Kernel 8.0: Patches XU*8.0*608, 607, and 672

Lock Manager Supplement to Patch Description

October 2019

Department of Veterans Affairs (VA)

Office of Information and Technology (OIT)

Enterprise Program Management Office (EPMO)

Revision History

Date	Revision	Description	Author
10/28/2019	1.0	Initial documentation for the Kernel Lock Manager Class 3 software to Class 1: **Kernel Patches XU*8.0*608, 607, and 672**	• Developers: REDACTED • Tech Writer: REDACTED

REF: For the current patch history related to this software, see the Patch Module (i.e., **Patch User Menu** [A1AE USER]) on FORUM.

Table of Contents

List of Figures

List of Tables

Orientation

Acknowledgments

The Kernel Lock Manager was originally developed by REDACTED. Without his efforts and support this product would *not* exist!

How to Use this Manual

The *Kernel Lock Manager Supplement to Patch Description* document for Kernel Patches XU*8.0*608, 607, and 672 describes the "*how to*" information of the Kernel Lock Manager functionality.

Intended Audience

The intended audience of this manual is the following stakeholders:

- Enterprise Program Management Office (EPMO)—VistA legacy development teams.
- System Administrators—System administrators at Department of Veterans Affairs (VA) sites who are responsible for computer management and system security on the VistA M Servers.
- Information Security Officers (ISOs)—Personnel at VA sites responsible for system security.
- Product Support (PS).

Disclaimers

Software Disclaimer

This software was developed at the Department of Veterans Affairs (VA) by employees of the Federal Government in the course of their official duties. Pursuant to title 17 Section 105 of the United States Code this software is *not* subject to copyright protection and is in the public domain. VA assumes no responsibility whatsoever for its use by other parties, and makes no guarantees, expressed or implied, about its quality, reliability, or any other characteristic. We would appreciate acknowledgement if the software is used. This software can be redistributed and/or modified freely provided that any derivative works bear some notice that they are derived from it, and any modified versions bear some notice that they have been modified.

CAUTION: To protect the security of VistA systems, distribution of this software for use on any other computer system by VistA sites is prohibited. All requests for copies of Kernel for *non*-VistA use should be referred to the VistA site's local Office of Information Field Office (OIFO).

Documentation Disclaimer

This manual provides an overall explanation of using the VistA Services Assembler (VSA) Wizard and the functionality contained in the VistA Services Assembler (VSA) Phase 2 software; however, no attempt is made to explain how the overall VistA programming system is integrated and maintained. Such methods and procedures are documented elsewhere. We suggest you look at the various VA Internet and Intranet SharePoint sites and websites for a general orientation to VistA. For example, visit the Office of Information and Technology (OIT) Enterprise Program Management Office (EPMO) Intranet website.

DISCLAIMER: The appearance of any external hyperlink references in this manual does *not* constitute endorsement by the Department of Veterans Affairs (VA) of this website or the information, products, or services contained therein. The VA does *not* exercise any editorial control over the information you find at these locations. Such links are provided and are consistent with the stated purpose of this VA Intranet Service.

Documentation Conventions

This manual uses several methods to highlight different aspects of the material:

- Various symbols are used throughout the documentation to alert the reader to special information. The following table gives a description of each of these symbols:

Table 1: Documentation Symbol Descriptions

Symbol	Description
ℹ	**NOTE / REF:** Used to inform the reader of general information including references to additional reading material.
⚠	**CAUTION / RECOMMENDATION / DISCLAIMER:** Used to caution the reader to take special notice of critical information.

- Descriptive text is presented in a proportional font (as represented by this font).
- Conventions for displaying TEST data in this document are as follows:
  - The first three digits (prefix) of any Social Security Numbers (SSN) begin with either "**000**" or "**666**".
  - Patient and user names are formatted as follows:
    - *<Application Name/Abbreviation/Namespace>*PATIENT,*<N>*
    - *<Application Name/Abbreviation/Namespace>*USER,*<N>*

Where:

- *<Application Name/Abbreviation/Namespace>* is defined in the Approved Application Abbreviations document.
- *<N>* represents the first name as a number spelled out and incremented with each new entry.

For example, in Kernel (XU or KRN) test patient and user names would be documented as follows:

KRNPATIENT,ONE; KRNPATIENT,TWO; KRNPATIENT,THREE; … KRNPATIENT,14; etc.

KRNUSER,ONE; KRNUSER,TWO; KRNUSER,THREE; … KRNUSER,14; etc.

- "Snapshots" of computer online displays (i.e., screen captures/dialogues) and computer source code is shown in a *non*-proportional font and may be enclosed within a box.
  - User's responses to online prompts are **bold** typeface and highlighted in yellow (e.g., **<Enter>**).
  - Emphasis within a dialogue box is **bold** typeface and highlighted in blue (e.g., STANDARD LISTENER: RUNNING).
  - Some software code reserved/key words are **bold** typeface with alternate color font.
  - References to "**<Enter>**" within these snapshots indicate that the user should press the <**Enter**> key on the keyboard. Other special keys are represented within **< >** angle brackets. For example, pressing the **PF1** key can be represented as pressing **<PF1>**.
  - Author's comments are displayed in italics or as "callout" boxes.

  NOTE: Callout boxes refer to labels or descriptions usually enclosed within a box, which point to specific areas of a displayed image.

- All uppercase is reserved for the representation of M code, variable names, or the formal name of options, field/file names, and security keys (e.g., XUPROGMODE security key).

  NOTE: Other software code (e.g., Delphi/Pascal and Java) variable names and file/folder names can be written in lower or mixed case (i.e., CamelCase).

Documentation Navigation

This document uses Microsoft® Word's built-in navigation for internal hyperlinks. To add **Back** and **Forward** navigation buttons to your toolbar, do the following:

1. Right-click anywhere on the customizable Toolbar in Word 2007 or higher (*not* the Ribbon section).
2. Select **Customize Quick Access Toolbar** from the secondary menu.

3. Select the **drop-down arrow** in the “Choose commands from:” box.
4. Select **All Commands** from the displayed list.
5. Scroll through the command list in the left column until you see the **Back** command (green circle with arrow pointing left).
6. Select/Highlight the **Back** command and press **Add** to add it to your customized toolbar.
7. Scroll through the command list in the left column until you see the **Forward** command (green circle with arrow pointing right).
8. Select/Highlight the **Forward** command and press **Add** to add it to your customized toolbar.
9. Select **OK**.

You can now use these **Back** and **Forward** command buttons in your Toolbar to navigate back and forth in your Word document when clicking on hyperlinks within the document.

NOTE: This is a one-time setup and is automatically available in any other Word document once you install it on the Toolbar.

How to Obtain Technical Information Online

Exported VistA M Server-based software file, routine, and global documentation can be generated through the use of Kernel, MailMan, and VA FileMan utilities.

NOTE: Methods of obtaining specific technical information online is indicated where applicable under the appropriate topic.

REF: See the *Kernel 8.0 & Kernel Toolkit 7.3 Technical Manual* for further information.

Help at Prompts

VistA M Server-based software provides online help and commonly used system default prompts. Users are encouraged to enter question marks at any response prompt. At the end of the help display, you are immediately returned to the point from which you started. This is an easy way to learn about any aspect of the software.

Obtaining Data Dictionary Listings

Technical information about VistA M Server-based files and the fields in files is stored in data dictionaries (DD). You can use the **List File Attributes** [DILIST] option on the **Data Dictionary Utilities** [DI DDU] menu in VA FileMan to print formatted data dictionaries.

REF: For details about obtaining data dictionaries and about the formats available, see the “List File Attributes” chapter in the “File Management” section in the *VA FileMan Advanced User Manual*.

Assumptions

This manual is written with the assumption that the reader is familiar with the following:

- VistA computing environment:
    - Kernel—VistA M Server software
    - VA FileMan data structures and terminology—VistA M Server software
- Microsoft Windows environment
- M programming language

Reference Materials

Readers who wish to learn more about Kernel should consult the following documents:

- *Kernel Release Notes*
- *Kernel Installation Guide*
- *Kernel 8.0 & Kernel Toolkit 7.3 Systems Management Guide*
- *Kernel 8.0 & Kernel Toolkit 7.3 Developer's Guide*
- *Kernel 8.0 & Kernel Toolkit 7.3 Technical Manual*
- *Kernel Security Tools Manual*
- Kernel VA Intranet website

    This site contains other information and provides links to additional documentation.

VistA documentation is made available online in Microsoft Word format and in Adobe Acrobat Portable Document Format (PDF). The PDF documents *must* be read using the Adobe Acrobat Reader, which is freely distributed by Adobe Systems Incorporated at the following website: http://www.adobe.com/

VistA documentation can be downloaded from the VA Software Document Library (VDL) website: http://www.va.gov/vdl/

REF: Kernel manuals are located on the VDL at: http://www.va.gov/vdl/application.asp?appid=10

VistA documentation and software can also be downloaded from the Product Support (PS) anonymous directories.

1 Systems Management Guide Insert—Lock Manager Utility

1.1 Kernel Lock Manager Overview

The Kernel Lock Manager utility is based on the original Class 3 VistA Lock Manager software developed by T. M. This software has been updated to Class 1 software via the following Kernel patches:

- **XU*8.0*608—**Contains all the software components that make up the Kernel Lock Manager, which includes the XULM LOCK DICTIONARY (#8993) file.
- **XU*8.0*607—**Populates the XULM LOCK DICTIONARY (#8993) file, which is included in Patch XU*8.0*608. It requires the KIDS enhancement Patch XU*8.0*672.
- **XU*8.0*672—**Enhances the Kernel Installation and Distribution System (KIDS) to allow applications to distribute entries in the XULM LOCK DICTIONARY (#8993) file as KIDS components.

The principle use of the Kernel Lock Manager utility is to assist users in locating locks held by a process that has become dissociated from an active user. Once located, this utility kills the process that owns the lock, thereby releasing the locks held by that process.

The principal advantages of the Kernel Lock Manager utility over the existing Caché utilities include the following functionality:

- Ability to use the Lock Manager from within Veterans Health Information Systems and Technology Architecture (VistA).
- Cross-Node capabilities—No longer need to log into multiple nodes, even if the process that holds the lock is on a different node than the one you currently logged onto. This is accomplished by using the RPC Data Broker to execute Remote Procedure Calls (RPCs) on the other nodes to obtain the lock table and to terminate processes.
- Built-in VistA expertise via the new XULM LOCK DICTIONARY (#8993) file—This file provides in-depth details about the following:
    - Locks
    - Files that the locks reference
    - Processes that hold the locks
- Extendible Lock dictionary—Ability to add information about locks is included in the initial release of the Lock Dictionary. The LOCK TEMPLATE component was added to KIDS via Kernel Patch XU*8.0*672. It allows application developers to add to the Lock Dictionary and distribute their additions via KIDS.

1.2 Configuration

There are two steps to configuring the Lock Manager:

1. Entering Site Parameters
2. Add Lock Manager Users

1.2.1 Entering Site Parameters—Edit Lock Manager Parameters Option

Use the **Edit Lock Manager Parameters** [XULM EDIT PARAMETERS] option to update the Lock Manager parameters in the XULM LOCK MANAGER PARAMETER S (#8993.1) file.

To edit the Lock Manager parameter, perform the following procedure:

1. From the **Lock Manager Menu** [XULM LOCK MANAGER MENU], select the **Edit Lock Manager Parameters** [XULM EDIT PARAMETERS] option.
2. At the "APPLICATION STATUS:" prompt, set the application status to **ENABLED**.
3. For each node in the system configuration, do the following:

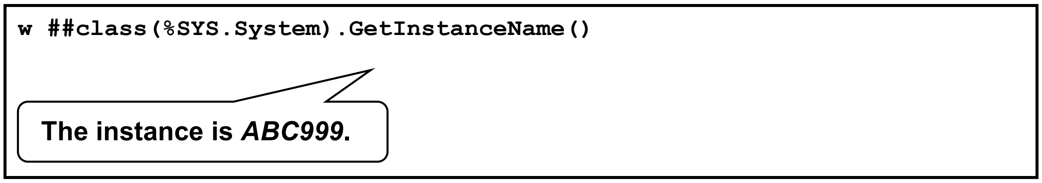
    a. At the "Select NODES:" prompt, enter Caché instance. The name can be obtained by logging onto each node and entering at the M prompt:

    ```
    w ##class(%SYS.System).GetInstanceName()
    ```

    The returned value is the Caché instance name.

    In the example in Figure 1, the instance is named *ABC999*:

    Figure 1: Sample Code Using GetInstanceName Library Call to Get Instance Name



```
w ##class(%SYS.System).GetInstanceName()
```

The instance is *ABC999*.

    b. At the "TCP/IP ADDRESS:" prompt, enter the **IP address**.

    c. At the "BROKER PORT:" prompt, enter the **port number of the Broker running on that node**. Either the RPC Broker port or the M-to-M port can be used, but the RPC Broker port is recommended and is more widely available.

    d. The "SHORT DISPLAY NAME" prompt is optional. If the node's name is over **8** characters long, it is necessary at times to display a shortened version. The default is to display only the last **8** characters. If the result is *not* satisfactory, you can enter a shortened name for the node to use as an alternative. **This pertains especially to Linux systems.**

Figure 2: Edit Lock Manager Parameters Option [XULM EDIT PARAMETERS]—Editing Site Parameters

```
Select Operations Management Option: LOCK <Enter> Lock Manager Menu


   LM     Kernel Lock Manager
   EDIT   Edit Lock Dictionary
   LOG    View Lock Manager Log
   SITE   Edit Lock Manager Parameters
   PURG   PURGE LOCK MANAGER LOG


Select Lock Manager Menu Option: SITE <Enter> Edit Lock Manager Parameters
APPLICATION STATUS: ENABLED// <Enter>
Select NODES: YYYYYYYY// <Enter>
  TCP/IP ADDRESS: 99.9.99.99// <Enter>
  BROKER PORT: 9999// <Enter>
  SHORT DISPLAY NAME: NODEX// <Enter>
```

1.2.2 Add Lock Manager Users

The following steps give a user access to the Lock Manager:

1. Assign XULM LOCKS Security Key
2. Assign XULM RPC BROKER CONTEXT Option
3. Assign XULM SYSTEM LOCKS Security Key

1.2.2.1 Assign XULM LOCKS Security Key

To assign the XULM LOCKS security key, perform the following procedure:

1. From the **Systems Manager Menu** [EVE], select the **Menu Management** [XUMAINT] menu.
2. At the “Select Menu Management Option:” prompt, select the **Key Management** [XUKEYMGMT] menu.
3. At the “Select Key Management Option:” prompt, select the **Allocation of Security Keys** [XUKEYALL] option.
4. At the “Allocate key:” prompt, enter **XULM LOCKS** security key.
5. At the “Another key:” prompt, press **Enter** to complete your entries.
6. At the “Holder of key:” prompt, enter the user’s name.

7. At the “Another holder:” prompt, enter any additional user names that need access to the Lock Manager. When complete, press **Enter**.
8. At the “You are allocating keys. Do you wish to proceed? YES//” prompt, press **Enter** to accept the **YES** default response.

Figure 3: Adding Lock Manager Users by Assigning XULM LOCKS Security Key

```
Select Systems Manager Menu Option: MENU <Enter> Management


          Edit options

          Key Management ...

          Secure Menu Delegation ...

          Restrict Availability of Options

          Option Access By User

          List Options by Parents and Use

          Fix Option File Pointers

          Help Processor ...

   OPED   Screen-based Option Editor

          Display Menus and Options ...

          Edit a Protocol

          Menu Rebuild Menu ...

          Out-Of-Order Set Management ...

          See if a User Has Access to a Particular Option

          Show Users with a Selected primary Menu


Select Menu Management Option: KEY <Enter> Management


          Allocation of Security Keys

          De-allocation of Security Keys

          Enter/Edit of Security Keys

          All the Keys a User Needs

          Change user’s allocated keys to delegated keys

          Delegate keys

          Keys For a Given Menu Tree

          List users holding a certain key

          Remove delegated keys

          Show the keys of a particular user


Select Key Management Option: ALLOC <Enter> ation of Security Keys
```

```
Allocate key: XULM LOCKS

Another key: <Enter>

Holder of key: XUUSER,ONE <Enter>      OX         TECHNICAL WRITER

Another holder: <Enter>

You've selected the following keys:

XULM LOCKS

You've selected the following holders:

XUUSER,ONE

You are allocating keys.  Do you wish to proceed? YES// <Enter>

XULM LOCKS being assigned to:
     XUUSER,ONE
```

1.2.2.2 Assign XULM RPC BROKER CONTEXT Option

The **KERNEL LOCK MANAGER** [XULM RPC BROKER CONTEXT] option is the context option the RPC Broker uses for the Lock Manager when making remote procedure calls.

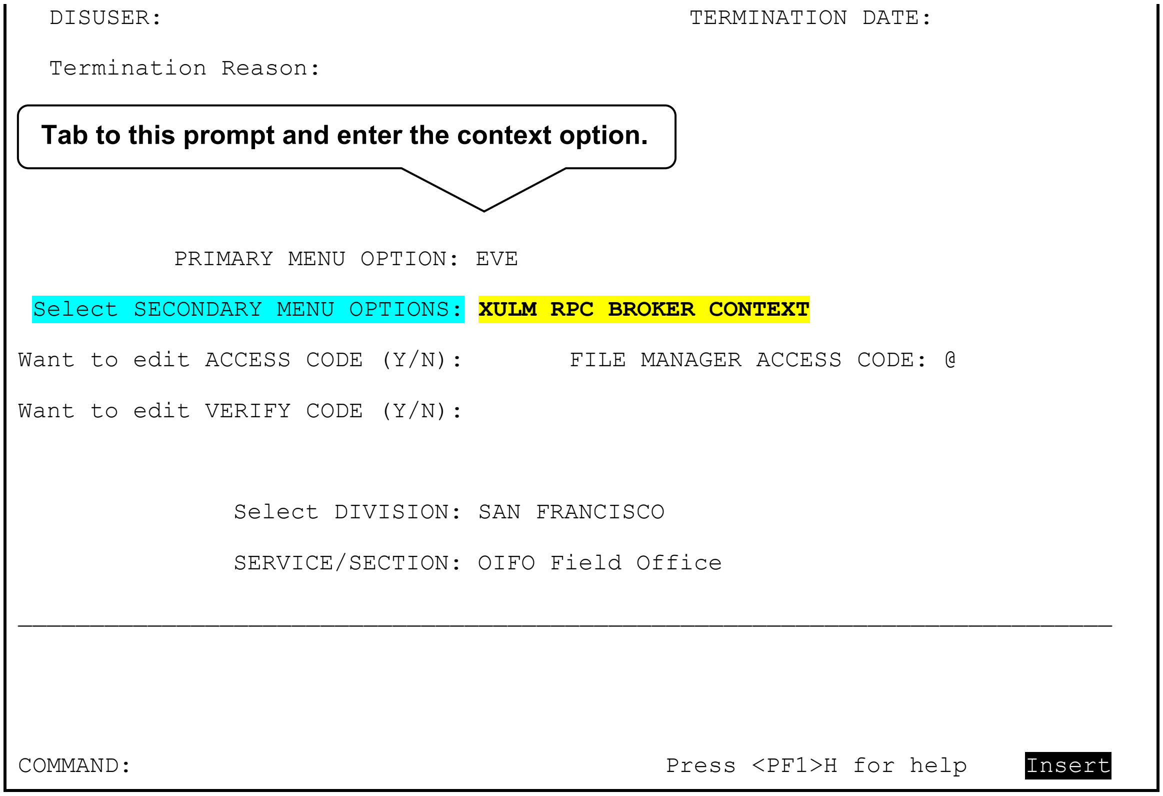
To assign the **XULM RPC BROKER CONTEXT** option for each user, perform the following procedure:

1. From the **Systems Manager Menu** [EVE], select the **User Management** [XUSER] menu.
2. At the "Select User Management Option:" prompt, select the **Edit an Existing User** [XUSEREDIT] option.
3. At the "Select NEW PERSON NAME:" prompt, enter the user's name.
4. In the "Edit an Existing User" main screen, tab down to the "Select SECONDARY MENU OPTIONS:" prompt, enter the **XULM RPC BROKER CONTEXT** option.
5. (Optional) In the "SECONDARY MENU OPTIONS" popup screen, tab to "SYNONYM:" prompt and enter a synonym for this context option.

6. Tab to the “COMMAND:” prompt, enter **CLOSE**. The “SECONDARY MENU OPTIONS” popup screen closes.
7. Tab to the “COMMAND:” prompt, enter **EXIT**. The “Edit an Existing User” main screen closes.

Figure 4: Assigning XULM RPC BROKER CONTEXT Option—Sample User Entries and System Responses (1 of 2)

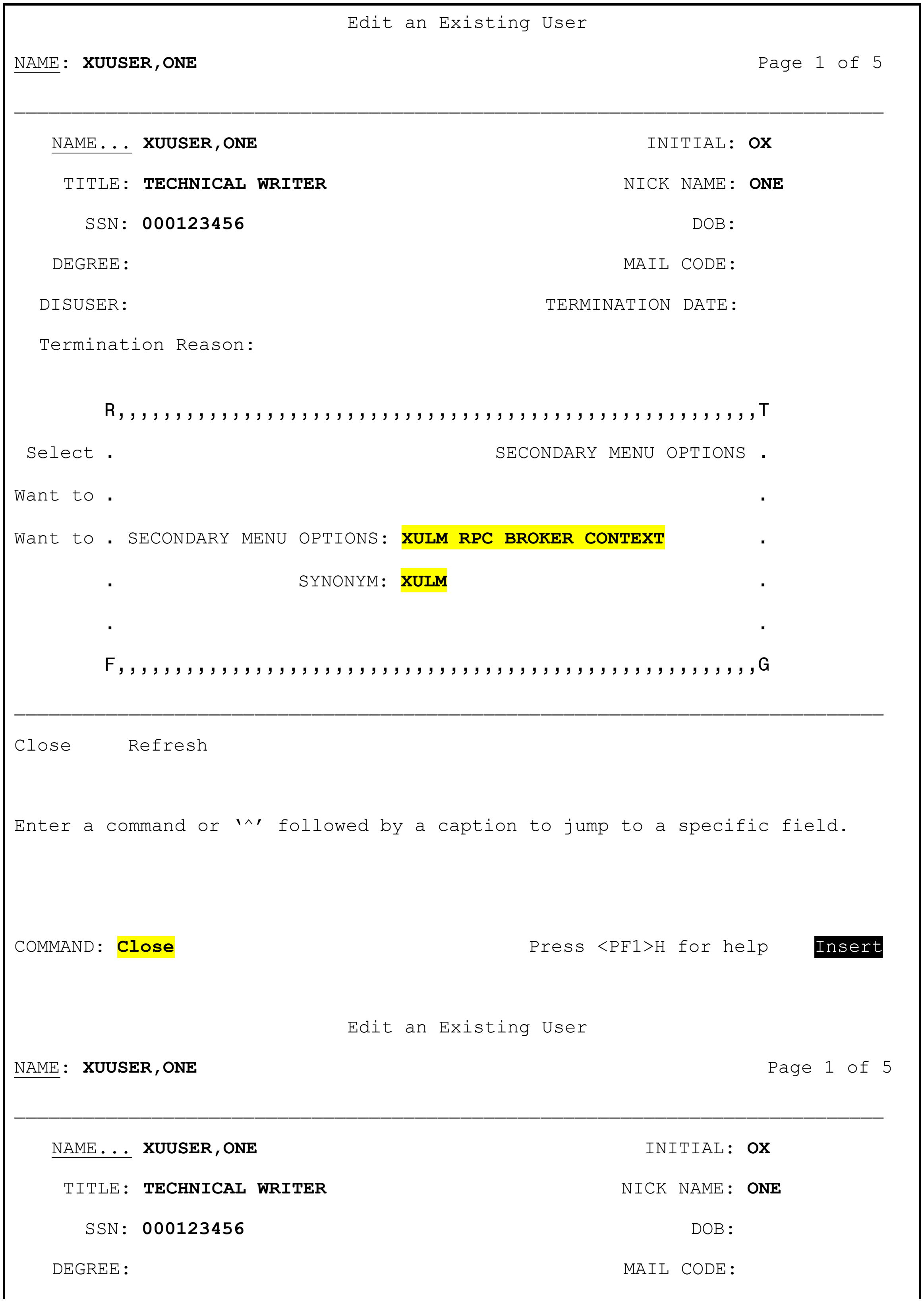
```
Select Systems Manager Menu Option: USER <Enter> Management


          Add a New User to the System
          Grant Access by Profile
          Edit an Existing User
          Deactivate a User
          Reactivate a User
          List users
          User Inquiry
          Switch Identities
          File Access Security ...
          Clear Electronic signature code
   OAA    OAA Trainee Registration Menu ...
          Electronic Signature Block Edit
          List Inactive Person Class Users
          Manage User File ...
          Person Class Edit
          Print Patch Report
          Reprint Access agreement letter


Select User Management Option: EDIT <Enter> an Existing User


Select NEW PERSON NAME: XUUSER <Enter> XUUSER,ONE       OX        TECHNICAL
WRITER


                             Edit an Existing User
NAME: XUUSER,ONE                                                    Page 1 of 5
_____________________________________________________________________________
   NAME... XUUSER,ONE                                         INITIAL: OX
    TITLE: TECHNICAL WRITER                                 NICK NAME: ONE
      SSN: 000123456                                              DOB:
   DEGREE:                                                  MAIL CODE:
```



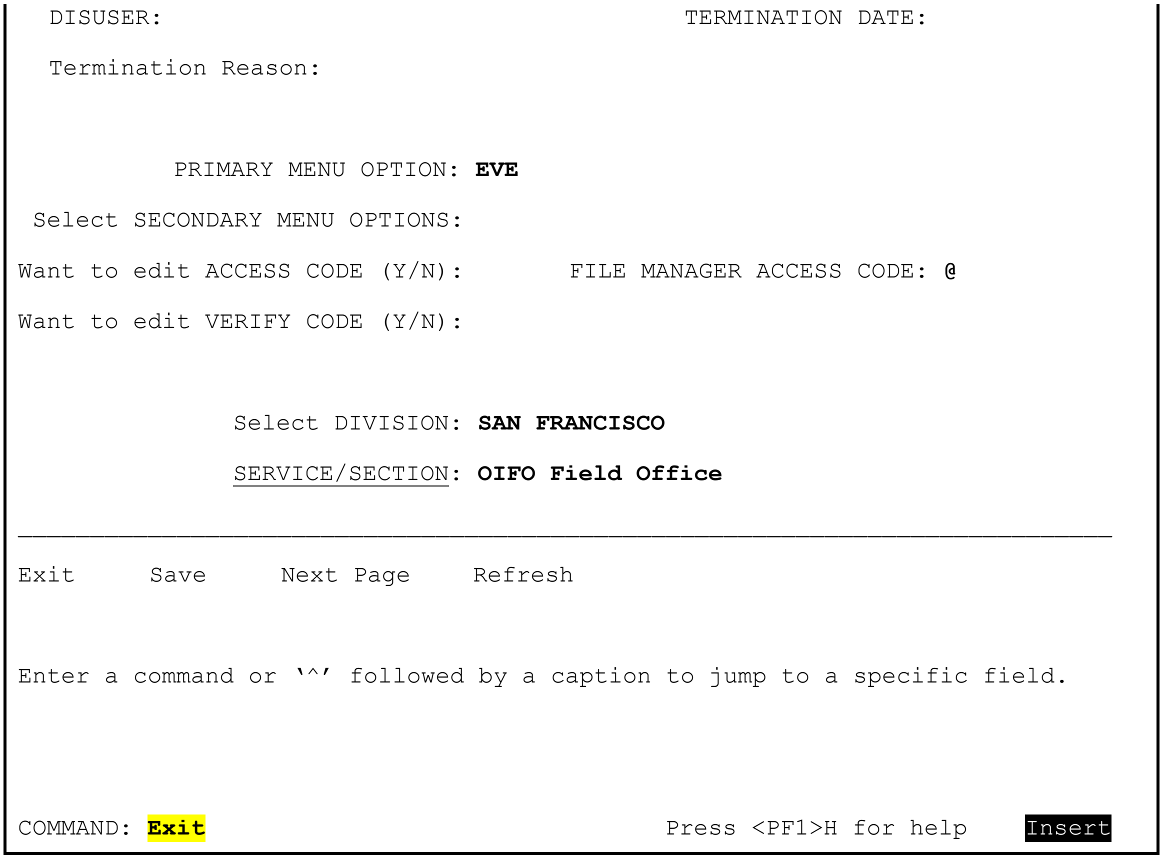
Tab to this prompt and enter the context option.

```
  DISUSER:                                    TERMINATION DATE:

  Termination Reason:



           PRIMARY MENU OPTION: EVE

 Select SECONDARY MENU OPTIONS: XULM RPC BROKER CONTEXT

Want to edit ACCESS CODE (Y/N):       FILE MANAGER ACCESS CODE: @

Want to edit VERIFY CODE (Y/N):


               Select DIVISION: SAN FRANCISCO

               SERVICE/SECTION: OIFO Field Office
____________________________________________________________________________



COMMAND:                                      Press <PF1>H for help    Insert
```

Figure 5: Assigning XULM RPC BROKER CONTEXT Option—Sample User Entries and System Responses (2 of 2)



```
                               Edit an Existing User
NAME: XUUSER,ONE                                                     Page 1 of 5
_______________________________________________________________________________
   NAME... XUUSER,ONE                                        INITIAL: OX
    TITLE: TECHNICAL WRITER                                NICK NAME: ONE
      SSN: 000123456                                             DOB:
   DEGREE:                                                 MAIL CODE:
  DISUSER:                                          TERMINATION DATE:
  Termination Reason:

        R,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,T
 Select .                                   SECONDARY MENU OPTIONS .
Want to .                                                           .
Want to . SECONDARY MENU OPTIONS: XULM RPC BROKER CONTEXT           .
        .                SYNONYM: XULM                              .
        .                                                           .
        F,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,G
_______________________________________________________________________________
Close     Refresh

Enter a command or '^' followed by a caption to jump to a specific field.


COMMAND: Close                                  Press <PF1>H for help    Insert

                               Edit an Existing User
NAME: XUUSER,ONE                                                      Page 1 of 5
_______________________________________________________________________________
   NAME... XUUSER,ONE                                        INITIAL: OX
    TITLE: TECHNICAL WRITER                                NICK NAME: ONE
      SSN: 000123456                                             DOB:
   DEGREE:                                                 MAIL CODE:
```



```
   DISUSER:                                    TERMINATION DATE:
   Termination Reason:

            PRIMARY MENU OPTION: EVE
  Select SECONDARY MENU OPTIONS:
Want to edit ACCESS CODE (Y/N):       FILE MANAGER ACCESS CODE: @
Want to edit VERIFY CODE (Y/N):

                Select DIVISION: SAN FRANCISCO
                SERVICE/SECTION: OIFO Field Office
_____________________________________________________________________________
Exit     Save     Next Page    Refresh

Enter a command or '^' followed by a caption to jump to a specific field.


COMMAND: Exit                                   Press <PF1>H for help    Insert
```

1.2.2.3 Assign XULM SYSTEM LOCKS Security Key

CAUTION: Use discretion when assigning this security key; deleting a system lock can result in database corruption!

To assign the XULM SYSTEM LOCKS security key, perform the following procedure:

1. From the **Systems Manager Menu** [EVE], select the **Menu Management** [XUMAINT] menu.
2. At the “Select Menu Management Option:” prompt, select the **Key Management** [XUKEYMGMT] menu.
3. At the “Select Key Management Option:” prompt, select the **Allocation of Security Keys** [XUKEYALL] option.
4. At the “Allocate key:” prompt, enter **XULM SYSTEM LOCKS** security key.
5. At the “Another key:” prompt, press **Enter** to complete your entries.
6. At the “Holder of key:” prompt, enter the user’s name.
7. At the “Another holder:” prompt, enter any additional user names that need access to the Lock Manager. When complete, press **Enter**.

8. At the “You are allocating keys. Do you wish to proceed? YES//” prompt, press **Enter** to accept the **YES** default response.

Figure 6: Adding Lock Manager Users by Assigning XULM SYSTEM LOCKS Security Key

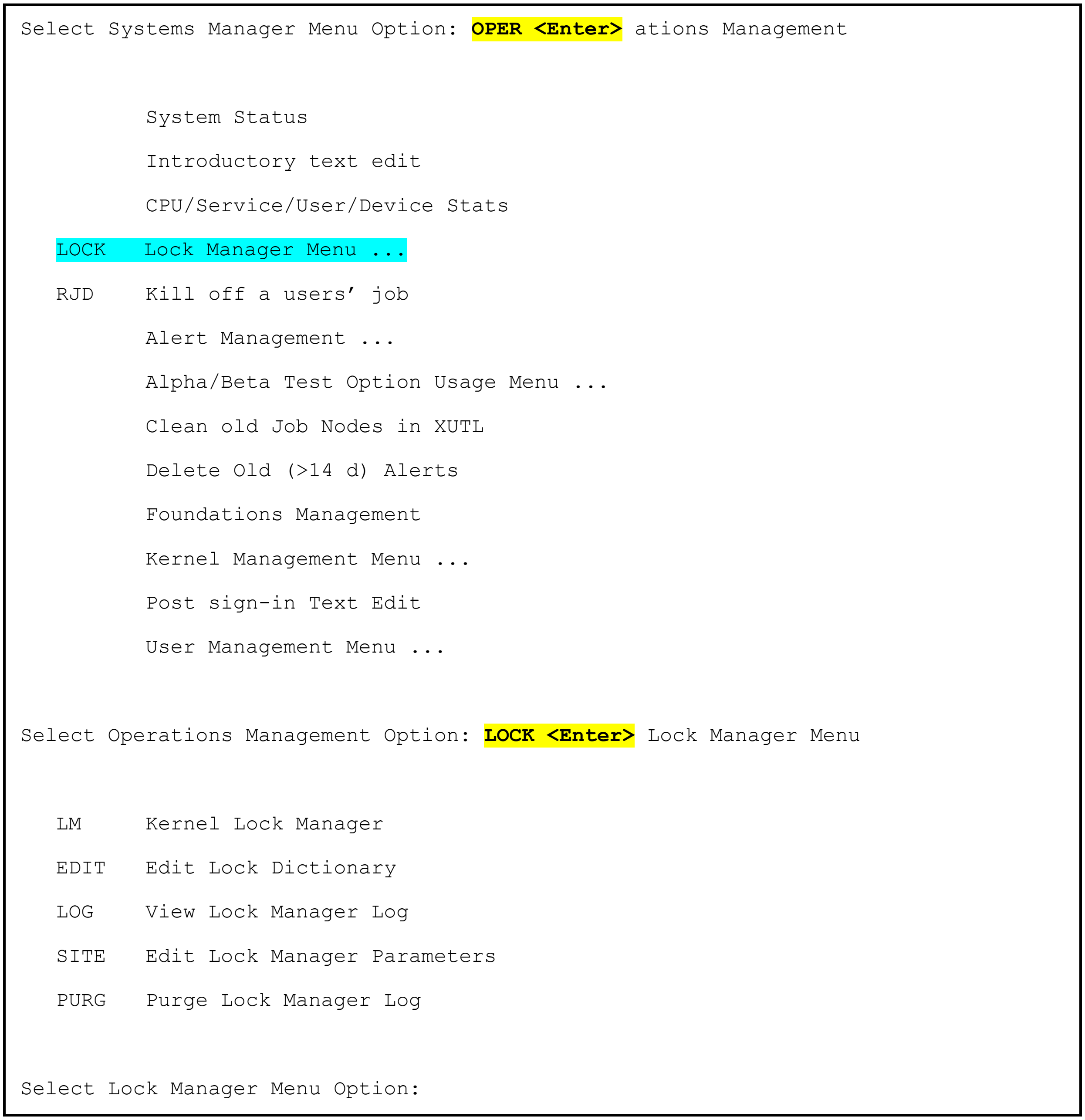
```
Select Systems Manager Menu Option: MENU <Enter> Management


          Edit options
          Key Management ...
          Secure Menu Delegation ...
          Restrict Availability of Options
          Option Access By User
          List Options by Parents and Use
          Fix Option File Pointers
          Help Processor ...
   OPED   Screen-based Option Editor
          Display Menus and Options ...
          Edit a Protocol
          Menu Rebuild Menu ...
          Out-Of-Order Set Management ...
          See if a User Has Access to a Particular Option
          Show Users with a Selected primary Menu

Select Menu Management Option: KEY <Enter> Management


          Allocation of Security Keys
          De-allocation of Security Keys
          Enter/Edit of Security Keys
          All the Keys a User Needs
          Change user’s allocated keys to delegated keys
          Delegate keys
          Keys For a Given Menu Tree
          List users holding a certain key
          Remove delegated keys
          Show the keys of a particular user

Select Key Management Option: ALLOC <Enter> ation of Security Keys
```

```
Allocate key: XULM SYSTEM LOCKS

Another key: <Enter>

Holder of key: XUUSER,ONE <Enter>      OX          TECHNICAL WRITER

Another holder: <Enter>

You've selected the following keys:

XULM SYSTEM LOCKS

You've selected the following holders:

XUUSER,ONE

You are allocating keys.  Do you wish to proceed? YES// <Enter>

XULM SYSTEM LOCKS LOCKS being assigned to:
     XUUSER,ONE
```

1.3 Options

The **Lock Manager Menu** [XULM LOCK MANAGER MENU] is located on the **Operations Management** [XUSITEMGR] menu:

Figure 7: Lock Manager Menu [XULM LOCK MANAGER MENU]



```
Select Systems Manager Menu Option: OPER <Enter> ations Management


          System Status

          Introductory text edit

          CPU/Service/User/Device Stats

   LOCK   Lock Manager Menu ...

   RJD    Kill off a users' job

          Alert Management ...

          Alpha/Beta Test Option Usage Menu ...

          Clean old Job Nodes in XUTL

          Delete Old (>14 d) Alerts

          Foundations Management

          Kernel Management Menu ...

          Post sign-in Text Edit

          User Management Menu ...


Select Operations Management Option: LOCK <Enter> Lock Manager Menu


   LM     Kernel Lock Manager

   EDIT   Edit Lock Dictionary

   LOG    View Lock Manager Log

   SITE   Edit Lock Manager Parameters

   PURG   Purge Lock Manager Log


Select Lock Manager Menu Option:
```

The **Lock Manager Menu** [XULM LOCK MANAGER MENU] includes the options listed in Table 2:

Table 2: Lock Manager—Options

Option Name	Option Menu Text	Description
XULM LOCK MANAGER	**Kernel Lock Manager**	Use this option to display the Lock Table and terminate processes that hold problem locks. This option is locked with the XULM LOCKS security key.
XULM EDIT LOCK DICTIONARY	**Edit Lock Dictionary**	User this option to add entries to the Lock Dictionary or edit existing entries.
XULM VIEW LOCK MANAGER LOG	**View Lock Manager Log**	Use this option to view the Kernel Lock Manager Log.
XULM EDIT PARAMETERS	**Edit Lock Manager Parameters**	Use this option to edit the site parameters for the Kernel Lock Manager.
XULM PURGE LOCK MANAGER LOG	**Purge Lock Manager Log**	Use this option to purge the Lock Manger Log of old entries.

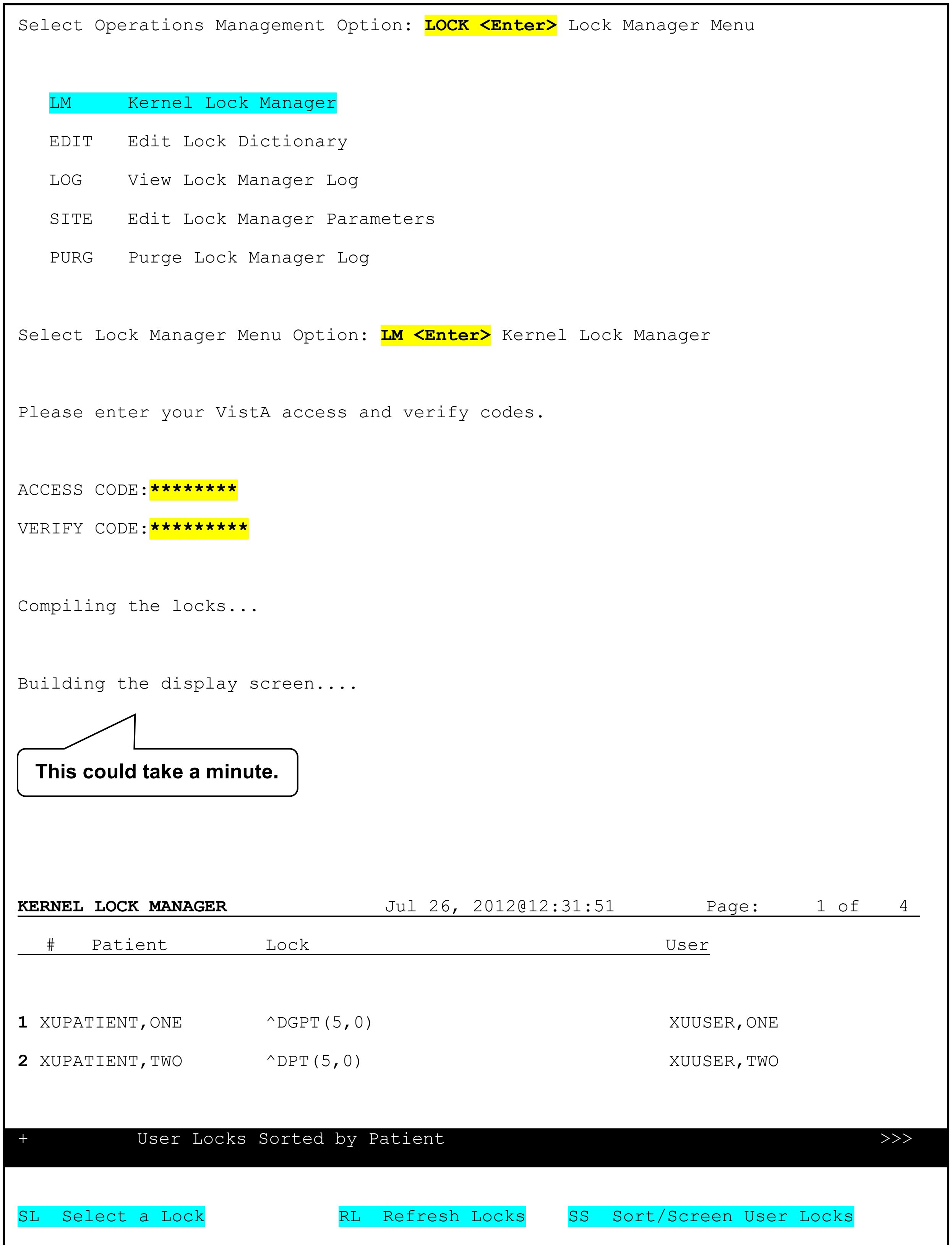
1.4 Using the Lock Manager

1.4.1 List Locks Screen

Use the **Kernel Lock Manager** [XULM LOCK MANAGER] option to view the lock table and the processes that own the locks. This option is locked with the XULM LOCKS security key.

Upon entering the option, you may be asked to enter your Access and Verify code. The Lock Manager uses these codes to query each node for information regarding locks and processes, via the RPC Data Broker. However, if the system consists of the single node on which you are already logged onto, you are *not* asked to enter your Access and Verify code.

Figure 8: Using Kernel Lock Manager Option [XULM LOCK MANAGER]—Sample User Entries and Report



```
Select Operations Management Option: LOCK <Enter> Lock Manager Menu


   LM      Kernel Lock Manager
   EDIT    Edit Lock Dictionary
   LOG     View Lock Manager Log
   SITE    Edit Lock Manager Parameters
   PURG    Purge Lock Manager Log


Select Lock Manager Menu Option: LM <Enter> Kernel Lock Manager


Please enter your VistA access and verify codes.


ACCESS CODE:********
VERIFY CODE:*********


Compiling the locks...


Building the display screen....
```

This could take a minute.

```
KERNEL LOCK MANAGER                 Jul 26, 2012@12:31:51        Page:      1 of    4
   #    Patient          Lock                                   User


1 XUPATIENT,ONE        ^DGPT(5,0)                               XUUSER,ONE
2 XUPATIENT,TWO        ^DPT(5,0)                                XUUSER,TWO

+          User Locks Sorted by Patient                                           >>>

SL   Select a Lock            RL   Refresh Locks      SS   Sort/Screen User Locks
```

```
GO   Go To a List Entry         SYS  System Locks      SN   Select Node

Select Action: Next Screen//
```

The main “User Locks” screen contains only user locks, as opposed to system locks. System locks are those locks used by infrastructure applications, such as the Kernel and HL7 packages, and are generally not of interest to users of the Lock Manager. In order to see the system locks, you can use the **SYS—System Locks** action.

Table 3 lists the actions available on the “List Locks” screen.

Table 3: Lock Manager—Actions

Lock Action	Description
SL—Select a Lock	This action allows a user to select a lock from the list. It then displays a new screen with detailed information about the lock.
GO—Go To a List Entry	This List Manager action asks the user where he/she wants to go to on the list and then shifts the display to that location.
RL—Refresh Locks	This action rebuilds the list of locks by reading the lock table.
SYS—System Locks	This action displays the list of the system locks. System locks are generally ignored within the Lock Manager. They are locks held by infrastructure packages, such as the Kernel or the HL7 package. **NOTE:** Only holders of the XULM SYSTEM LOCKS security key can use this option.
SS—Sort/Screen User Locks	This action provides the user with several options for how the list locks should be displayed. The options include sorting the list by the following: • Patient Name • User Name • Lock string, or screening the entries by lock reference, which means that only locks that relate to a specific file are included in the display.
SN—Select Node	This action allows the user to select either a single computer node or all computer nodes. If the user selects a single node, then the display of locks includes only locks placed by processes running on that node.

1.4.2 Single Lock Details Screen

Use the **SL—Select a Lock** action to view the lock details (Figure 9). The detailed information includes the following information:

- Node Information
- Lock ID
- Process ID (decimal and Hex)—Process that owns the lock.
- User Name
- Task Information
- Lock Usage
- File References—Files that the lock references
- Other locks held by process

Figure 9: Select a Lock Action—Sample Detailed Lock Information

```
DETAILED LOCK INFORMATION    Jul 27, 2012@10:30:47         Page:    1 of    2

Node: AABC999

Lock:  ^DGBT(392,3120311.080346,0)

Full Reference: ^[^"^_$1$DGA4:[ XXX.YYY]"]DGBT(392,3120311.080346,0)

Process ID (decimal): 542188409

Process ID (hex): 20512379

User Name: XUUSER,ONE                                 DUZ: 53

Task Information:

    Task#: 3808610

    Started: Jul 27, 2012@10:26:29

    Option:

    Description: No Description (%ZTLOAD)

Lock Usage:

This lock is on a record in the BENEFICIARY TRAVEL CLAIM file (#392).

File References:

   PATIENT FILE RECORD:

      Patient Name:  XUPATIENT,ONE

      Sex:  FEMALE

      DOB:  Mar 03, 1955

      SSN:  000567987

   BENEFICIARY TRAVEL CLAIM FILE RECORD:

     Claim Dt/Tm:  Mar 11, 2012@08:03:46

      Account#:  111 CAR,TRAINS, AND PLACES

      Patient Name:  XUPATIENT,ONE

      Sex:  FEMALE

      DOB:  Mar 03, 1955

      SSN:  000567987


Other locks held by process:

       ^%ZTSCH("TASK",3808610)


       ^DPT(27,0)
```

```
+          Enter ?? for more actions                                    >>>
KILL  Terminate this Process

Select Action: Next Screen//
```

1.4.2.1 Terminate this Process Action

Use the **KILL—Terminate this Process** action to terminate the process, thereby releasing all the locks held by it.

CAUTION: This action is irreversible! Before terminating a process, examine all the information provided on the screen. Do *not* terminate the process unless you are sure the user is no longer active.

Do *not* terminate a system process unless you have the expertise to ascertain the effect. Incorrectly terminating a system process could have adverse effects on multiple users or applications.

When a process is terminated, an entry is made in the XULM LOCK MANAGER LOG (#8993.2) file. It consists of the following data:

- User's Name
- Date/Time of Action
- Detailed Lock Information

1.5 Managing the Lock Manager

Table 4 reviews the various management functions available within the Lock Manager and the corresponding option where the function can be performed.

Table 4: Lock Manager—Management Functions

Function	Option
Enable/Disable the Lock Manager	**Edit Lock Manager Parameters** [XULM EDIT PARAMETERS]
Edit IP address and port numbers of RPC Data Broker on the system nodes.	**Edit Lock Manager Parameters** [XULM EDIT PARAMETERS]
Edit the list of system locks. System locks are generally excluded from view within the Lock Manager, which makes it easier for users to review the lock table.	**Edit Lock Manager Parameters** [XULM EDIT PARAMETERS]
View the Lock Manager: use log that records	**View Lock Manager Log**

Function	Option
each instance of a process being terminated.	[XULM VIEW LOCK MANAGER LOG]
Purge the Lock Manager use log.	**Purge Lock Manager Log** [XULM PURGE LOCK MANAGER LOG]
Add or Edit entries in the Lock Dictionary.	**Edit Lock Dictionary** [XULM EDIT LOCK DICTIONARY]

1.6 Maintaining the Lock Dictionary

1.6.1 Adding Lock Templates—Edit Lock Dictionary Option

Use the **Edit Lock Dictionary** [XULM EDIT LOCK DICTIONARY] option to add to or edit entries in the XULM LOCK DICTIONARY (#8993) file.

A "Lock Template" is a description of the lock. It looks like the entry in the lock table, except that it can contain a variable in place of a subscript. A variable is used when the actual subscript value is *not* known in advance. Usually, it represents the internal entry number (IEN) of the record that is being locked. Variables are important, because they can be used in M code (see Figure 10).

To add an entry to the XULM LOCK DICTIONARY (#8993) file, perform the following procedure:

1. From the **Lock Manager Menu** [XULM LOCK MANAGER MENU] at the "Select Lock Manager Menu Option:" prompt, select the **Edit Lock Dictionary** [XULM EDIT LOCK DICTIONARY] option.
2. At the "Enter response: E//" prompt, enter one of the following values related to entries in the lock dictionary:

   - **E**—Edit an existing entry.
   - **D**—Delete an existing entry.
   - **A**—Add a new entry.

   In this example, the user is adding a new entry; so, she selected **A—Add a new entry**.
3. At the "LOCK TEMPLATE:" prompt, enter a lock template based on the following rules:

   - Locks are almost always on a global; though, it is allowable to lock a local variable. For the case of a global lock, enter a space as the first character, since VA FileMan does *not* allow a caret (**^**) as the first character (e.g., **^DGCR(399,IEN**; this sample includes a leading space before the **^**).
   - Subscripts that are *not* variables should include quotes unless they are numbers.
   - Variables should start with a letter and should not be quoted.
4. At the "GLOBAL LOCK?: YES//" prompt, press **Enter** to accept the **YES** default. Locks are usually on globals, but it is possible to lock a local variable too.

5. At the “XULM LOCK DICTIONARY GLOBAL LOCK?: YES//” prompt, press **Enter** to accept the **YES** default.

6. At the “XULM LOCK DICTIONARY PACKAGE:” prompt, enter the package that is responsible for the lock (e.g., Integrated Billing [sample]).

7. At the “PARTIAL MATCH ALLOWED?:” prompt, enter **YES**. This means that a lock table entry with additional subscripts is still considered as matching the Lock template. For example, by answering **YES** to this prompt the lock on **^DGCR(399,1,0)** would be considered a match; otherwise, the additional subscript **0** would rule it out as a match.

8. At the “Edit? NO//” prompt, enter a description for the purpose of the Lock template.

9. (Optional) At the “Executable check logic for variable IEN (optional):” prompt, enter M code to verify that the variable IEN has a permissible value. It should set **Y=0** if *not* OK, and **Y=1** if OK. For example:

   ```
   S Y=$S($D(^DGCR(399,IEN,0)):1,1:0)
   ```

   In this example, you can check that the record actually exists. If the check fails, then the Lock template is ruled *not* to match the lock. The M code should set **Y=1** if the value is acceptable, or **0** if the value is *not* acceptable. Setting **Y=0** means that the lock table entry is considered *not* to match the Lock template.

10. (Optional) At the “Select FILE:” prompt, you can enter a file that is related to the lock (e.g., PATIENT [#2] file) in some way. Either the lock is on a record in the file, or a record in the file can be navigated to based on the information contained within the lock.

    If you enter a file, then you can enter M code that returns identifiers from a record in that file that can help users identify the problem lock. If there are identifiers that you would like to display to the user, first select the file, and then enter the M code that retrieves the identifiers from the file.

    Users of the Lock Manager search for the problem lock by the file or files that it is related to. Entering “COMPUTABLE FILE REFERENCES” is what makes this possible. Most locks of interest are related in some way to a particular patient, so entries in the Lock Dictionary should almost always contain a computable file reference to the PATIENT (#2) file, but other computable file references should also be included when appropriate.

11. At the “Are you adding ‘*XXXXXXXX*’ as a new COMPUTABLE FILE REFERENCES (the *nXX* for this XULM LOCK DICTIONARY)? No//” prompt, enter **YES**.

12. At the “COMPUTABLE FILE REFERENCES FILE: *XXXXXXXX*//” prompt, press **Enter** to accept the default response.

13. At the “Enter MUMPS code that returns identifiers for the file:” prompt, enter M code that returns identifiers for the file references. In order to return identifiers for the PATIENT (#2) file, the application should call the PAT^XULMU API. It takes the patient DFN as the input. For example:

    ```
    D PAT^XULMU($P($G(^DGCR(399,IEN,0)),"^",2))
    ```

14. At the “Edit? NO//” prompt, enter **YES** and then enter a description to list the identifiers that are returned for this file reference (e.g., Name, Sex, Date of Birth [DOB], and Social Security Number [SSN]).

15. At the “Select FILE:” prompt, enter another computable file identifier (e.g., BILL/CLAIMS [#399] file).

16. At the “Are you adding ‘*XXXXXXXX*’ as a new COMPUTABLE FILE REFERENCES (the *nXX* for this XULM LOCK DICTIONARY)? No//” prompt, enter **YES**.

17. At the “COMPUTABLE FILE REFERENCES FILE: //” prompt, press **Enter**.

18. At the “Enter MUMPS code that returns identifiers for the file. MUMPS CODE:” prompt, enter M code that returns identifiers for the file references. This file returns identifiers from the PATIENT (#2) file as well as the bill number. In order to obtain the patient identifiers when the referenced file is *not* the PATIENT (#2) file use the ADDPAT^XULMU API. The input parameter is the patient DFN. For example:

    ```
    N ND S ND=$G(^DGCR(399,IEN,0)),ID(“IEN″)=IEN D ADDPAT^XULMU(+$P(ND,″^″,2)) S
    ID(0)=ID(0)+1,ID(ID(0))=″BILL NUMBER:″_$P(ND,″^″)
    ```

19. At the “Edit? NO//” prompt, enter **YES** and then enter a description to list the identifiers that are returned for this file reference (e.g., Name, Sex, Date of Birth [DOB], Social Security Number [SSN], and Bill Number).

Figure 10: Adding New Entry to XULM LOCK DICTIONARY (#8993) File—Sample ^DGCR(399,IEN) Template

```
Select Operations Management Option: LOCK MANAGER MENU


     LM     Kernel Lock Manager

     EDIT   Edit Lock Dictionary

     LOG    View Lock Manager Log

     SITE   Edit Lock Manager Parameters

     PURG   Purge Lock Manager Log


Select Lock Manager Menu Option: EDIT LOCK DICTIONARY




Do you want to edit an existing entry in the lock dictionary or add a new one?


     Select one of the following:


     E      Edit an entry

     D      Delete an entry

     A      Add a new entry


Enter response: E// ADD A NEW ENTRY


* You cannot enter the ‘^’ prefix when selecting a lock template. **

LOCK TEMPLATE: ^DGCR(399,IEN)

LOCK TEMPLATE: _^DGCR(399,IEN)
```

VA FileMan does *not* allow ^ as the first character! Re-enter the value with a leading space.

```
LOCK TEMPLATE: ^DGCR(399,IEN)// <Enter>

GLOBAL LOCK?: YES// <Enter>
```

```
     XULM LOCK DICTIONARY GLOBAL LOCK?: YES// <Enter> YES

     XULM LOCK DICTIONARY PACKAGE: INTEGRATED BILLING

PARTIAL MATCH ALLOWED?: YES

What is the purpose of this lock?:
  No existing text
  Edit? NO// YES

This lock is on a record in the BILL/CLAIMS file (#399).

You can optionally enter MUMPS code to verify that the variable IEN
has a permissible value. It should set Y=0 if not ok, Y=1 if ok.

Executable check logic for variable IEN (optional): S
Y=$S($D(^DGCR(399,IEN,0)):1,1:0)

You can display file identifiers for the locked record, or for a record in
another file related to the locked record.  Most locks are related to a
specific patient, so most entries in the lock dictionary should include a
file reference to the PATIENT file (#2) and to the file of the locked record,
and perhaps other files as well.

If you would like to include file references, first select the file, and then
enter the MUMPS code that will retrieve the file identifiers from that file.

Select FILE: 2 <Enter> PATIENT
  Are you adding 'PATIENT' as
     a new COMPUTABLE FILE REFERENCES (the 1ST for this XULM LOCK DICTIONARY)? No
// YES
     COMPUTABLE FILE REFERENCES FILE: PATIENT// <Enter>
     Enter MUMPS code that returns identifiers for the file:

D PAT^XULMU($P($G(^DGCR(399,IEN,0)),″^″,2))

     List the identifiers that are returned for this file reference.

     Identifiers:
     No existing text
     Edit? NO// YES

Returns the patient’s name, sex, date of birth, and Social Security Number.

Select FILE: 399 <Enter> BILL/CLAIMS

     Are you adding ‘BILL/CLAIMS ‘ as a new COMPUTABLE FILE REFERENCES (the 2ND for
this XULM LOCK DICTIONARY)? No// YES
     COMPUTABLE FILE REFERENCES FILE: // <Enter>

Enter MUMPS code that returns identifiers for the file.

     MUMPS CODE: N ND S ND=$G(^DGCR(399,IEN,0)),ID(“IEN”)=IEN D
ADDPAT^XULMU(+$P(ND,″^″,2)) S ID(0)=ID(0)+1,ID(ID(0))=″BILL NUMBER:″_$P(ND,″^″)

List the identifiers that are returned for this file reference.

     Identifiers:
     No existing text
     Edit? NO// YES

This file reference returns the patient name, date of birth, sex,
Social Security Number, and BILL NUMBER.
```

1.6.2 Exporting Lock Templates

Entries in the Lock Dictionary can be included in a KIDS distribution. The KIDS enhancement that adds LOCK TEMPLATES as a new component are released in Kernel Patch XU*8.0*672.

1.7 Viewing and Purging Lock Manager Logs

1.7.1 View Lock Manager Log Option

Use the **View Lock Manager Log** [XULM VIEW LOCK MANAGER LOG] option to display the entries for the terminated lock processes in the XULM LOCK MANAGER LOG (#8993.2) file.

To view the Lock Manager log, perform the following procedure:

1. From the **Lock Manager Menu** [XULM LOCK MANAGER MENU], select the **View Lock Manager Log** [XULM VIEW LOCK MANAGER LOG] option.
2. At the “Select XULM LOCK MANAGER LOG DATE/TIME PROCESS TERMINATED:” prompt, enter a specific date/time or two question marks (**??**) to get a list.
3. .At the “DEVICE:” prompt, enter a device to display the log for the specific entry selected.

Figure 11: View Lock Manager Log Option [XULM VIEW LOCK MANAGER LOG]—Sample User Entries and Report

```
Select Lock Manager Menu Option: VIEW <Enter> Lock Manager Log

Kernel Lock Manager Log


Select XULM LOCK MANAGER LOG DATE/TIME PROCESS TERMINATED: ??


   Choose from:

   JUN 18, 2012@17:14:23

   JUN 18, 2012@17:22:32

   JUN 18, 2012@17:33:27

   JUN 19, 2012@09:03:58

   JUN 19, 2012@09:04:43

   JUN 19, 2012@09:45:49

   JUN 19, 2012@11:04:16

   JUN 19, 2012@11:06:47

   JUN 19, 2012@12:33:43

   JUN 19, 2012@12:35:36

   JUN 19, 2012@12:47:21

   JUN 19, 2012@12:48:48

   JUN 19, 2012@12:50:42

   JUN 19, 2012@12:53:16

   JUN 19, 2012@12:55:59

   JUN 20, 2012@06:40:46

   JUN 24, 2012@09:14:55

   JUN 24, 2012@09:21:43

   JUN 24, 2012@09:22:50

^

Select XULM LOCK MANAGER LOG DATE/TIME PROCESS TERMINATED: JUNE 18 <Enter> JUN 18,
2012

     1   6-18-2012@17:14:23

     2   6-18-2012@17:22:32

     3   6-18-2012@17:33:27

CHOOSE 1-3: 1 <Enter> 6-18-2012@17:14:23
```

```
DEVICE: <Enter>  Telnet Terminal    Right Margin: 80// <Enter>

XULM LOCK MANAGER LOG LIST                      AUG 14,2012  16:12    PAGE 1
--------------------------------------------------------------------------------


DATE/TIME PROCESS TERMINATED: JUN 18, 2012@17:14:23
  THE TERMINATOR: XUUSER,ONE
 PROCESS DESCRIPTION:
 Lock:  ^DGBT(1,0)
 Node: AABC999
 Full Reference: ^["^^_$1$DGA4:[NXT.NXT]"]DGBT(1,0)
 Process ID (decimal): 540943078
 Process ID (hex): 203E22E6
 User Name: UNKNOWN                                    DUZ:
 Task Information: not available
 Lock Usage:  not available
 File References: not available

 Other locks held by process:
        ^DGPT(1,0)

        ^DPT(4,0)

<Enter>

XULM LOCK MANAGER LOG LIST                      AUG 14,2012  16:12    PAGE 2
--------------------------------------------------------------------------------


        ^DPT(5,0)

Select XULM LOCK MANAGER LOG DATE/TIME PROCESS TERMINATED:
```

1.7.2 Purge Lock Manager Log Option

Use the **Purge Lock Manager Log** [XULM PURGE LOCK MANAGER LOG] option to purge the Lock Manager log.

To purge the Lock Manager log, perform the following procedure:

1. From the **Lock Manager Menu** [XULM LOCK MANAGER MENU], select the **Purge Lock Manager Log** [XULM PURGE LOCK MANAGER LOG] option.
2. At the "How many days of data should be retained: (0-365): 30//" prompt, enter the number of days to *retain* the log data (e.g., **30** days). Any log data older than the value entered is purged (e.g., **31** or more days). The default is **30** days with a maximum of **1** year (**365** days).
3. When the data purge is complete, the system displays: **DONE!**

Figure 12: Purge Lock Manager Log Option [XULM PURGE LOCK MANAGER LOG]—Sample User Entries and System Responses

```
Select Lock Manager Menu Option: PURG <Enter> Purge Lock Manager Log

How many days of data should be retained:  (0-365): 30// 364

DONE!

Enter RETURN to continue or '^' to exit:
```

2 Developer's Guide Insert—Lock Manager

2.1 Application Programming Interface (API)—Housekeeping

When an application terminates, there may be housekeeping required. A prime example is the need to delete temporary data kept in the **^TMP** and **^XTMP** globals. An application that is terminated by the Lock Manager does *not* have the opportunity to do its own housecleaning, but the Lock Manager can do it for the application if it registers a housecleaning routine via the APIs described below.

2.1.1 CLEANUP^XULMU(): Execute the Housecleaning Stack

Reference Type: Supported

Category: Lock Manager

ICR #: 5832

Description: The CLEANUP^XULMU API executes the housecleaning stack set by the process identified by **DOLLARJ**. Entries are executed in the first-in-first-out (FIFO) order, with the last entry added being the first to be executed, and **last** being the last entry executed. If the **last** parameter is *not* passed in, then the entire stack is executed.

NOTE: This API was released with Kernel Patch XU*8.0*608.

Format: `CLEANUP^XULMU([last])`

Input Parameters: last: (optional) This is the last entry that is executed. If *not* passed in, then the entire housecleaning stack is executed.

Output: returns: None.

2.1.1.1 Examples

2.1.1.1.1 Example 1

An application can execute the entire housecleaning stack with the code shown in Figure 13:

Figure 13: CLEANUP^XULMU API—Example 1

```
DO CLEANUP^XULMU
```

2.1.1.1.2 Example 2

If an application is called by another application, then the first application may have already placed entries of its own on the stack. So, the **last** parameter needs to be passed, with **last** being the first entry placed on the stack. It is the last entry executed, since that stack is executed in FIFO order.

Figure 14: CLEANUP^XULMU API—Example 2

```
DO CLEANUP^XULMU(last)
```

2.1.2 SETCLEAN^XULMU(): Register a Cleanup Routine

Reference Type: Supported

Category: Lock Manager

ICR #: 5832

Description: The SETCLEAN^XULMU API registers a cleanup routine that should be executed when the process is terminated by the Kernel Lock Manager. An entry is created on a stack kept for the process. The location is:

^XTMP("XULM CLEANUP_"_$J)

Where **$J** uniquely identifies the process.

A process can call SETCLEAN^XULMU repeatedly, and each time a new entry is placed on the stack.

CAUTION: Once an application calls SETCLEAN, upon exiting it *must* either execute its housecleaning stack or delete it using the APIs CLEAN or UNCLEAN.

NOTE: This API was released with Kernel Patch XU*8.0*608.

Format: `SETCLEAN^XULMU(rtn,.var)`

Input Parameters: rtn: (required) The routine to be executed when the process is terminated.

.var: (required) An input array containing a list of variables that should be defined when the routine is executed. It is up to the application to ensure that all the required variables are defined when CLEANUP^XULMU is called.

Output: returns: Returns: An integer that identifies the entry created on the stack. The application needs to retain the value in order to either execute the entry on the housecleaning stack or to remove it.

2.1.2.1 Example

Suppose the application has a cleanup routine CLEANUP^XXAPP, and it needs to be executed with DFN defined with its present valued. The application would use this API as shown in Figure 15:

Figure 15: SETCLEAN^XULMU API—Example

```
N VAR,CLEANUP

S VAR(“DFN”)=DFN

S CLEANUP=$$SETCLEAN^XULMU(“CLEANUP^XXAPP”,.VAR)
```

The application’s housekeeping stack would look like Figure 16:

Figure 16: SETCLEAN^XULMU API—Sample Stack

```
^XTMP(“XULM CLEANUP”,$J,1,“ROUTINE”)=“CLEANUP^XXAPP”

^XTMP(“XULM CLEANUP”,$J,1,“VARIABLES”,”DFN”)=1000061
```

2.1.3 UNCLEAN^XULMU(): Remove Entries from the Housecleaning Stack

Reference Type: Supported

Category: Lock Manager

ICR #: 5832

Description: The UNCLEAN^XULMU API removes entries from the housecleaning stack set by calling the SETCLEAN^XULMU(): Register a Cleanup Routine API. Entries are removed in First-In-First-Out (FIFO) order. If the **last** parameter is *not* passed in, then the entire stack is deleted; otherwise, just the entries back to **last** are removed.

NOTE: This API was released with Kernel Patch XU*8.0*608.

Format: `UNCLEAN^XULMU([last])`

Input Parameters: last: (optional) Identifies the last entry on the housekeeping stack to remove. Entries are removed in FIFO order. Therefore, the first entry removed is the last entry that was added, and the last entry removed is **last**. If *not* passed in, the entire housecleaning stack is deleted.

Output: returns: None.

2.1.3.1 Examples

2.1.3.1.1 Example 1

The example in Figure 17 would remove the entire housecleaning stack:

Figure 17: UNCLEAN^XULMU API—Example 1

```
DO UNCLEAN^XULMU
```

2.1.3.1.2 Example 2

If an application is called by another application, then the first application may have already placed entries of its own on the stack. So, the parameter **last** needs to be passed, with **last** being the first entry placed on the stack. It is the last entry deleted, since that stack is executed in first-in-first-out (FIFO) order.

Figure 18: UNCLEAN^XULMU API—Example 2

```
DO UNCLEAN^XULMU(last)
```

2.2 Application Programming Interface (API)—Lock Dictionary

2.2.1 ADDPAT^XULMU(): Add Patient Identifiers for a Computable File Reference

Reference Type: Supported

Category: Lock Manager

ICR #: 5832

Description: The ADDPAT^XULMU API is very similar to the PAT^XULMU(): Get a Standard Set of Patient Identifiers API, except that it is used to *add* the patient identifiers for a computable file reference for a file that is *not* the PATIENT (#2) file. The computable file references can include additional identifiers. For example, a computable file reference for a billing file can contain the bill number as an identifier as well as the patient identifiers returned by the ADDPAT^XULMU API.

NOTE: This API was released with Kernel Patch XU*8.0*608.

Format: `ADDPAT^XULMU(dfn)`

Input Parameters: dfn: (required) The IEN of a record in the PATIENT (#2) file.

Output: returns: Returns: **ID(0)**: If *not* defined at the point the ADDPAT^XULMU API is called, it is initially set to **zero** (**0**). When the ADDPAT^XULMU API returns, the **ID(0)** is incremented by **4**.

```
ID(ID(0)+1)=<patient name>
ID(ID(0)+2)=<patient sex>
ID(ID(0)+3)=<patient date of birth>
ID(ID(0)+4)=<patient Social Security Number>
```

2.2.2 PAT^XULMU(): Get a Standard Set of Patient Identifiers

Reference Type: Supported

Category: Lock Manager

ICR #: 5832

Description: The PAT^XULMU API is for use within the M code for a computable file reference to the PATIENT (#2) file. It returns a standard set of patient identifiers.

NOTE: This API was released with Kernel Patch XU*8.0*608.

Format: `PAT^XULMU(dfn)`

Input Parameters: dfn: (required) The IEN of a record in the PATIENT (#2) file.

Output: returns: Returns the following variables:

- **ID("IEN")**=DFN
- **ID(0)=4**
- **ID(1)**=*<patient name>*
- **ID(2)**=*<patient sex>*
- **ID(3)**=*<patient date of birth>*
- **ID(4)**=*<patient Social Security Number>*

2.2.2.1 Example

Assuming that **DFN** is a variable defined within the Lock template, then the M code for a computable file reference to the PATIENT (#2) file is shown in Figure 19:

Figure 19: PAT^XULMU API—Example

```
DO PAT^XULMU(DFN)
```