The view will move to the screen that contains that text.
- n = Repeat previous search.
- q = Quit the browser.

UNIT TEST CHECKLIST

(1) Module	Function Key
(✓) Show Version	: Shift-F10 :
(✓) Comment/Bug	: Shift-F3 :
(✓) Fast Access	: F5 :
(✓) Back	: F4 or Esc :
(✓) Main Menu	: Shift-F4 :
(✓) Key Template	: Ctrl-F1 :
(✓) Context Prnt	: F11 :
(✓) Query Print	: Ctrl-F11 :
(✓) Oracle Print	: Shift-F11 :

() Block Title: Report Log
Block Table: REPORTLOG

(☒) Help : F1 :	(☒) Key Help : Shift-F1 :
(☒) Online Doc : F2 :	(☒) Dis Journal : Shift-F2 :
(☒) Related Data : F3 :	(☒) Edit Text : F7 :
(☒) Spec Func 1 : Shift-F5 :	(☒) Spec Func 2 : Ctrl-F5 :
(☒) Home/End : Home/End :	(☒) Prv/Nxt Block: PgDn/PgUp:
(☒) Scroll Up/Dn : Shift-Up/Dn:	(☒) Prv/Nxt Field: Tab/Sh-Tb:
(☒) Prv/Nxt Rec : Up/Down :	(☒) Commit Form : F10 :
(☒) Clear Field : F8 :	(☒) Clear Record : Shift-F8 :
(☒) Clear Form : Shift-F8 :	(☒) Exit System : Ctrl-F4 :
(☒) Enter Query : F12 :	(☒) Count Query : Ctrl-F12 :
(☒) Correct Query Fields? :	(☒) Uppercase and LOVs? :
() Execute Qry : Shift-F12:	(☒) Correct Query Order? :
(☒) Insert Rec : Insert :	(☒) Add record at end :
(☒) Mandatory checks OK? :	(☒) Validation checks OK? :
(☒) List of Values OK? :	(☒) Duplicate Fld: Shift-F6 :
(☒) Duplicate Rec: F6 :	(☒) Duplicate checks OK? :
(☒) Optional fields marked? :	(☒) Correct fields uppercase:
(☒) Delete Record: Delete :	

TEST PLAN

Testing of the transaction is performed at four levels:

1. Developer testing.
2. Secondary testing.
3. End user testing.
4. Production release & verification.

As each applicable section of this test plan is completed, the designated individual responsible for the section will sign the plan in the appropriate places. Each check off item will be completed. The status of the transaction will be updated in the data dictionary as the transaction progresses through the development and test steps.

Developer Testing

As each individual transaction is generated, the developer performs tests against the transaction to insure that it meets specifications and works properly. Upon successful completion of this testing, the transaction is made available for secondary testing.

Test status conditions are:

- (✓) 1 - Transaction under development, untested.
 - (✓) Transaction added to menu and security.
 - (✓) Transaction moved to development directory (addmod).
- (✓) 2 - Development complete, initial unit testing in progress.
 - (✓) Help added
 - (✓) Unit Test checklist completed.
 - (✓) Specific test plan added (if needed).
 - (✓) Any bugs found in Unit test fixed.
 - () Updated test booklet printed.
- () 3 - Initial unit testing complete.
 Developer TA Date Complete 1/20/93

Secondary Testing

Secondary testing is performed by a person external to the development team. Tests are performed using a standardized test checklist. Initial testing is performed to ensure that the transaction meets the functional requirements of the transaction and satisfies the test plan narrative that follows.

- (✓) 4 - Secondary testing in progress.
- (✓) 5 - Secondary testing complete, problems found. Resolution in progress.
- () 6 - Unable to resolve problem at this time.
- (✓) 7 - Problem resolution complete.
- (✓) 8 - Secondary testing complete, no problems found.
 Tester TA Date Complete 1/26/93

When problems are found:

1. The tester passes the checklist to the project manager for review and action.
2. The project manager assigns the checklist to the developer of the transaction for correction of the problems.
3. The developer corrects the described problems, records the corrective actions taken and the number of hours of corrective effort used, tests the transaction for correct operation, and returns the checklist to the project manager.
4. The project manager returns the checklist to the tester for retesting.

The above sequence of events repeats until the tester signs off the transaction as passing the test requirements.

End User Testing

The transaction is tested by end users for a predetermined time period. This testing is done in the development environment. If no problems are encountered, or all problems are resolved at the end of this time period, the module is ready for Project Manager review.

- () 9 - End User testing in progress.
- () 10 - Problem found, resolution in progress.
- () 11 - Problem resolution complete.
Developer _____ Date Complete _____
- () 12 - Resolution verified.
Tester _____ Date Complete _____

Production Status

- (✓) 13 - Ready for Project Manager review.
() The Project Manager reviews all test results and determines that the transaction has completed testing.
- () 14 - The transaction is ready to be placed into production.
(✓) Software files moved to production directory.
(✓) Transaction added to production menu.
(✓) Transaction access given to production roles.
(✓) Validation test performed in production.
() Production documentation up to date.
- () 15 - The transaction is in production

System Architect JA Date Complete 1/26/93

Module Test Plan

- () Display transaction help. Verify that the narrative adequately describes the purpose and functions of the transaction in end-user terms.
- () Transaction help describes:
 - () Purpose of the transaction
 - () Functions performed by the transaction
 - () Where multiple block, what happens in each block.

() 11 - Problem resolution complete.

- Developer _____ Date Complete _____

() 12 - Resolution verified.

Tester _____ Date Complete _____

Production Status

() 13 - Ready for Project Manager review.

() The Project Manager reviews all test results and determines that the transaction has completed testing.

() 14 - The transaction is ready to be placed into production.

() Software files moved to production directory.

() Transaction added to production menu.

() Transaction access given to production roles.

() Validation test performed in production.

() Production documentation up to date.

() 15 - The transaction is in production

System Architect _____ Date Complete _____

Module Test Plan

(X) Display transaction help. Verify that the narrative adequately describes the purpose and functions of the transaction in end-user terms.

(X) Transaction help describes:

(X) Purpose of the transaction

(X) Functions performed by the transaction

(X) Where multiple block, what happens in each block.

Block Title: Report Log

Block Table: REPORTLOG

Block Usage: Undefined

Block Help

This block is a list of all report output files owned by the current user. See the transaction help for a full description of the functionality of this block.

Block Test Plan

(X) Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms:

Block Test Plan

- (✓) Block help describes:
 - (✓) Purpose of the block
 - (✓) Options available in the block
 - (✓) The relationship of the block to the transaction or to the principal parent block of the transaction
 - (✓) Navigation to other blocks of the transaction
- (✓) Where block synchronization is used, ensure that the block data is in sync with the parent block.
- (✓) Query a single record:
 - (✓) A single record is retrieved.
 - (✓) Pressing the F12 key clears the record and permits entering a second query.
 - (✓) The second query successfully retrieves the desired data.
- (✓) Query multiple records using wild cards:
 - (✓) Pressing F12 clears the block and makes the block ready for a query.
 - (✓) A query composed of a trailing % wildcard retrieves the desired data.
 - (✓) A query using a prefix % wildcard retrieves the correct data.
 - (✓) A query using both prefix and suffix % wildcards retrieves the correct data.
 - (✓) A query using the underscore (_) position wildcard retrieves the correct data.
- (✓) Verify that the data queries in the block are ordered logically.
- () Store a series of records using the F6 [duplicate record] key to carry forward field values from the previously stored record.
 - () Attempting to store the duplicated record without change should result in an error.
 - () Change key fields and store the record. It should store successfully.
- (✓) Press the down arrow key until the cursor is at the last record. Verify that the cursor does not move to an empty record.

Field Prompt: SELECT (P,E,R,D)

Field Test Plan

- (✓) Display column help. Verify that the narrative adequately

- (✓) describes the column's purpose and functions in end-user terms.
- (✓) Enter an invalid value and press <F10>. An error should occur.
- () Enter a valid value and press <F10>. Verify that the appropriate pop-up appears.
 - (✓) Enter 'P' for print. 'Printer Destination' should appear.
 - (✓) Enter 'R' for Rename. 'Enter New Filename' should appear.
 - (✓) Enter 'D' for Delete. 'Enter Y to Delete' should appear.
 - (✓) Enter 'E' for Edit. Should enter report browser.

Field Prompt: Printer Destination

Field Test Plan

- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Enter an invalid value. An error should result.
- (✓) Enter a valid value and verify the the file is printed.
- (✓) Test LOV:
 - (✓) Select the first entry in the LOV.
 - () Select the last entry in the LOV.
 - (✓) Use the FIND option and select an entry.

Field Prompt: Enter New Filename

Field Test Plan

- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- () Enter a new filename and verify that the file is renamed.

Field Prompt: Enter Y to Delete F10 to process

Field Test Plan

- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Enter a value other than Y and press <F10>. The file should not be deleted.
- (✓) Enter Y and press <F10>. Verify that the file is deleted by exiting and reentering the form.

Field Prompt: Report File
 Field Usage: Queryable Only
 Field Name: RLOG_FILENAME

Field Help

Field Test Plan

- () Enter query mode and verify that this field is accessible.

2/12

- () Enter query mode and enter a valid value in this field.
- () Only records matching the entered value should be displayed.

Field Prompt: Report Date
 Field Usage: Queryable Only
 Field Name: RLOG_REPORT_DT

Field Help

Field Test Plan

- () Enter query mode and verify that this field is accessible.
- () Enter query mode and enter a valid value in this field.
- () Only records matching the entered value should be displayed.

Field Prompt: File Size
 Field Usage: Queryable Only
 Field Name: RLOG_SIZE

Field Help

Field Test Plan

- () Enter query mode and verify that this field is accessible.
- () Enter query mode and enter a valid value in this field.
- () Only records matching the entered value should be displayed.
- () Attempt to enter a decimal value. An error should occur.

2.24 GE6011R1 ONLINE DOCUMENTATION SUMMARY REPORT

Initial Test Date:

8/8/94

Verification Date:

8/8/94

- ☒ Verify that the report is accessible from the menu.
- ☒ Verify that the report is run correctly when requested.
- ☒ Verify that the report format is correct.
- ☒ Verify that the correct data is listed in the report.

2.25 GE6011U1 MAINTAIN ONLINE DOCUMENTATION

LEMIS Module Test Checklist Report

Name: MAINTAIN ONLINE DOCUMENTATION
Short Name: GE6011U1
Purpose: Maintain on-line documentation

Initial Test Date: 2/23 JC

Verification Date: 2/26 JC

Module Description

This module is used to maintain the online documentation header records, the textual information for each record, and the list of subordinate documents.

Module Help

This transaction performs maintenance on the on-line documentation for the system. Each individual documentation text group is identified and stored as a separate entry. The entries are then linked together to form a complete documentation picture.

To add a documentation item:

- 1) If a record is displayed on the screen press <Insert> to open an empty record for input.
- 2) Enter the application ID for this documentation item. LOV available
- 3) Enter a 20 character identifier for this item. This should be descriptive enough to identify this item easily.
- 4) Enter a brief title for this documentation item. Make the title understandable to the user, because it will be displayed for the users selection.
- 5) Enter the document type. LOV is available. The document type determines what type of entry this will be. It will also limit

Module Help

-
- where this item may be attached in the documentation hierarchy.
- 6) Enter the ownership of this item. LOV is available.
 - 7) Press <F10> to commit this record.
 - 8) The related text pop-up will appear. Press <PageDown> to move to the text area.
 - 9) Enter the text for this documentation item and press <F10> to save it.
 - 10) Press <F10> to commit the text.

To delete a document item:

- 1) If the desired item has been attached to any other documentation items, it will need to be unattached (see procedure below).
- 2) Press <F12> to enter query mode.
- 3) Enter the identifier for the desired documentation item.
- 4) Press <Shift-F12> to execute the query.
- 5) Press <F7> to access the text popup.
- 6) Press <Delete> to delete the text record.
- 7) If there are any subordinate documents, use the procedure below to remove them.
- 8) Press <F10> to commit the text deletion.
- 9) Press <Delete> to delete the document item.
- 10) If "Cannot delete Document Block while Related Documents exists" appears, then this item remains connected to other document items. This item must be unattached from all other document items before being deleted.
- 11) Press <F10> to commit the deletion.

Adding subordinate document items:

- 1) Press <F12> to enter query mode.
- 2) Enter the identifier for the desired document item.
- 3) Press <Shift-F12> to query the item.
- 4) Press <PageDown> to move to the "Related Documents" block.
- 5) Use the down arrow key or <Insert> to open an empty record.
- 6) Enter the display sequence. This will determine the order these items will appear in the "Related Topics" popup of the on-line documentation screen.
- 7) Enter the identifier of the document to be related to the current one. LOV is available. Only documents with appropriate document types may be entered here.
- 8) Press <F10> to save the connection.

To remove a subordinate document item:

- 1) Navigate to the "Document Block", Press <F12> to enter query mode.
- 2) Enter the identifier for the desired document item.
- 3) Press <Shift-F12> to query the item.
- 4) Press <PageDown> to move to the "Related Documents" block.
- 5) Use the arrow keys to move the cursor to the subordinate document to remove.

11/11

Module Help

- 6) Press <Delete> to remove the connection.
7) Press <F10> to commit the change.

UNIT TEST CHECKLIST

(X) Module Function Key

(X) Show Version : Shift-F10 :
(X) Comment/Bug : Shift-F3 :
(X) Fast Access : F5 :
(X) Back : F4 or Esc :
(X) Main Menu : Shift-F4 :
(X) Key Template : Ctrl-F1 :
(X) Context Prnt : F11 :
(X) Query Print : Ctrl-F11 :
(X) Oracle Print : Shift-F11 :

(X) Block Title: Document Block
Block Table: DOCUMENT_HDR

(X) Help : F1 :	(X) Key Help : Shift-F1 :
(X) Online Doc : F2 :	(X) Dis Journal : Shift-F2 :
(X) Related Data : F3 :	(X) Edit Text : F7 :
(X) Spec Func 1 : Shift-F5 :	(X) Spec Func 2 : Ctrl-F5 :
(X) Home/End : Home/End :	(X) Prv/Nxt Block: PgDn/PgUp:
(X) Scroll Up/Dn : Shft-Up/Dn:	(X) Prv/Nxt Field: Tab/Sh-Tb:
(X) Prv/Nxt Rec : Up/Down :	(X) Commit Form : F10 :
(X) Clear Field : F8' :	(X) Clear Record : Shift-F8 :
(X) Clear Form : Shift-F8 :	(X) Exit System : Ctrl-F4 :
(X) Enter Query : F12 :	(X) Count Query : Ctrl-F12 :
(X) Correct Query Fields? :	(X) Uppercase and LOVs? :
(X) Execute Qry : Shift-F12:	(X) Correct Query Order? :
(X) Insert Rec : Insert :	(X) Add record at end :
(X) Mandatory checks OK? :	(X) Validation checks OK? :
(X) List of Values OK? :	(X) Duplicate Fld: Shift-F6 :
(X) Duplicate Rec: F6 :	(X) Duplicate checks OK? :
(X) Optional fields marked? :	(X) Correct fields uppercase:
(X) Delete Record: Delete :	

(✓) Block Title: Related Documents
Block Table: DOCUMENT_NEST

(✓) Help : F1	:	(✓) Key Help : Shift-F1	:
(✓) Online Doc : F2	:	(✓) Dis Journal : Shift-F2	:
(✓) Related Data : F3	:	(✓) Edit Text : F7	:
(✓) Spec Func 1 : Shift-F5	:	(✓) Spec Func 2 : Ctrl-F5	:

(✓) Home/End : Home/End	:	(✓) Prv/Nxt Block: PgDn/PgUp:	:
(✓) Scroll Up/Dn : Shift-Up/Dn:	:	(✓) Prv/Nxt Field: Tab/Sh-Tb:	:
(✓) Prv/Nxt Rec : Up/Down	:	(✓) Commit Form : F10	:

(✓) Clear Field : F8	:	(✓) Clear Record : Shift-F8	:
(✓) Clear Form : Shift-F8	:	(✓) Exit System : Ctrl-F4	:

(✓) Enter Query : F12	:	(✓) Count Query : Ctrl-F12	:
(✓) Correct Query Fields?	:	(✓) Uppercase and LOVs?	:
(✓) Execute Qry : Shift-F12:	:	(✓) Correct Query Order?	:

(✓) Insert Rec : Insert	:	(✓) Add record at end	:
(✓) Mandatory checks OK?	:	(✓) Validation checks OK?	:
(✓) List of Values OK?	:	(✓) Duplicate Fld: Shift-F6	:
(✓) Duplicate Rec: F6	:	(✓) Duplicate checks OK?	:
(✓) Optional fields marked?	:	(✓) Correct fields uppercase:	:

(✓) Delete Record: Delete	:		:

TEST PLAN

Testing of the transaction is performed at four levels:

1. Developer testing.
2. Secondary testing.
3. End user testing.
4. Production release & verification.

As each applicable section of this test plan is completed, the designated individual responsible for the section will sign the plan in the appropriate places. Each check off item will be completed. The status of the transaction will be updated in the data dictionary as the transaction progresses through the development and test steps.

Developer Testing

As each individual transaction is generated, the developer performs tests against the transaction to insure that it meets specifications and works properly. Upon successful completion of this testing, the transaction is made available for secondary testing.

Test status conditions are:

- (✓) 1 - Transaction under development, untested.
 (✓) Transaction added to menu and security.
 (✓) Transaction moved to development directory (addmod).
 (✓) 2 - Development complete, initial unit testing in progress.
 (✓) Help added
 (✓) Unit Test checklist completed.
 (✓) Specific test plan added (if needed).
 (✓) Any bugs found in Unit test fixed.
 (✓) Updated test booklet printed.
 (✓) 3 - Initial unit testing complete.
 Developer [Signature] Date Complete 2/22/93

Secondary Testing

Secondary testing is performed by a person external to the development team. Tests are performed using a standardized test checklist. Initial testing is performed to ensure that the transaction meets the functional requirements of the transaction and satisfies the test plan narrative that follows.

- (✓) 4 - Secondary testing in progress.
 (✓) 5 - Secondary testing complete, problems found. Resolution in progress.
 () 6 - Unable to resolve problem at this time.
 (✓) 7 - Problem resolution complete.
 (✓) 8 - Secondary testing complete, no problems found.
 Tester [Signature] Date Complete 2/26/93

When problems are found:

1. The tester passes the checklist to the project manager for review and action.
2. The project manager assigns the checklist to the developer of the transaction for correction of the problems.
3. The developer corrects the described problems, records the corrective actions taken and the number of hours of corrective effort used, tests the transaction for correct operation, and returns the checklist to the project manager.
4. The project manager returns the checklist to the tester for retesting.

The above sequence of events repeats until the tester signs off the transaction as passing the test requirements.

End User Testing

The transaction is tested by end users for a predetermined time period. This testing is done in the development environment. If no problems are encountered, or all problems are resolved at the end of this time period, the module is ready for Project Manager review.

- () 9 - End User testing in progress.
- () 10-- Problem found, resolution in progress.
- () 11 - Problem resolution complete.
Developer _____ Date Complete _____
- () 12 - Resolution verified.
Tester _____ Date Complete _____

Production Status

- (/) 13 - Ready for Project Manager/System Architect review.
() The Project Manager or System Architect reviews all test results and determines that the transaction has completed testing.
Architect/Manager A Date Complete 2/26/93
- () 14 - The transaction is ready to be placed into production.
(/) Software files moved to production directory.
(/) Transaction added to production menu.
(/) Transaction access given to production roles.
(/) Validation test performed in production.
() Production documentation up to date.
- (/) 15 - The transaction has been put into production
By A Date Done 2/26/92

Module Test Plan

- (/) Display transaction help. Verify that the narrative adequately describes the purpose and functions of the transaction in end-user terms.
(/) Transaction help describes:
(/) Purpose of the transaction
(/) Functions performed by the transaction
(/) Where multiple block, what happens in each block.
- () Verify that the text popup appears when a document header record is committed without adding text.

Block Title: Document Block
Block Table: DOCUMENT_HDR
Block Usage: Full Function

Block Help

This table stores the header information for the on-line documentation text for the system. The actual text for each document is stored in the LEMIS_TEXT table.

Block Test Plan

-
- (/) Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms:
 - (/) Block help describes:
 - () Purpose of the block
 - () Options available in the block
 - (/) The relationship of the block to the transaction or to the principal parent block of the transaction
 - (/) Navigation to other blocks of the transaction
 - (/) Where block synchronization is used, ensure that the block data is in sync with the parent block.
 - (/) Query a single record:
 - () A single record is retrieved.
 - (/) Pressing the F12 key clears the record and permits entering a second query.
 - (/) The second query successfully retrieves the desired data.
 - (/) Query multiple records using wild cards:
 - () Pressing F12 clears the block and makes the block ready for a query.
 - (/) A query composed of a trailing % wildcard retrieves the desired data.
 - (/) A query using a prefix % wildcard retrieves the correct data.
 - (/) A query using both prefix and suffix % wildcards retrieves the correct data.
 - (/) A query using the underscore (_) position wildcard retrieves the correct data.
 - (/) Verify that the data queries in the block are ordered logically.
 - (/) Store a series of records using the F6 [duplicate record] key to carry forward field values from the previously stored record.
 - (/) Attempting to store the duplicated record without change should result in an error.
 - () Change key fields and store the record. It should store successfully.
 - (/) Update a record:
 - () The change count should increment.
 - () The date updated should become the current date.

- () An error should occur if the operator owner identifier does not match the data owner identifier.
- (✓) Delete a record from the block:
 - () If the record has dependent records, the delete should fail unless the help text indicates that a cascading delete is permitted.
 - () Use the clear rcd key (CTRL/F8).
 - () For a single record block, the block should clear.
 - () For a multi-record block, the record pointed to by the cursor should clear.
 - (✓) Use the delete key.
 - () For a single record block, the block should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should appear in the block.
 - (✓) For a multiple record block, the entry should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should move up to fill the cleared space.

Field Prompt: Appl

Field Usage: Updateable

Field Name: IDENT_APPLICATION

Field Help

This field contains the two character Application ID.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (✓) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.

- (/) Change the field, committing after each change.
- (/) Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (/) Enter query mode and verify that this field is accessible.
- (/) Enter query mode and enter a valid value in this field.
 - (/) Only records matching the entered value should be displayed.
- () Attempt to delete an existing data field value. *Either the field result will not be updatable or an error will result*
 - () Use the space bar to attempt to delete the field data.
 - () Use clear field (F8) to attempt to delete the field data.
- (/) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- (/) Test LOV:
 - (/) Select the first entry in the LOV.
 - (/) Select the last entry in the LOV.
 - (/) Use the FIND option and select an entry.

OK
890
Delete allowed

Field Prompt: Ident
 Field Usage: Updateable
 Field Name: IDENT_DOCUMENT

Field Help

 Enter a unique identifier for this document item.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

-
- (/) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
 - (/) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (/) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
 - (/) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - (/) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (/) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
 - (/) Enter query mode and verify that this field is accessible.
 - (/) Enter query mode and enter a valid value in this field.
 - (/) Only records matching the entered value should be displayed.
 - () Attempt to delete an existing data field value. An error should result.
 - () Use the space bar to attempt to delete the field data.
 - () Use clear field (F8) to attempt to delete the field data.

Repeat 3
890

- ☒ Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- ☒ Enter a duplicate value. An error should occur.
- ☐ From LOV, select a value already present in the database. An error should occur.

N/A

Field Prompt: Title
 Field Usage: Updateable
 Field Name: TITLE_DOCUMENT

Field Help

 Enter the title for this document item. This title will be displayed in the document selection list and at the top of the screen when the document is being displayed.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

-
- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
 - (✓) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
 - (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
 - (✓) Enter query mode and verify that this field is accessible.
 - (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
 - (✓) Attempt to delete an existing data field value. An error should result.
 - () Use the space bar to attempt to delete the field data.

gao Report

Repeat
890

() Use clear field (F8) to attempt to delete the field data.

(✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.

Field Prompt: Doc Type
Field Usage: Updateable
Field Name: TYPE_DOCUMENT

Field Help

Enter the two character document type code. List of values is available.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

- (X) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (X) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (X) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (X) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (X) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (X) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (X) Enter query mode and verify that this field is accessible.
- (X) Enter query mode and enter a valid value in this field.
 - (X) Only records matching the entered value should be displayed.
- () Attempt to delete an existing data field value. An error should result.
 - () Use the space bar to attempt to delete the field data.
 - () Use clear field (F8) to attempt to delete the field data.

Field Clear
Repeat 890

(✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.

(✓) Test LOV:

(✓) Select the first entry in the LOV.

(✓) Select the last entry in the LOV.

(✓) Use the FIND option and select an entry.

Field Prompt:

Field Usage: Queryable Only

Field Name: DESCRIPTION

Field Help

A 1-40 character description of the document type code.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

(✓) Enter query mode and verify that this field is accessible.

(✓) Enter query mode and enter a valid value in this field.

(✓) Only records matching the entered value should be displayed.

Field Prompt: Chngs
Field Usage: Non Updateable
Field Name: COUNT_CHANGE

Field Help

This indicates how many times this record has been changed.

This data field:

- Provides a default value. To accept the default value, press the ENTER key or TAB key.
- Must contain a valid value. A blank is not allowed.
- Must contain a numeric value (0-9), a comma (,), plus (+), minus (-), or period (.).

Field Test Plan

- (✓) Attempt to store alphabetic characters in the field. An error should result.
- (✓) Begin the field entry with a space. An error should result.
- (✓) Attempt to enter a longer data value than allowed.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Attempt to change data in the field. An error should result.
 - (✓) Use the spacebar to attempt to change data.
 - (✓) Use the clear field key (F8) to attempt to change data.
- (✓) Attempt to delete data in the field. An error should result.
 - (✓) Use the spacebar to attempt to delete data.
 - (✓) Use the clear field key (F8) to attempt to delete data.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- () Attempt to enter a decimal value. An error should occur.
- (✓) Accept the default.
- (✓) Modify each default value with a valid alternative.
- (✓) Modify each default value with an invalid alternative.
- (✓) Attempt to delete an existing data field value. An error should.

Fixed
89 decimal allowed

result.

- ☒ Use the space bar to attempt to delete the field data.
- ☒ Use clear field (F8) to attempt to delete the field data.
- ☒ Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- ☒ Test LOV:
 - ☐ Select the first entry in the LOV.
 - ☐ Select the last entry in the LOV.
 - ☐ Use the FIND option and select an entry.

Field Prompt: Owner
 Field Usage: Updateable
 Field Name: IDENT_OWNER

Field Help

The field contains the ROLE identifier of the owner of the data item. The value entered is that of the current role of the logged on user that is placing the data item in the database. Once set, the role value may not be modified by the user. If you are adding a data item, the role value displayed is the role under which you are currently operating. You may change the role identifier to some other valid value (use LOV to see a list of valid values). CAUTION: If you change the default role identifier value to one for which you are not authorized, you will not be able to view or update the data after it is stored.

This data field:

- Provides a default value. To accept the default value, press the ENTER key or TAB key.
- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (✓) Enter data beyond the length of the field:
 - () If autoskip is in effect, cursor will skip to the next field.
 - (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (✓) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.

894
 888
 error received

- ☐ Random value within the range should be accepted.
- ☒ Enter query mode and verify that this field is accessible.
- ☒ Enter query mode and enter a valid value in this field.
- ☒ Only records matching the entered value should be displayed.
- ☒ Accept the default.
- ☒ Modify each default value with a valid alternative.
- ☐ Modify each default value with an invalid alternative.
- ☐ Attempt to delete an existing data field value. An error should result.
 - ☐ Use the space bar to attempt to delete the field data.
 - ☐ Use clear field (F8) to attempt to delete the field data.
- ☒ Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.

890
Delete allowed

- Rel
Field Prompt:

Field Usage: Updateable

Field Name: LEVEL_DATA_VALID

Field Help

This is the release level of this record. If set to 9 this item will be accessible to all applications, if set to 0 it will only be accessible to the owning application.

This data field:

- Provides a default value. To accept the default value, press the ENTER key or TAB key.
- Must contain a valid value. A blank is not allowed.
- Must contain a numeric value (0-9), a comma (,), plus (+), minus (-), or period (.).

Field Test Plan

-
- (/) Attempt to store alphabetic characters in the field. An error should result.
 - (/) Begin the field entry with a space. An error should result.
 - (/) Attempt to enter a longer data value than allowed.
 - (/) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
 - (/) Change the field in an existing record. Verify that the change occurred by displaying the record on the screen before and after the changes were made.
 - (/) Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
 - (/) Enter query mode and verify that this field is accessible.
 - (/) Enter query mode and enter a valid value in this field.
 - (/) Only records matching the entered value should be displayed.
 - (/) Attempt to enter a decimal value. An error should occur.

- (X) Accept the default.
- (X) Modify each default value with a valid alternative.
- () Modify each default value with an invalid alternative.
- () Attempt to delete an existing data field value. An error should result.
 - () Use the space bar to attempt to delete the field data.
 - () Use clear field (F8) to attempt to delete the field data.
- (X) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- (X) Attempt to store an invalid code. An error should result.

890
Deleted
all 890

Block Title: Related Documents
 Block Table: DOCUMENT_NEST
 Block Usage: Full Function

Block Help

This table stores the related documents to the on-line help documents stored in DOCUMENT_HDR.

Block Test Plan

- (✓) Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms:
 - (✓) Block help describes:
 - (✓) Purpose of the block
 - (✓) Options available in the block
 - (✓) The relationship of the block to the transaction or to the principal parent block of the transaction
 - (✓) Navigation to other blocks of the transaction
- () Where block synchronization is used, ensure that the block data is in sync with the parent block.
- (✓) Query a single record:
 - (✓) A single record is retrieved.
 - (✓) Pressing the F12 key clears the record and permits entering a second query.
 - (✓) The second query successfully retrieves the desired data.
- () Query multiple records using wild cards:
 - (✓) Pressing F12 clears the block and makes the block ready for a query.
 - (✓) A query composed of a trailing % wildcard retrieves the desired data.
 - (✓) A query using a prefix % wildcard retrieves the correct data.
 - (✓) A query using both prefix and suffix % wildcards retrieves the correct data.
 - (✓) A query using the underscore (_) position wildcard retrieves the correct data.
- (✓) Verify that the data queries in the block are ordered logically.
- (✓) Store a series of records using the F6 [duplicate record] key to carry forward field values from the previously stored record.

- (X) Attempting to store the duplicated record without change should result in an error.
- () Change key fields and store the record. It should store successfully.
- (X) Update a record:
 - (X) The change count should increment.
 - () The date updated should become the current date.
 - () An error should occur if the operator owner identifier does not match the data owner identifier.
- (X) Delete a record from the block:
 - () If the record has dependent records, the delete should fail unless the help text indicates that a cascading delete is permitted.
 - () Use the clear rcd key (CTRL/F8).
 - () For a single record block, the block should clear.
 - () For a multi-record block, the record pointed to by the cursor should clear.
 - (X) Use the delete key.
 - () For a single record block, the block should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should appear in the block.
 - (X) For a multiple record block, the entry should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should move up to fill the cleared space.

Field Prompt: Seq
 Field Usage: Updateable
 Field Name: DISPLAY_SEQ

Field Help

 This is the sequence that the these items will appear in the list of related documents.

Field Test Plan

- (X) Attempt to store alphabetic characters in the field. An error should result.
- (X) Begin the field entry with a space. An error should result.
- (X) Attempt to enter a longer data value than allowed.
- (X) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (X) Change the field in an existing record. Verify that the change

occurred by displaying the record on the screen before and after the changes were made.

- (X) Change the field, committing after each change.
- () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
 - () Value above valid range should display message COM00101 or COM00103 (if no LOV).
 - () Minimum value within the range should be accepted.
 - () Maximum value within the range should be accepted.
 - () Random value within the range should be accepted.
- (X) Enter query mode and verify that this field is accessible.
- (X) Enter query mode and enter a valid value in this field.
 - (X) Only records matching the entered value should be displayed.
- () Attempt to enter a decimal value. An error should occur. *B92*
- () Attempt to delete an existing data field value. An error should result.
 - () Use the space bar to attempt to delete the field data.
 - () Use clear field (F8) to attempt to delete the field data.
- (X) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.

OK decimal allowed

Delete allowed

Field Prompt: Ident Document
Field Usage: Queryable Only
Field Name: IDENT_DOCUMENT

Field Help

Enter a unique identifier for this document item.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

- (/) Enter query mode and verify that this field is accessible.
- (/) Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.
- (/) Enter a duplicate value. An error should occur.
- (/) From LOV, select a value already present in the database. An error should occur.

Field Prompt: Title Document
Field Usage: Queryable Only
Field Name: TITLE_DOCUMENT

Field Help

Enter the title for this document item. This title will be displayed in the document selection list and at the top of the screen when the document is being displayed.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

-
- (X) Enter query mode and verify that this field is accessible.
 - (X) Enter query mode and enter a valid value in this field.
 - (X) Only records matching the entered value should be displayed.
 - (X) Enter a duplicate value. An error should occur.
 - (X) From LOV, select a value already present in the database. An error should occur.

Application: GENERIC Application Maintenance

Transaction: GE6011U1 Maintain Online Documentation

Document ID : 889 Priority: LOW Date: 23-FEB-93

Status : HANDLED Date: 24-FEB-93 Class:
Type : PROBLEM
Assigned To: KEN ATKINS
Entered By : LORRAINE CAMPBELL
System ID : 8180 Oracle ROWID: 00000A4C.0017.000B
Last Error : -0
Last Msg : FRM-40400 Transaction complete -- 1 records posted an
Table : DOCUMENT_HDR
Block : DOCHDR
Field : DOCHDR.IDENT_APPLICATION

DESCRIPTION

No text popup appears when I do anything.

CORRECTIVE ACTION

2/24/93 - FLAG_TEXT was not last field. - Fixed.

K
LC
2/26/93

Application: GENERIC Application Maintenance

Transaction: GE6011U1 Maintain Online Documentation

=====

Comment ID :	890	Priority: LOW	Date: 23-FEB-93
--------------	-----	---------------	-----------------

=====

Status	: HANDLED	Date: 24-FEB-93	Class:
Type	: PROBLEM		
Assigned To:	KEN ATKINS		
Entered By :	LORRAINE CAMPBELL		
System ID :	8192	Oracle ROWID: 00000A4D.0002.000B	
Last Error :	-0		
Last Msg	: FRM-40350 Query caused no records to be retrieved.		
Table	: DOCUMENT_HDR		
Block	: DOCHDR		
Field	: DOCHDR.IDENT_APPLICATION		

DESCRIPTION

test booklet states that an error should result when attempting to delete a data field value. Field is updatable, therefore no error results when using space bar or F8.

CORRECTIVE ACTION

2/24/93 - Problem with test book. Fixed.

OK
R

Application: GENERIC Application Maintenance

Transaction: GE6011U1 Maintain Online Documentation

Comment ID : 891 Priority: LOW Date: 24-FEB-93

Status : HANDLED Date: 24-FEB-93 Class:
Type : PROBLEM
Assigned To: KEN ATKINS
Entered By : LORRAINE CAMPBELL
System ID : 8203 Oracle ROWID: 00000A4D.0000.000B
Last Error : -0
Last Msg : FRM-40400 Transaction complete -- 1 records posted an
Table : DOCUMENT_HDR
Block : DOCHDR
Field : DOCHDR.COUNT_CHANGE

DESCRIPTION

no error occurred when entering a decimal value. documentation states that an error should occur

CORRECTIVE ACTION

2/24/93 - This field should not have been enterable. Fixed.

OK
RE

Application: GENERIC Application Maintenance

Transaction: GE6011U1 Maintain Online Documentation

=====

Comment ID :	892	Priority: LOW	Date: 24-FEB-93
--------------	-----	---------------	-----------------

=====

Status	: HANDLED	Date: 24-FEB-93	Class:
Type	: PROBLEM		
Assigned To:	KEN ATKINS		
Entered By :	LORRAINE CAMPBELL		
System ID :	2541	Oracle ROWID: 000000C2.003A.000B	
Last Error :	-0		
Last Msg :	FRM-40400 Transaction complete -- 1 records posted an		
Table	: DOCUMENT_NEST		
Block	: DOCNST		
Field	: DOCNST.DISPLAY_SEQ		

DESCRIPTION

a decimal value was allowed here. Documentation states that it should not be allowed.

CORRECTIVE ACTION

2/24/93 - Field set to integer.

OK 2/26/93
k

Application: LEMIS

Liquid Effluent Monitoring Information System

Transaction: GE6011U1 Maintain Online Documentation

```
=====
Comment ID : 894      Priority: LOW      Date: 24-FEB-93
=====
Status      : HANDLED      Date: 24-FEB-93      Class: OPERATIONAL
Type        : COMMENT
Assigned To : KEN ATKINS
Entered By  : KEN ATKINS
System ID   :              Oracle ROWID:
Last Error  : -
Last Msg    : -
Table       :
Block       :
Field       : IDENT_OWNER
=====
```

DESCRIPTION

Recieved FRM-40735 error when selecting 'TEST'.

CORRECTIVE ACTION

2/24/93 - Fixed. Added predicate to limit validation and lov to current application.

OK
2/26/93

2.26 GE6013U1 DOCUMENT TYPE MAINT

LEMIS Module Test Checklist Report

Name: GE6013U1 - DOCUMENT TYPE MAINT
 Short Name: GE6013U1
 Purpose: Maintain the document type code table
 Initial Test Date: 12/18/92 *lc*
 Verification Date: 2/3/93 *lc*

Help	Disp	----- Security -----			
	Fkeys	Ins	Del	Query	Upd
()	()	()	()	()	()

Block: TYPE_DOCUMENT
 Title: Document Type

Help	Scroll	Text	Navigate	Ins	Del	Upd	Qry
()	()	()	()	()	()	()	()

Data Element	Datatype	Dis Len	Help	LOV	Val/ Range	Not Null	Ins
TYPE_DOCUMENT	CHAR	2	()	()	()	()	()
DESCRIPTION	CHAR	30	()	()	()	()	()

Module Description

Module Help

This transaction maintains the document type code table used to validate the DOCUMENT_TYPE data field in the DOCUMENT_HDR table. Valid codes are two characters long and upper case.

To store a new document type code:

Module Help

1. Use the down arrow key to move the cursor to a blank line.

If all of the spaces are filled and more document type codes are to be added, press the down arrow key to the bottom entry and then continue to press the down arrow key until a blank entry appears. The already entered codes will scroll upward, leaving space for an additional code to be entered.

2. Enter the appropriate document type code.
3. Enter a 1-30 character description of the code.
4. To add more codes, press the down arrow key to the next blank space and repeat the entry process.
5. Press the COMMIT key (F10) to store the selected codes in the database.

To update an existing code:

1. Press the down arrow key until the cursor displays the entry to be updated.
2. If the code is to be changed, enter the replacement code over the existing code. CAUTION: Changing a document type code will cause the replacement value to be entered everywhere in the database where the original value was used.
3. If the description is to be changed, TAB to the description field and make the change.
4. Press the COMMIT key (F10) to update the selected codes in the

To delete an existing code:

1. Press the down arrow key until the cursor is located at the code to be deleted.
2. Press the DELETE key. NOTE: If the code is being used by other data in the database, it cannot be deleted. An error message will be displayed.
3. If no error occurs, press the COMMIT key (F10) to delete the code from the database.

Module Test Plan

TEST PLAN

Testing of the transaction is performed at five levels:

1. Developer testing.
2. Secondary testing.
3. System testing.
4. End user testing.
5. Production certification.

Module Test Plan

As each applicable section of this test plan is completed, the designated individual responsible for the section will sign the plan in the appropriate places. Each check off item will be completed. The status of the transaction will be updated in the data dictionary as the transaction progresses through the development and test steps.

Developer Testing

As each individual transaction is generated, the developer performs tests against the transaction as defined in the detail test plan. Upon successful completion of this testing, the transaction is made available for secondary testing.

Test status conditions are:

- () 1 - Transaction under development, untested.
 - () Transaction added to system and security files.
 - () Specific test plan/test results/test case added.
 - () Test plan narrative script loaded.
- () 2 - Development complete, initial unit testing in progress.
 - () Help added
 - () LOV tested
- () 3 - Initial unit testing complete.
 Developer Date Complete 2/18/93

Secondary Testing

Secondary testing is performed by a person external to the development team. Tests are performed using a standardized test checklist. Initial testing is performed to ensure that the transaction meets the functional requirements of the transaction and satisfies the test plan narrative that follows.

- () 4 - Secondary testing in progress.
- () 5 - Secondary testing complete, problems found.
- () 6 - Problem resolution in progress.
- () 7 - Problem resolution complete.
- () 8 - Secondary testing complete, no problems found.
 Tester Date Complete 2/23/93

When problems are found:

1. The tester passes the checklist to the project manager for review and action.
2. The project manager assigns the checklist to the developer of the transaction for correction of the problems.
3. The developer corrects the described problems, records the corrective actions taken and the number of hours of corrective effort used, tests the transaction for correct operation, and returns the

Module Test Plan

checklist to the project manager.

4. The project manager returns the checklist to the tester for retesting.

The above sequence of events repeats until the tester signs off the transaction as passing the test requirements.

System Testing

System testing is performed by a second independent tester. This tester repeats the test plan scenario, the test cases, and any other tests desired under end-user operating conditions.

- () 9 - System testing in progress.
- () 10 - System testing complete, problems found.
- () 11 - Problem resolution in progress.
- () 12 - Problem resolution complete.
Developer _____ Date Complete _____
- () 13 - System testing complete, no problems found.
Tester ~~RA~~ Date Complete ~~2/3/93~~

When problems are found, the same corrective action and retest event sequence as defined in secondary testing above are performed.

End User Testing

The transaction is tested by end users. When appropriate, problem resolution takes place as in previous test segments.

Production Status

- () 19 - The Project Manager reviews all test results and determines that the transaction has successfully completed testing.
 - () Reviews tests results.
 - () Reviews problem reports to ensure all corrections done.
 - () Updates NOTES text to reflect testing results.
 - Project Manager RA Date Complete _____
- () 20 - The transaction is placed into the production environment.
 - () Software files in production directories.
 - () DMR 550 updated to add transaction.
 - () DMR 900 updated to add transaction.
 - () Transaction added to production usage menu.
 - System Architect AS Date Complete 2/3/93

Module Test Case

- (X) Attempt to store duplicate records in the database. Duplicate

records are those where the unique identifier or any unique field contains duplicate values over 2 or more rows.

- (X) Duplicate row insertion attempts result in error message COM00102 [Duplicate value, enter new value or F1 for help]
- (X) Display transaction help. Verify that the narrative adequately describes the purpose and functions of the transaction in end-user terms.
 - () Transaction help describes:
 - () Purpose of the transaction
 - () Functions performed by the transaction
 - () Where multiple page, what happens on each page
- (X) Add a new record by using the insert key.
 - (X) The insert key clears all blocks of the entry screen.
 - (X) A new record can be successfully added and committed.
- (X) Add a new record by scrolling past the last entry using the down arrow key.
 - 744 (X) Synchronization of all blocks occurs while scrolling to the end of the string of retrieved records.
 - (X) All fields of the transaction blank when the end of the string of retrieved records is reached.
 - (X) A new data record can be successfully added.
- (X) Query a single record.
 - (X) A single record is retrieved.
 - (X) Pressing ~~SHIFT~~/F12 clears the record and permits entering a second query.
 - (X) The second query successfully retrieves the desired data.
- (X) Query multiple records using wild cards.
 - (X) Pressing F12 clears the screen and makes the transaction ready for a query.
 - (X) A query composed of a trailing % wildcard retrieves the desired data. Verify by running the same query through SQL*Plus.
 - (X) A query using a prefix % wildcard retrieves the correct data. Verify by running the same query under SQL*Plus.
 - (X) A query using both prefix and suffix % wildcards retrieves the correct data. Verify by running the same query through
- (X) Change each field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.

- (☒) Change each field individually, committing after each change.
- (☐) Change each field, as per the field test plan, for both valid and invalid values.
- (☒) Delete an existing record.
 - (☐) The selected record is deleted from the database. Verify by using SQL*Plus to attempt to retrieve the record.
- (☐) Display the -JN journal record rows for the table to verify that the records changed and deleted above are entered into the journal table correctly.
 - (☐) The _JN journal table for the affected table should contain rows for the initial addition of the record to the database, each change made to the database, and the deletion of the record from the database.
- (☐) Use SQL*Plus to verify all data fields are stored correctly, including ID_ keys, date-updated, owner role, and any other autogenerated fields.
 - (☐) All data fields are correctly stored in the database.

Block: TYPE_DOCUMENT
 Title: Document Type

Block Description

Block Test Case

Block Test Results Expected

Field Name: TYPE_DOCUMENT

Field Description

Field Test Case

- (☒) Begin field with a space.
- (☒) Enter data beyond length of field.
 - (☐) If autoskip is in effect, cursor will skip to the next field.
 - (☒) If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the

Block: TYPE_DOCUMENT
Title: Document Type

Field Name: TYPE_DOCUMENT

Field Test Case

- last position of the field.
- (/) Set keyboard on lower case and enter data. Data should be converted to upper case.
- (/) Clear an existing field and attempt to update the record.

Field Test Results Expected

- (/) Display table/block help. Verify that the narrative adequately describes the purpose and functions of the table/block in end-user terms.
 - () Table/block help describes:
 - () Purpose of the table/block
 - () Functions performed by the table/block
- (/) Add a new record by using the insert key.
 - () The insert key clears all fields of the entry block.
 - () A new record can be successfully added and committed.
- (/) Add a new record by scrolling past the last entry using the down arrow key.
 - () All fields of the table/block blank when the end of the string of retrieved records is reached.
 - () A new data record can be successfully added.
- (/) Query a single record.
 - () A single record is retrieved.
 - () Pressing SHIFT/F12 clears the record and permits entering a second query.
 - () The second query successfully retrieves the desired data.
- (/) Query multiple records using wild cards.
 - () Pressing F12 clears the block and makes the block ready for a query.
 - () A query composed of a trailing % wildcard retrieves the desired data. Verify by running the same query through
 - () A query using a prefix % wildcard retrieves the correct data.

Block: TYPE_DOCUMENT
 Title: Document Type

Field Name: TYPE_DOCUMENT

Field Test Results Expected

- Verify by running the same query under SQL*Plus.
- () A query using both prefix and suffix % wildcards retrieves the correct data. Verify by running the same query through SQL*Plus.
 - (/) Change each field in an existing record. Verify that the change after the changes were made.
 - () Change each field individually, committing after each change.
 - () Change each field, as per the field test plan, for both valid and invalid values.
 - (/) Delete an existing record.
 - () The selected record is deleted from the database. Verify by using SQL*Plus to attempt to retrieve the record.
 - () Display the -JN journal record rows for the table to verify that the records changed and deleted above are entered into the journal table correctly.
 - () The _JN journal table for the affected table should contain rows for the initial addition of the record to the database, each change made to the database, and the deletion of the record from the database.
 - () Use SQL*Plus to verify all data fields are stored correctly, including ID_ keys, date-updated, owner role, and any other autogenerated fields.
 - () All data fields are correctly stored in the database.
 - (/) Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user
 - (/) Field help describes:
 - (/) Purpose of the field
 - () LOV reference if available
 - () Valid range of values where present
 - (/) Required indicator when field is NOT NULL
 - (/) Default value when appropriate
 - (/) Change the field in an existing record. Verify that the change

Block: TYPE_DOCUMENT
Title: Document Type

Field Name: TYPE_DOCUMENT

Field Test Results Expected

occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and

- (/) Change the field, committing after each change.
- () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
- (/) Store a series of records using the SHIFT/F7 [duplicate field] key to carry forward field values from the previously stored record.
- () Use SQL*Plus to verify all data fields are processed correctly.
- (/) Any alphanumeric characters may be entered into the field.
- (/) Delete an existing value.
 - (/) For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.
- (/) Attempt to bypass the field. An error message should tell you that the field data must be entered.
- () Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should result.
- (/) Attempt to enter data as lower case. An error should result.

Block: TYPE_DOCUMENT
 Title: Document Type

Field Name: DESCRIPTION

Field Description

Field Test Case

- (/) Begin field with a space.
- () Enter data beyond length of field.
 - () If autoskip is in effect, cursor will skip to the next field.
 - (/) If no autoskip, cursor will remain at end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (/) Set keyboard on lower case and enter data. Data should be converted to upper case.
- (/) Clear an existing field and attempt to update the record.

Field Test Results Expected

- (/) Display field help. Verify that the narrative adequately describes the purpose and functions of the field in end-user terms.
 - () Field help describes:
 - () Purpose of the field
 - () LOV reference if available
 - () Valid range of values where present
 - () Required indicator when field is NOT NULL
 - () Default value when appropriate
- (/) Change the field in an existing record. Verify that the change occurred by displaying the record on screen before and after the changes were made; display the record using SQL*Plus before and after the changes were made.
 - () Change the field, committing after each change.
 - () Change the field where a range of valid values exists:
 - () Value below valid range should display message COM00101 or COM00103 (if no LOV).
- (/) Store a series of records using the SHIFT/F7 [duplicate field] key to carry forward field values from the previously stored record.

Block: TYPE_DOCUMENT
Title: Document Type

Field Name: DESCRIPTION

Field Test Results Expected

- () Display the -JN journal record rows for the table to verify that the records changed and deleted above are entered into the journal table correctly.
- () The _JN journal table for the affected table should contain rows for the initial addition of the record to the database, each change made to the database, and the deletion of the record from the database.
- () Use SQL*Plus to verify all data fields are processed correctly.
- () All data fields are correctly stored in the database.
- (✓) Any alphanumeric characters may be entered into the field.
- (✓) Delete an existing value.
- (✓) For NOT NULL fields, the COM00107 message [field value may not be deleted] is displayed at commit.
- (✓) Attempt to bypass the field. An error message should tell you that the field data must be entered.
- (✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should result.
- (✓) Attempt to enter data as lower case. An error should result.

Application: GENERIC Application Maintenance

Transaction: GE6013U1 Maintain Document Types

=====

Comment ID : 744 Priority: LOW Date: 18-DEC-92

=====

Status : FIXED Date: 18-JAN-93 Class:

Type : PROBLEM

Assigned To: KEN ATKINS

Entered By : LORRAINE CAMPBELL

System ID : 7626 Oracle ROWID: 00000A80.0010.000B

Last Error : -0

Last Msg : FRM-40400 Transaction complete -- 2 records posted an

Table : TYPE_DOCUMENT

Block : TYPDOC

Field : TYPDOC.TYPE_DOCUMENT

DESCRIPTION

synchronization not working if entries are suppose to be in alpha
order

CORRECTIVE ACTION

01/18/93 - Now sorted by ident*OK*

2.27 GE6015U1 MAINTAIN DOCUMENT START POINT

LEMIS Module Test Checklist Report

Name: MAINTAIN DOCUMENT START POINT
Short Name: GE6015U1
Purpose: To maintain the Online documentation start po
Initial Test Date: 2/22 LC
Verification Date: 2/22 LC

Module Description

This module maintains the on-line documentation starting point for each transaction and block within that transaction.

Module Help

This transaction maintains the on-line documentation starting point for every transaction and block in the system. The records in this table link each transaction / block to a document. If there are no such linkages, the on-line document display screen starts at the root document for the current application.

To add a starting point for a transaction:

- 1) Navigate to an open record by using the arrow keys or pressing <Insert>.
- 2) Enter the desired transaction name. List of values is available. The description of each transaction is displayed below the transaction and block name.
- 3) Enter the desired block name. If "ALL" is entered as the block name, then this is to be the starting document for all blocks within the transaction. The block name will be the same as the table type code for the blocks table.
- 4) Enter the starting document ID. List of values is available. The

Module Help

description of the starting document will be displayed below the document ID.

UNIT TEST CHECKLIST

() Module Function Key

☒ Show Version : Shift-F10 :
☒ Comment/Bug : Shift-F3 :
☒ Fast Access : F5 :
☒ Back : F4 or Esc :
☒ Main Menu : Shift-F4 :
☒ Key Template : Ctrl-F1 :
☒ Context Prnt : F11 :
☒ Query Print : Ctrl-F11 :
☒ Oracle Print : Shift-F11 :

 () Block Title: Online Document Starting Point
 Block Table: DOCUMENT_START

☒ Help : F1	☐ Key Help : Shift-F1 :
☒ Online Doc : F2	☒ Dis Journal : Shift-F2 :
☒ Related Data : F3	☒ Edit Text : F7 :
☒ Spec Func 1 : Shift-F5	☒ Spec Func 2 : Ctrl-F5 :
☒ Home/End : Home/End	☒ Prv/Nxt Block: PgDn/PgUp:
☒ Scroll Up/Dn : Shft-Up/Dn:	☒ Prv/Nxt Field: Tab/Sh-Tb:
☒ Prv/Nxt Rec : Up/Down	☒ Commit Form : F10 :
☒ Clear Field : F8	☒ Clear Record : Shift-F8 :
☒ Clear Form : Shift-F8	☒ Exit System : Ctrl-F4 :
☒ Enter Query : F12	☒ Count Query : Ctrl-F12 :
☒ Correct Query Fields?	☒ Uppercase and LOVs? :
☒ Execute Qry : Shift-F12:	☒ Correct Query Order? :
☒ Insert Rec : Insert	☒ Add record at end :
☒ Mandatory checks OK?	☒ Validation checks OK? :
☒ List of Values OK?	☒ Duplicate Fld: Shift-F6 :
☒ Duplicate Rec: F6	☒ Duplicate checks OK? :
☒ Optional fields marked?	☒ Correct fields uppercase:
<hr/>	
☒ Delete Record: Delete	:

TEST PLAN

Testing of the transaction is performed at four levels:

1. Developer testing.
2. Secondary testing.
3. End user testing.
4. Production release & verification.

As each applicable section of this test plan is completed, the designated individual responsible for the section will sign the plan in the appropriate places. Each check off item will be completed. The status of the transaction will be updated in the data dictionary as the transaction progresses through the development and test steps.

Developer Testing

As each individual transaction is generated, the developer performs tests against the transaction to insure that it meets specifications and works properly. Upon successful completion of this testing, the transaction is made available for secondary testing.

Test status conditions are:

- (✓) 1 - Transaction under development, untested.
 - () Transaction added to menu and security.
 - (✓) Transaction moved to development directory (addmod).
 - (✓) 2 - Development complete, initial unit testing in progress.
 - (✓) Help added
 - (✓) Unit Test checklist completed.
 - (✓) Specific test plan added (if needed).
 - (✓) Any bugs found in Unit test fixed.
 - (✓) Updated test booklet printed.
 - (✓) 3 - Initial unit testing complete.
- Developer AC Date Complete 2/19/93

Secondary Testing

Secondary testing is performed by a person external to the development team. Tests are performed using a standardized test checklist. Initial testing is performed to ensure that the transaction meets the functional requirements of the transaction and satisfies the test plan narrative that follows.

- (✓) 4 - Secondary testing in progress.
 - () 5 - Secondary testing complete, problems found. Resolution in progress.
 - () 6 - Unable to resolve problem at this time.
 - () 7 - Problem resolution complete.
 - (✓) 8 - Secondary testing complete, no problems found.
- Tester AC Date Complete 2/22/93

When problems are found:

1. The tester passes the checklist to the project manager for review and action.

2. The project manager assigns the checklist to the developer of the-transaction for correction of the problems.
3. The developer corrects the described problems, records the corrective actions taken and the number of hours of corrective effort used, tests the transaction for correct operation, and returns the checklist to the project manager.
4. The project manager returns the checklist to the tester for retesting.

The above sequence of events repeats until the tester signs off the transaction as passing the test requirements.

End User Testing

The transaction is tested by end users for a predetermined time period. This testing is done in the development environment. If no problems are encountered, or all problems are resolved at the end of this time period, the module is ready for Project Manager review.

- () 9 - End User testing in progress.
 () 10 - Problem found, resolution in progress.
 () 11 - Problem resolution complete.
 Developer _____ Date Complete _____
 () 12 - Resolution verified.
 Tester _____ Date Complete _____

Production Status

- (✓) 13 - Ready for Project Manager/System Architect review.
 () The Project Manager or System Architect reviews all test results and determines that the transaction has completed testing.
 Architect/Manager A Date Complete 2/23/93
 () 14 - The transaction is ready to be placed into production.
 (✓) Software files moved to production directory.
 (✓) Transaction added to production menu.
 (✓) Transaction access given to production roles.
 (✓) Validation test performed in production.
 () Production documentation up to date.
 (✓) 15 - The transaction has been put into production
 By A Date Done 2/23/93

Module Test Plan

- (✓) Display transaction help. Verify that the narrative adequately describes the purpose and functions of the transaction in end-user terms.

- (/) Transaction help describes:
 - (/) Purpose of the transaction
 - (/) Functions performed by the transaction
 - () Where multiple block, what happens in each block.

Block Title: Online Document Starting Point
 Block Table: DOCUMENT_START
 Block Usage: Full Function

Block Help

This table determines the starting point for on-line documentation when it is accessed from within a transaction. Each transaction may start at any place within on-line documentation. Each block in a transaction may also start at any document.

Block Test Plan

- (/) Display block help. Verify that the narrative adequately describes the purpose and functions of the block in end-user terms:
 - () Block help describes:
 - (/) Purpose of the block
 - (/) Options available in the block
 - (/) The relationship of the block to the transaction or to the principal parent block of the transaction
 - (/) Navigation to other blocks of the transaction
- (/) Where block synchronization is used, ensure that the block data is in sync with the parent block.
- (/) Query a single record:
 - (/) A single record is retrieved.
 - (/) Pressing the F12 key clears the record and permits entering a second query.
 - (/) The second query successfully retrieves the desired data.
- (/) Query multiple records using wild cards:
 - (/) Pressing F12 clears the block and makes the block ready for a query.
 - (/) A query composed of a trailing % wildcard retrieves the desired data.
 - (/) A query using a prefix % wildcard retrieves the correct data.
 - (/) A query using both prefix and suffix % wildcards retrieves the correct data.

- (✓) A query using the underscore (_) position wildcard retrieves the correct data.
- (✓) Verify that the data queries in the block are ordered logically.
- (✓) Store a series of records using the F6 [duplicate record] key to carry forward field values from the previously stored record.
 - (✓) Attempting to store the duplicated record without change should result in an error.
 - (✓) Change key fields and store the record. It should store successfully.
- (✓) Update a record:
 - (✓) The change count should increment.
 - () The date updated should become the current date.
 - () An error should occur if the operator owner identifier does not match the data owner identifier.
- (✓) Delete a record from the block:
 - () If the record has dependent records, the delete should fail unless the help text indicates that a cascading delete is permitted.
 - () Use the clear rcd key (CTRL/F8).
 - () For a single record block, the block should clear.
 - () For a multi-record block, the record pointed to by the cursor should clear.
 - (✓) Use the delete key.
 - () For a single record block, the block should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should appear in the block.
 - (✓) For a multiple record block, the entry should clear. If multiple occurrences of the block's table have been retrieved, the next record in the set should move up to fill the cleared space.

Field Prompt: Transaction
 Field Usage: Non Updateable
 Field Name: NAME_TRANSACTION

Field Help

The 8 character transaction name.

Field Test Plan

- (✓) Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- (✓) Enter data beyond the length of the field:

Field Test Plan

- () If autoskip is in effect, cursor will skip to the next field.
- (✓) If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- (✓) Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- (✓) Attempt to change data in the field. An error should result.
 - (✓) Use the spacebar to attempt to change data.
 - (✓) Use the clear field key (F8) to attempt to change data.
- (✓) Attempt to delete data in the field. An error should result.
 - (✓) Use the spacebar to attempt to delete data.
 - (✓) Use the clear field key (F8) to attempt to delete data.
- (✓) Enter query mode and verify that this field is accessible.
- (✓) Enter query mode and enter a valid value in this field.
 - (✓) Only records matching the entered value should be displayed.
- (✓) Attempt to delete an existing data field value. An error should result.
 - (✓) Use the space bar to attempt to delete the field data.
 - (✓) Use clear field (F8) to attempt to delete the field data.
- (✓) Enter data into the field, backspace the data out of the field and attempt to TAB to the next field. An error should occur.
- (✓) Test LOV:
 - (✓) Select the first entry in the LOV.
 - (✓) Select the last entry in the LOV.
 - (✓) Use the FIND option and select an entry.
- () Enter a duplicate value. An error should occur.
- (✓) From LOV, select a value already present in the database. An error should occur.

Duplication

Field Prompt:
Field Usage: Queryable Only
Field Name: DESCR TRANSACTION

Field Help

A brief description of the transaction. This field will be used as the transaction name in the menus, and on any reports. The description may be entered in mixed case.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.

Field Test Plan

-
- (X) Enter query mode and verify that this field is accessible.
 - (X) Enter query mode and enter a valid value in this field.
 - (X) Only records matching the entered value should be displayed.
 - (X) Enter a duplicate value. An error should occur.
 - (X) From LOV, select a value already present in the database. An error should occur.
- NA

Field Prompt: Block Name
 Field Usage: Non Updateable
 Field Name: BLOCK_NAME

Field Help

The block name of the block that this will be the starting document for. The combination of transaction and block name will determine which starting document will be used. If "ALL" is entered here this will define a starting document for ALL blocks within a transaction.

Field Test Plan

- () ✓ Verify that the field accepts character or numeric data within the limits imposed by validation criteria.
- () ✓ Enter data beyond the length of the field:
 - () ✓ If autoskip is in effect, cursor will skip to the next field.
 - () ✓ If no autoskip, cursor will remain at the end of the field and overlay characters, leaving the last character keyed in the last position of the field.
- () ✓ Display column help. Verify that the narrative adequately describes the column's purpose and functions in end-user terms.
- () ✓ Attempt to change data in the field. An error should result.
 - () ✓ Use the spacebar to attempt to change data.
 - () ✓ Use the clear field key (F8) to attempt to change data.
- () ✓ Attempt to delete data in the field. An error should result.
 - () ✓ Use the spacebar to attempt to delete data.
 - () ✓ Use the clear field key (F8) to attempt to delete data.
- () Enter query mode and verify that this field is accessible.
- () Enter query mode and enter a valid value in this field.
 - () Only records matching the entered value should be displayed.
- () Accept the default.
- () Modify each default value with a valid alternative.
- () Modify each default value with an invalid alternative.
- () ✓ Attempt to delete an existing data field value. An error should result.
 - () ✓ Use the space bar to attempt to delete the field data.
 - () ✓ Use clear field (F8) to attempt to delete the field data.
- () ✓ Enter data into the field, backspace the data out of the field and

Duplication

attempt to TAB to the next field. An error should occur.

(☒) Enter a duplicate value. An error should occur.

(☐) From LOV, select a value already present in the database. An error should occur.

Field Prompt: Doc Ident
Field Usage: Queryable Only
Field Name: IDENT_DOCUMENT

Field Help

Enter a unique identifier for this document item.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

(☒) Enter query mode and verify that this field is accessible.

(☒) Enter query mode and enter a valid value in this field.

(☒) Only records matching the entered value should be displayed.

Field Prompt: Title
Field Usage: Queryable Only
Field Name: TITLE_DOCUMENT

Field Help

Enter the title for this document item. This title will be displayed in the document selection list and at the top of the screen when the document is being displayed.

This data field:

- May contain any character combination.
- Must contain a valid value. A blank is not allowed.
- All input is converted to upper case.

Field Test Plan

(☒) Enter query mode and verify that this field is accessible.

(☒) Enter query mode and enter a valid value in this field.

(☒) Only records matching the entered value should be displayed.

This page intentionally left blank.

3.0 LIBRARY

3.1 LI5051R1 CONSTITUENT/METHOD REPORT

Initial Test Date: 20 5/11/94 Verification Date: 5/11/94

- (✓) Verify that the report is accessible from the menu.
- (✓) Verify that the report is run correctly when requested.
- (✓) Verify that the report format is correct.
- (✓) Verify that the correct data is listed in the report.

3.2 LI5053R1 ANALYTICAL METHODS LISTING

Initial Test Date: 20 5/11/94 Verification Date: 5/11/94

- (✓) Verify that the report is accessible from the menu.
- (✓) Verify that the report is run correctly when requested.
- (✓) Verify that the report format is correct.
- (✓) Verify that the correct data is listed in the report.

3.3 LI5055R1 METHOD/CONSTITUENTS REPORT

Initial Test Date: 20 5/11/94 Verification Date: 5/11/94

- (✓) Verify that the report is accessible from the menu.
- (✓) Verify that the report is run correctly when requested.
- (✓) Verify that the report format is correct.
- (✓) Verify that the correct data is listed in the report.

3.4 LI5056R1 LABORATORY/METHOD REPORT

Initial Test Date: 20 5/11/94 Verification Date: 5/11/94

- (✓) Verify that the report is accessible from the menu.
- (✓) Verify that the report is run correctly when requested.
- (✓) Verify that the report format is correct.
- (✓) Verify that the correct data is listed in the report.

This page intentionally left blank.