

Chapter Com

Common Operations



Com.0 Introduction

M3UI programs support common operations that are available in many tools. These common operations deal with activities that have general usefulness, rather than applying to the work of a specific tool. Such activities include manipulating objects; reading, writing, and managing files; printing; and exiting a tool.

Because each tool uses different terminology for the kinds of entities it creates, the names of some common operations vary between tools. However, the effect of each operation is the same in all tools. For example, the **Read Model** operation reads network models in the Network Editor, probe files in the Probe Editor, and data sets in **m3_edmap**, but the effect is the same—to read a saved item into the current tool. For brevity, this chapter uses the following general terms:

- *model* represents any entity that can be written to or read from disk, such as network models, analysis configurations, packet formats, and simulation sequences.
- *object* represents any graphical component of a model that can be manipulated in a tool, such as nodes and links.

Com.1 Common Operations Action Buttons

Quick Guide to Common Operations		
Button	Name	Description
	Copy Selected Objects	Copies selected objects to the clipboard
	Cut Selected Objects	Copies selected objects to the clipboard, then deletes the original objects
	Paste Clipboard Objects	Copies objects on the clipboard to a specified location in the tool window
	Clear Model	Clears the current model from the tool window
	Delete Models	Deletes saved models from disk storage

Quick Guide to Common Operations (Continued)		
Button	Name	Description
	Write Model	Saves a model to disk
	Read Model	Loads a model from disk into the current tool
	Set/Reset User Lock	Toggles the user lock on a model file
	Print Graphics	Saves selected graphic images on an "in-memory" printer page for later printing via the Print Page system operation
	Print Report	Generates and prints a report describing all or selected objects in the current tool
	Edit/View Comment Log	Displays and permits editing of the comment log for a model
	Produce Ema Code	Generates External Model Access (EMA) code for the model currently in the tool
	Exit Tool	Quits the currently active tool

The Clipboard

The copy, cut, and paste operations use a *clipboard*, or hidden buffer, to temporarily store selected objects. The clipboard belongs to the tool environment rather than to a particular tool window. Thus, you can use the clipboard to copy objects within a tool window or from one tool window to another of the same type. Objects in the clipboard are lost when the program is exited, or when new objects are copied into it.

Com.1.0 Copy Selected Objects

This operation copies objects in a tool window and places them on the clipboard. The original objects are not affected.

To copy objects...

- 1) Select one or more objects to be copied by left-clicking on them. Selection markers appear at the corners of the objects.
- 2) Activate the **Copy Selected Objects** action button.
 - ➔ The objects are copied to the clipboard.

Com.1.1 Cut Selected Objects

This operation deletes objects from a tool window and places them on the clipboard. Any dependent objects (such as links) associated with the cut objects are also deleted; any open dialog boxes associated with the cut objects are closed.

To cut objects...

- 1) Select one or more objects to be cut by left-clicking on them. Selection markers appear at the corners of the objects.
- 2) Activate the **Cut Selected Objects** action button.
 - ➔ The objects are copied to the clipboard and deleted from the tool window.

Com.1.2 Paste Clipboard Objects

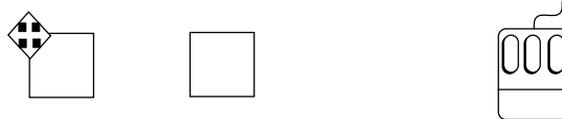
This operation allows you to place a copy of objects on the clipboard at a specified location within a tool window. The objects on the clipboard are unaffected and can be pasted repeatedly at different locations or in different tool windows. This operation does

not replace links that were connected to the clipboard objects unless they were selected as a group (for example, as an entire network model).

✓ To paste objects...

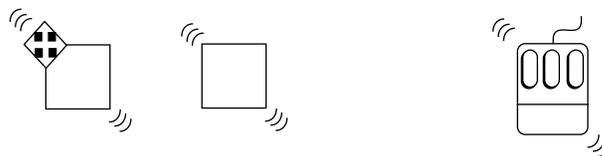
1) Activate the **Paste Clipboard Objects** action button.

➔ The cursor changes to four arrows and outlines of each object on the clipboard appear in the tool window.



2) Move the mouse until the object outlines reach the desired location.

➔ The cursor and outlines follow the mouse's movement.

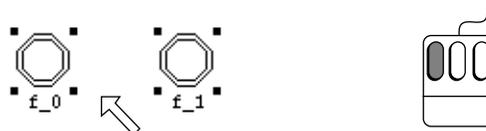


3) Click the left mouse switch.

➔ If the name of an object being pasted is identical to that of an object already in the tool window, a dialog box appears, prompting you to enter a suffix that will be appended to the pasted object's name to maintain uniqueness.

a) Select **OK** to accept the default naming suffix, or type a new suffix for the name of the object being pasted.

➔ The objects are pasted in the specified location and the cursor resumes its usual appearance. If necessary, a suffix is appended to the name of each pasted object.



Com.1.3 Clear Model

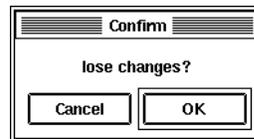


This operation removes the currently-displayed model from the tool, restoring the tool to the state that it has when it is first opened. It does not delete the model from disk.

✓ To clear a model from the tool window...

1) Activate the **clear Model** action button.

➔ If there have been no changes to the currently active model since it was last saved, the model is erased from the tool window. If there have been changes, a **Confirm** dialog box pops up.



a) Click **OK** in the dialog box or press **<Return>**.

➔ The model is erased from the tool window.

Com.1.4 Delete Models



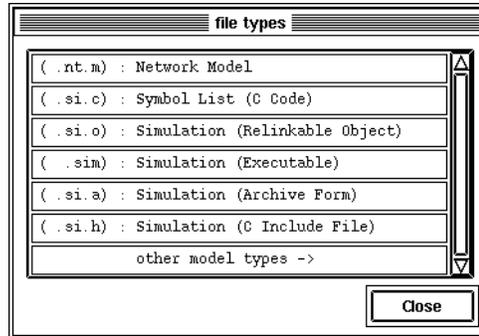
This operation erases files from disk storage. This includes all types of files created by the current tool or by the program.

CAUTION: There is no way to recover a file once you have deleted it.

✓ To delete a model from disk...

1) Activate the **Delete Models** action button.

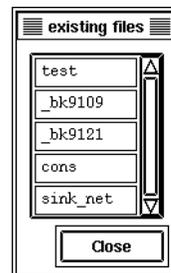
➔ A dialog box appears enumerating the common file types related to the current tool.



2) Make one of the following choices:

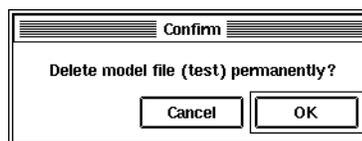
- Select a file type to delete models of that type.
- Select **other model types ->** to see a list of all file types, then select a file type from the expanded list.

➔ A dialog box appears with a list of files of the selected type.

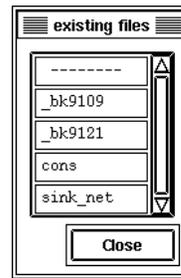


3) Select a file to be deleted.

➔ A **Confirm** dialog box appears.



- 4) Click **OK** or press **<Return>** to delete the file.
 - ➔ The file is deleted and the file list reappears with an indication of the deleted file.



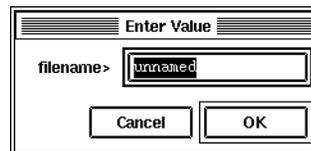
- 5) Repeat steps 3 and 4 for each file to be deleted, then select **Close**.

Com.1.5 Write Model

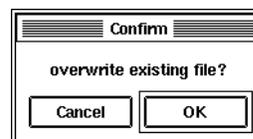
This operation saves the model currently in the tool window by writing it to disk. Model file names are automatically given a suffix to identify their type.

To write a model to disk...

- 1) Activate the **write Model** action button.
 - ➔ A dialog box pops up for entering the model name.



- 2) Type a name for the model and select **OK**, or just select **OK** to use the default name highlighted in the text entry area.
 - a) If the name is already in use, a **confirm** dialog box pops up. To overwrite the existing file, click **OK**. To avoid overwriting the existing file, click on **cancel** and repeat step 2 with a different name.



➔ The program saves the file.

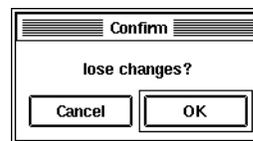
Com.1.6 Read Model

This operation reads models from disk and displays them in the tool window. The new model replaces any model already in the window.

 **To read a model from disk...**

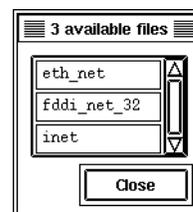
- 1) Activate the **Read Model** action button.

➔ If a model with unsaved changes is in the tool window, a **Confirm** dialog box appears.



- 2) Left-click **OK** or press **<Return>** to continue the operation.

➔ A list of available models appears.



- 3) Select a model from the list.

➔ The program deletes any model in the tool window and loads the new model.

Troubleshooting

If an error occurs when reading in a file, the tool issues a warning beep and the message display shows an error message. If an error occurs while writing the model file, the tool may be unable to read it in later. In such cases, the file header information may have become corrupted. Header information can be viewed and modified using the **m3_manfile** program. For more information on using **m3_manfile**, refer to section *Prg.11* in the *OPNET External Interfaces Manual*.

Com.1.7 Set/Reset User Lock

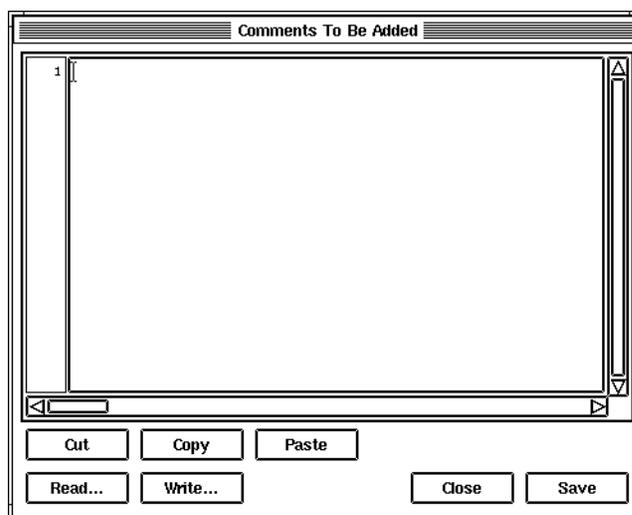
This operation allows you to restrict file access so that models cannot be easily overwritten by others. When the user lock is set, only the user who set it can overwrite the file; all other users will have read-only access. When the lock is reset, the file is unlocked and can be overwritten by other users. This operation toggles the state of the user lock between set and unset.

Files remain locked or unlocked even after an **opnet** session has terminated. Locked files can be overwritten by the user account that set them and therefore do not provide complete security. User locks function only as warning systems and are recommended, but not required.

If you read in a model for which the user lock has been set (by another user), the message display indicates that the file is read-only. You will not be able to overwrite the file, but can write it to disk under a different file name.

✓ To toggle the user lock...

- 1) If the model has not been saved, write it to disk. (User locks apply only to files on disk; if no version of the model exists on disk, an error message appears.)
- 2) Activate the **Set/Reset User Lock** action button.
 - ➔ If unlocked, the file for the current model is locked. The operation is confirmed in the message display. If already locked, the file for the current model is unlocked and a comments editing pad appears.



- 3) If desired, add comments to the file's comment log. Refer to *Chapter Dbox* in the *OPNET MIL 3 User Interface Manual* for instructions on entering and saving text.

User locks can also be set and reset from the command line using the `m3_manfile` program. See the *Prg.10* section of the *OPNET External Interfaces Manual*.

Com.1.8 Print Graphics



This operation captures a selected portion of a model's graphic image and saves it in a PostScript file for later printing. For a detailed description of this operation and instructions on using it, see section *Pub.1.1 Print Graphics* in the *OPNET MIL 3 User Interface Manual*.

Com.1.9 Print Report



This operation creates a report which fully describes the contents of the current model and saves it in a PostScript file. This operation will also spool the file to a networked PostScript printer if the program has been configured to do so. For a detailed description of this operation and instructions on using it, see section *Pub.2 Reports* in the *OPNET MIL 3 User Interface Manual*. For information on configuring print spooling, see section *Prog.1* in the *OPNET External Interfaces Manual*.

Com.1.10 Edit/View Comment Log



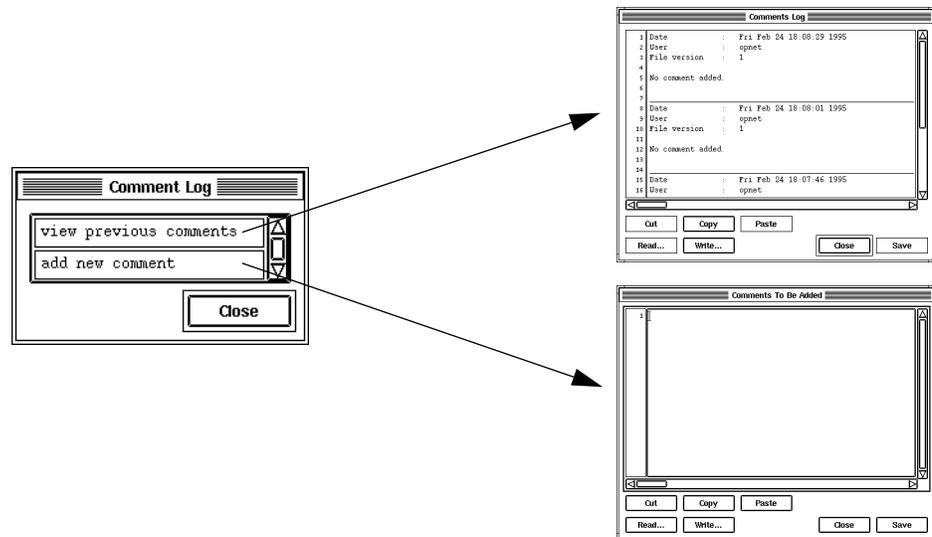
This operation allows you to maintain free-form, descriptive information concerning the evolution of a model. The comment log stores each comment along with the date and time it was entered, the model version, and the name of the user who added it. This enables users to notify each other of recent changes and to archive historical information.

Comments can be added or viewed, but not modified. When adding comments, you enter information into a text edit pad. When viewing comments, you see a view-only text edit pad listing all comments that have been previously added to the file, starting with the most recent modification. While using the comment log, you cannot change or delete previously added comments; to do so, use the `m3_manfile` program described in section *Prg.11* of the *OPNET External Interfaces Manual*.

M3UI programs automatically prompt for comments when the user lock is reset, as described in section *Com.1.7 Set/Reset User Lock*.

✓ **To view or edit a comment log...**

- 1) Make sure the model whose comments you want to access is in the active tool window.
- 2) Activate the **Edit/View Comment Log** action button.
 - ➔ The **Comment Log** dialog box pops up.
- 3) Choose the desired operation:
 - To see the existing comments, select **view previous comments** from the dialog box.
 - To add a new comment, select **add new comment**.
 - ➔ An edit pad for the selected operation pops up.



- 4) Read or add comments, then close the pad. For instructions on using edit pads, refer to *Chapter Dbox* of the *OPNET MIL 3 User Interface Manual*.

Com.1.11 Produce Ema Code



This operation allows you to generate External Model Access (**EMA**) source code that represents the model currently present in the active tool. **Ema** code is in text format,

allowing you to edit it with an ordinary text editor. For more information about **Ema**, see the *External Model Access* chapter of the *OPNET External Interfaces Manual*.

CAUTION: **Ema** source code files for all models have the same suffix, “.em.c”, regardless of model type. Be sure that the **Ema** file you create will not overwrite a previously-created file you want to keep.

✓ **To produce Ema code...**

- 1) Make sure the model for which you want to produce **Ema** code is active. If not, use the **Read Model** operation to display the model in the tool window.
- 2) Activate the **Produce Ema Code** action button.
 - ➔ The program generates the **Ema** code and stores it in the file <model name>.em.c.

Com.1.12 Exit Tool



This operation allows you to exit from a tool window.

CAUTION: Do not confuse the **Exit Tool** action button with the larger **Exit Program** button in the tool button panel. The **Exit Program** button causes the program to quit. Refer to *Chapter Pwin* of the *OPNET MIL 3 User Interface Manual*.

Exit Buttons



Exit Tool button



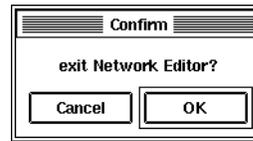
Exit Program button

✓ **To exit a tool window...**

- 1) Make sure the tool window to be closed is active. If not, left-click in the window to make it active.

2) Activate the **Exit Tool** action button.

➔ If the tool window contains unsaved objects, the **Confirm** dialog box appears.



3) If necessary, left-click **OK** to confirm closing the tool.

➔ The program closes the tool window, erases the tool's action buttons, and exits the tool. If other tool windows were open in the background, the one that was most recently open becomes active and its action buttons are displayed.

