

# Nets Users Guide

---

Copyright (C) 1999-2005  
Mike McCauley and Hugh Irvine

Users Guide for the Nets network inventory  
and management system.  
For Nets Revision 2.5

---

## 1.0 Introduction

---

Nets is a graphical source code product providing a flexible and extensible environment for maintaining essential computer and data communications network inventory, configuration and cost information in a platform and database independent manner.

Nets provides a physical model of the network, from which it is simple and straightforward to derive and display any number of textual, logical or graphical views.

This document is the Users Guide for the Nets system and describes the different screens and operations that are used to enter data, search for information and run reports.

Instructions and examples are also provided for operations such as importing and exporting databases, importing and using drawings and icons, using the network browser, reviewing notes associated with any network element stored in the database and perusing the audit trail to verify database operations.

---

## 2.0 Overview

---

Nets is a platform independent and database independent application that enables Managers and Engineers to easily maintain accurate and up-to-date network data.

Nets is delivered with a standard set of database objects together with editing, viewing and reporting tools which provide the framework required to maintain a complete, detailed inventory of any computer network.

Nets includes an autodiscovery tool that can be used to populate the basic Nets database from the output of the **nmap** network scanner.

Nets is based on a set of fundamental network elements (represented as database objects), including the following:

- Locations - Points of Presence (POPs), telco exchanges, co-location facilities, etc.
- Racks - equipment racks used to house devices
- Devices - servers, workstations, routers, switches, patch panels, etc.
- Slots - devices may contain any number of slots
- Interfaces - those device elements that connect to links (ethernet, serial, FDDI, etc.)
- Cards - interfaces may be contained on cards, which may be inserted into slots
- Links - LAN, WAN or cable plant connections
- Protocols - typically TCP/IP, but any protocol is supported (IPX, Appletalk, etc.)
- Addresses - protocol addresses configured on interfaces (such as IP V4)
- Address Maps - graphical representations of IP address blocks and usage
- DNS Names - the DNS names that correspond to interface addresses
- Graphics - maps, drawings, digital photos and schematic diagrams of the network

---

## Overview

---

All of these network elements and the database objects and operations associated with them are described in this manual. The complete set of Nets database objects and their definitions can be found in the Nets Database Schema.

---

### **3.0 Nets Database Objects**

---

The Nets database defines a number of types of Nets objects. All of the objects of a given type are stored in a database table, which contains a number of columns corresponding to the data definitions for a Nets object. The Nets system provides a number of standard tools to create, modify, delete and report on every Nets object in the database.

#### **3.1 Object Naming**

Every Nets database object has a name which is used to identify the object within the Nets system. The object names are simply symbolic strings and can be set to whatever is most relevant to the network that is being managed. Duplicate names are permitted.

It is helpful when entering data into Nets, to follow a standard object naming scheme so that subsequent searches and use of the data in the database are as simple and meaningful as possible (especially in a multi-user environment).

If a standard naming scheme is already in use within the organisation, it should be used (and possibly extended) so that the objects in Nets have similar (or identical) names to the real network objects.

If a naming scheme does not exist within the organisation, it is strongly recommended that a naming scheme derived from the existing network Domain Name System (DNS) definitions be adopted at the outset.

The network objects that are included in the example data set (called “*exampledata.dat*” in the Nets distribution) and that are used in this manual follow the DNS scheme. The domain name in the examples is “*irvine.com.au*”, and objects are typically named as follows:

```
interface.device.irvine.com.au
```

or

```
portnumber.patchpanel.rack.irvine.com.au
```

The example data does not contain any Points of Presence (POP’s), but if it did, the names would look like this:

```
interface.device.pop.irvine.com.au
```

It may also be useful to use the device internal names for interfaces, for example:

```
Serial1/1/3:17.device.pop.irvine.com.au
```

```
FastEthernet0/1.20.device.pop.irvine.com.au
```

Obviously, not all of these names will appear in the DNS zone files, however keeping the Nets names in a similar, related hierarchy will certainly simplify Nets use and operation. Note that some Nets names will map directly to DNS names, but it is important to remember that the name spaces are in fact different and distinct.

### 3.2 Hints for using Objects

The majority of the data in the Nets database will relate to real interfaces, devices, links, etc. However, there are some cases in which the use of “pseudo-objects” can simplify Nets usage. Pseudo-objects are virtual objects used as place-holders in the Nets system.

One example is keeping track of telecommunications links that have been ordered on a POP by POP basis. This will be much easier if the ordered links are connected to interfaces that are part of a pseudo-device at the POP location. This would be accomplished by creating a pseudo-device with interfaces connected to those links:

```
interface.telco-orders.pop.irvine.com.au
```

Once the telecommunications links have been installed, the links can be transferred to the corresponding interfaces of the real devices.

Another example of the use of pseudo-objects is the creation of sms, pager and email gateways used for automatic notification of Nets events. The telephone numbers and/or email addresses of these entities will be used to send messages when required:

```
sms.gw.irvine.com.au
```

```
pager.gw.irvine.com.au
```

```
email.gw.irvine.com.au
```

**Note:** the gateways mentioned above are not included with this release of Nets.

There are many other situations in which this technique can be used.

## 4.0 Starting Nets

---

When Nets has been permanently installed, the main Nets user interface can be run with the following command:

```
netsmain.pl
```

On Windows, `netsmain.pl` can also be run by just double-clicking it in Windows Explorer. A Windows Shortcut to `netsmain.pl` can also be double-clicked.

`Netsmain.pl` understands a range of command line arguments:

```
netsmain.pl [-h] [-d] [-u username] [-p password]
            [-dbsource dbi:xxx:yyy] [-dbusername name] [-dbauth auth]
            [-c rcfile] [-home dir]
            [configname=value] ... [scriptfilename] ...
```

- `-h`  
Prints some command line help to stdout and exits.
- `-d`  
Turns debugging on. Messages are printed to stdout. This is equivalent to setting the Nets `LogLevel` to 4.
- `-u username`  
Attempts to log in to Nets with the given username. If the username/password combination authenticates, the user will be logged in without presenting a login dialog.
- `-p password`  
Attempts to log in to Nets with the given password.
- `-dbsource dbi:xxx:yyy`  
Overrides `DBSource` in `Sql.pm` and the `NETS_DBSOURCE` environment variable. Specifies the database source name that Nets will use.
- `-dbusername name`  
Overrides `DBUsername` in `Sql.pm` and the `NETS_DBUSERNAME` environment variable. Specifies the SQL user name that Nets will use to log in to the database.
- `-dbauth auth`  
Overrides `DBAuth` in `Sql.pm` and the `NETS_DBAUTH` environment variable. Specifies the SQL user password that Nets will use to log in to the database.
- `-c rcfile`  
Reads the named Nets RC file, interpreting lines of the form `name=value` and using them to set Nets configuration variables.
- `-home dir`  
Specifies an alternate directory to override the value for `NetsDir`.
- `configname=value`  
Overrides configuration variable settings in `Site.pm`. Any number of variable settings can be specified.
- `scriptfilename`

Starts the Nets Scripting Language script running. Any number of script file names can be specified.

#### **4.1 Nets Users**

Nets provides a system of user names and logins, allowing various levels and ways of controlling access to Nets and its data.

Nets is delivered with 2 standard Nets users:

- **ADMINISTRATOR**

By default, this user has access to all parts of the Nets system and can change anything. The default password is ADMIN (this should be changed as soon as possible after installation).

- **DEFAULT**

This is the default user name for any user who does not have their own Nets user name. It will be used if there is not a specific Nets user for the logged in user name. The default password is DEFAULT.

When the Nets main program (netsmain.pl) starts up, it logs in to the Nets system using the following logic:

1. On Unix, the user name defaults to the users Unix username. On Windows, it defaults to 'Windows User'.
2. If NETS\_USERNAME and/or NETS\_PASSWORD environment variables are set, they are used.
3. If -u and/or -p arguments are specified on the command line, they will be used as the user name and password.
4. If there is both a user name and a password resulting from the above steps, Nets will attempt to log the user in silently (i.e. without presenting a Login dialog). If the named user (or DEFAULT) exists and the password is correct, the user will be logged in to Nets, else the next step is followed.
5. The Login dialog is presented, allowing the user to enter their Nets username and password.
6. If the entered user name (or DEFAULT) exists and the password is correct, the user will be logged in to Nets.
7. After logging in, the user's details will be added to the CURRENT\_NETSUSERS table in the Nets database for licensing purposes.

As can be seen from the above, Nets will try to log users in with a user name, and will fall back to the DEFAULT user name (if it exists). This means that it is possible to have specific user names and privileges for some users, and a generic login for other users.

## 4.2 Nets Login Screen

When a Nets user starts Nets with no parameters, the following login screen is displayed, into which the username and password must be entered:

---

**FIGURE 1.**

Nets Login Screen



A Nets user must enter a valid username and password to enter the system.

### 4.3 Nets Main Screen

Once a Nets user has successfully logged on, they will be presented with the Nets Main Screen. The Main Screen contains a File Menu, an Edit Menu, a View Menu and a Help Menu. The Main Screen will always be present while Nets is running.

---

**FIGURE 2.**

Nets Main Screen



The Nets username and the hostname of the user's workstation is displayed in the title bar and the message area of the Main Screen.

All of the screens in the Nets system have a similar format, with at a minimum a File Menu, an Edit Menu, a View Menu and a Help Menu.

In addition, the Nets Main Window contains shortcut buttons for the following editors:

- Locations
- Devices
- Interfaces
- Links
- Drawings

All of the Nets object editors are described in Section 7.0 on page 27.

## 5.0 Nets User Interface

---

The Nets user interface has a number of common elements that are used throughout the Nets system. These elements include standard menus and menu items, text and numerical data entry and display areas, object selectors and the like.

**FIGURE 3.** Example Nets Screen



The screen shown above illustrates the main elements of the user interface.

The standard menus are to the left under the title bar, while the help menu is to the right.

The left hand window shows the list of available objects and the scroll bar and scrolling arrows allow access to longer lists. Clicking on any list item will show the details for the corresponding database object in the display areas to the right of the list window.

The data fields are used to enter new data as well as display current data. The data entered into the data entry fields can be composed of strings, numbers, IP addresses, latitudes and longitudes, dates and times, etc. Some data fields have verification routines associated with them so that only data in the correct format can be entered.

### 5.1 Entering Dates and Times

Nets allows dates to be entered in a variety of formats, and there are many fast shorthand ways of entering dates in the past and the future. All times and dates are entered and displayed in the local time zone.

All dates stored and displayed in Nets include both the date and the time. Wherever a date can be entered, the time can also be entered. If a time is not specified, it always defaults to midnight at the beginning of the indicated day. If a time is entered, it must be in the format hh:mm(:ss). The seconds are optional. Times are in 24 hour time format. Some examples are:

- 11:55
- 16:20:30
- 08:08
- 09:5
- 00:00 (i.e. midnight)

Dates are stored and displayed in Nets including the full 4 digit year. When entering dates, shorthand years can be used, such as “99” or “00”. Nets will make an educated guess about which year is meant.

During installation, the Nets administrator will have chosen the appropriate date format for local use. This controls how dates are printed in Nets, and also how exact dates must be entered.

Many of the shorthand date formats allow the use of contractions. For example, instead of “1 week”, meaning 1 week in the future, “1w” is acceptable, and instead of “today”, “tod” is acceptable.

**TABLE 1.**

Permitted Date/Time Formats

General date format	Meaning	Examples
now	The date and time of right now	now
today	The date of today, time defaults to midnight at the beginning of today.	tod today 10:55
tomorrow	The date of tomorrow	tom 9:20
<i>n</i> minutes	The number of minutes in the future.	1mi 3 minutes 5mins
<i>n</i> minutes ago	The number of minutes in the past.	2m ag 3 minutes ago 66mins ago
<i>n</i> hours	The number of hours in the future.	1h 3hours 5h

**TABLE 1.** Permitted Date/Time Formats

General date format	Meaning	Examples
<i>n</i> hours ago	The number of hours in the past.	2hag 3 hours ago 6h ago
<i>n</i> days	The number of days in the future. "1 day" is the same as "tomorrow"	1d 3day 5d 12:00
<i>n</i> days ago	The number of days in the past. "1 day ago" is yesterday	2dag 3 day ago 6d ago
<i>n</i> weeks	The number of weeks in the future. If today is Wednesday, then "2 weeks" would be wednesday 2 weeks from today. "1 week" is the same as "7 days"	2 w 3 weeks 15:00
<i>n</i> weeks ago	The number of weeks in the past. "1 week ago" is the same as "7 days ago"	3wag 6week ago 3w ag 21:30
<i>n</i> months	This day of the month a number of months in the future.	2m 3month
<i>n</i> months ago	This day of the month a number of months in the past.	1mag 4 mon ago
<i>n</i> years	Today's date a number of years in the future. "1 year" is the same as "12 months"	1y 2 year
<i>n</i> years ago	Today's date a number of years in the past. "1 year ago" is the same as "12 months ago"	2yag 3 y ago
<i>dd/mm/yyyy</i>	(Only if the Nets administrator has chosen the local date format as "dd/mm/yyyy") The exact date given.	12/5/99 13:00 1/1/00 5/9/2001 08:00 30/12/1999
<i>mm/dd/yyyy</i>	(Only if the Nets administrator has chosen the local date format as "mm/dd/yyyy") The exact date given.	5/12/99 13:00 1/1/00 9/5/2001 08:00 12/30/1999
<i>yyyymmdd</i>	The exact date given.	20000101 12:00 20011225

**5.1.1 Entering Time Intervals**

Some data fields require an time interval to be entered to specify a period of time, as opposed to an exact date. time intervals can be intered in a variety of ways. Some examples are:

- 10
- 1h10m5s
- 1:10:5

**TABLE 2.** Permitted Time Interval Formats

Time interval format (N is an integer)	Meaning	Examples
<i>n</i>	an exact number of seconds	10 12455
<i>ns</i>	Exact number of seconds	10s 12455s
<i>nm</i>	Exact number of minutes	5m 60m
<i>nh</i>	Exact number of hours	2h 155h
<i>nhmm</i>	Hours and minute	1h55m 10h5m
<i>nmns</i>	Minutes and seconds	5m55s 1m5s
<i>n:n</i>	Minutes and seconds	5:55 1:5
<i>nhmmns</i>	Hours minutes and seconds	1h55m10s 5h5m5s
<i>n:n:n</i>	Hours minutes and seconds	1:55:10 5:5:5

**5.1.2 Using Dates in Searches**

A number of list screens allow searches for items by date. These screens include “Date from” and “to” fields. For example, on the View->Search for Object->Installed Packages screen, searches can be performed using these dates, and the date range searched can be limited by entering a start date/time and/or an end date/time. If neither the start nor the end date/time are entered, then all the matching date/times will be listed.

The dates and times entered may be any of the supported date/time formats. See “Entering Dates and Times” on page 11.

**TABLE 3.** Example Date Ranges in Searches

date from	to	Meaning
		All dates
now		Any time from right now
	now	Any time up until right now
	1 week	Any time up until midnight at the beginning of the day 1 week from today
1 d ago	1d	Any time from midnight at the beginning of yesterday to midnight at the end of today

**TABLE 3.** Example Date Ranges in Searches

date from	to	Meaning
1/1/99	tod 12:00	Any time from midnight at the beginning of January 1 1999 to midday today
now	1y	Any time from right now to 1 year from now
tod 8:00	tod 10:55	Any time from 8 am today to 10:55 am today.
tod	tom	Any time today (i.e from midnight at the beginning of today to midnight at the end of today)

### 5.2 Nets Selector Menus

Many Nets screens employ selector menus as a means of simplifying data entry by allowing the user to directly select an object from a related table.

---

**FIGURE 4.** Example Selector



This figure shows the Address Family selector from the previous screen (Figure 3 on page 10). Selectors provide direct access to data objects contained in other database tables. Clicking on the selection bar will allow a particular object from another table to be referenced from this object, while clicking on the arrow button (->) will display the editor for the object.

These common user interface elements have been employed to provide standard editor screens and lister screens for all Nets database objects. The standard features of the editor screens and lister screens are described below.

### 5.3 Nets Editors

The Nets editors are used for all manual data entry into the database and also for viewing existing data. All Nets editors have the same format, with a File menu, an Edit menu, a View menu and a Help menu. In addition, a number of editors also have an optional Tools menu that provides useful functions relating to the specific database object that the editor is used for.

All editors also contain these buttons:

- OK  
Saves the current data in the editor to the database.  
**Note:** Nets maintains internal record numbers to avoid data collisions if more than one Nets user edits the same database object at the same time.
- Cancel  
Cancel this operation. The window will disappear.
- Help  
Display the help screen for this editor.

Some database objects (Device, Interface and Link) have more data fields that can fit comfortably on a single screen. The editors for these objects contain sub-panels, selectable by tabs, that group related information together for ease of access.

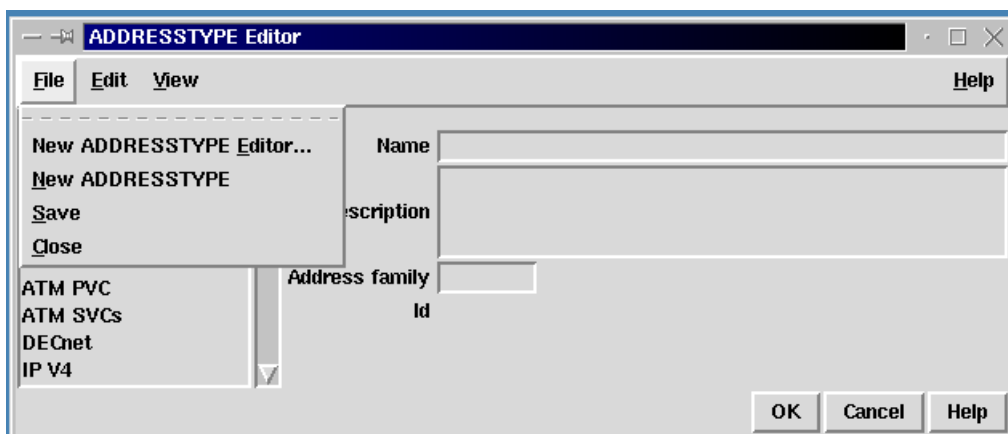
The common features of each editor menu are described in the following sections:

#### 5.3.1 Editor File Menu

All object editors have a File menu, as shown below.

---

**FIGURE 5.** Example Editor File Menu



Every editor File menu contains:

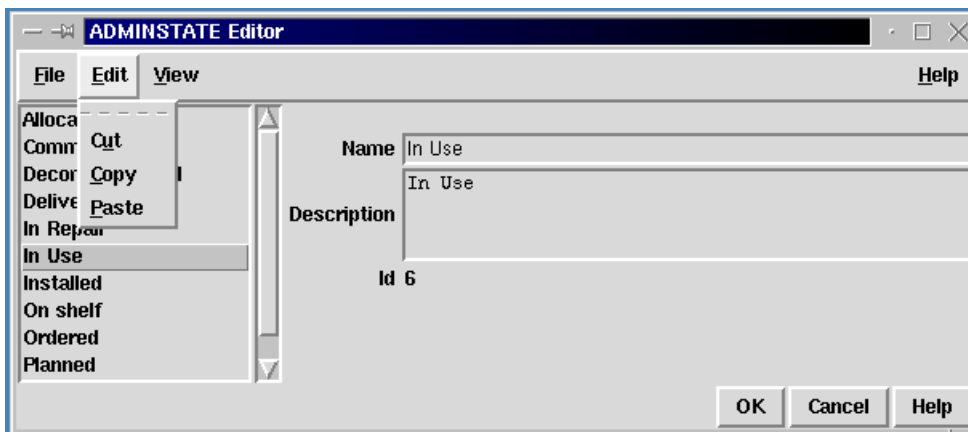
- New \*OBJECT\* Editor...  
Creates a new editor for this database object. A new window will appear.
- New \*OBJECT\*  
Clears the editor fields in preparation for entering data for a new object.
- Save  
Save the newly entered or changed data to the database.
- Close  
Close this editor. The window will disappear.

### 5.3.2 Editor Edit Menu

All object editors have an Edit menu, as shown below.

---

FIGURE 6. Example Editor Edit Menu



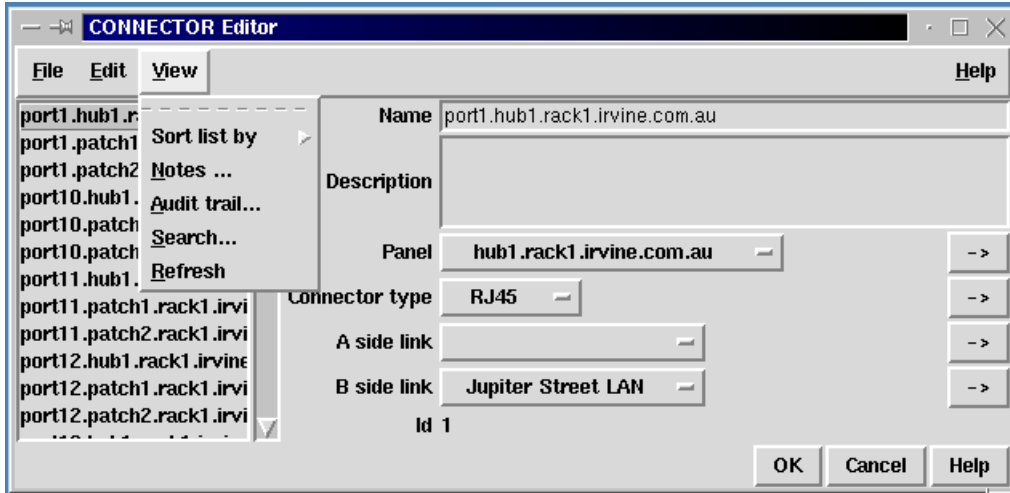
Every editor Edit menu contains:

- Cut  
Deletes this database object.  
The selected object will be deleted from the database as soon as Cut is selected.  
**Caution - this action is immediate and cannot be undone.**
- Copy  
Copy the contents of the data fields of this editor to the clipboard.
- Paste  
Paste the contents of the clipboard into the data fields of this editor. This is very useful for quickly creating new identical objects in the database, or for copying similar fields from one object type to another.

5.3.3 Editor View

All object editors have a View menu, as shown below.

FIGURE 7. Example Editor View Menu



Every editor View menu contains:

- Sort list by  
Allows the list of objects in the editor list window to be sorted by various criteria, as shown below.

TABLE 4. Sort Criteria in Object Lists

Sort criteria	Meaning	Examples
Alphabetic	Sort in alphabetic order	alpha beta
Alphabetic (reverse)	Sort in reverse alphabetic order	beta alpha
IP Address	Sort by IP address (numerical, right to left)	10.1.1.2 10.1.1.11
Id	Sort by internal Id number	3 7
Natural	Sort by	
Numeric	Sort in numerical order	1 2

Different sorts can produce identical results when used on some data lists.

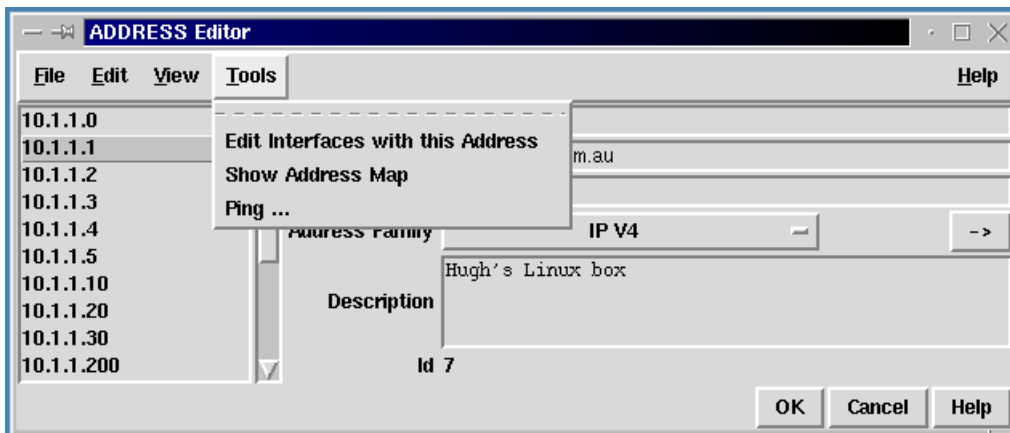
- Notes...  
Every Nets database object can have any number of notes attached to it. This menu item will bring up the Notes Editor to allow a new note to be added to the object, and existing notes to be examined.
- Audit trail...  
Every operation in the Nets system will result in an entry being created in the audit trail. This menu item will bring up the audit trail for this object, showing the date and time, action and Nets user responsible for each operation to this object.
- Search...  
This menu item will bring up the Lister screen for this type of object to allow searching for a particular object. If an object is double-clicked in the lister, it will be displayed in the corresponding object editor.
- Refresh  
This menu item will refresh the contents of this editor by re-reading data from the database, usually required after editing a related database object in a different editor.

#### 5.3.4 Editor Tools Menu

Some object editors have a Tools menu, as shown below.

---

**FIGURE 8.** Example Editor Tools Menu



The Tools Menu provides one or more useful shortcuts to related Nets database objects and their corresponding editors or displays. Additional tools may be displayed if any locally defined tools or packages have been installed in the Nets system.

### 5.4 Nets Listers

The Nets listers are used to list and search for objects in the database. All Nets listers have the same format, with a File menu, an Edit menu, a View menu and a Help menu.

Lister screens are divided into two parts. The top section allows search criteria to be specified, including such things as:

- from and to dates
- object type
- object id
- who by

The top section of all listers also contain this button:

- Search  
Searches for this type of object with the specified criteria.

The bottom section of the screen displays the search results and double-clicking on any line will display the relevant object from the database in its corresponding editor.

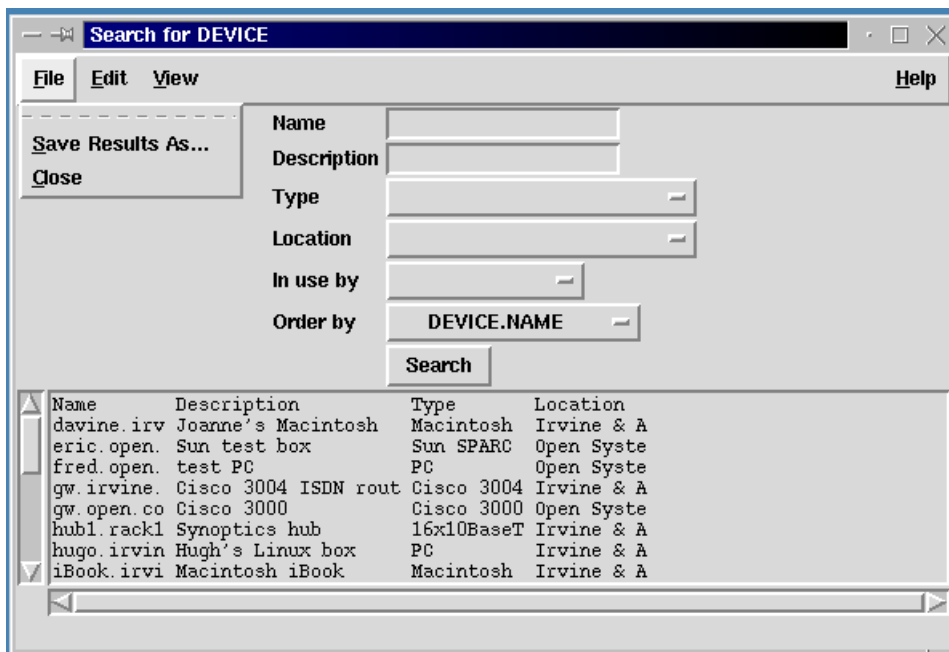
The common features of each Lister menu are described in the following sections:

#### 5.4.1 Lister File Menu

All object listers have a File menu, as shown below.

---

FIGURE 9. Example Lister File Menu



Every lister File menu contains:

- Save Results As...  
Save the search results to the specified file.
- Close  
Close this lister. The window will disappear.

#### 5.4.2 Lister Edit

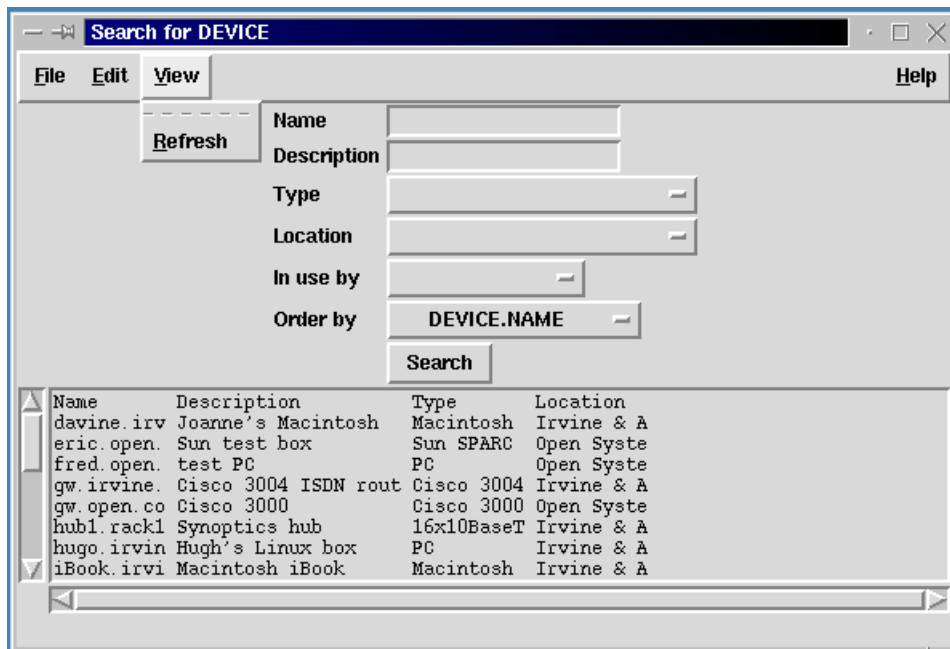
There are no Edit menu items currently defined for listers.

#### 5.4.3 Lister View Menu

All object listers have a View menu, as shown below.

---

FIGURE 10. Example Lister View Menu



Every lister View menu contains:

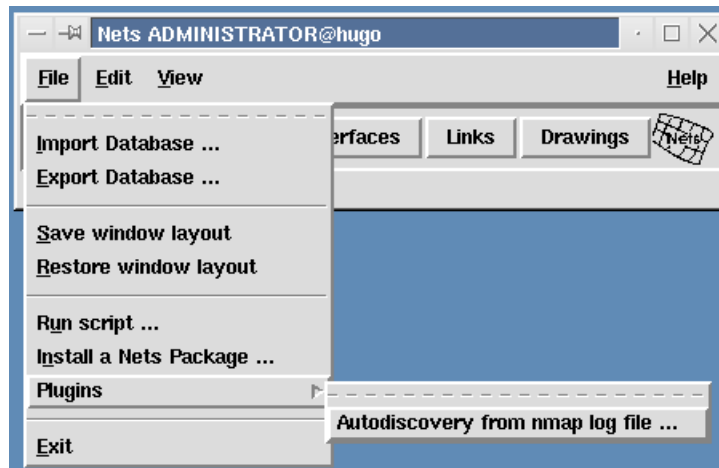
- Refresh  
This menu item will refresh the contents of this lister by re-reading object data from the database. It is usually required after editing a related database object in a corresponding editor.

## 6.0 Nets File Menu

---

The Nets File Menu allows the user to initiate fundamental Nets actions, such as importing and exporting databases, saving and restoring window layouts, running scripts, installing packages and exiting Nets. Some of these actions are controlled by the user's permissions. Refer to the Nets User editor for details. (Section 7.27 on page 58).

**FIGURE 11.** Nets Main Screen File Menu



This screen shows the Nets File Menu.

The File menu items are described in the following sections.

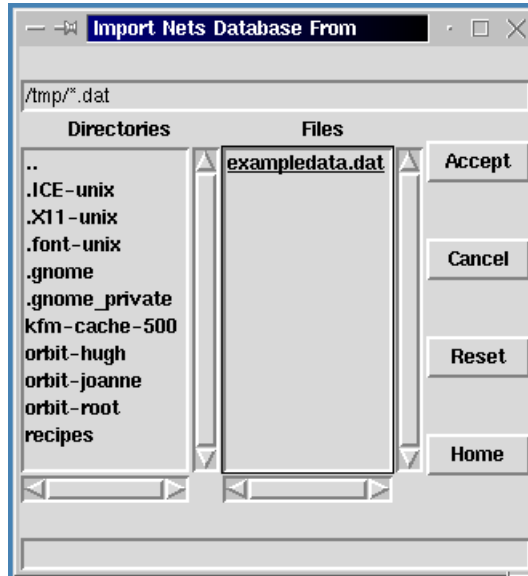
### 6.1 Import Database

Nets users can import data into the database from an external file. This is useful for importing bulk data from another system, or initial loading of the database.

---

FIGURE 12.

Nets Import Database



This screen shows the file selection dialog to choose a file to import.

The top data field shows the directory and display filter currently in use, while the bottom data field is used to enter a new file name to create in the current directory. The Directories list allows a different directory to be selected, while the Files list allows an existing file to be selected and overwritten.

**Note:** double-clicking on a file name will select and over-write the file directly.

## 6.2 Export Database

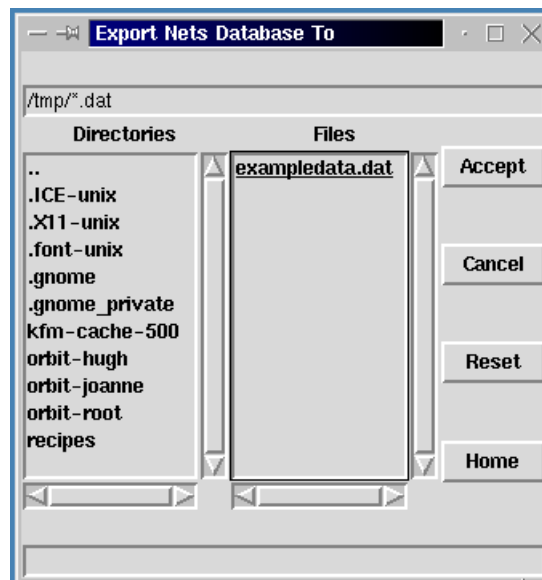
Nets users can export data from the database to an external file. This is useful for exporting bulk data to another system, or making portable backups of the database.

**Note:** every record in every table will be exported from the database to a file, in a format suitable for use with a subsequent Import Database.

---

FIGURE 13.

Nets Export Database



This screen shows the file selection dialog to choose a file to export data to.

### 6.3 Save Window Layout

Nets users can open as many different windows as desired, and position them on the screen as appropriate. The screen layout can be saved to the database on a per user basis with this menu item. The next time the user logs in, the Nets windows will be restored to their positions when they were last saved.

### 6.4 Restore Window Layout

Nets users can restore a previously saved window layout at any time by using this menu item. This can be useful if a user wishes to restore a normal window layout that has been changed during a Nets session.

### 6.5 Run Script

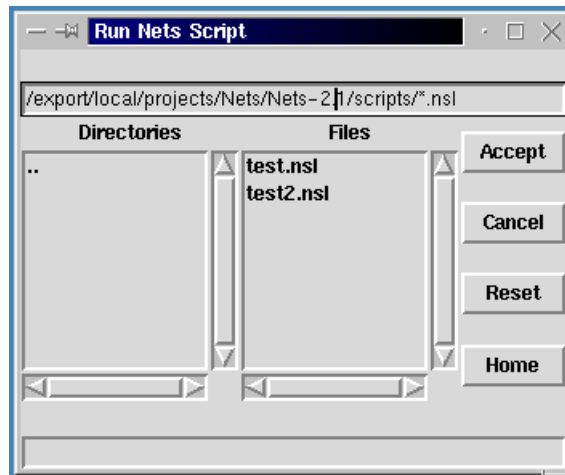
This menu item is used to run pre-defined Nets scripts. Nets scripts are background tasks that operate while the Nets program would otherwise be idle. Nets scripts are identified by the suffix .nsl (Nets Scripting Language). Refer to the Nets Developers Guide for further details regarding scripts.

**Note:** only users with appropriate permissions will be able to run scripts.

---

FIGURE 14.

Nets Run Script



This screen shows the selection dialog to choose a Nets script to run.

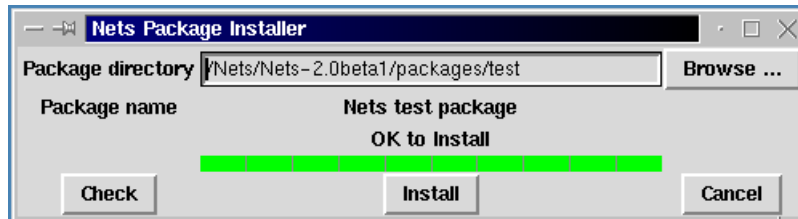
## 6.6 Install a Nets Package

This menu item is used to install Nets packages.

**Note:** only users with appropriate permissions will be able to install packages.

---

**FIGURE 15.** Nets Install Package



This screen shows the Nets package installer.

A Nets package is a directory containing a 'DESCRIPTION' file, together with the various package components and the instructions for installing them.

The Nets package installer is controlled by the following buttons:

- Browse ... - used to select a package to install
- Check - used to verify package contents before installation
- Install - used to perform the installation
- Cancel - used to exit the package installer before doing an installation

Nets packages are a convenient way for users and vendors to distribute additional Nets add-ons for installation into the Nets system. Refer to the Nets Developers Guide for further information regarding package development and contents.

## 6.7 Plugins

The Plugins menu item cascades to a list of the Nets Plugins that are installed on this system. Nets Plugins are typically installed from Nets Packages, one of which is the Autodiscovery from nmap log file ... (part of the standard Nets distribution).

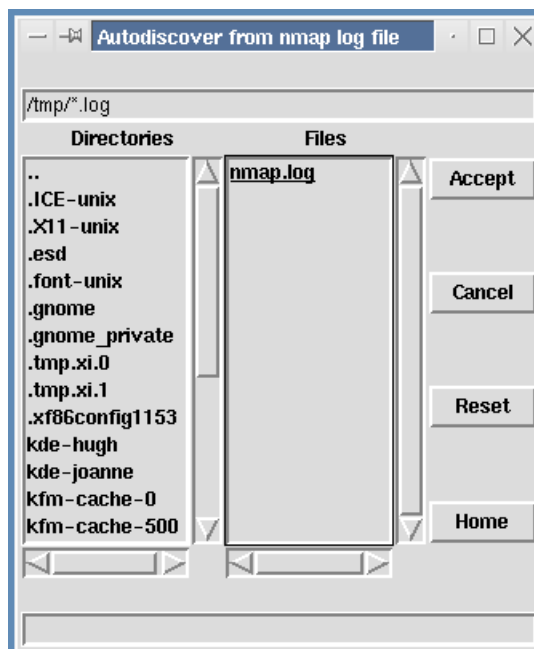
### 6.7.1 Autodiscovery from nmap log file

Nets provides a means to populate the database with the output of the **nmap** scanner.

---

FIGURE 16.

This screen is used to input the list of network elements from an nmap scan.



This screen shows the file selector to import the results of an **nmap** scan.

The **nmap** scanner can be found at the following URL:

<http://www.insecure.org/nmap/>

The output from **nmap** should be generated as follows

(using the appropriate network addresses):

```
nmap -O 10.1.1.* >nmap.log
```

**Note:** the output from an **nmap** scan will not provide a complete network inventory, however it will provide basic information that can be used to populate the database.

## 7.0 Nets Edit Menu

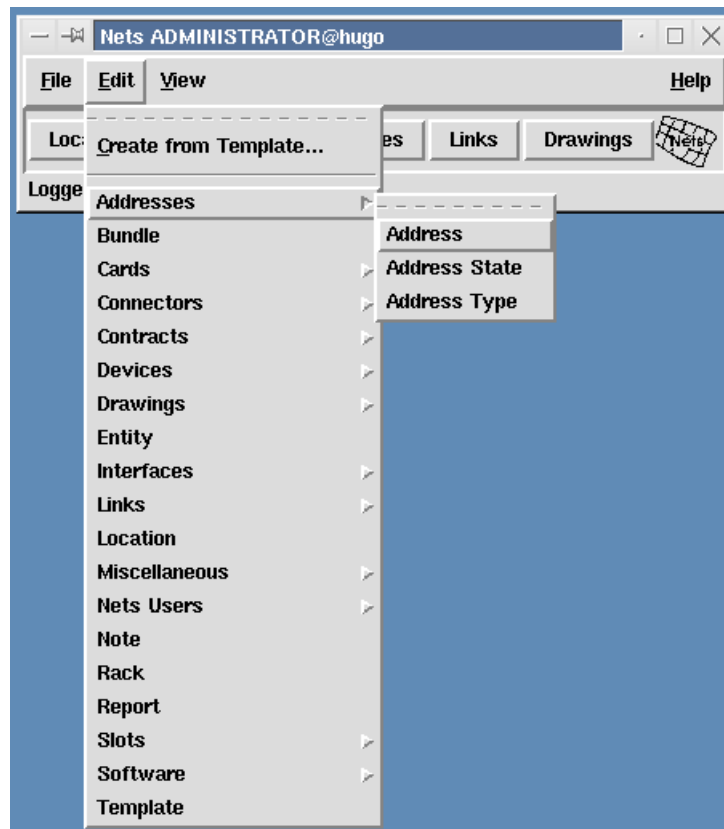
---

The Nets Edit Menu provides direct access to all Nets object editors. The Edit Menu also contains a “Create from Template...” item which allows a user to create multiple Nets objects at the same time, as defined by a Template file. This is a convenient method for combining many Nets edit operations into a single operation.

**Note:** many Nets Edit items group together two or more related Nets objects in cascaded sub-menus for ease and speed of access.

---

**FIGURE 17.** Nets Edit Menu



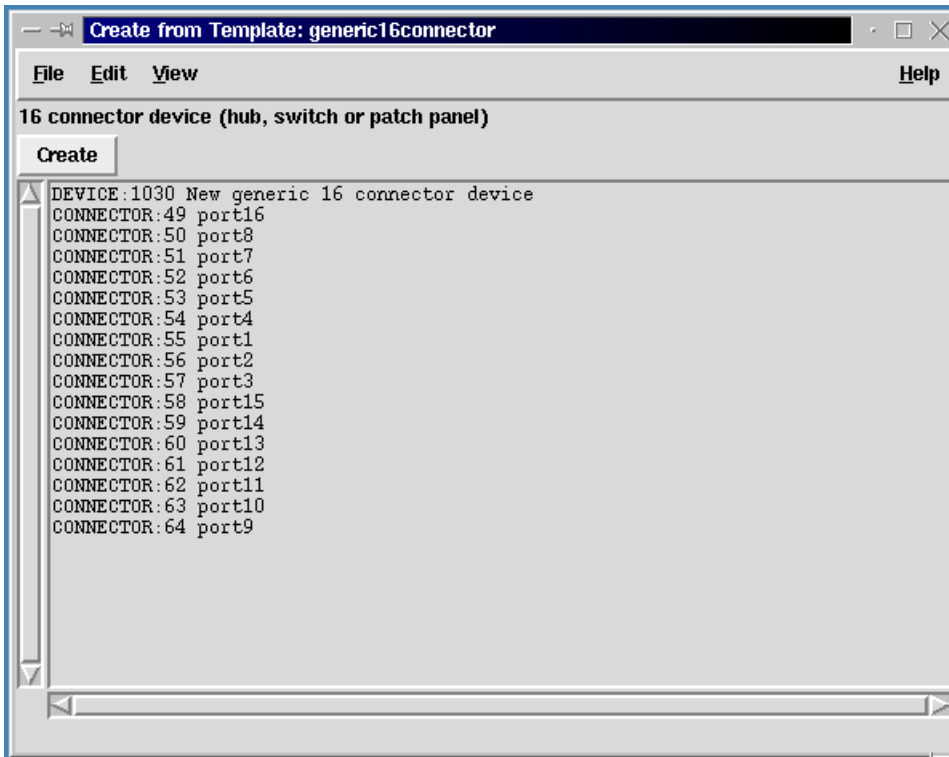
This screen shows the Nets Edit Menu.

### 7.1 Create from Template (multiple objects)

To simplify data entry into Nets, Template files can be created by combining typical groups of Nets objects into a single file. Template files are usually used to create complicated devices with many interfaces all at once. There are several standard Templates included in the Nets distribution, and additional Templates can be created as required.

---

FIGURE 18. Nets Create from Template



This screen shows one of the standard Nets Templates (generic16connector) being used to create a new 16 port switch. The same Template can be used to create any 16 port device, including hubs, switches or patch panels.

**Note:** double-clicking on any line will display the corresponding object in the appropriate editor for that object.

## 7.2 Address (ADDRESS object)

One of the important uses for Nets is keeping track of protocol addresses. This is most often used with IP V4 addresses, but any addresses from any protocol family can be entered into the system.

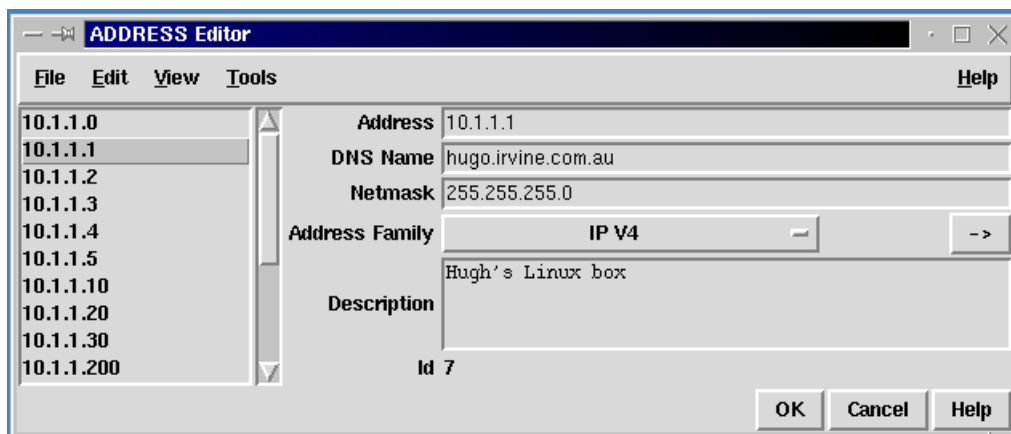
When working with IP V4 addresses, it is useful to enter CIDR block definitions and subnet definitions as well as individual host addresses, as the Nets Address Map display will show all of this information through the use of different coloured display segments.

CIDR blocks are IP address allocations, either obtained from an IP address registry (ARIN, RIPE-NCC, APNIC, etc.), or internally used private address space (10.0.0.0 for example). Subnets (as defined by netmasks) are the local subdivisions of the IP address space that are used for local LAN segments, point-to-point WAN links, etc.

IP V4 addresses are also usually associated with DNS names, and Nets can be used to store the address to DNS name mappings.

---

**FIGURE 19.** Nets Address Editor



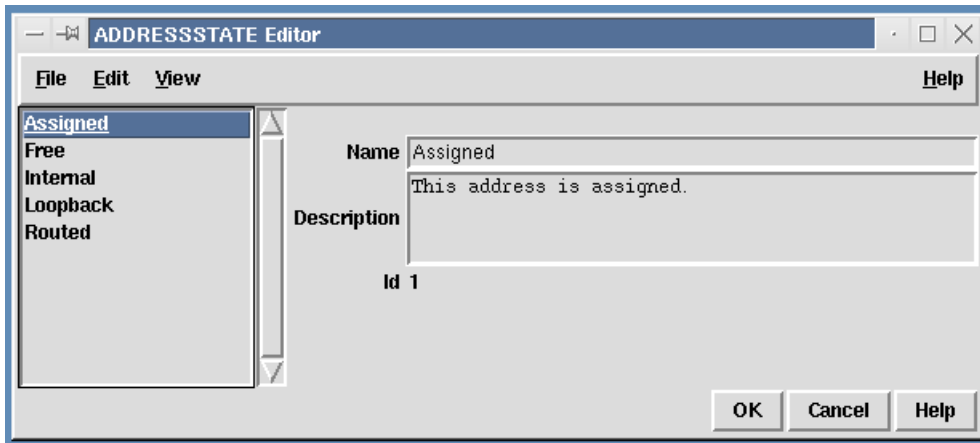
This screen shows the Nets Address Editor.

### 7.3 Address State (ADDRESSSTATE object)

Address State is used to indicate whether an address is allocated, free, in use, etc.

---

FIGURE 20. Nets Address State Editor



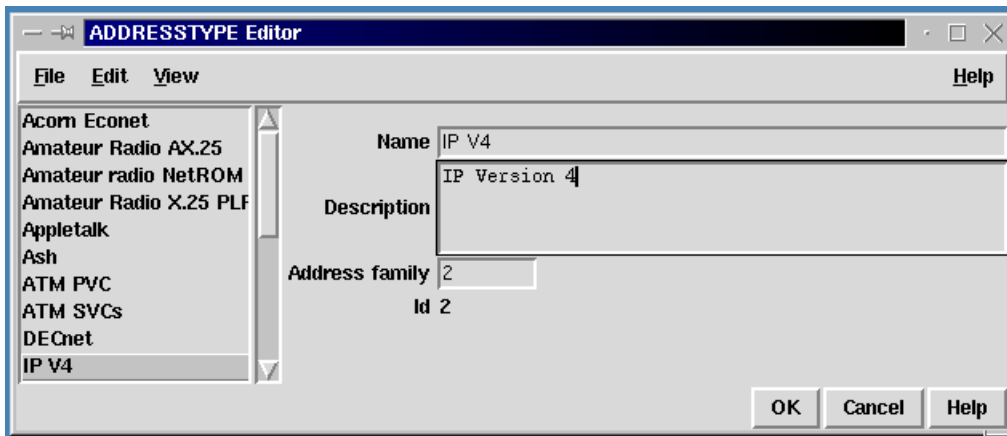
This screen shows the Address State Editor.

### 7.4 Address Type (ADDRESSTYPE object)

The Address Type refers to the specific network protocol that an Address belongs to. This will usually be IP V4, but any protocol can be defined, in addition to the standard set of protocols included with Nets.

---

FIGURE 21. Nets Address Type Editor



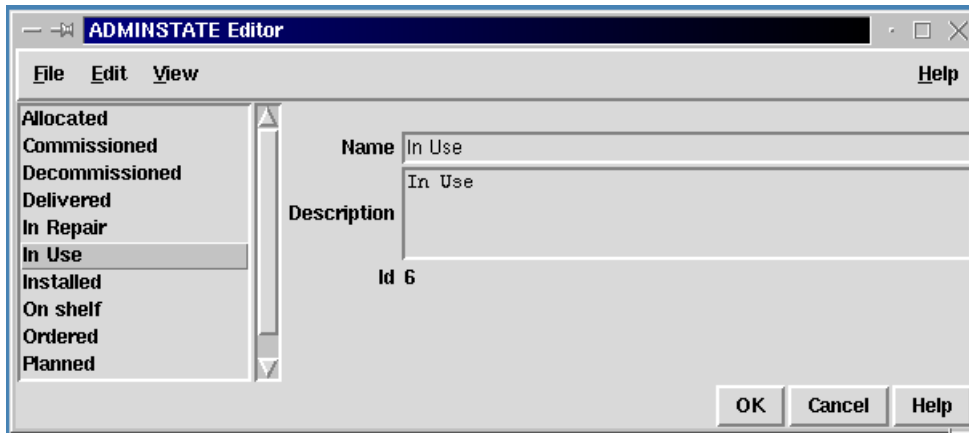
This screen shows the Address Type Editor.

### 7.5 Administrative State (ADMINSTATE object)

Nets can be used to track a variety of administrative states. Additional administrative states can be defined on a per-site basis as required. Administrative states are used in the provisioning panels of Devices, Interfaces and Links to describe the current state of the provisioning process.

---

**FIGURE 22.** Nets Administrative State Editor



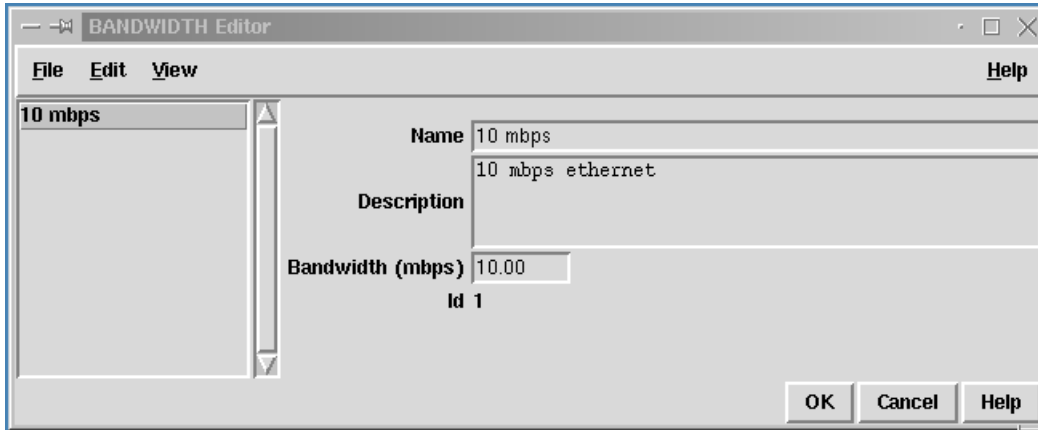
This screen shows the Administrative State Editor.

### 7.6 Bandwidth (BANDWIDTH object)

The bandwidth object is used to define different values to describe the data transmission capacity of telecommunications links. Refer to link editor in Section 7.24 on page 54.

---

FIGURE 23. Nets Bandwidth Editor



The Bandwidth field for links and virtual circuits (sub-links) can be used to track capacity utilisation, which is useful to assist with circuit provisioning.

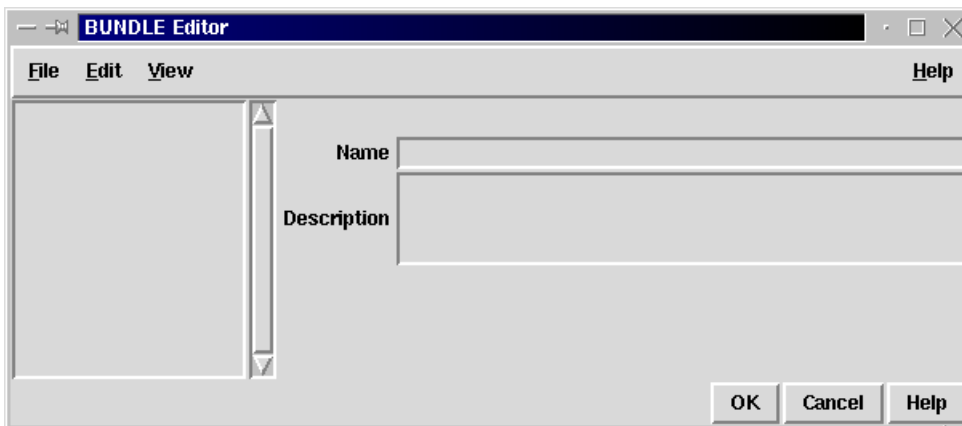
### 7.7 Bundle (BUNDLE object)

Bundles refer to cable bundles, as are typically used in equipment rooms. The physical cables that correspond to Links are usually found in cable bundles.

**Note:** the use of this object is optional and may not be required in all cases.

---

FIGURE 24. Nets Bundle Editor

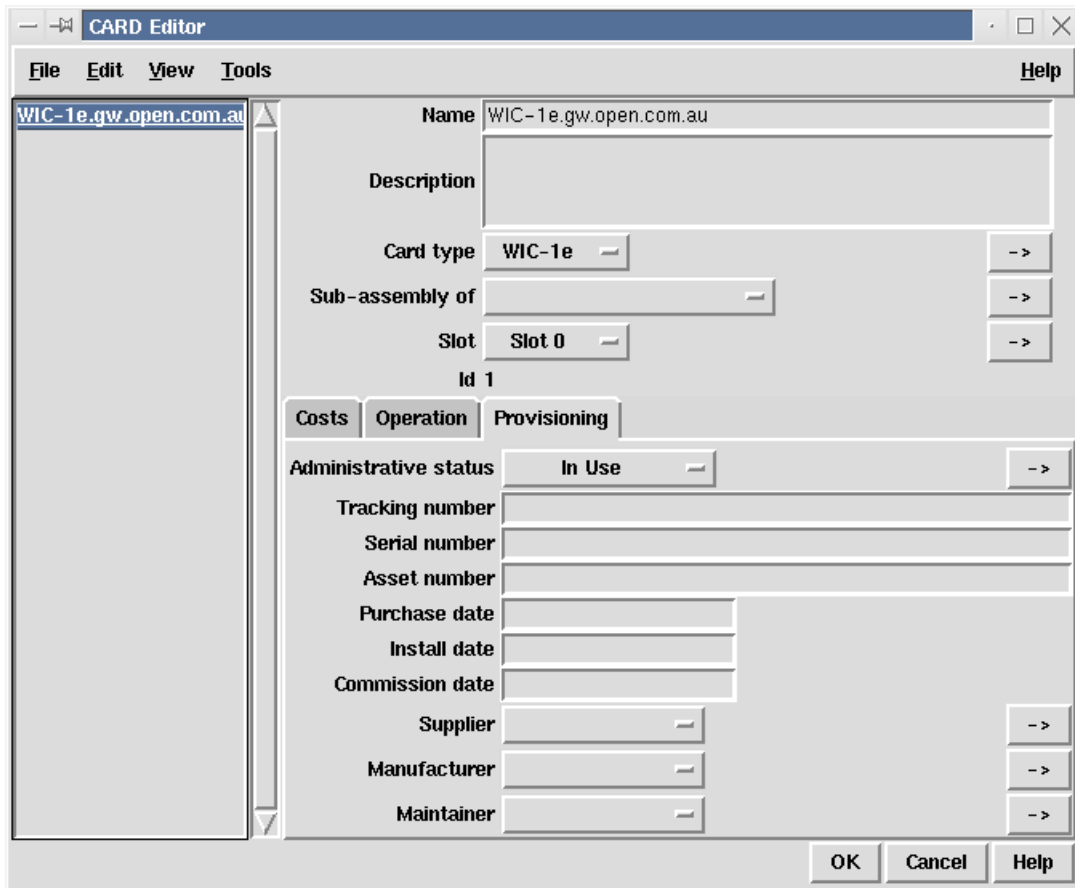


### 7.8 Card (CARD object)

Cards are optional objects that can be used when one or more interfaces are built on a card that can be installed in a device slot. See the Interface Editor for details on how an interface can be related to a card.

---

**FIGURE 25.** Nets Card Editor



This screen shows the Nets Card Editor.

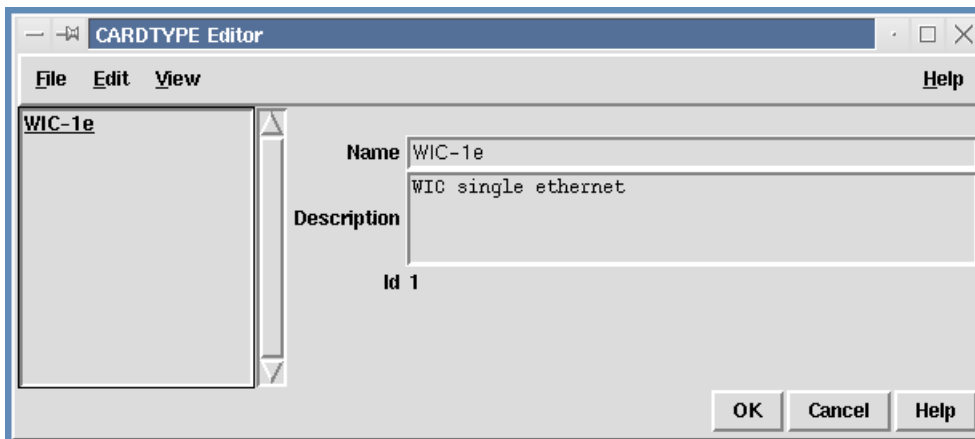
**Note:** Cards (and Slots) are optional and may not be required in all cases.

### 7.9 Card Type (CARDTYPE object)

Card Types are used to indicate what type of cards are in use.

---

**FIGURE 26.** Nets Card Type Editor



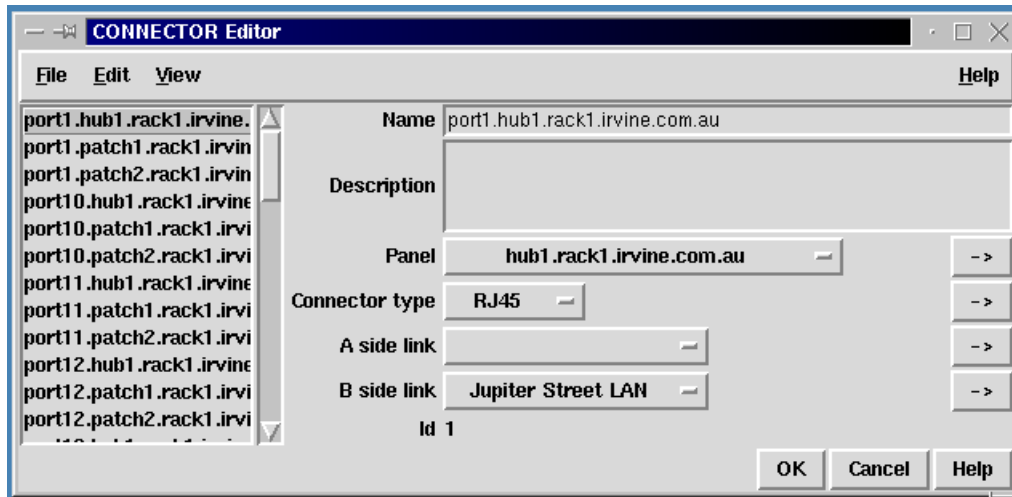
This screen shows the Nets Card Type Editor.

### 7.10 Connector (CONNECTOR object)

Connectors are used to connect two links, either individually, or more generally in groups corresponding to switches, hubs or patch panels. See the Nets Device Editor for details regarding the grouping of multiple connectors into a device.

---

FIGURE 27. Nets Connector Editor



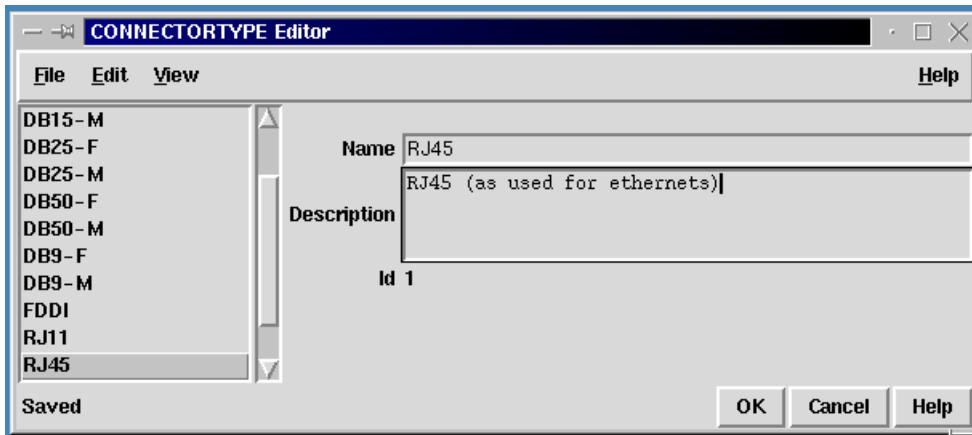
This screen shows the Nets Connector Editor. Notice the hierarchical naming scheme employed to simplify the grouping of Nets objects.

### 7.11 Connector Type (CONNECTORTYPE object)

Connectors can be of any type, many of which are pre-defined in Nets.

---

FIGURE 28. Nets Connector Type Editor



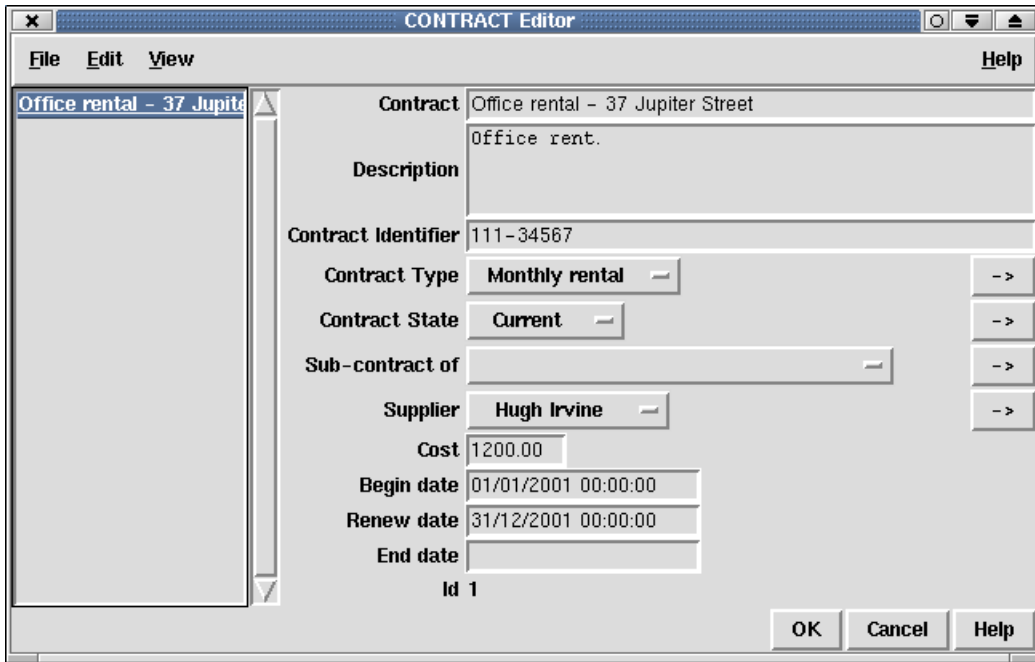
This screen shows the Nets Connector Type Editor. Additional connector types can be defined as required.

### 7.12 Contract (CONTRACT object)

Contract details can be maintained in Nets and Devices, Interfaces, Links and Locations may all refer to contracts for the purposes of leases, rental, maintenance, etc.

---

FIGURE 29. Nets Contract Editor



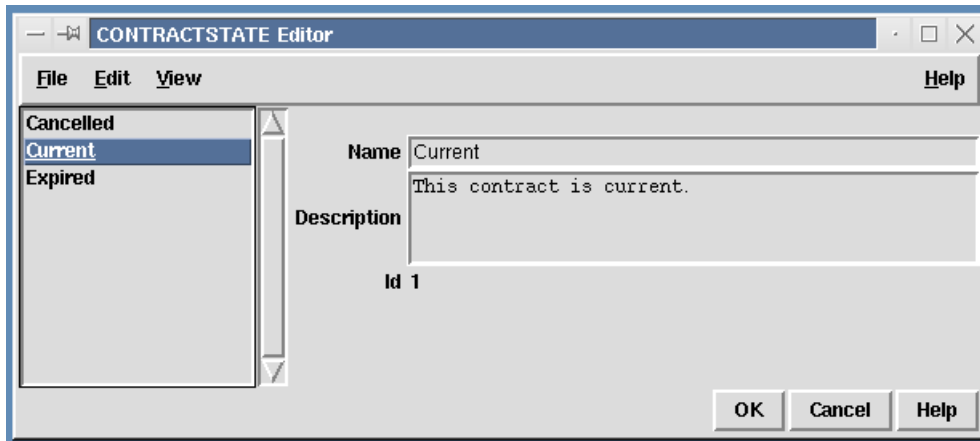
This screen shows the Nets Contract Editor.

### 7.13 Contract State (CONTRACTSTATE object)

Contracts may be in different states, which may be defined here.

---

**FIGURE 30.** Nets Contract State Editor



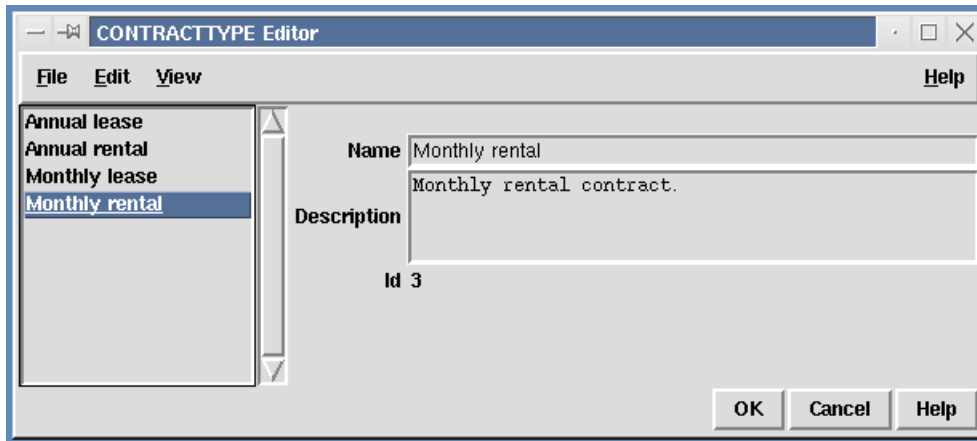
This screen shows the Nets Contract State Editor.

### 7.14 Contract Type (CONTRACTTYPE object)

Contracts may be of different types, which may be defined here.

---

**FIGURE 31.** Nets Contract Type Editor



This screen shows the Nets Contract Type Editor.

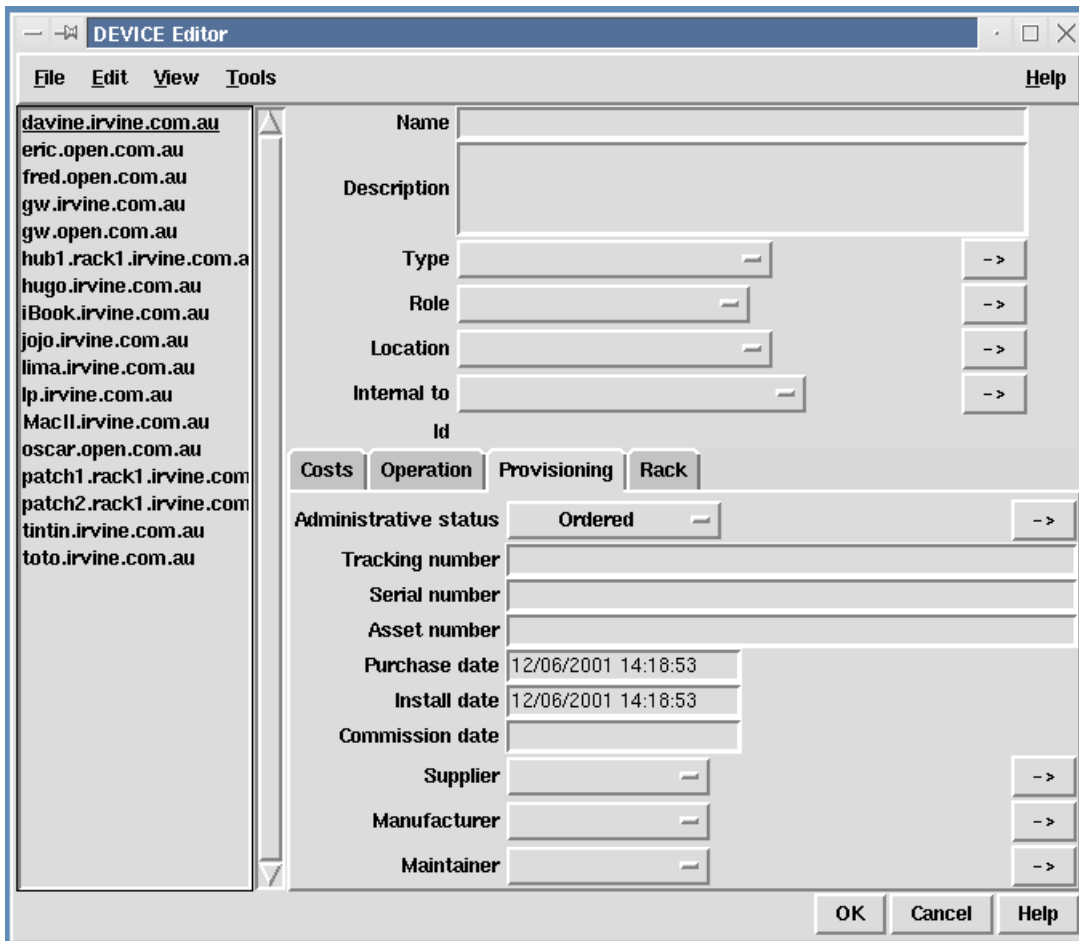
### 7.15 Device (DEVICE object)

Devices are one of the most important objects in Nets. Devices are typically network elements such as workstations, servers, routers, switches, hubs, patch panels, etc. Pseudo-devices can also be used as place-holders for tracking equipment and/or telecommunications links that have been ordered but not yet delivered or installed.

Devices can be either active or passive. Active devices incorporate one or more interfaces, together with the corresponding network addresses and links. Passive devices such as switches, hubs and patch panels contain only connectors that connect to links.

Some devices contain both interfaces and connectors, and some devices can be included inside other devices. An example of such a device would be an intelligent switch with multiple VLAN support and a protocol address to connect to it for administrative purposes.

FIGURE 32. Nets Device Editor



This screen shows the Nets Device Editor. Note the “Costs”, “Operation”, “Provisioning” and “Rack” panels that contain groups of related data for ease of access.

The Device Editor contains the following data fields:

- 7.15.1 Name**  
This is the symbolic name of this device.
- 7.15.2 Description**  
A brief description of this device and its function.
- 7.15.3 Type**  
The type of this device (server, router, switch, etc.).
- 7.15.4 Role**  
The role of this device (border router, NAS, etc.).
- 7.15.5 Location**  
The location where this device is installed.
- 7.15.6 Internal to**  
Refers to the parent device that this device is part of.
- 7.15.7 Costs - Purchase cost**  
Purchase cost of this device.
- 7.15.8 Costs - Installation cost**  
Installation cost for this device.  
Optional. Used to track installation costs.
- 7.15.9 Costs - Fixed recurring cost**  
Fixed recurring cost for this device.  
Optional. Used to track maintenance costs for example.
- 7.15.10 Costs - Variable recurring cost**  
Variable recurring cost for this device.  
Optional. Used to track usage charges for example.
- 7.15.11 Costs - Depreciation rate**  
Depreciation rate for this device.  
Optional. Used for asset depreciation calculations.
- 7.15.12 Costs - Maintenance contract**  
Maintenance contract for this device.

- 7.15.13 Operation - Administrative interface**  
The administrative interface to access this device.  
Optional. Used for remote access to router loopback interfaces for example.
- 7.15.14 Operation - Operational state**  
The current operational state of this device.
- 7.15.15 Operation - In use by**  
The entity (customer) that this device is being used by.
- 7.15.16 Provisioning - Administrative status**  
The administrative status of this device (planned, ordered, installed, etc.).
- 7.15.17 Provisioning - Tracking number**  
Tracking number for this device.
- 7.15.18 Provisioning - Serial number**  
Serial number of this device.
- 7.15.19 Provisioning - Asset number**  
The asset number of this device.
- 7.15.20 Provisioning - Purchase date**  
The date this device was purchased.
- 7.15.21 Provisioning - Install date**  
The date this device was installed.
- 7.15.22 Provisioning - Commission date**  
The date this device was commissioned.
- 7.15.23 Provisioning - Supplier**  
The supplier of this device.
- 7.15.24 Provisioning - Manufacturer**  
The manufacturer of this device.
- 7.15.25 Provisioning - Maintainer**  
The maintainer of this device.
- 7.15.26 Rack - Rack**  
The rack this device is installed in.
- 7.15.27 Rack - Voltage**  
The voltage required by this device.
- 7.15.28 Rack - Power**  
The power required by this device.

**7.15.29 Rack - Position**

The position of this device in the rack (RU).

**Note:** either top to bottom, or bottom to top, depending on local preference.

**7.15.30 Rack - Height**

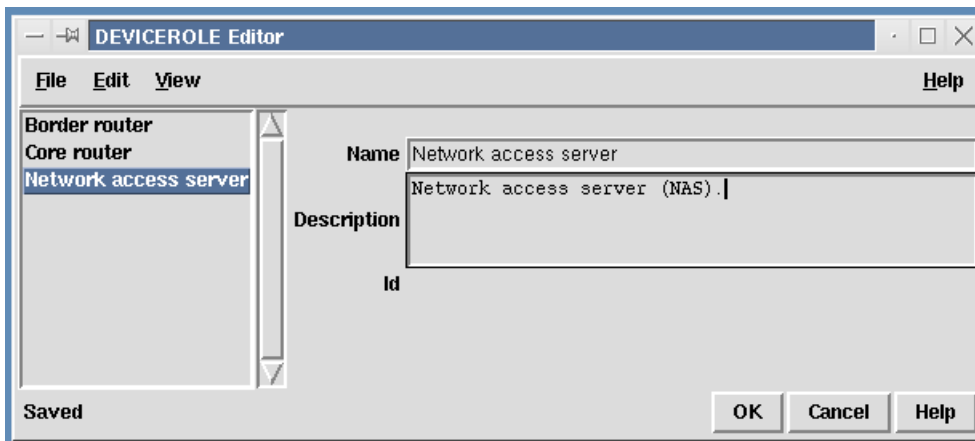
The height of this device in the rack (RU).

### 7.16 Device Role (DEVICEROLE object)

Devices may perform different roles, which are defined here.

---

FIGURE 33. Nets Device Role Editor



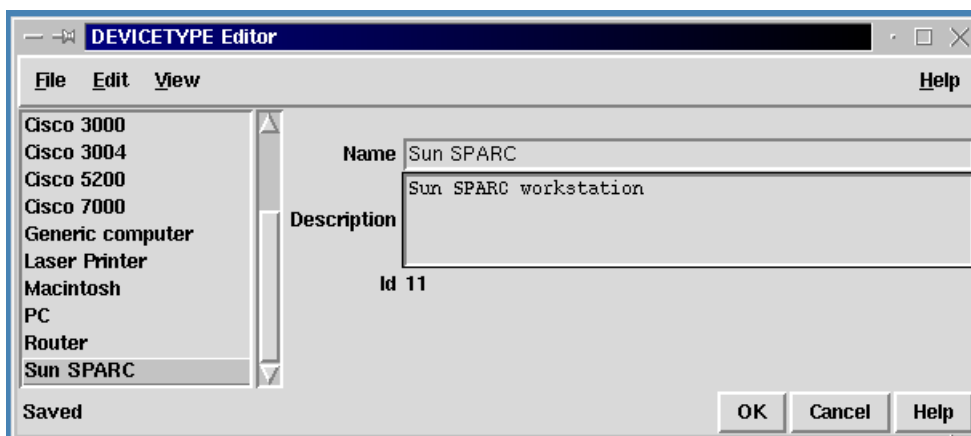
This screen shows the Device Role Editor.

### 7.17 Device Type (DEVICETYPE object)

Nets defines devices according to device type. Additional device types can be added to Nets at any time using this screen.

---

FIGURE 34. Nets Device Type Editor



The Nets Device Type Editor.

### 7.18 Drawing Object (DOBJ object)

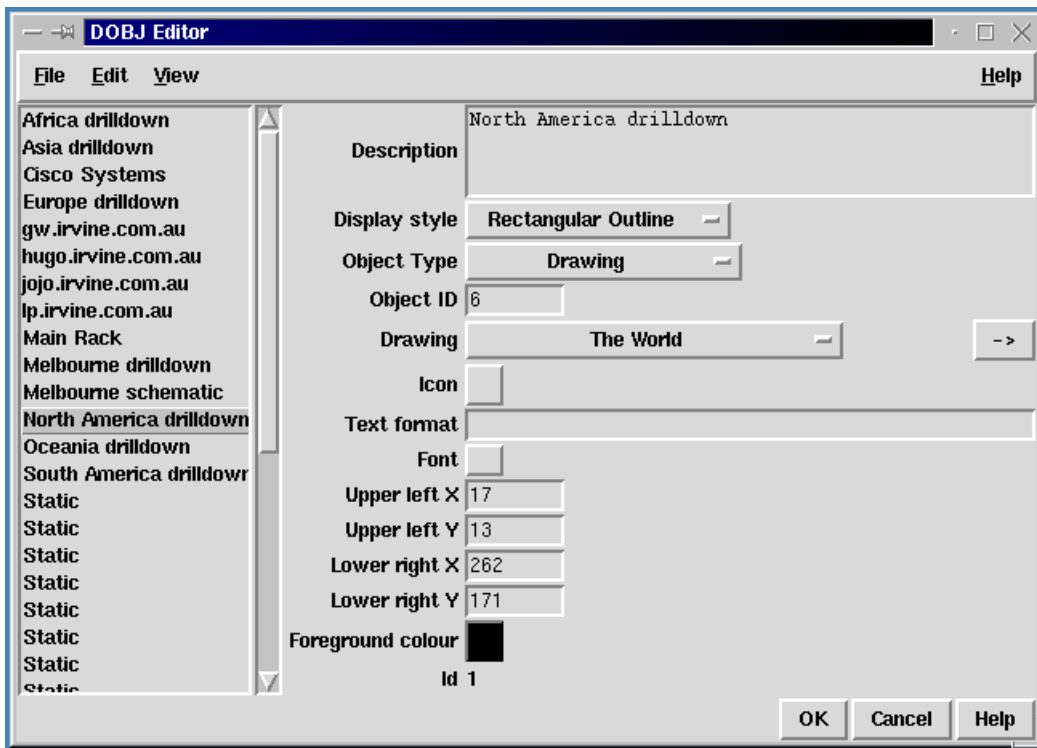
Nets Drawing Objects are the visual elements added to maps, schematics and digital photographs to represent network elements (or other drawings).

Nets drawing objects include icons, text, lines and click-through areas. Each drawing object can represent any Nets object (including other drawings), thereby allowing great flexibility in defining and using groups and sets of drawings.

**Note:** it is much simpler and more straight-forward to create and manipulate drawing objects using the tools provided in the drawing viewer. Refer to Section 9.3.4 on page 108.

---

FIGURE 35. Nets Drawing Object Editor



This screen shows the Nets Drawing Object Editor. For examples of how to define and use drawing objects, refer to Section 9.0 on page 94 of this manual.

### 7.19 Drawing (DRAWING object)

Nets drawings are the backgrounds upon which drawing objects are placed. Nets drawings can be prepared in any drawing program, can be scanned from maps or schematic diagrams, or can be imported from any digital camera. Nets supports drawing images in GIF, XBM or XPM formats in the standard release, and additional formats can be supported with the use of suitable Tk modules (available from CPAN, [www.cpan.org](http://www.cpan.org)). Refer to the Nets Installation and Administration Guide for details.

---

FIGURE 36. Nets Drawing Editor



This screen shows the Nets Drawing Editor. For examples of how to define and use drawings, refer to Section 9.0 on page 94 of this manual.

**Note:** Filename refers to a file in the <NetsDir/images/drawings> directory.

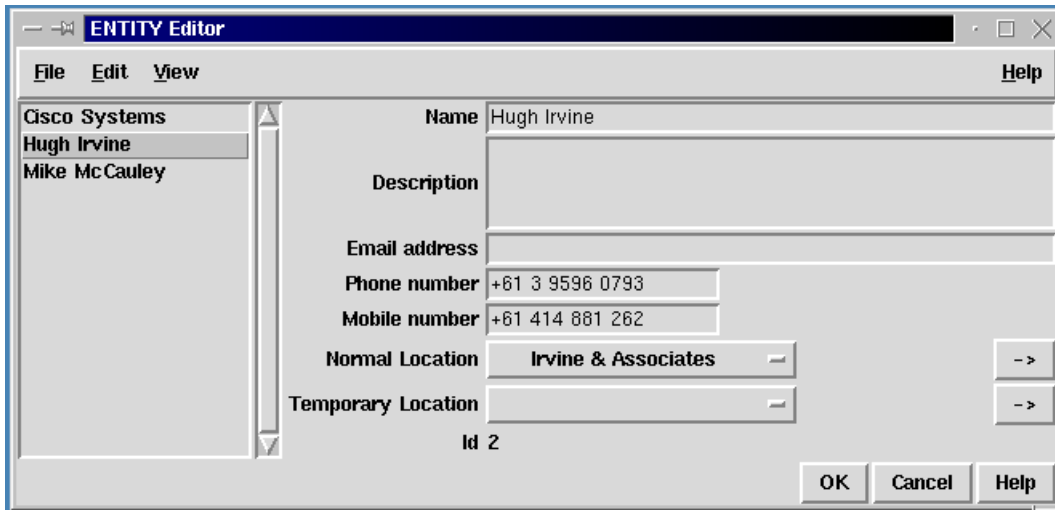
### 7.20 Entity (ENTITY object)

Nets entities are people and/or companies for which contact details are kept.

**Note:** pseudo-entities can be used for sms, pager and email gateways.

---

**FIGURE 37.** Nets Entity Editor



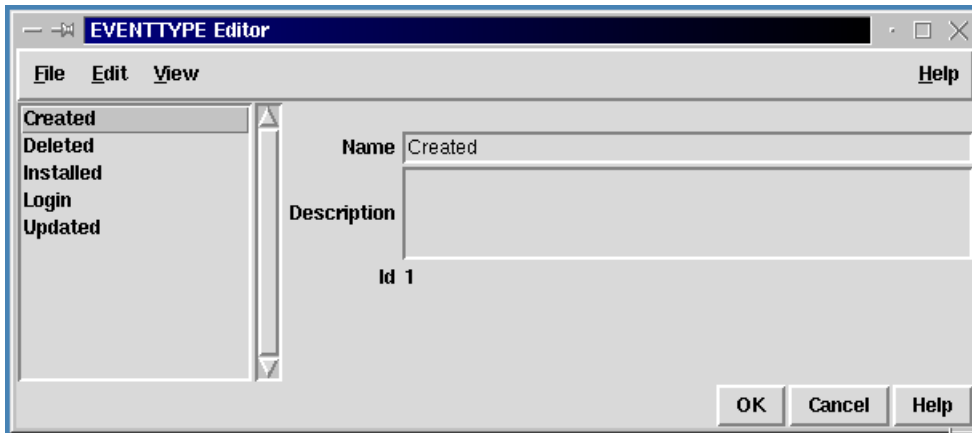
This screen shows the Nets Entity Editor.

### 7.21 Event Type (EVENTTYPE object)

Event Types are used in the Nets audit trail to keep track of database activity.

---

**FIGURE 38.** Nets Event Type Editor



This screen shows the Nets Event Type Editor. Additional event types can be added at any time as required.

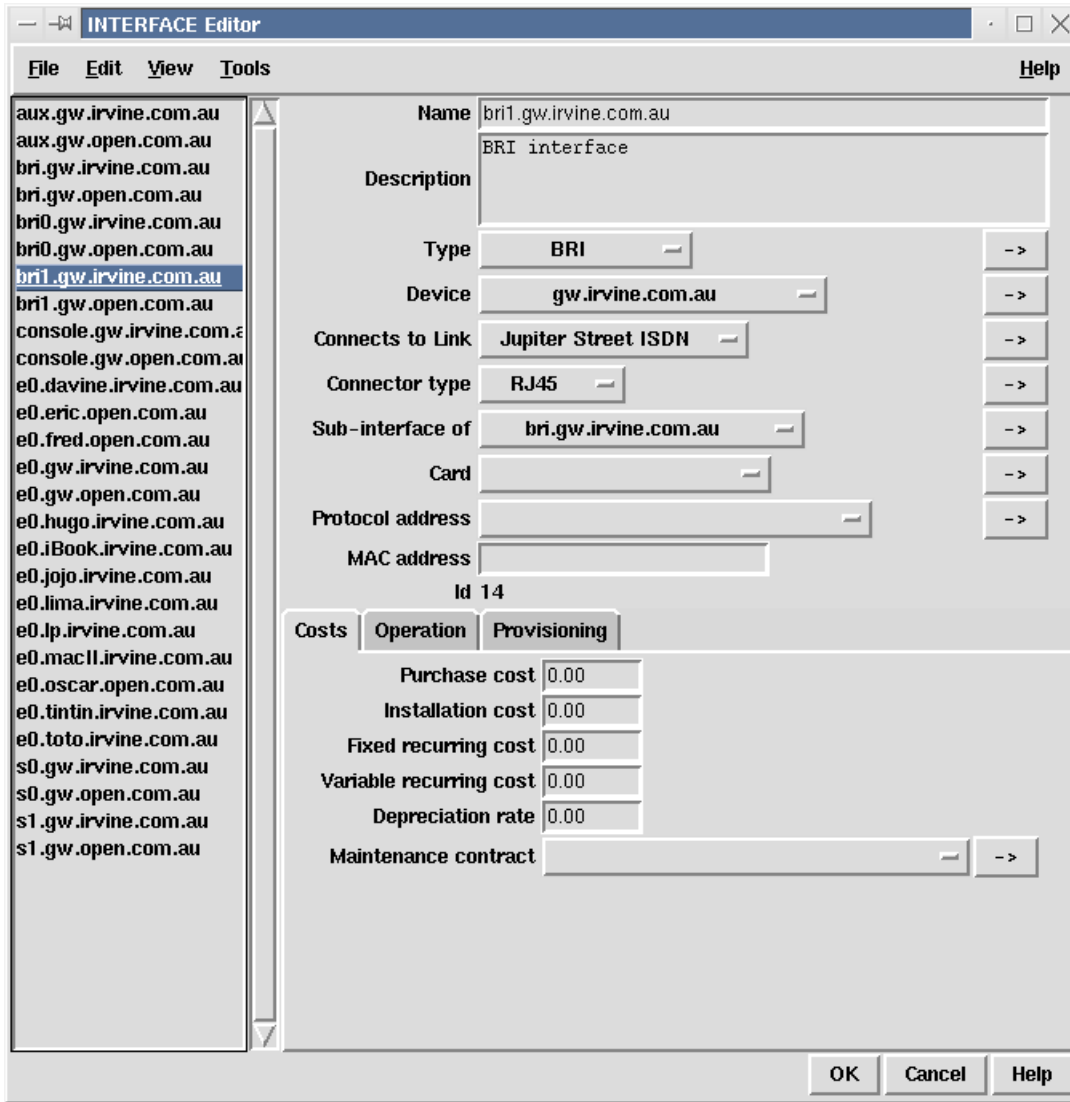
### **7.22 Interface (INTERFACE object)**

Interfaces are important Nets objects. Interfaces are the active network elements contained in devices that have network protocols and network addresses (such as IP V4) assigned to them, and that are used to connect to links.

Interfaces are usually created as parts of devices, and interfaces connect to LAN and/or WAN links. Interfaces are also configured with addresses (and DNS names), connector types and MAC addresses (optional).

Interfaces that support channelized data streams (such as Frame Relay, ATM, etc.) can have sub-interfaces associated with them that can be used to connect to virtual circuits that have been defined for the links connected to these interfaces.

FIGURE 39. Nets Interface Editor



This screen shows the Nets Interface Editor. Note the “Costs”, “Operation” and “Provisioning” panels that contain groups of related data for ease of access.

The Interface Editor contains the following data fields:

**7.22.1 Name**

This is the symbolic name of this interface.

**7.22.2 Description**

A brief description of this interface and its function.

- 7.22.3 Type**  
The type of this interface (ethernet, serial, ISDN, POS etc.).
- 7.22.4 Device**  
The device this interface is installed in.
- 7.22.5 Connects to Link**  
The link this interface connects to.
- 7.22.6 Connector type**  
The connector type used on this interface to connect to the link.
- 7.22.7 Sub-interface of**  
The parent interface that this interface is part of (for use with virtual circuits).
- 7.22.8 Card**  
The card that this interface is part of (optional).
- 7.22.9 Protocol address**  
The protocol address allocated to this interface.
- 7.22.10 MAC address**  
The media access control (MAC) address of this interface (optional).
- 7.22.11 Costs - Purchase cost**  
Purchase cost of this interface.
- 7.22.12 Costs - Installation cost**  
Installation cost for this interface.
- 7.22.13 Costs - Fixed recurring cost**  
Fixed recurring cost for this interface.
- 7.22.14 Costs - Variable recurring cost**  
Variable recurring cost for this interface.
- 7.22.15 Costs - Depreciation rate**  
Depreciation rate for this interface.
- 7.22.16 Costs - Maintenance contract**  
The maintenance contract for this interface.
- 7.22.17 Operation - Operational state**  
The current operational state of this interface.
- 7.22.18 Operation - In use by**  
The entity (customer) that this interface is being used by.
- 7.22.19 Provisioning - Administrative status**  
The administrative status of this interface (planned, ordered, installed, etc.).

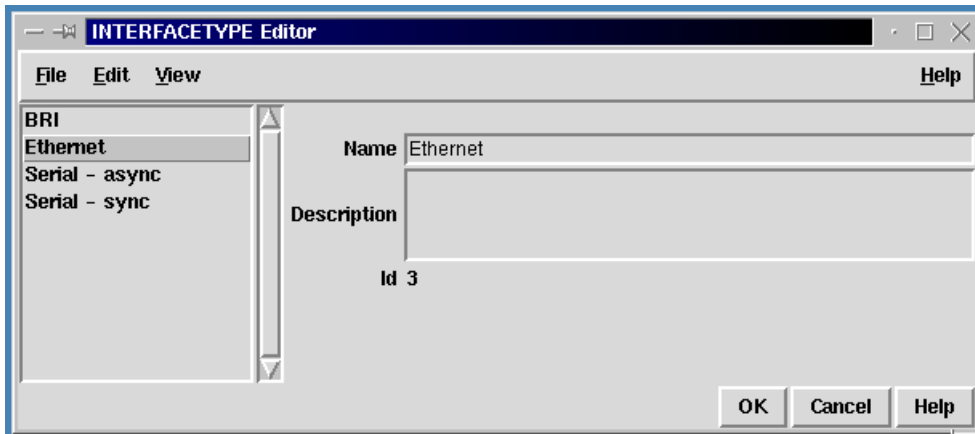
- 7.22.20 Provisioning - Tracking number**  
Tracking number for this interface.
- 7.22.21 Provisioning - Serial number**  
Serial number of this interface.
- 7.22.22 Provisioning - Asset number**  
The asset number of this interface.
- 7.22.23 Provisioning - Purchase date**  
The date this interface was purchased.
- 7.22.24 Provisioning - Install date**  
The date this interface was installed.
- 7.22.25 Provisioning - Commission date**  
The date this interface was commissioned.
- 7.22.26 Provisioning - Supplier**  
The supplier of this interface.
- 7.22.27 Provisioning - Manufacturer**  
The manufacturer of this interface.
- 7.22.28 Provisioning - Maintainer**  
The maintainer of this interface.

### 7.23 Interface Type (INTERFACETYPE object)

Interface Types are used to indicate what type a particular interface is. Additional interface types may be added at any time using this screen.

---

FIGURE 40. Nets Interface Type Editor



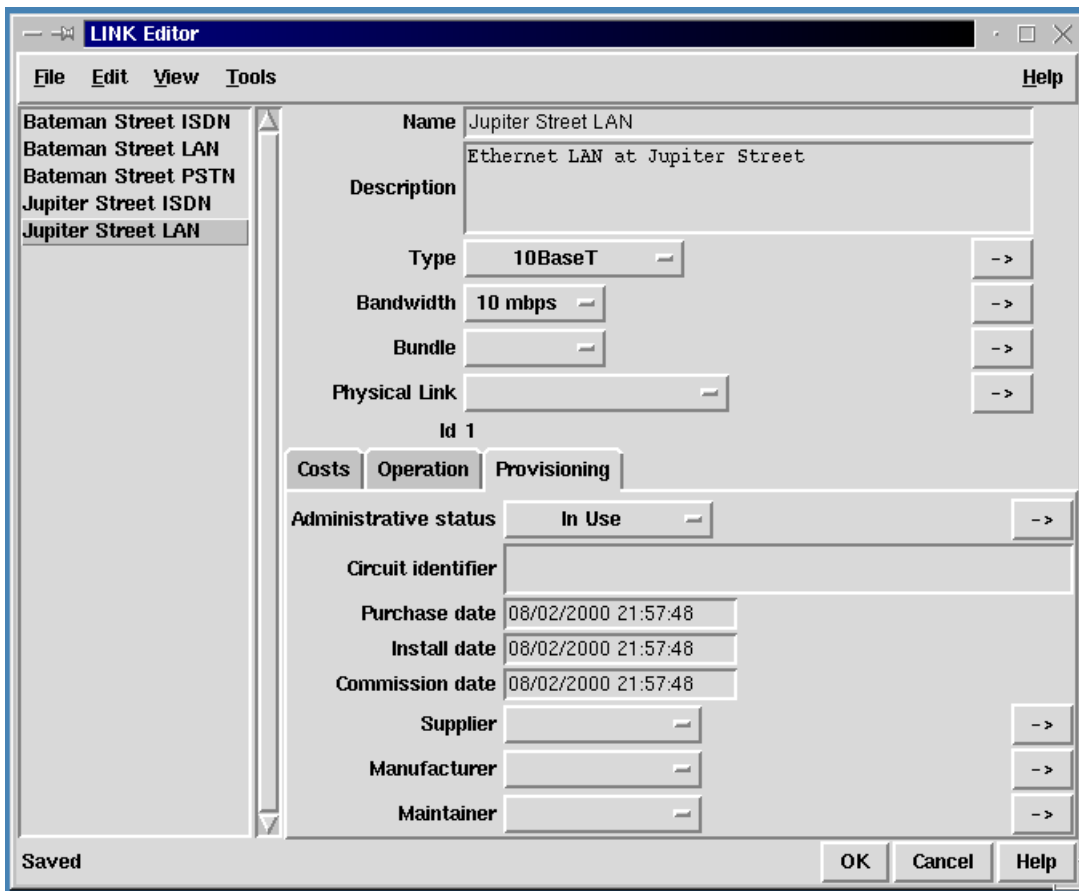
This screen shows the Nets Interface Type Editor.

### 7.24 Link (LINK object)

Links are important Nets objects that describe all LAN and WAN links in the network. Links are configured with a link type, supplier, circuit identifier and additional operational and provisioning information.

Links that describe virtual circuits can be associated with the physical links that carry them and the virtual circuits can be connected to the corresponding sub-interfaces that the parent interface contains.

FIGURE 41. Nets Link Editor



This screen shows the Nets Link Editor. Note the “Costs”, “Operation” and “Provisioning” panels that contain groups of related data for ease of access.

The Link Editor contains the following data fields:

- 7.24.1 Name**  
This is the symbolic name of this link.
- 7.24.2 Description**  
A brief description of this link and its function.
- 7.24.3 Type**  
The type of this link (ethernet, serial, ISDN, POS, etc.).
- 7.24.4 Bandwidth**  
The bandwidth capacity of this link.
- 7.24.5 Bundle**  
The cable bundle this link is part of. Refer to Section 7.6 on page 32 for details.
- 7.24.6 Physical Link**  
The link this virtual circuit is part of.
- 7.24.7 Costs - Purchase cost**  
Purchase cost of this link.
- 7.24.8 Costs - Installation cost**  
Installation cost for this link.
- 7.24.9 Costs - Fixed recurring cost**  
Fixed recurring cost for this link.
- 7.24.10 Costs - Variable recurring cost**  
Variable recurring cost for this link.
- 7.24.11 Costs - Depreciation rate**  
Depreciation rate for this link.
- 7.24.12 Costs - Maintenance contract**  
Maintenance contract for this link.
- 7.24.13 Operation - Operational state**  
The current operational state of this link.
- 7.24.14 Operation - In use by**  
The entity (customer) that this link is being used by.
- 7.24.15 Provisioning - Administrative status**  
The administrative status of this link (planned, ordered, installed, etc.).
- 7.24.16 Provisioning - Circuit identifier**  
The circuit identifier for this link.

**7.24.17 Provisioning - Purchase date**

The date this link was purchased.

**7.24.18 Provisioning - Install date**

The date this link was installed.

**7.24.19 Provisioning - Commission date**

The date this link was commissioned.

**7.24.20 Provisioning - Supplier**

The supplier of this link.

**7.24.21 Provisioning - Manufacturer**

The manufacturer of this link.

**7.24.22 Provisioning - Maintainer**

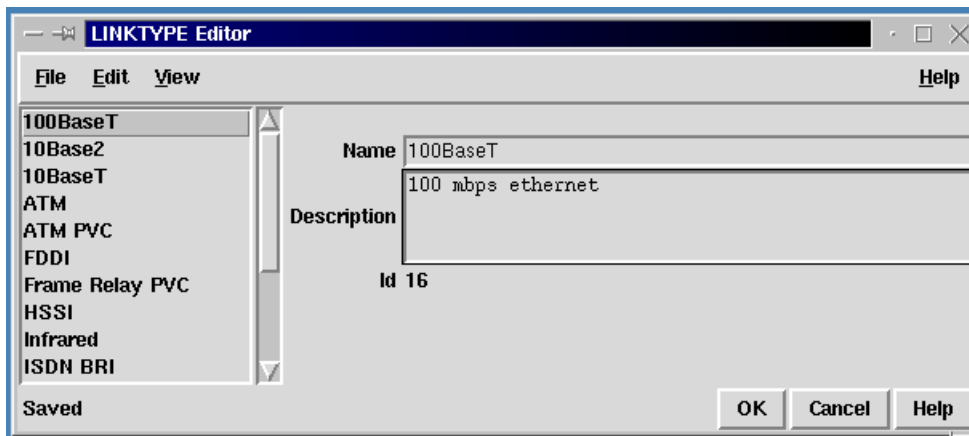
The maintainer of this link.

**7.25 Link Type (LINKTYPE object)**

Link types are used in links to describe the type of each link.

---

**FIGURE 42.** Nets Link Type Editor



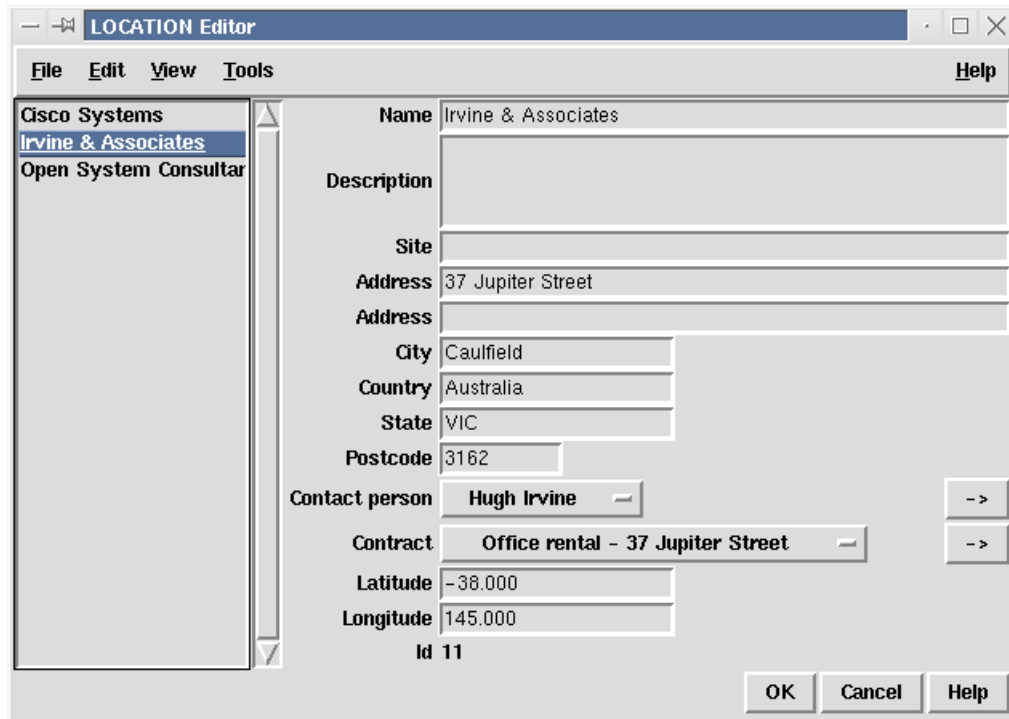
This screen shows the Nets Link Type Editor. Additional link types can be defined at any time as required.

### 7.26 Location (LOCATION object)

Locations are defined for all POP's, offices, hosting facilities, etc. that are part of the network. Address details and contact details are maintained, as well as latitude and longitude references for use with maps.

---

FIGURE 43. Nets Location Editor



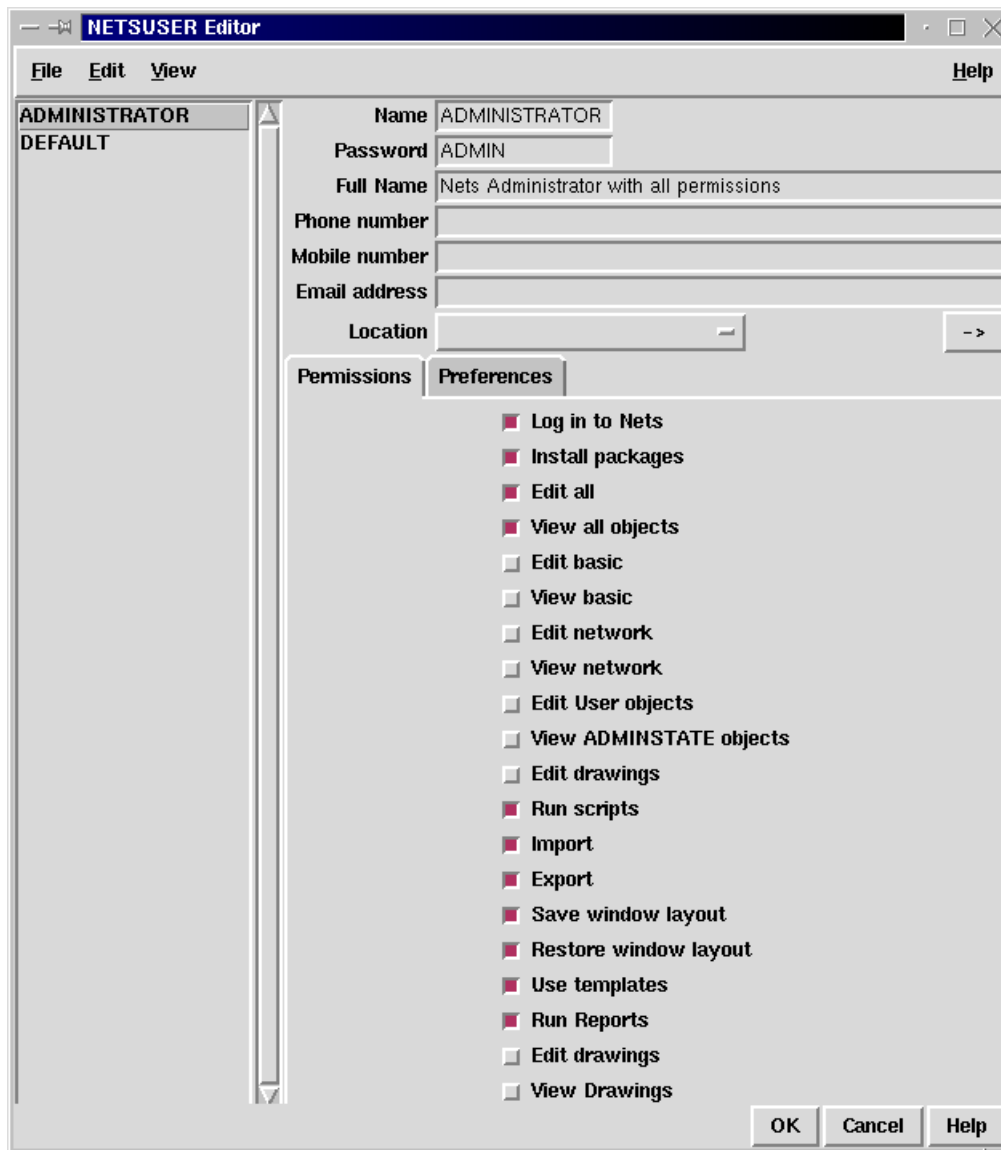
This screen shows the Nets Location Editor.

### 7.27 Nets User (NETSUSER object)

Nets users are those users who are permitted to log in to the Nets system. The Nets administrator is typically the only user who is allowed to access user information.

---

FIGURE 44. Nets User Editor



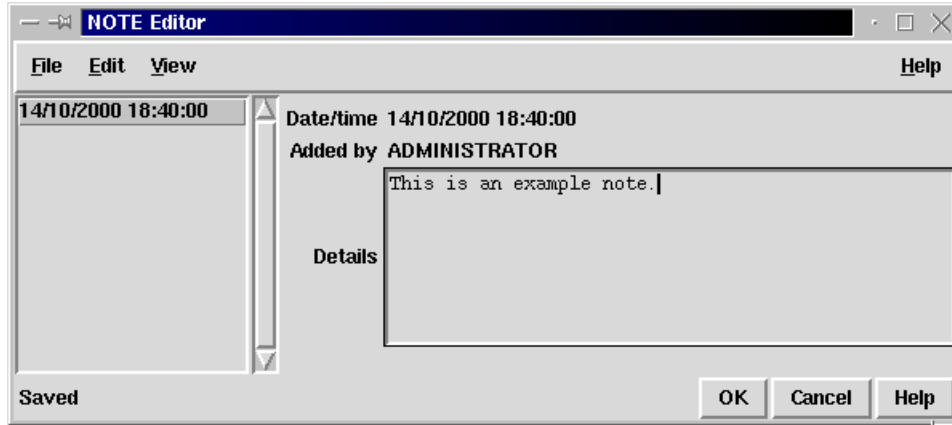
This screen shows the Nets User Editor.

### 7.28 Note (NOTE object)

Notes can be added to any Nets object through the View->Notes... menu item of any object editor. All notes associated with a particular object are time stamped and include the username of the Nets user who added the note.

---

FIGURE 45. Nets Note Editor



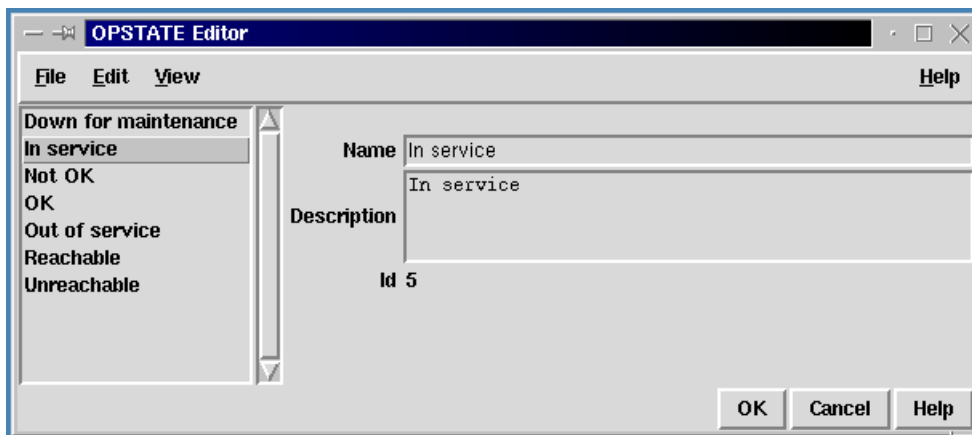
This screen shows the Nets Note Editor.

### 7.29 Operating State (OPSTATE object)

Nets can be used to track a variety of operating states. Operating state reflects the current state of a device, interface or link. Additional operating states can be defined on a per-site basis as required.

---

FIGURE 46. Nets Operating State Editor



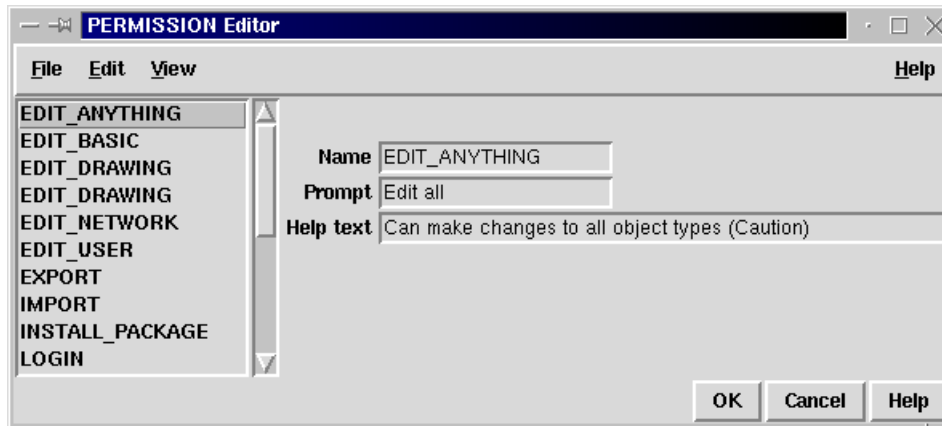
This screen shows the Nets Operating State Editor.

### 7.30 Permission (PERMISSION object)

Nets users are granted permissions when they are created. Site-specific permissions can be added with this editor if required to enable or disable access to particular operations.

---

FIGURE 47. Nets Permission Editor



This screen shows the Nets Permission Editor.

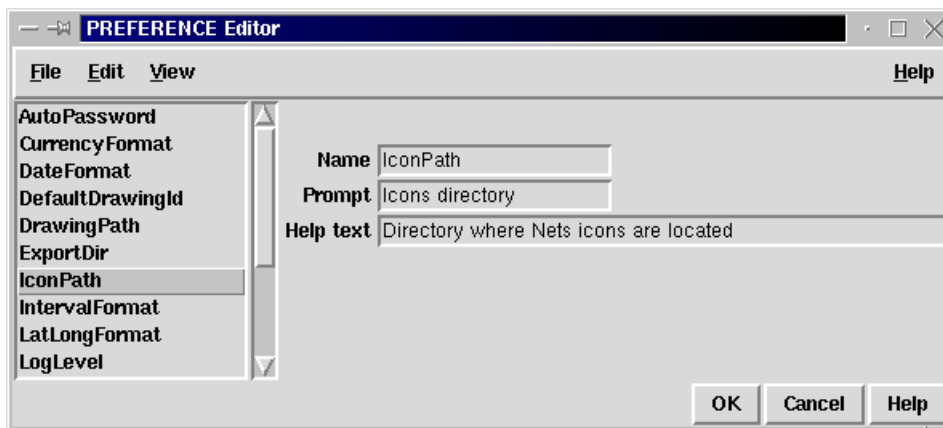
### 7.31 Preference (PREFERENCE object)

Nets users are given preferences when they are created. The set of preferences that are defined in Nets can be added to at any time as required.

**Note:** the default values for all preferences can be defined in the file Nets/Site.pm and can be over-ridden by various configuration files and run-time parameters.

---

FIGURE 48. Nets Preference Editor



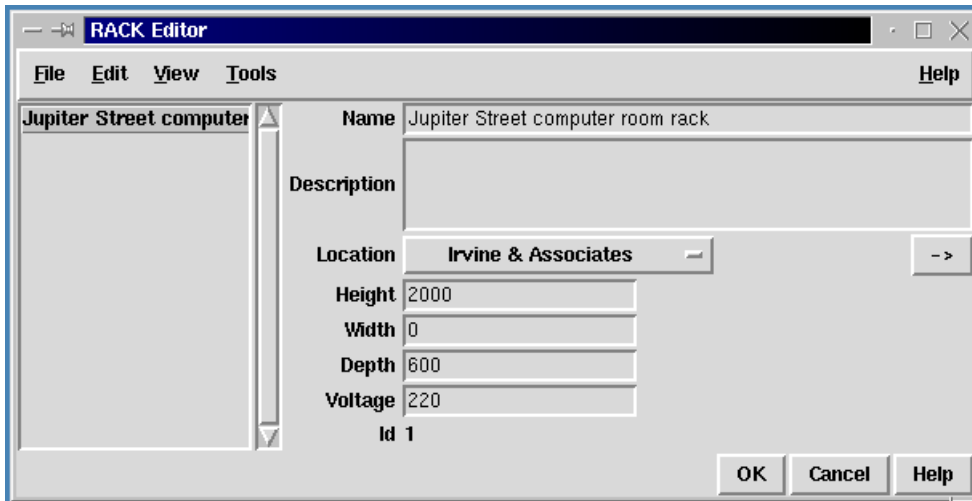
This screen shows the Nets Preference Editor.

### 7.32 Rack (RACK object)

Racks are usually used in locations such as telecommunications facilities and computer rooms to house network equipment. Racks are defined with height, width, depth and available power supply voltages.

---

FIGURE 49. Nets Rack Editor



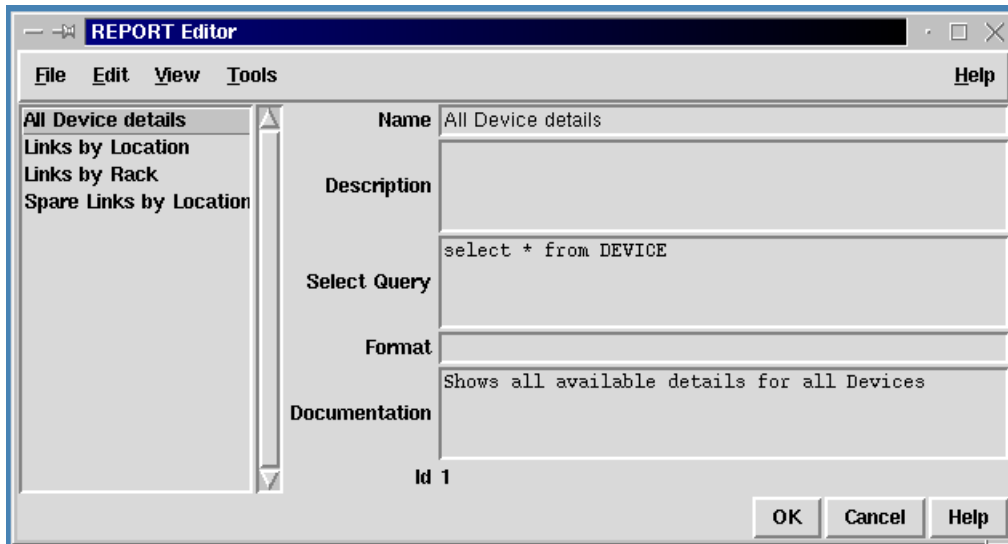
This screen shows the Nets Rack Editor.

### 7.33 Report (REPORT object)

Reports are used to generate useful summaries of the data stored in Nets. There are a set of standard reports included with Nets, however additional reports can be defined at any time as required.

---

FIGURE 50. Nets Report Editor



This screen shows the Nets Report Editor.

The Select Query field is where the SQL query to extract the data from the database for this report is defined. Any SQL query that is supported by the database can be used. Refer to the database documentation for SQL query details and syntax.

The Format field is where the SQL display format is defined for the query specified in the Select Query field above. A Format is only required if the default format is inadequate to display the results of the query defined above.

### 7.34 Slot (SLOT object)

Devices may optionally contain slots, which are defined here.

A device may have any number of slots, and the slots may also contain cards.

---

FIGURE 51. Nets Slot Editor



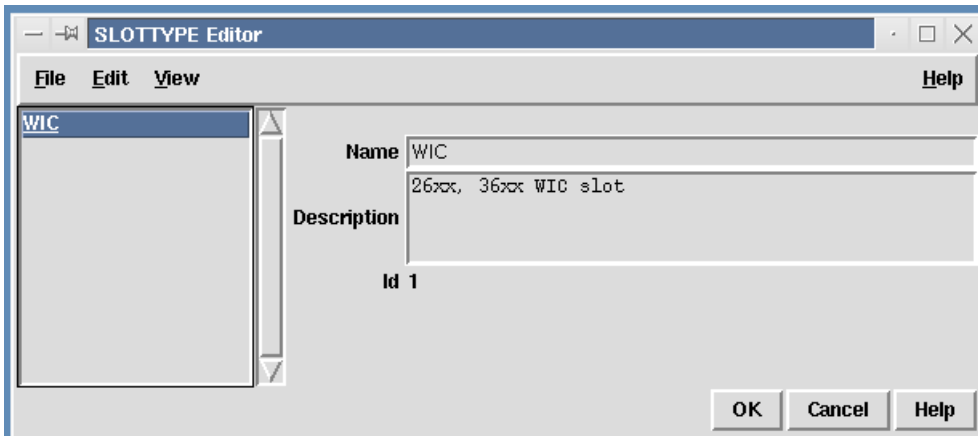
This screen shows the Nets Slot Editor.

### 7.35 Slot Type (SLOTTYE object)

Slots can be of different types. Additional slot types are defined here.

---

FIGURE 52. Nets Slot Type Editor



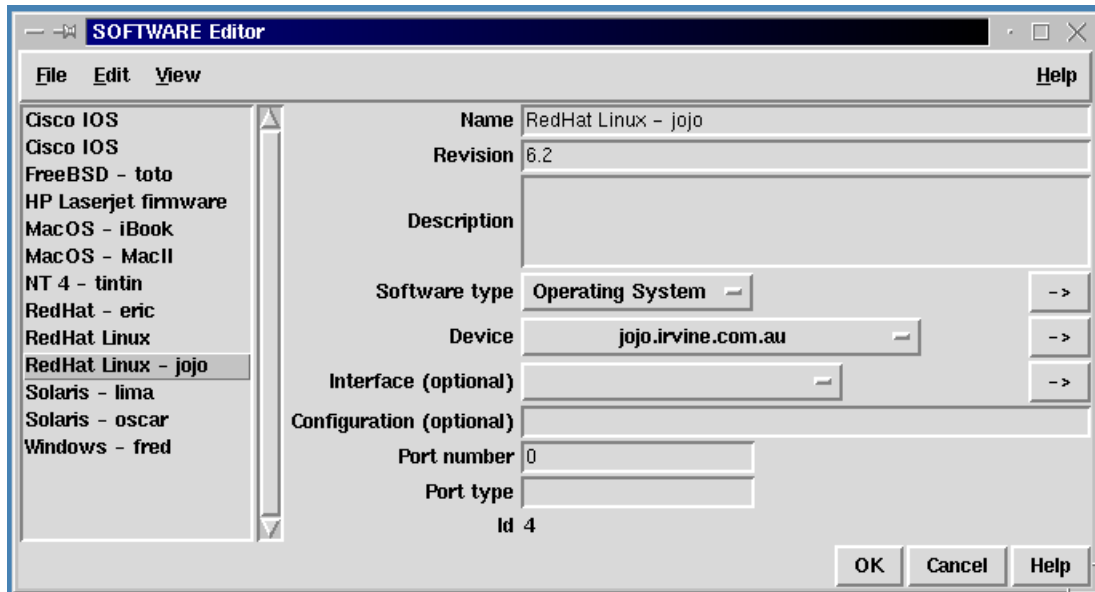
This screen shows the Nets Slot Type Editor.

### 7.36 Software (SOFTWARE object)

A software object is normally an operating system or a software service installed on a device (and optionally related to an interface). In addition, configuration information can be maintained for particular software services.

---

FIGURE 53. Nets Software Editor



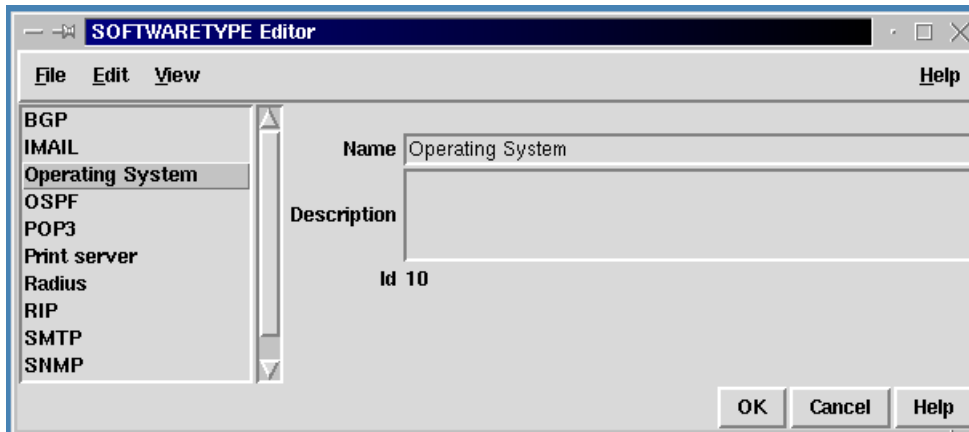
This screen shows the Nets Software Editor.

### 7.37 Software Type (SOFTWARETYPE object)

Software types are used to describe each type of software associated with Nets devices. Additional software types can be added at any time as required.

---

FIGURE 54. Nets Software Type Editor



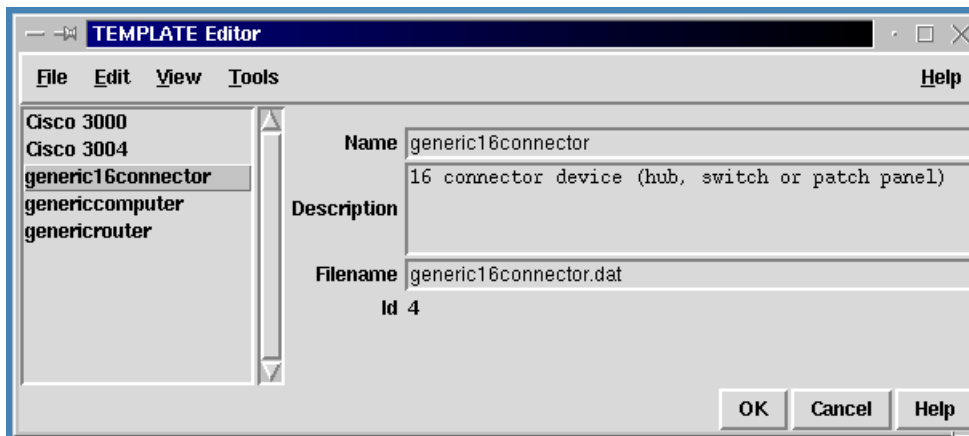
This screen shows the Nets Software Type Editor.

### 7.38 Template (TEMPLATE object)

Templates allow multiple database objects to be configured at the same time. Template files are stored in the <NetsDir/templates> directory and this screen is used to associate a Nets template name with its corresponding source file.

---

FIGURE 55. Nets Template Editor



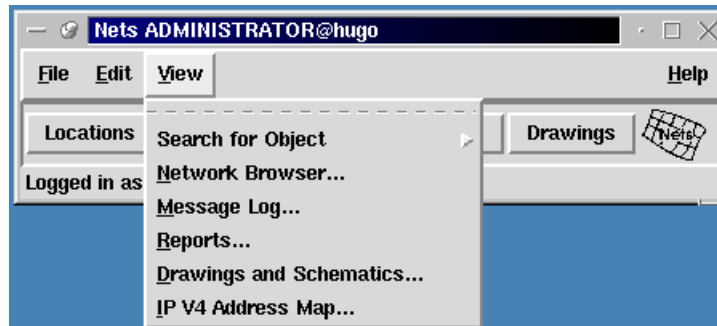
This screen shows the Nets Template Editor.

## 8.0 Nets View Menu

---

The Nets View Menu allows a user to search for a variety of Nets objects, to display the Nets Network Browser, to display the Message Log, to display Reports, to display Drawings and Schematics and finally to display IP V4 Address Maps.

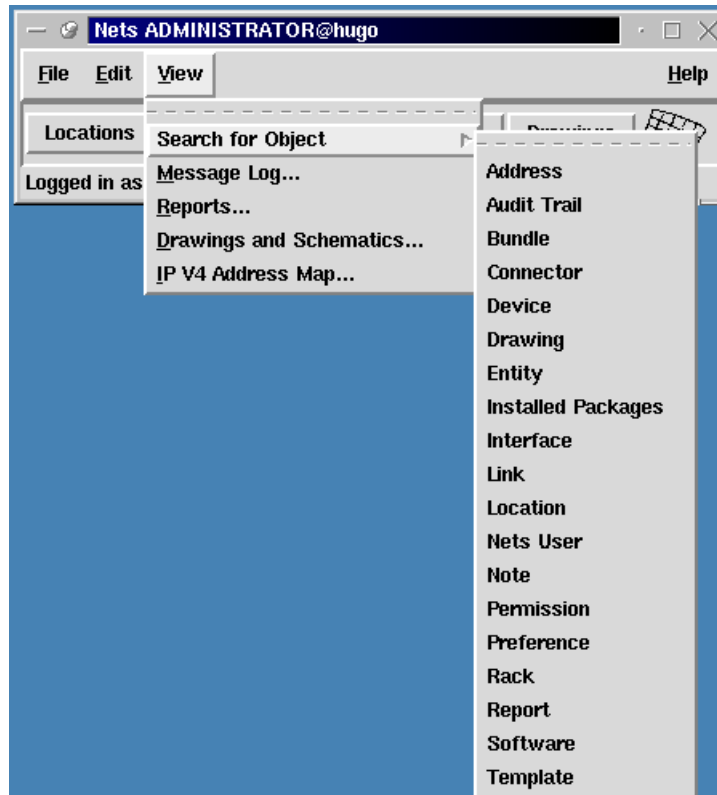
**FIGURE 56.** Nets Main Screen View Menu



This screen shows the Nets View Menu.

FIGURE 57.

Nets Object Listers

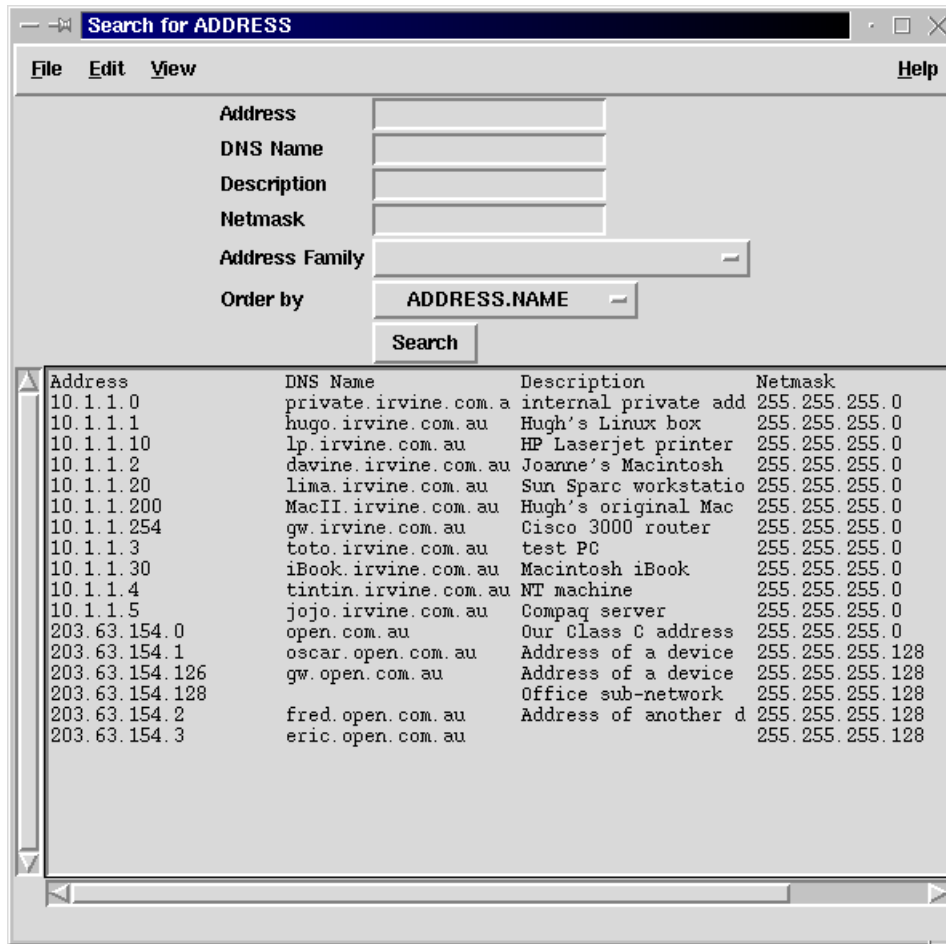


This screen shows the list of Nets objects that can be searched on and listed.

### 8.1 Address (ADDRESS object)

The Address screen allows searches to be performed on addresses stored in Nets. Double-clicking on any address will display that address in the address editor.

FIGURE 58. Nets Address Lister

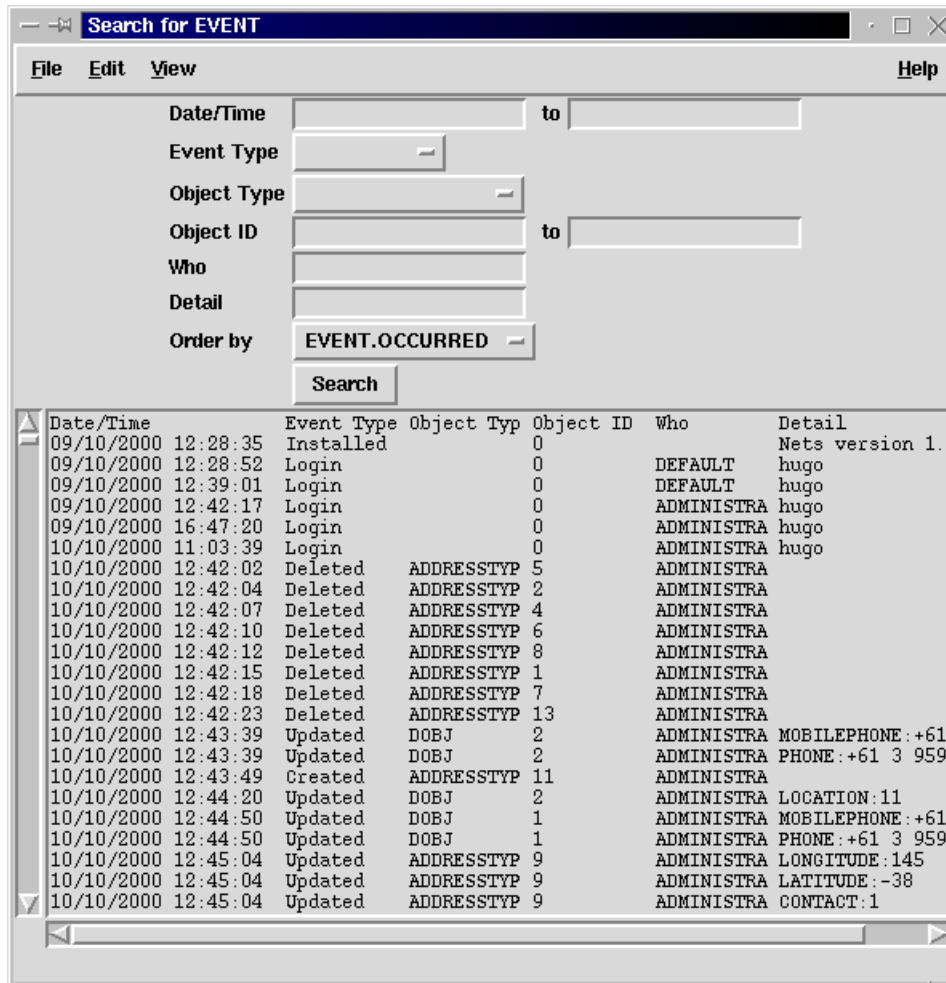


This screen shows the Nets Address Lister.

### 8.2 Audit Trail (EVENT object)

The Nets audit trail contains a list of every login, database operation and administrative action to allow review of any or all Nets activity.

FIGURE 59. Nets Audit Trail Lister



This screen shows the Nets Audit Trail Lister.

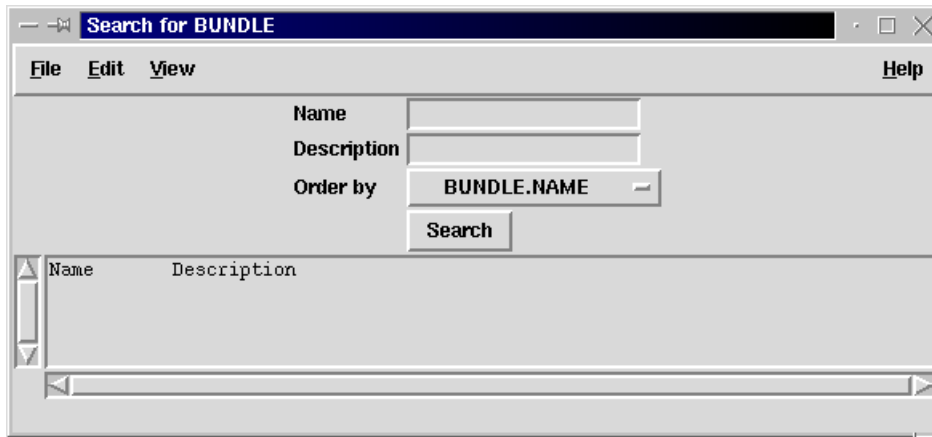
**Note:** the Detail column in the display above shows a brief description of each audit trail item (the event that occurred to cause this entry to be made in the audit trail).

### 8.3 Bundle (BUNDLE object)

Cable bundles stored in Nets can be searched for and/or listed with this screen.

---

**FIGURE 60.** Nets Bundle Lister



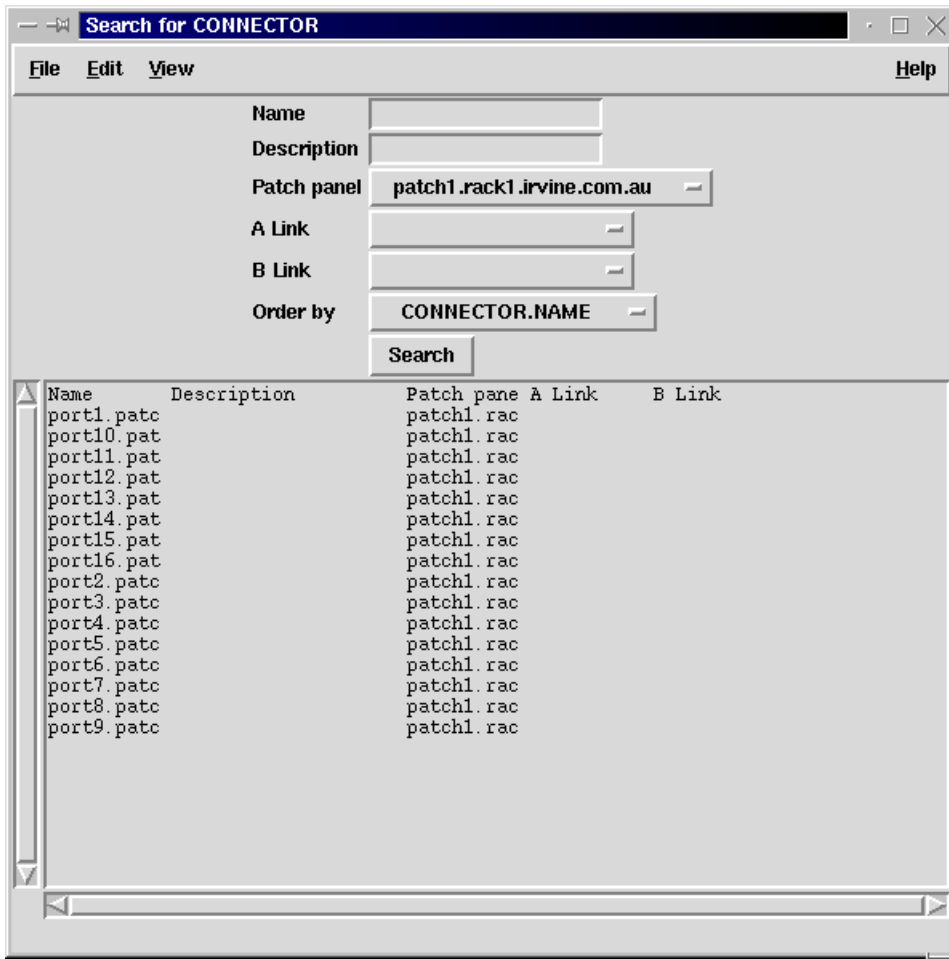
This screen shows the Nets Bundle Lister.

### 8.4 Connector (CONNECTOR object)

This screen can be used to search and/or list the connectors stored in Nets.

---

FIGURE 61. Nets Connector Lister



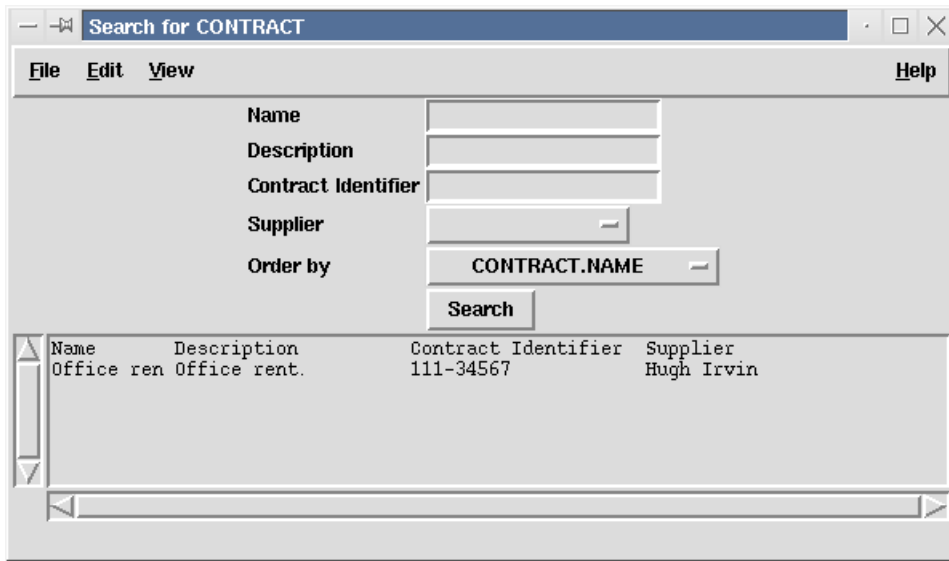
This screen shows the Nets Connector Lister.

### 8.5 Contract (CONTRACT object)

This screen can be used to search and/or list the contracts stored in Nets.

---

**FIGURE 62.** Nets Contract Lister



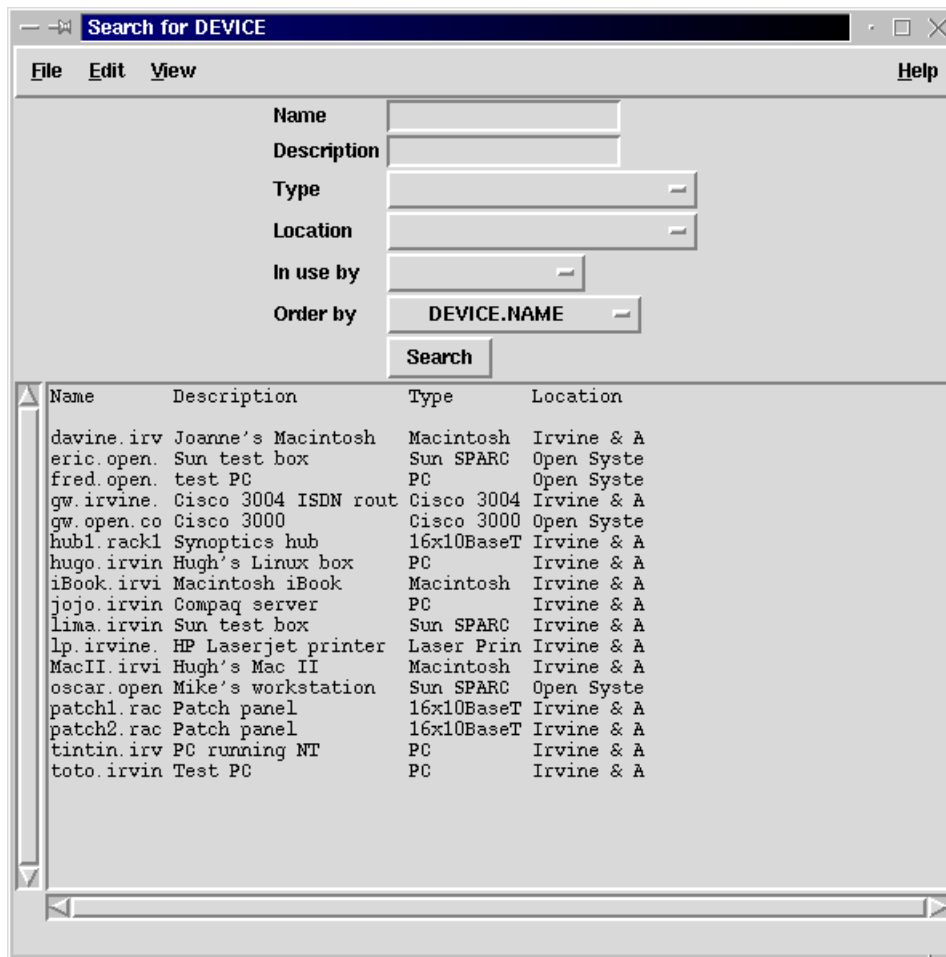
This screen shows the Nets Contract Lister.

### 8.6 Device (DEVICE object)

This screen is used to search and/or list the devices stored in Nets.

---

FIGURE 63. Nets Device Lister



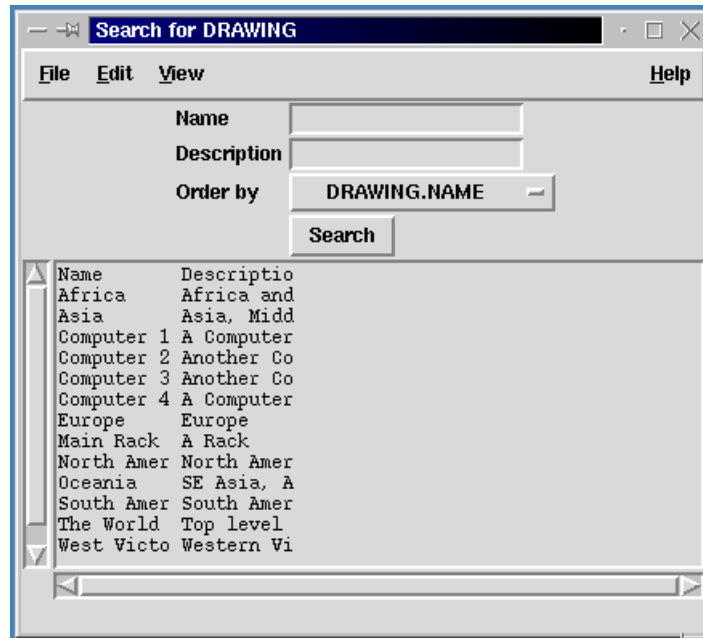
This screen shows the Nets Device Lister.

### 8.7 Drawing (DRAWING object)

This screen allows searching and/or listing the drawings stored in Nets.

---

**FIGURE 64.** Nets Drawing Lister



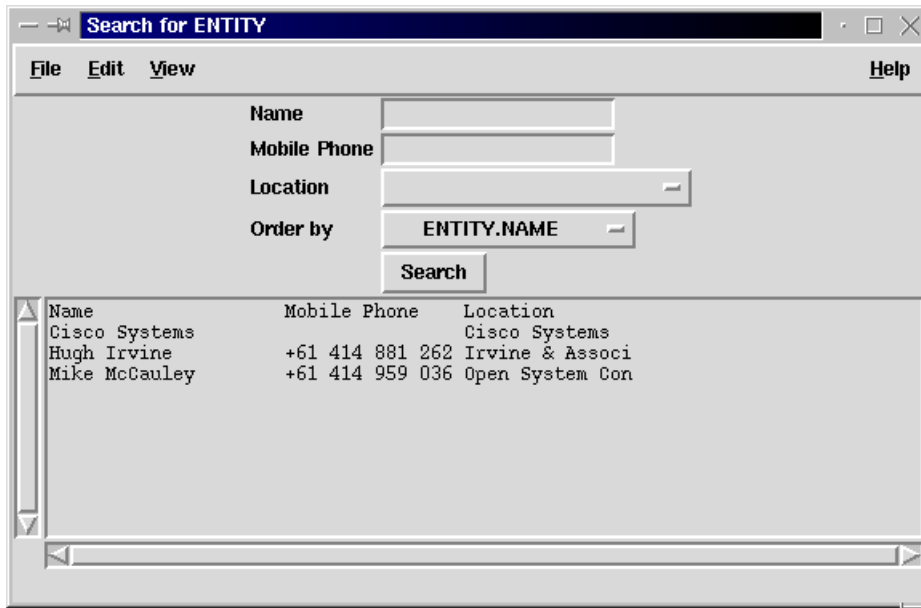
This screen shows the Nets Drawing Lister.

### 8.8 Entity (ENTITY object)

This screen allows searching and/or listing the entities stored in Nets.

---

FIGURE 65. Nets Entity Lister

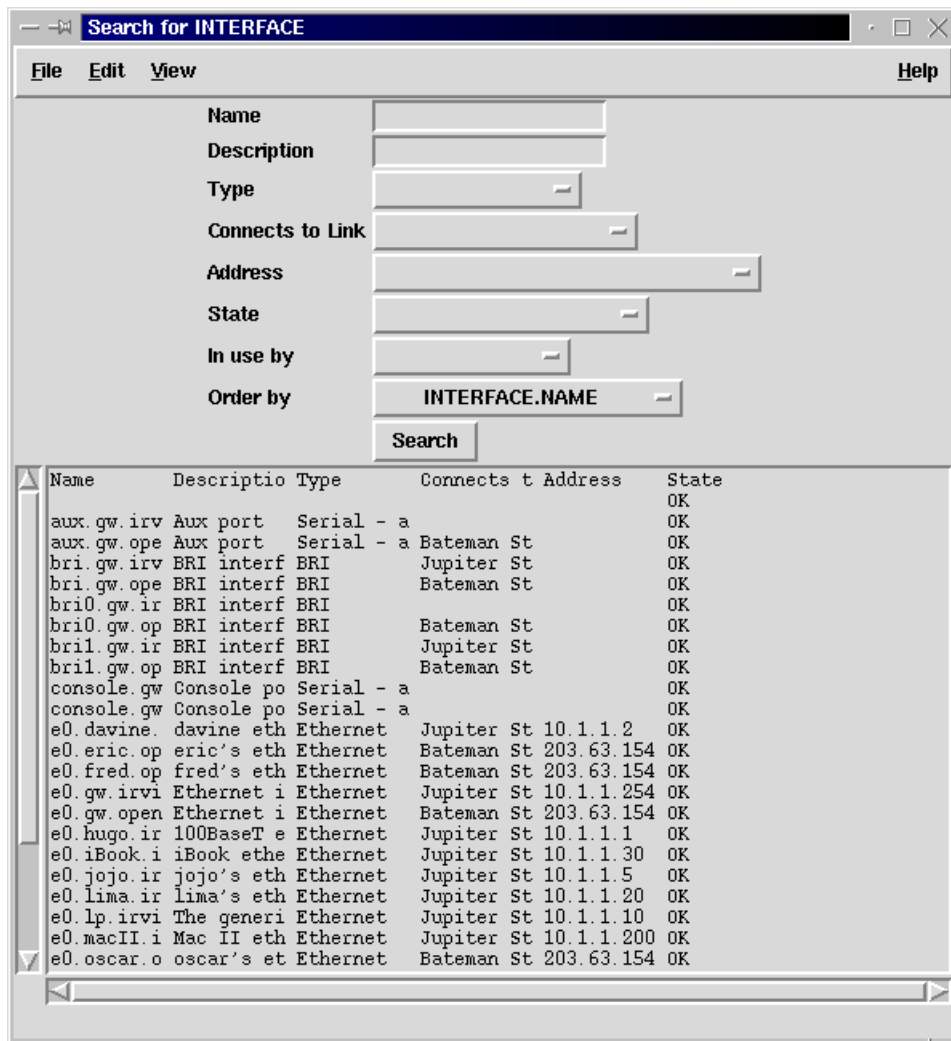


This screen shows the Nets Entity Lister.

### 8.9 Interface (INTERFACE object)

This screen allows searches and/or listing the interfaces stored in Nets.

FIGURE 66. Nets Interface Lister



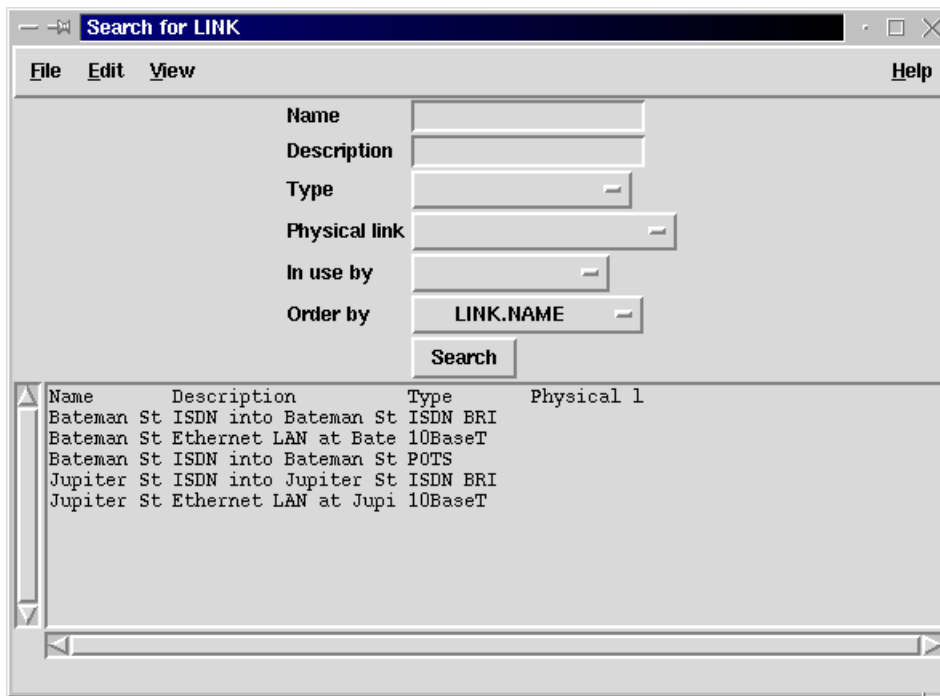
This screen shows the Nets Interface Lister.

### 8.10 Link (LINK object)

This screen allows the searching and/or listing of links stored in Nets.

---

FIGURE 67. Nets Link Lister



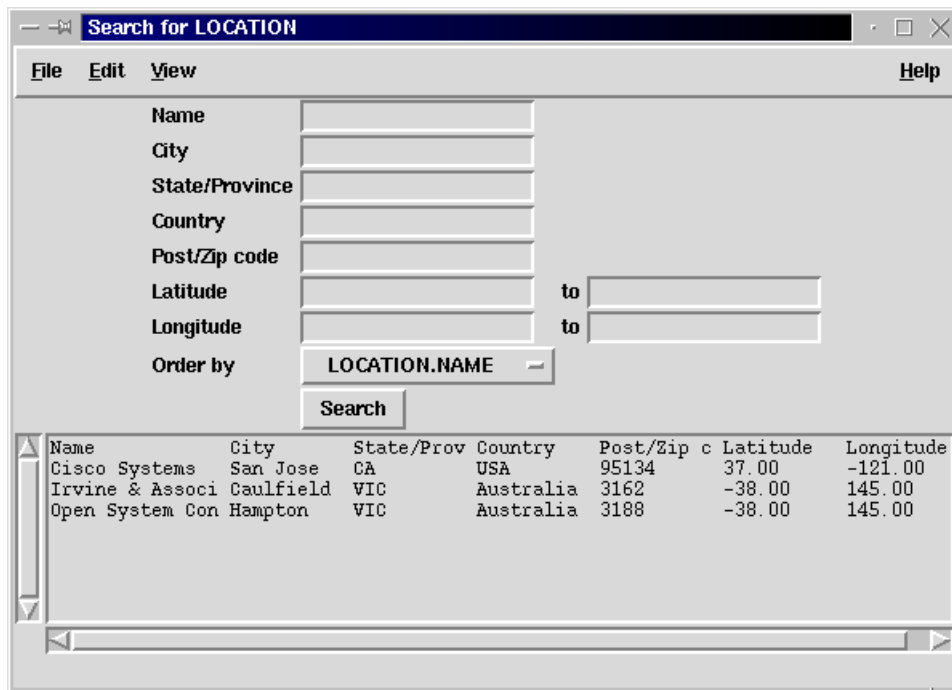
This screen shows the Nets Link Lister.

### 8.11 Location (LOCATION object)

This screen allows the searching and/or listing of locations stored in Nets.

---

FIGURE 68. Nets Location Lister



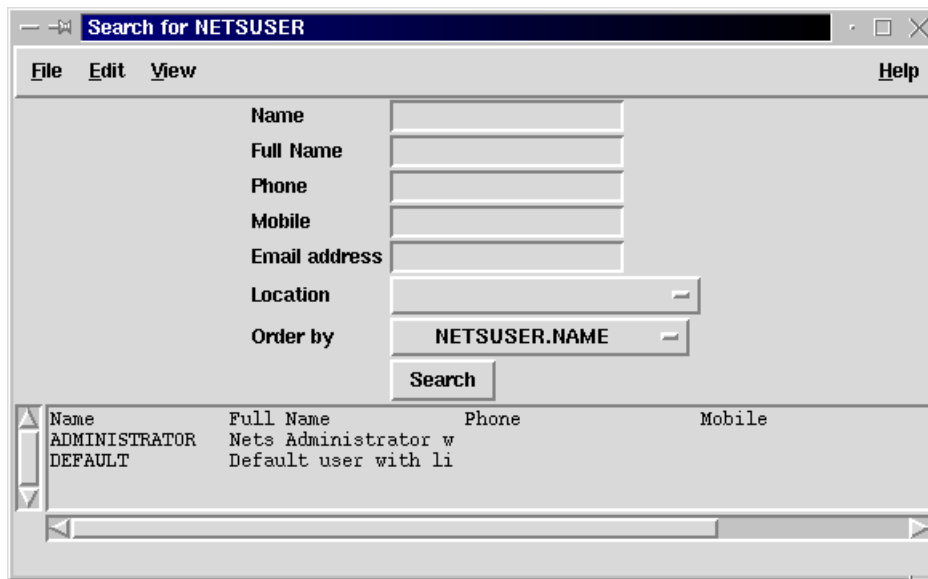
This screen shows the Nets Location Lister.

### 8.12 Nets User (NETSUSER object)

This screen allows the searching and/or listing of Nets users.

---

FIGURE 69. Nets User Lister



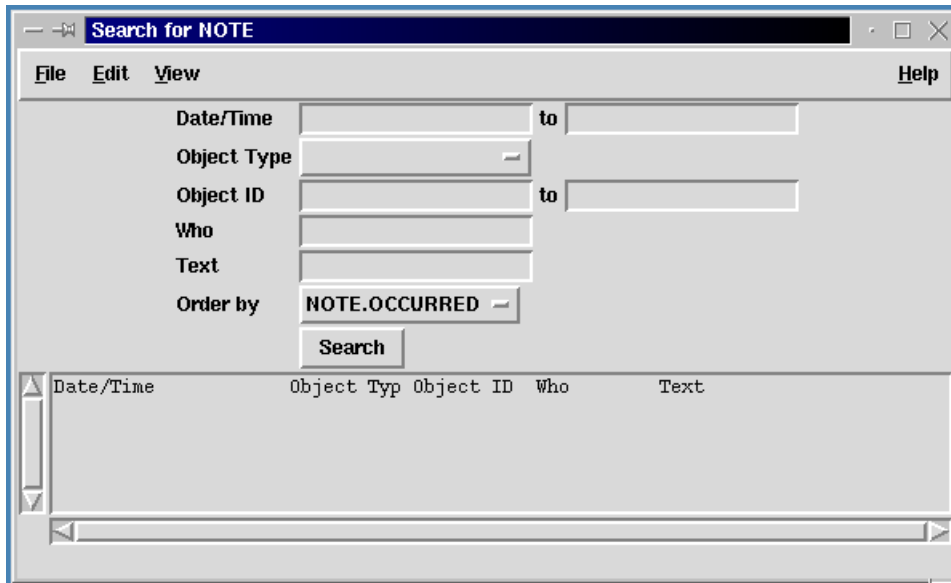
This screen shows the Nets User Lister.

### 8.13 Note (NOTE object)

This screen allows the searching and/or listing of notes stored in Nets.

---

**FIGURE 70.** Nets Note Lister



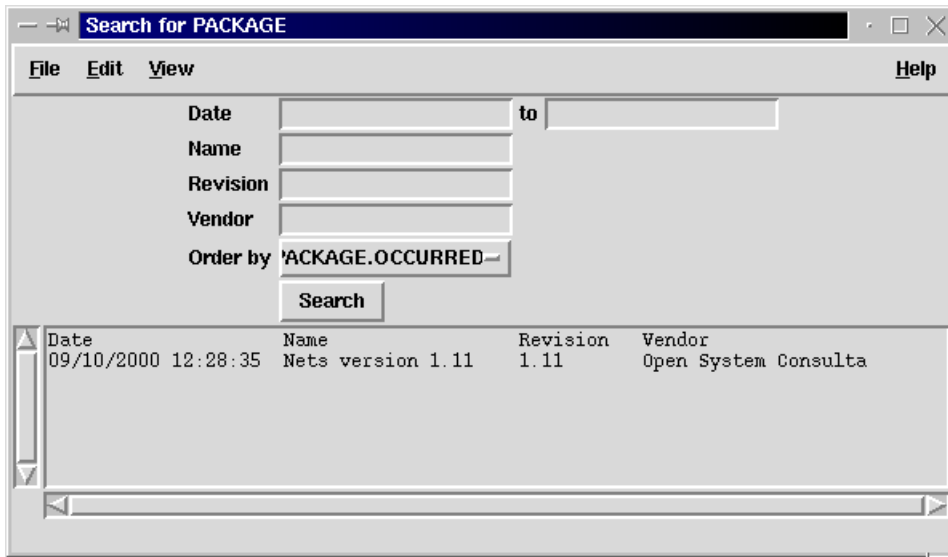
This screen shows the Nets Note Lister.

### 8.14 Package (PACKAGE object)

This screen allows searching and/or listing the packages installed in Nets.

---

FIGURE 71. Nets Package Lister

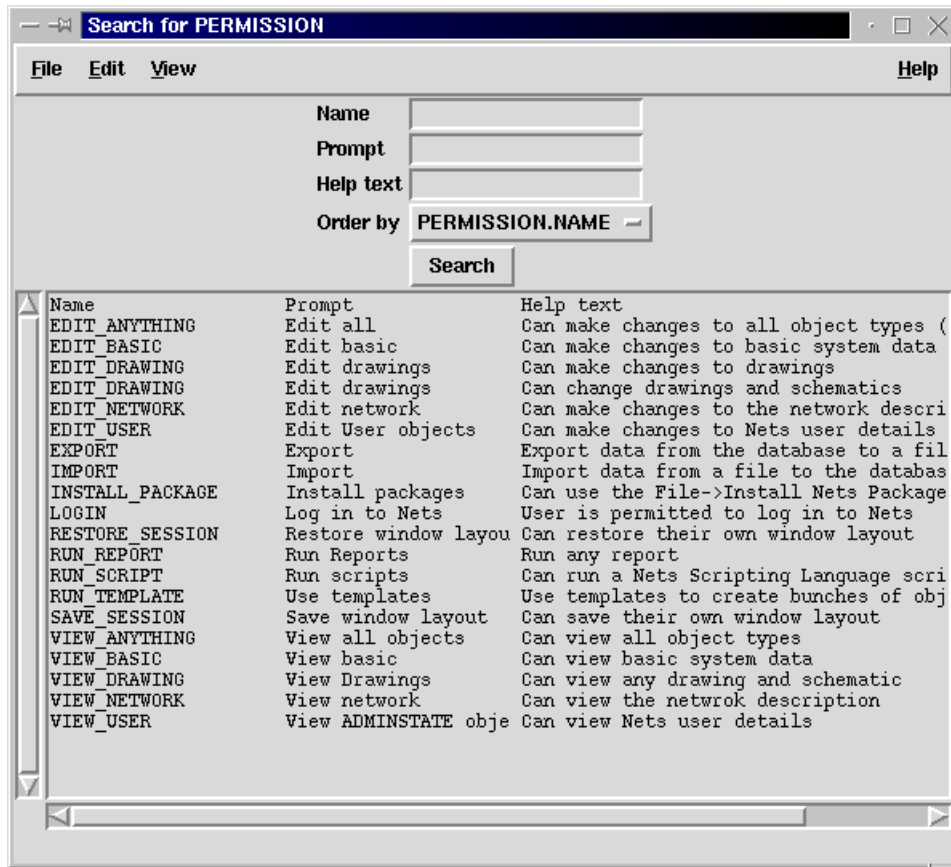


This screen shows the Nets Package Lister.

8.15 Permission (PERMISSION object)

This screen allows the searching and/or listing of the user permissions in Nets.

FIGURE 72. Nets Permission Lister

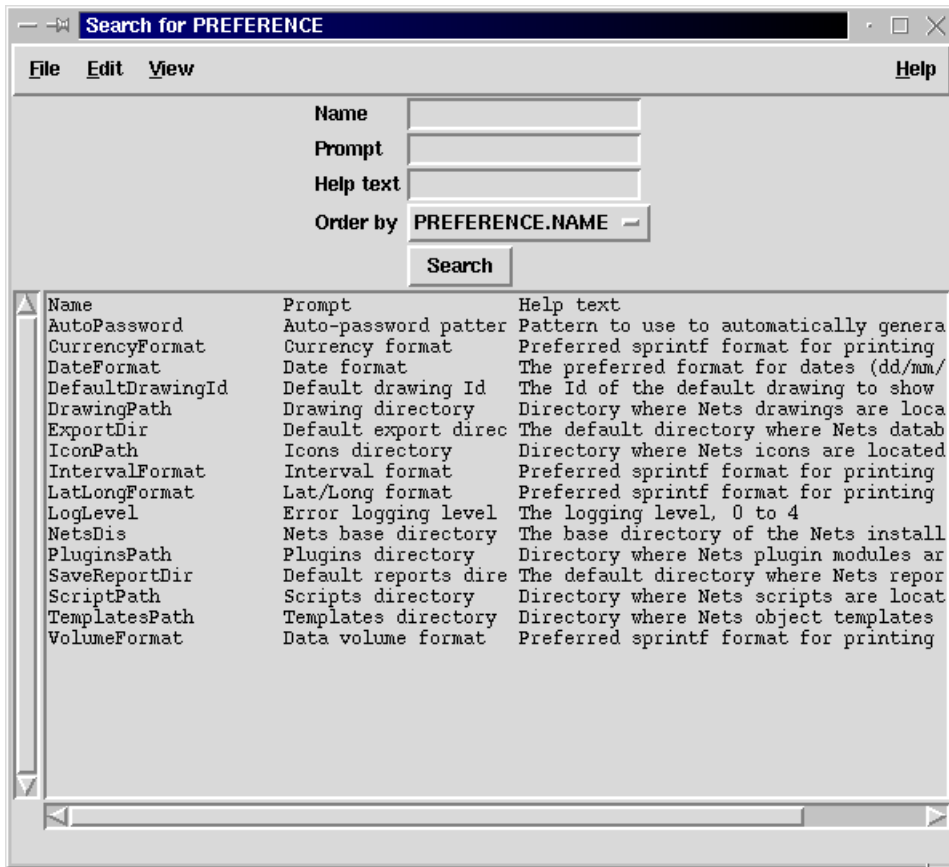


This screen shows the Nets Permission Lister.

### 8.16 Preference (PREFERENCE object)

This screen allows the searching and/or listing of Nets preferences.

FIGURE 73. Nets Preference Lister



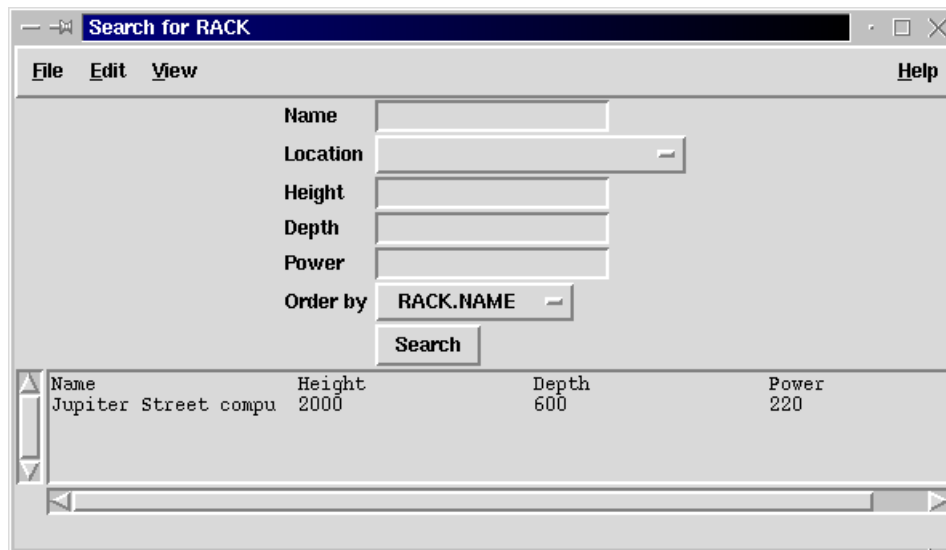
This screen shows the Nets Preference Lister.

### 8.17 Rack (RACK object)

This screen allows the searching and/or listing of the racks stored in Nets.

---

FIGURE 74. Nets Rack Lister



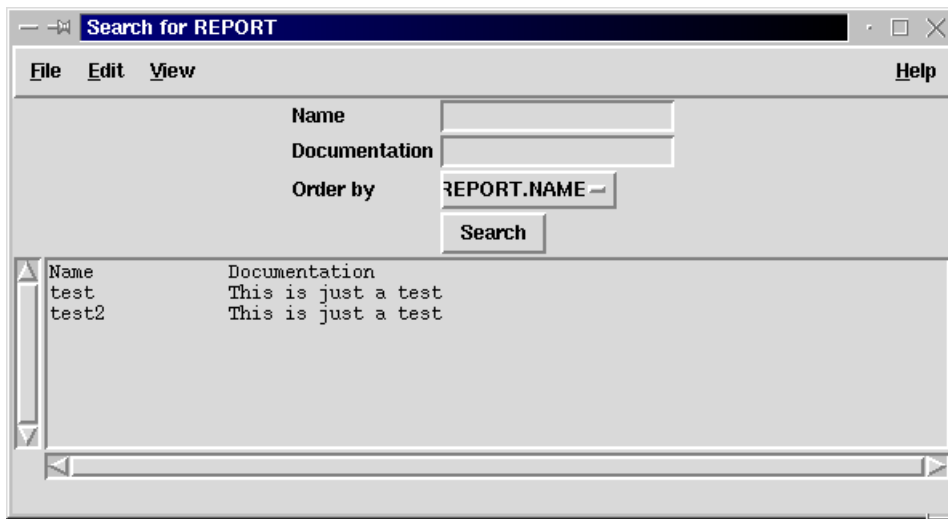
This screen shows the Nets Rack Lister.

### 8.18 Report (REPORT object)

This screen allows the searching and/or listing of the reports used in Nets.

---

FIGURE 75. Nets Report Lister

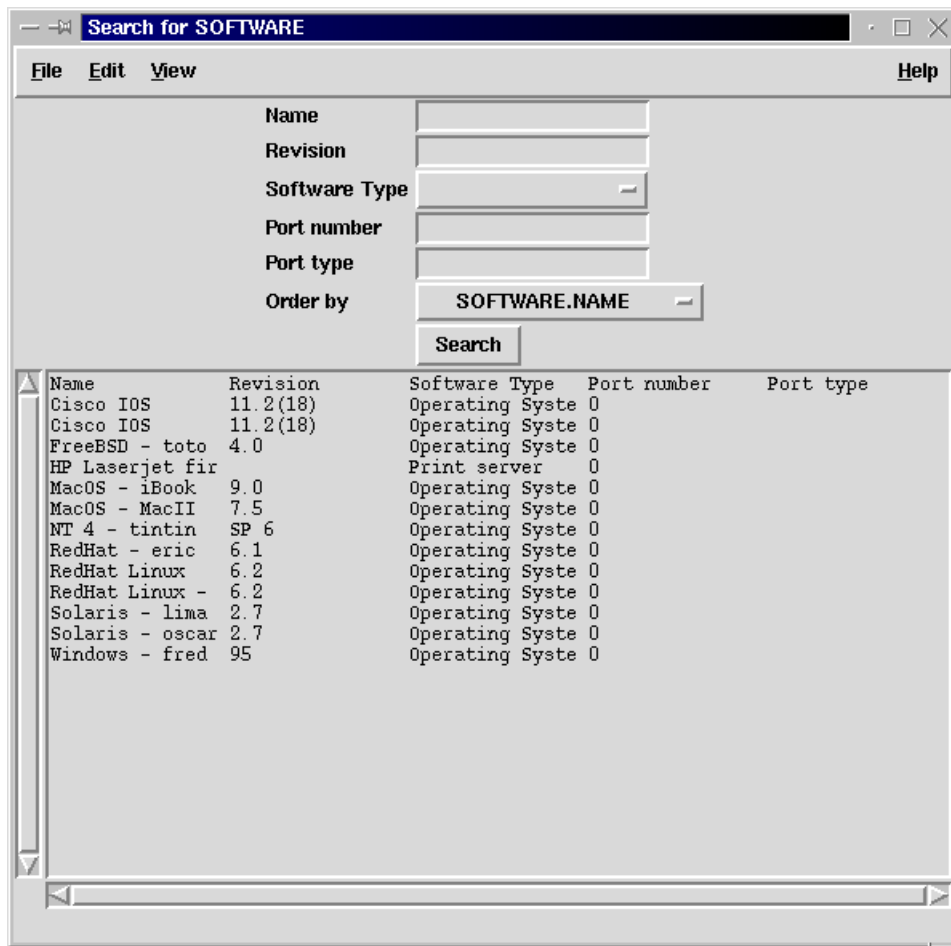


This screen shows the Nets Report Lister.

### 8.19 Software (SOFTWARE object)

This screen allows the searching and/or listing of software descriptions stored in Nets.

FIGURE 76. Nets Software Lister



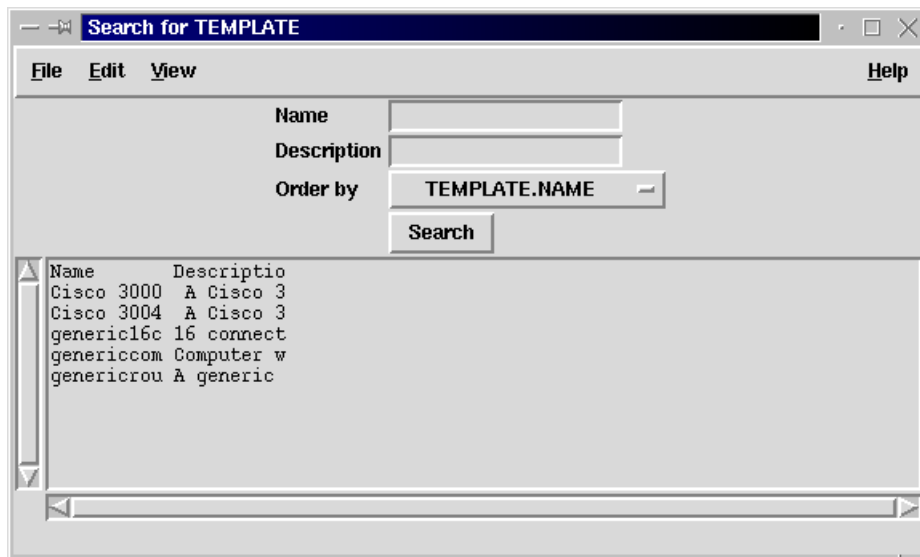
This screen shows the Nets Software Lister.

## 8.20 Template (TEMPLATE object)

This screen allows the searching and/or listing of Nets templates.

---

FIGURE 77. Nets Template Lister



This screen shows the Nets Template Lister.

### 8.21 Network Browser

The network browser provides an alternative means for perusing, displaying and searching for objects in the Nets database, in a hierarchical manner similar to a graphical file manager.

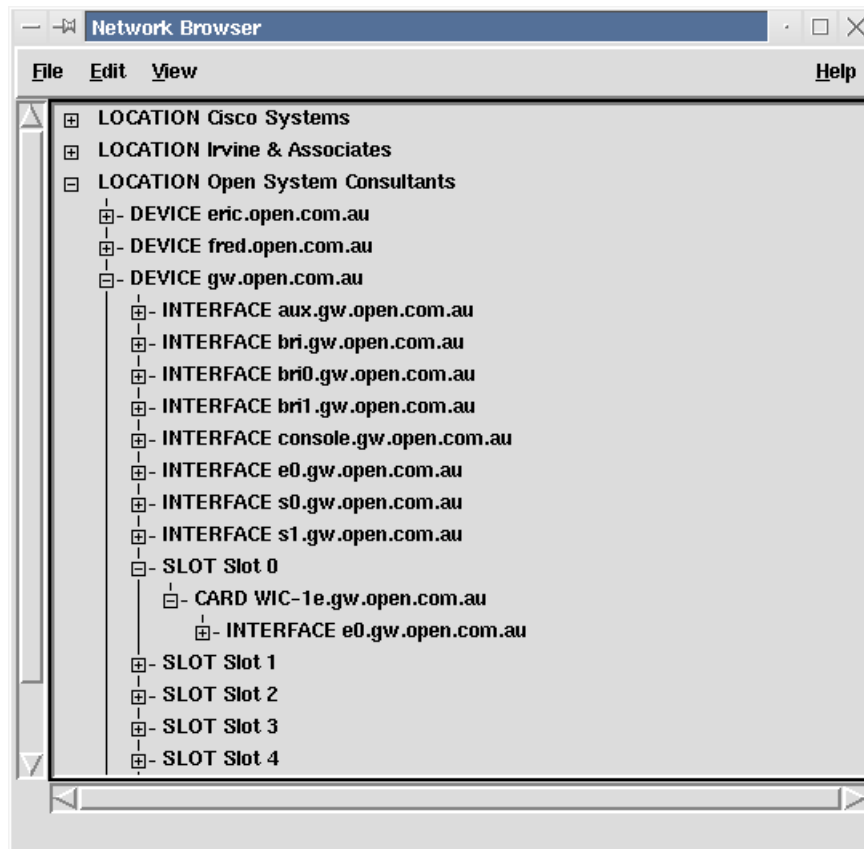
At the top level, the network browser displays all of the locations stored in the database with an expander box (shown with a “+”) next to it. Clicking on the expander box will display all of the objects related to the expanded object.

For example, expanding a location will display all of the devices and racks at that location, expanding a rack will display all of the devices in that rack, and expanding a device will display all of the slots, cards, interfaces and connectors in that device.

Double-clicking on any line in the network browser display will display the database object that the line corresponds to in the appropriate object editor.

---

**FIGURE 78.** Nets Network Browser



This screen shows the Nets Network Browser.

## 8.22 Message Log

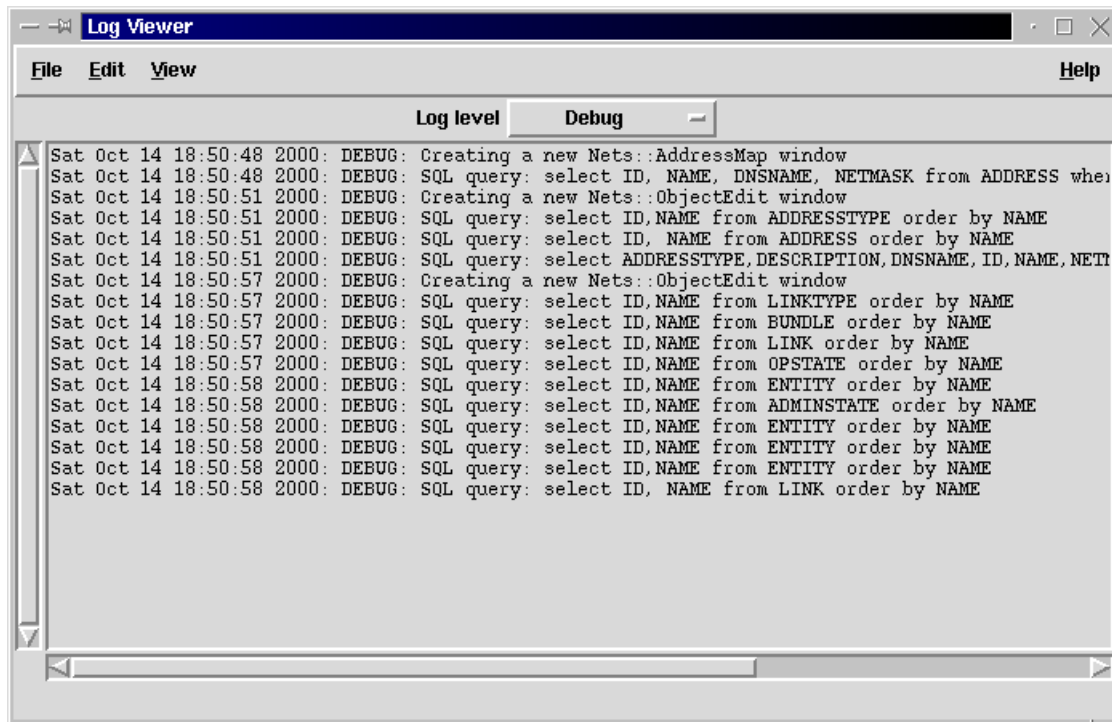
This screen allows the Nets message log to be viewed in real time. The message log shows the messages that Nets displays during operation. Messages are displayed at each of five log levels:

- Serious errors - only serious error conditions are logged.
- Warnings - warning conditions are logged.
- Notices - normal but significant conditions are logged.
- Info - informational messages are logged.
- Debug - detailed debugging information is logged.

**Note:** the message log level to be displayed can be altered by selecting the relevant item in the Log level option menu.

---

FIGURE 79. Nets Message Log Viewer



This screen shows the Nets Message Log.

### 8.23 Reports

This screen allows access to Nets reports. Nets is delivered with a number of standard reports included, and additional reports can be defined as required.

A report is selected by using the View->Reports menu item.

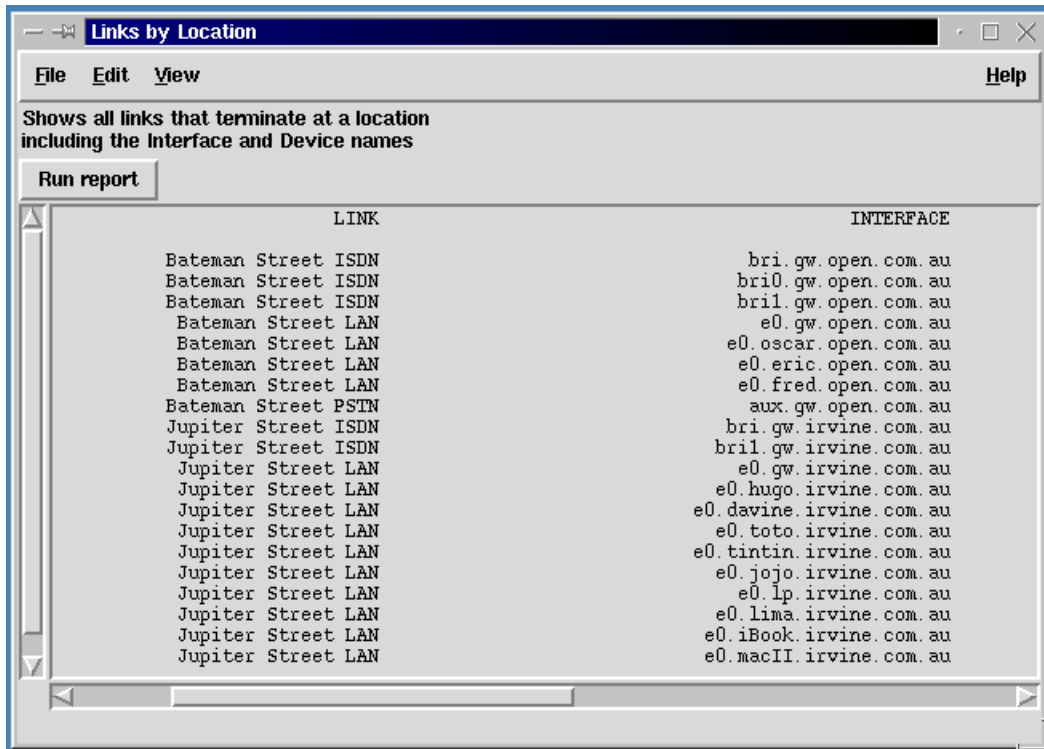
The standard Nets reports include:

- All Device details - show details of all devices.
- Links by Location - show all links by location.
- Spare Links by Location - show spare links by location.
- Links by Rack - show all links by rack.

**Note:** report results may be saved to a file with File->Save Results As...

---

FIGURE 80. Nets Reports



This screen shows the Nets report for all links by location.

## 8.24 Drawings and Schematics

This screen allows access to the drawings and schematics stored in Nets.

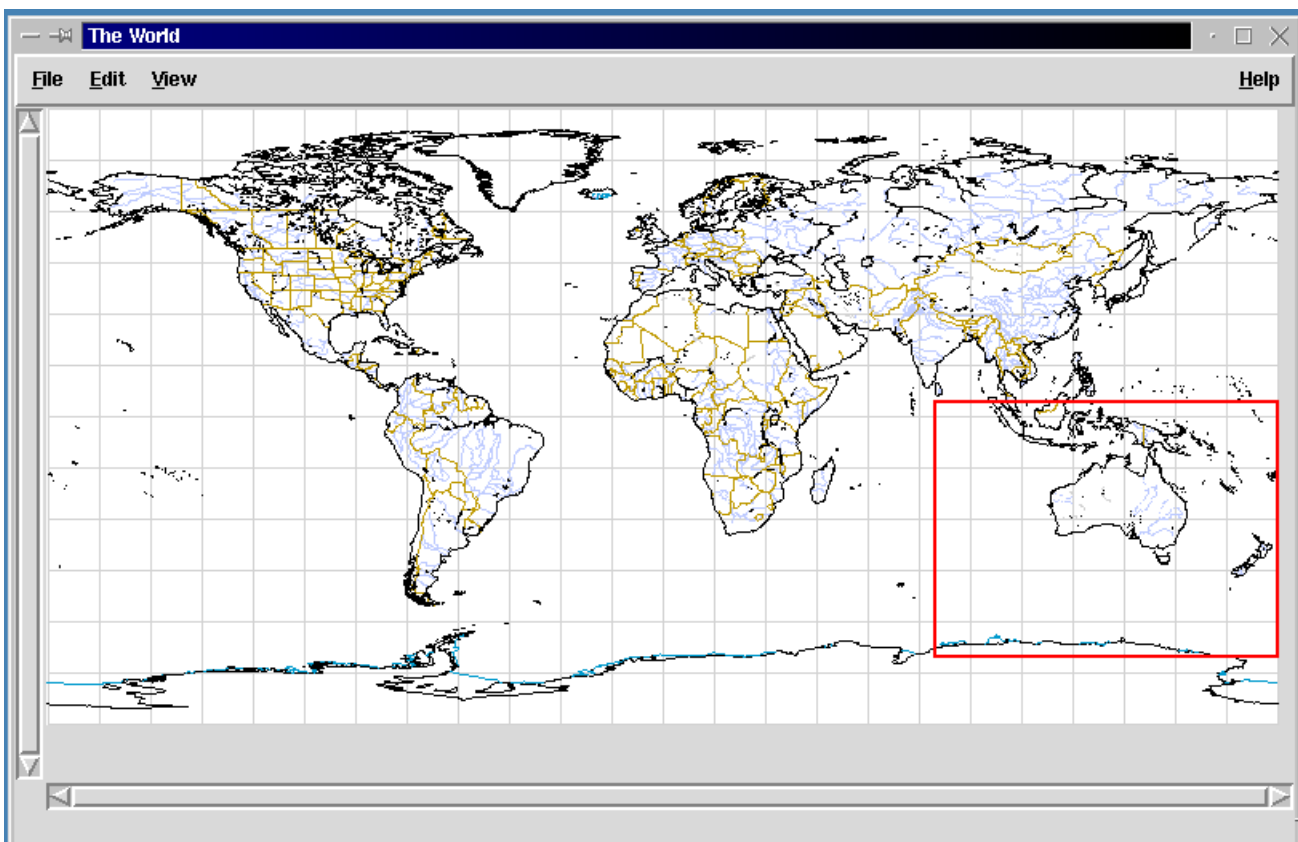
Drawings are used to display Nets network information in a variety of graphical formats. Drawings consist of backgrounds such as maps, schematic diagrams, scanned images, digital photographs, etc, and foreground objects such as drill-down areas, icons, text, lines, etc.

Foreground objects will be highlighted when the mouse pointer hovers over an area or an icon is clicked. Double-clicking an area or an icon will display the Nets object that is associated with it, including other drawings. Right-clicking an area or icon will show the meta-data associated with the corresponding drawing object (DOBJ).

The Drawings shortcut button in the Nets Main Screen also displays this screen.

---

**FIGURE 81.** Nets Drawings and Schematics



Refer to Section 9.0 on page 94 for more details regarding drawings and schematics.

### 8.25 IP V4 Address Map

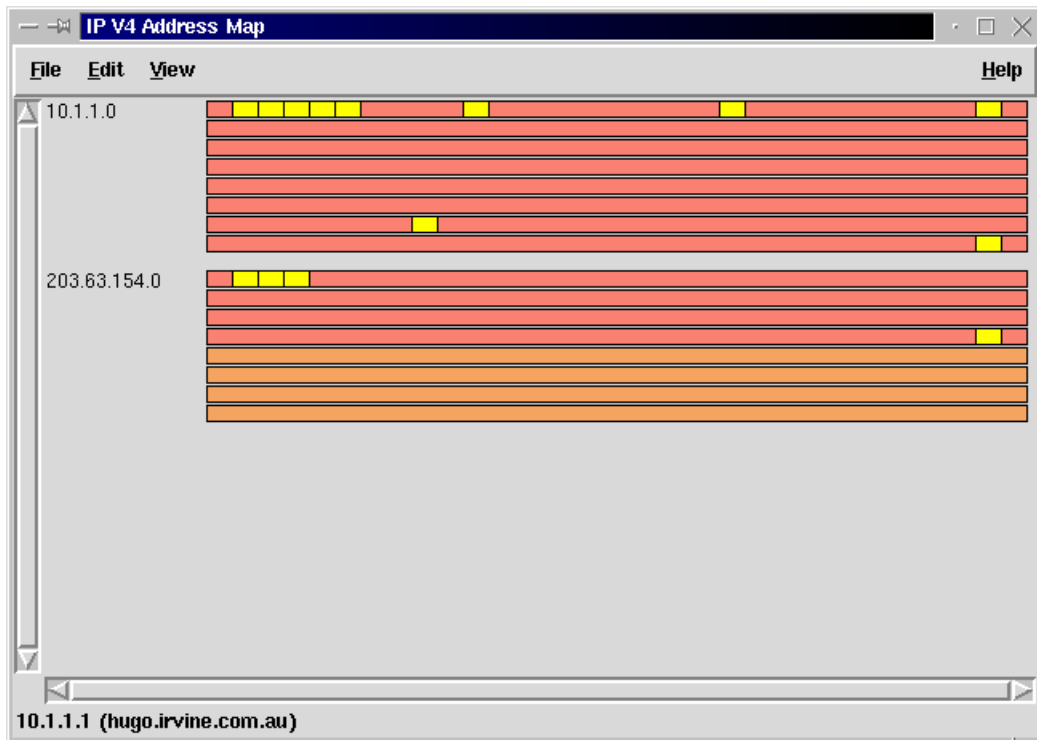
The IP V4 address map displays all CIDR (classless inter-domain routing) blocks, subnets and host addresses stored in the Nets IP V4 address table. Different size address blocks, subnets and host addresses are all displayed in different colours to aid in visualization.

CIDR blocks are displayed first, with subnets (defined by netmask) displayed on top of the CIDR blocks, and host addresses displayed on top of subnets. CIDR blocks and subnets are IP addresses for which the netmask part is all zero (ie. 10.1.1.0 and 255.255.255.0, or 10.1.1.0/24). Host addresses are IP addresses with a non-zero value in the subnet portion of the address (ie. 10.1.1.1 and 255.255.255.0 or 10.1.1.1/24).

Single-clicking an address or address block will display the address (and the associated DNS name) in the bottom left corner of the window. Double-clicking on any address or address block will display that address in the address editor.

---

**FIGURE 82.** Nets IP V4 Address Map



This screen shows the Nets IP V4 Address Map.

---

## **9.0 Drawings and Icons**

---

Nets supports a wide variety of graphical representations of database objects. Any image in GIF, XBM or XPM format can be used directly and additional image formats can be supported by adding the appropriate image module to the underlying Nets graphics sub-system (see the Nets Installation and Administration Guide for details).

A standard set of maps and icons is provided and additional images can be added as required. Images can be prepared in any external graphics program and imported into Nets using the Drawing editor as described below. Images can also be scanned maps, drawings and schematics or photographs imported directly from a digital camera.

### **9.1 Drawings and Drawing Objects**

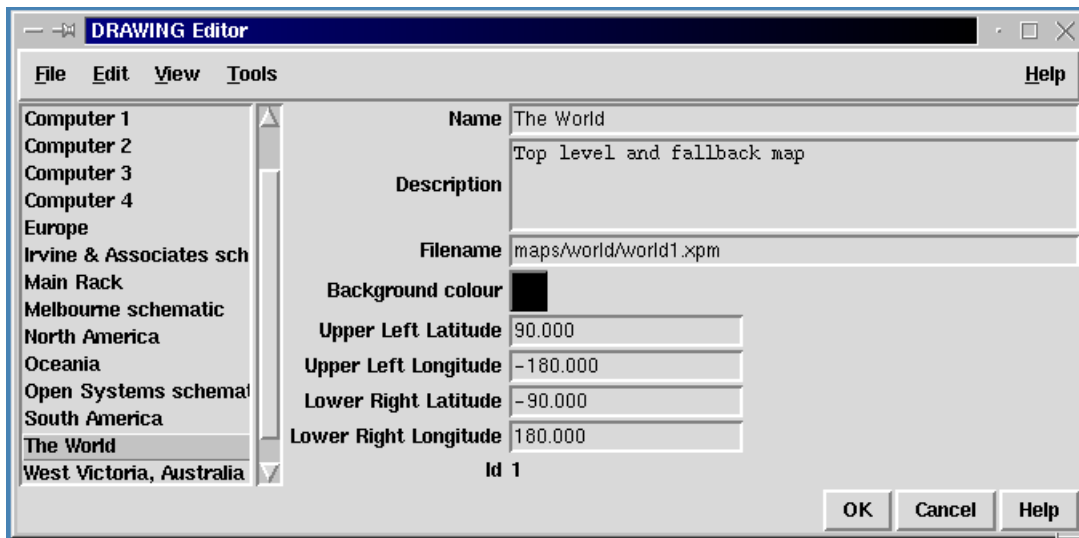
There are two fundamental elements that are used in Nets drawings, the drawings themselves (DRAWING objects) which can be considered as backgrounds, and the objects (DOBJ objects) that are displayed on those backgrounds.

## 9.2 Creating a New Drawing

A new drawing is created in the Drawing editor.

---

**FIGURE 83.** The Nets Drawing Editor



This example shows one of the drawings that is supplied in the Nets distribution.

The following information is used for each drawing:

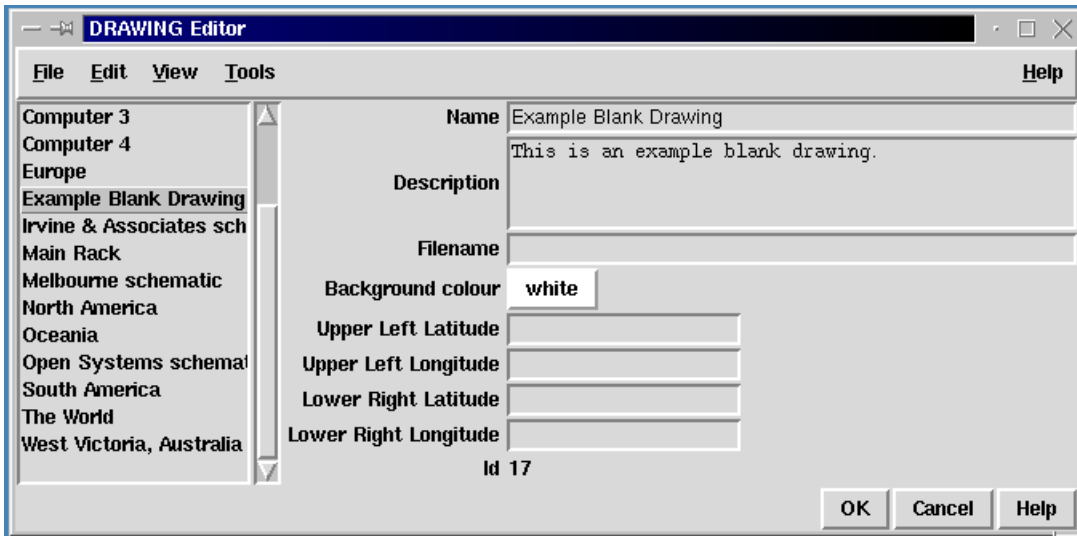
- Name - a name used to refer to this drawing in Nets
- Description - a brief description of this drawing
- Filename - the filename that contains the actual image (optional)
- Background colour - defines the background colour of a blank drawing (optional)
- Upper Left Latitude - used to define the upper left corner of a map (optional)
- Upper Left Longitude - used to define the upper left corner of a map (optional)
- Lower Right Latitude - used to define the lower right corner of a map (optional)
- Lower Right Longitude - used to define the lower right corner of a map (optional)

If latitudes and longitudes are provided on map backgrounds, locations that have been defined with latitudes and longitudes will be positioned automatically when added to the map with "New Object" described below.

If the Filename field is defined as the name of an image file, the contents of the file will be used as the background for the new drawing.

If the Filename field is left empty, a drawing with a blank background will be created.

FIGURE 84. Creating a Drawing with a Blank Background

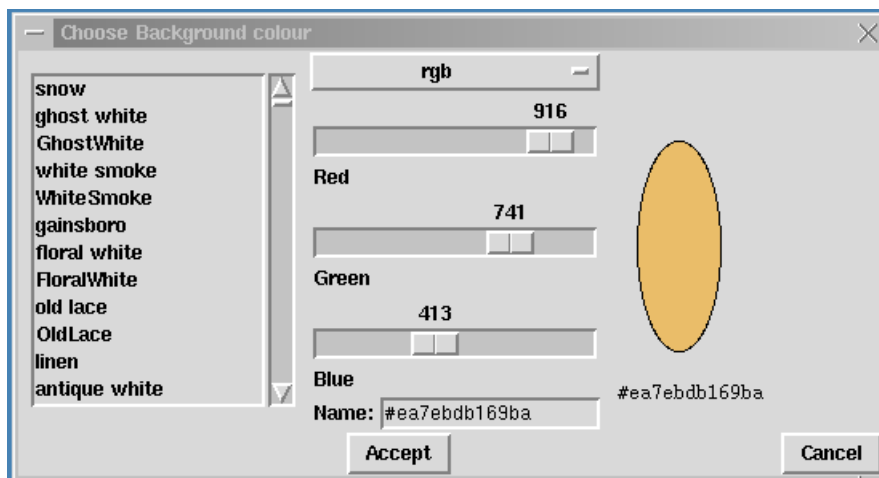


This screen shows an example blank background being created.

To change the background colour, click the the Background colour selector.

---

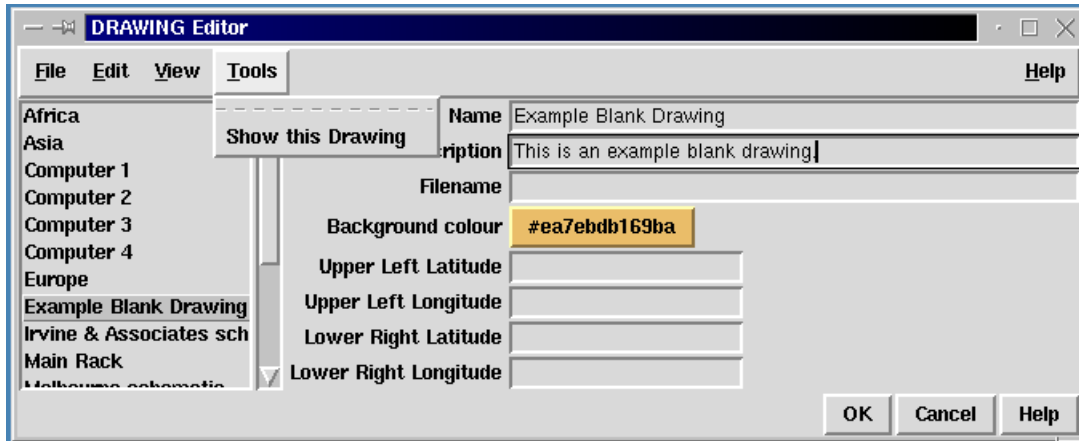
FIGURE 85. Background Colour Selector



Background colours can either be selected from the list in the left hand window, or defined directly by manipulating the Red, Green and Blue sliders.

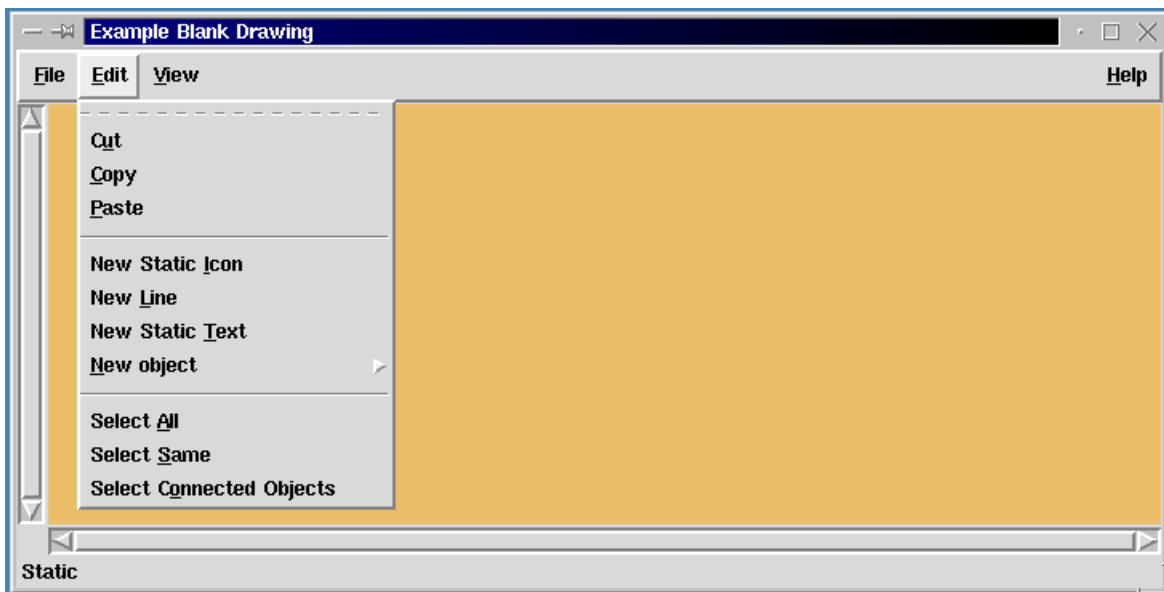
Once a drawing has been created and saved to the database with the OK button, it can be selected and displayed with the Tools->Show this Drawing menu item.

FIGURE 86. Display the Drawing



Drawing objects such as icons, lines, text, etc. can then be added to the drawing.

FIGURE 87. Add and Manipulate Objects



This screen shows the Edit Menu that is used to add drawing objects to a drawing.

### 9.3 Creating Drawing Objects

Drawing objects are the graphical elements that are displayed on drawings.

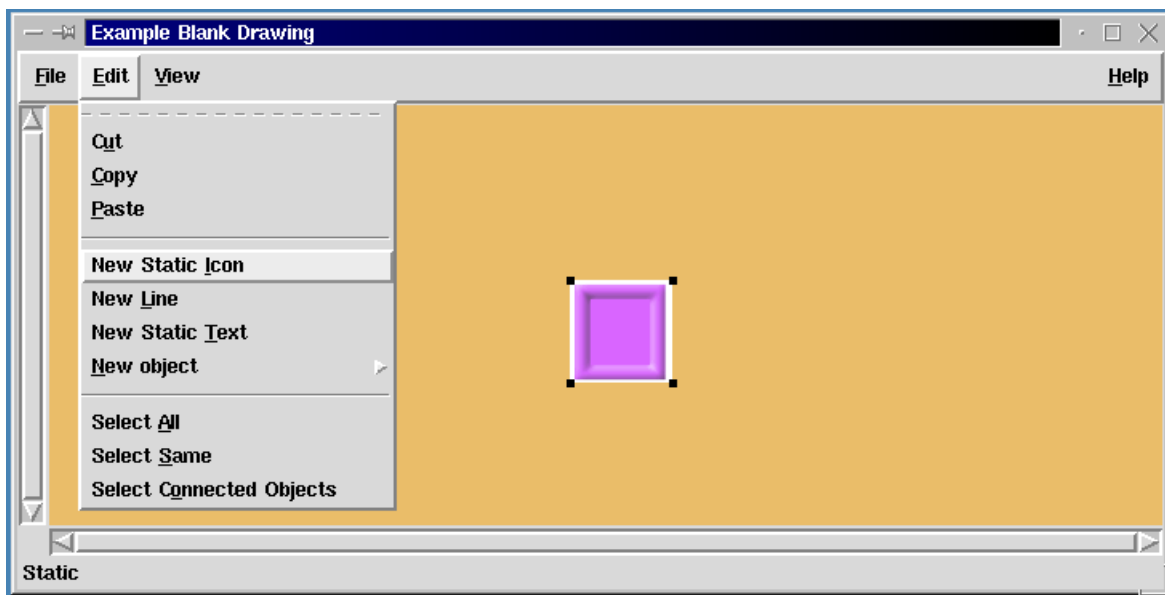
The following sections describe how to add icons, lines, text and database objects (graphical elements) to a drawing (background).

#### 9.3.1 New Static Icon

New static icons are added to a drawing as follows:

---

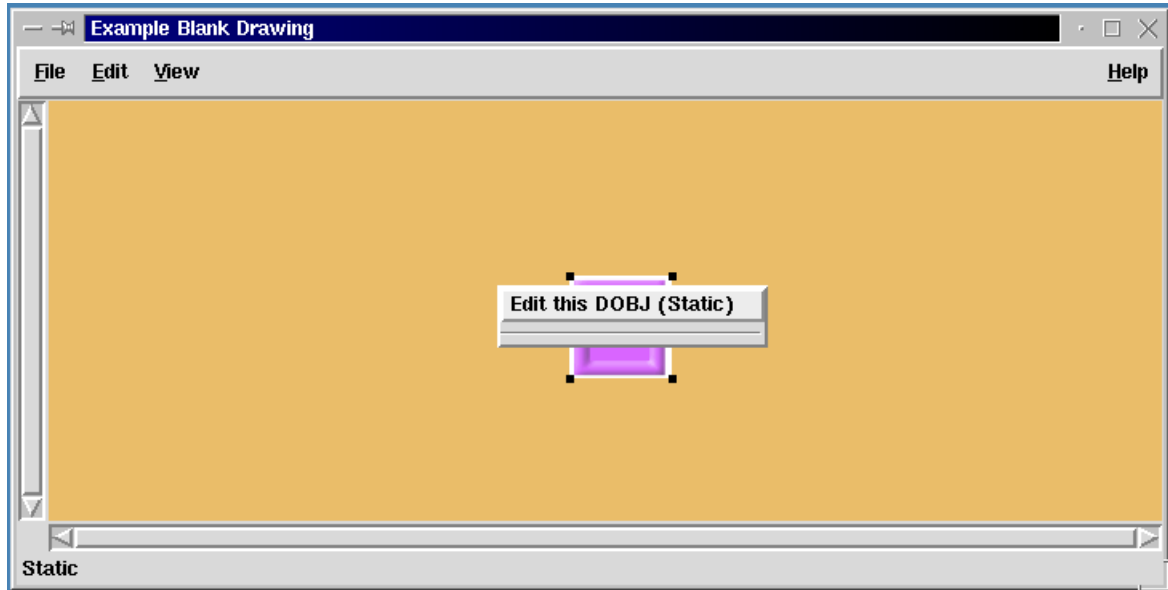
**FIGURE 88.** Adding a New Static Icon



This screen shows the Edit->New Static Icon selection and the result. A new default icon is created in the middle of the drawing. The icon can be repositioned on the drawing by dragging it to its correct location (click and hold the left mouse button and drag the icon to its new location, then release the left mouse button).

The drawing object corresponding to the icon can be changed by right-clicking to display the drawing object (Edit this DOBJ) selector. Refer to Figure 89 on page 99 and Figure 90 on page 100 for details.

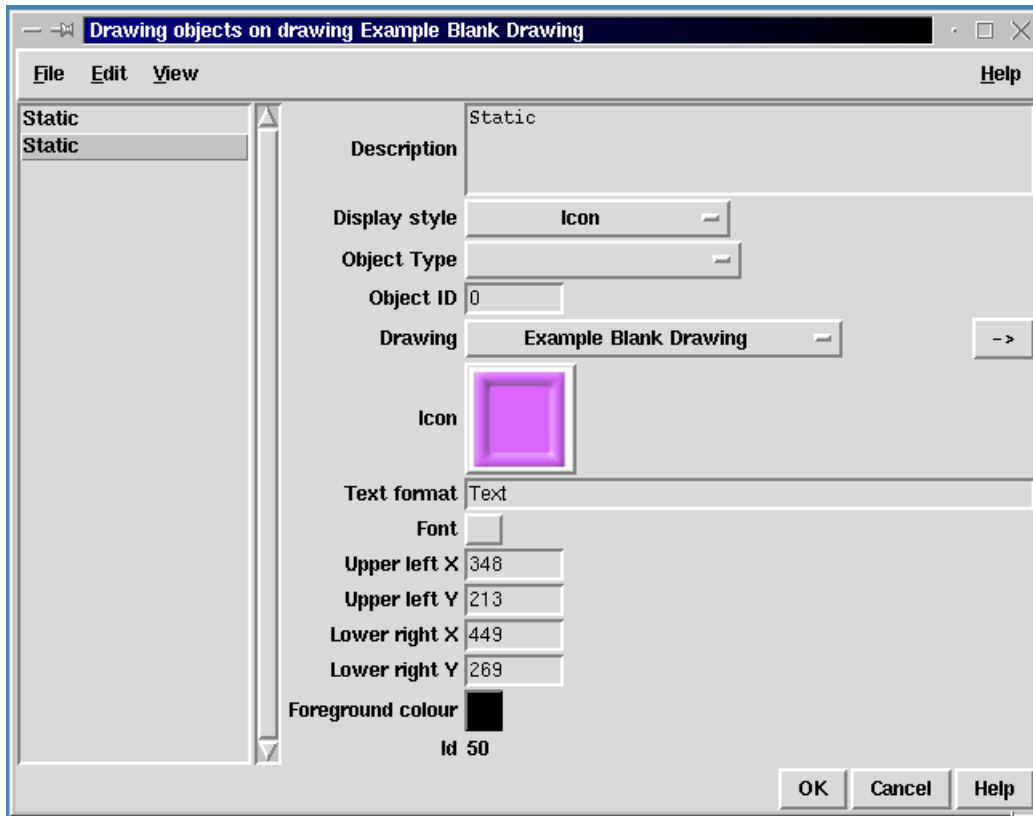
**FIGURE 89.** Editing a Static Icon



This screen shows the menu that appears when right-clicking a static icon.

Selecting "Edit this DOBJ" will cause the drawing object (DOBJ) editor to appear. Refer to Figure 90 on page 100 for details.

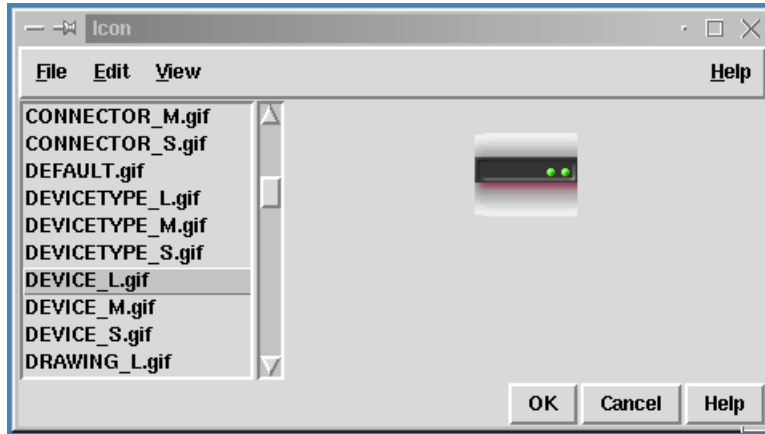
FIGURE 90. Drawing Object Editor



This screen shows the Drawing Object editor for the new icon.

The icon can be changed by clicking on the Icon field to display the icon selector screen. Refer to Figure 91 on page 101 for details.

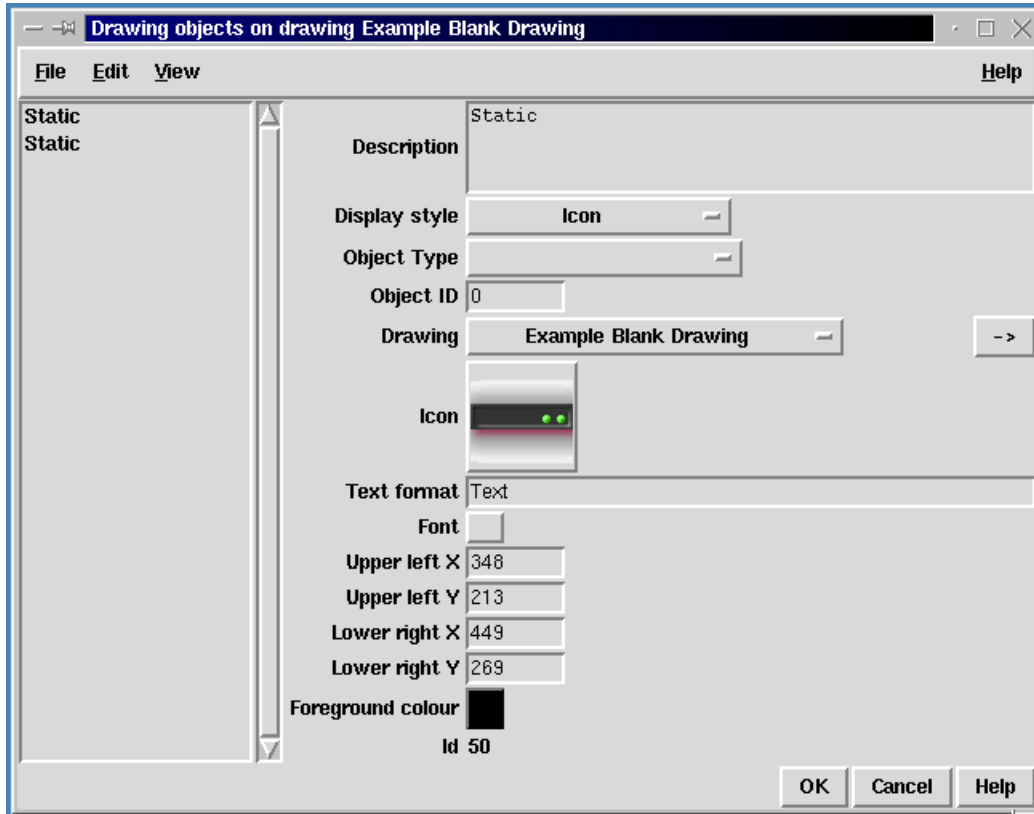
**FIGURE 91.** Icon Selector



This screen shows the list of icons available for use. The list is created automatically from the list of files present in the <NetsDir/images/icons> directory.

Clicking OK will change the icon for this drawing object.

FIGURE 92. Drawing Object Editor with New Icon



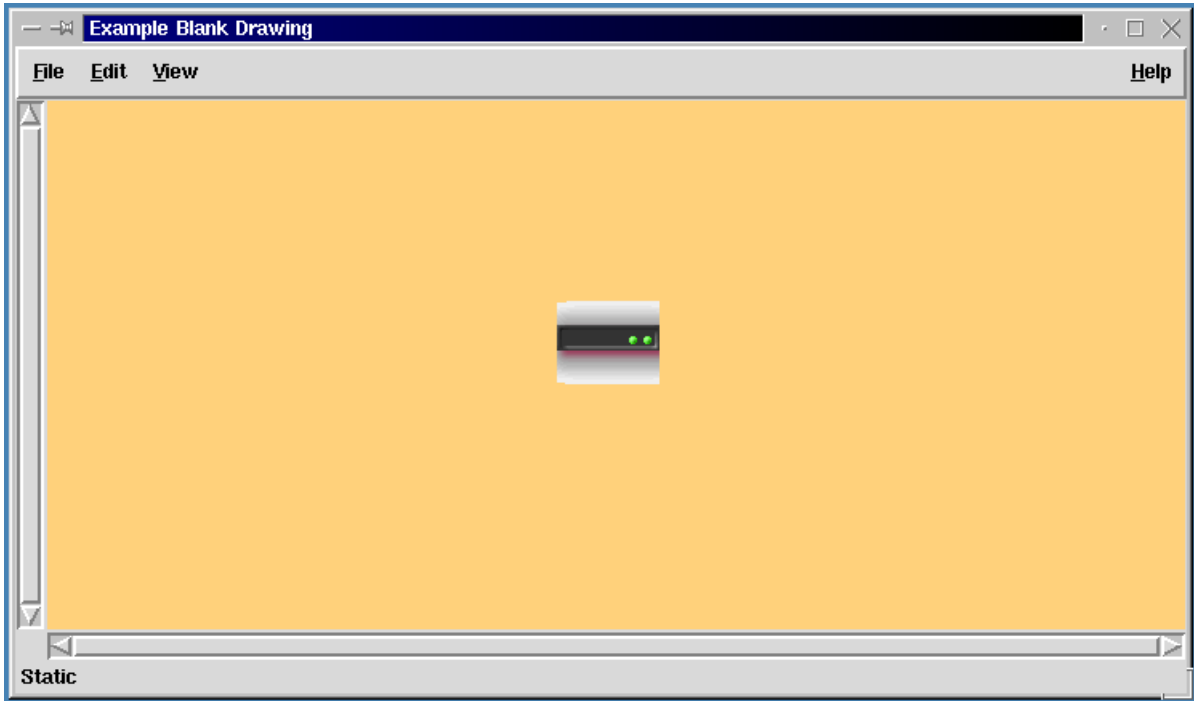
This screen shows the new icon selected for this drawing object.

The icon can be linked to any database object by selecting the appropriate Object Type and Object ID. If the Object Type and Object ID are left undefined, the icon will simply appear on the drawing.

Clicking OK will save the drawing object, and the new icon will appear on the drawing after View->Refresh in the drawing viewer to re-read the database information for the drawing.

The Object Type can be defined as "Drawing" to refer to other drawings. This can be used to create drill-down hierarchies of icons and drawings in any order and to any depth as required. Multiple hierarchies can also be defined to show geographical, logical, schematic and/or photographic representations of the Nets objects.

**FIGURE 93.** Blank Drawing with New Icon



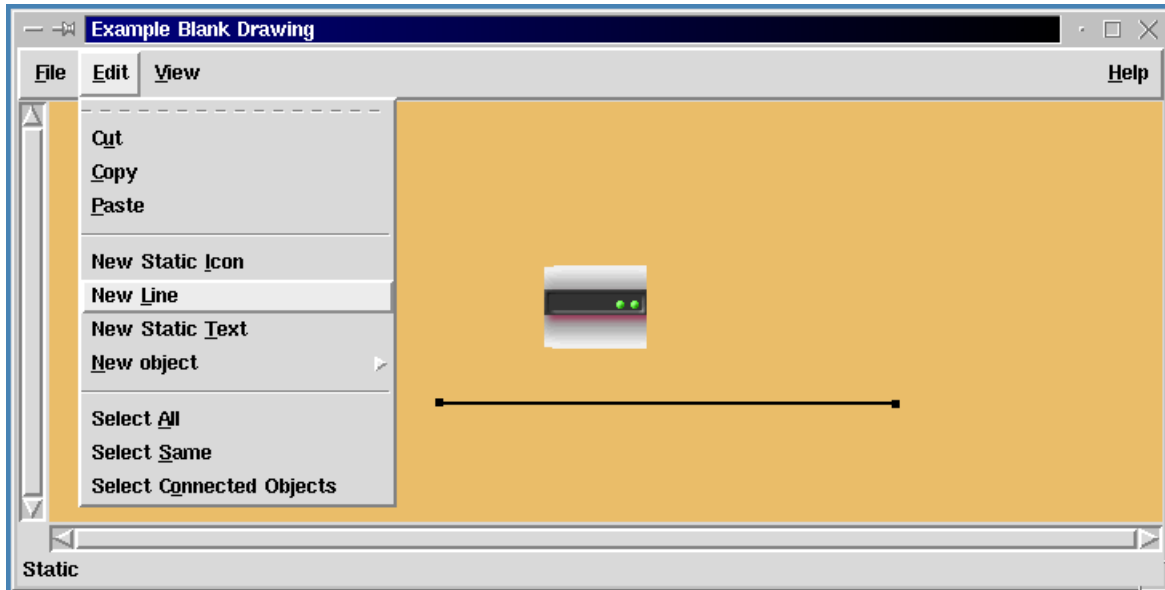
This screen shows the new icon displayed on the drawing.

### 9.3.2 New Line

New lines can be added to drawings in the same way as icons, by selecting Edit->New Line. A new line will be added to the drawing and it can be moved and adjusted as required. Lines are usually used to represent links.

---

**FIGURE 94.** New Line added to the Drawing



This screen shows the New Line selection and the result.

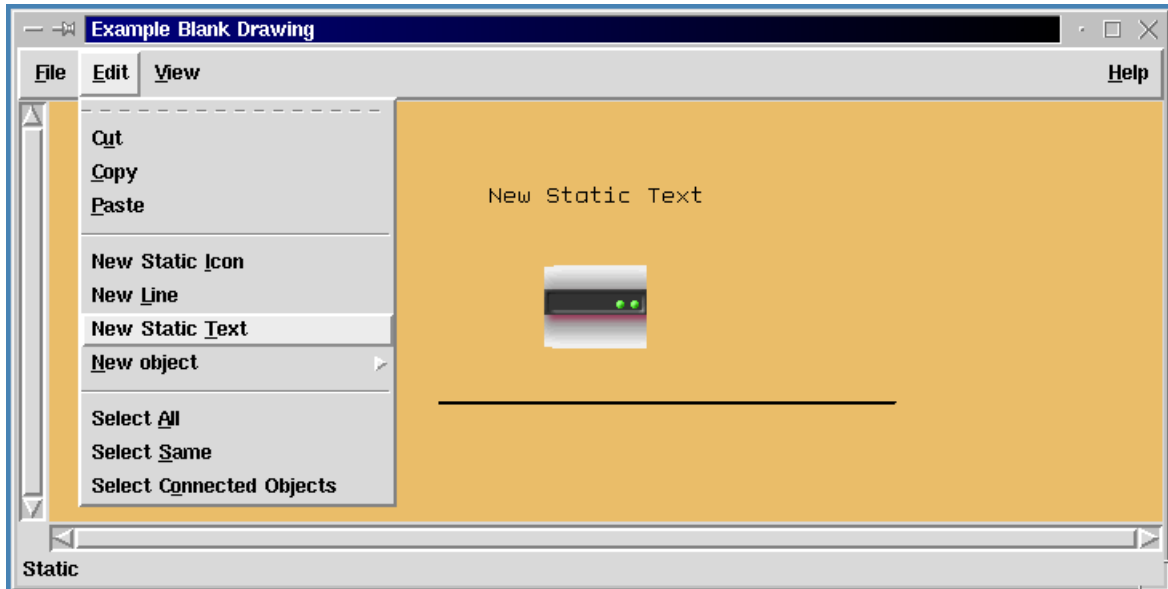
The drawing object associated with the new line can be edited in the same way as an icon (see Section 9.3.1 on page 98).

### 9.3.3 New Static Text

Text fields can be added to drawings with Edit->New Static Text.

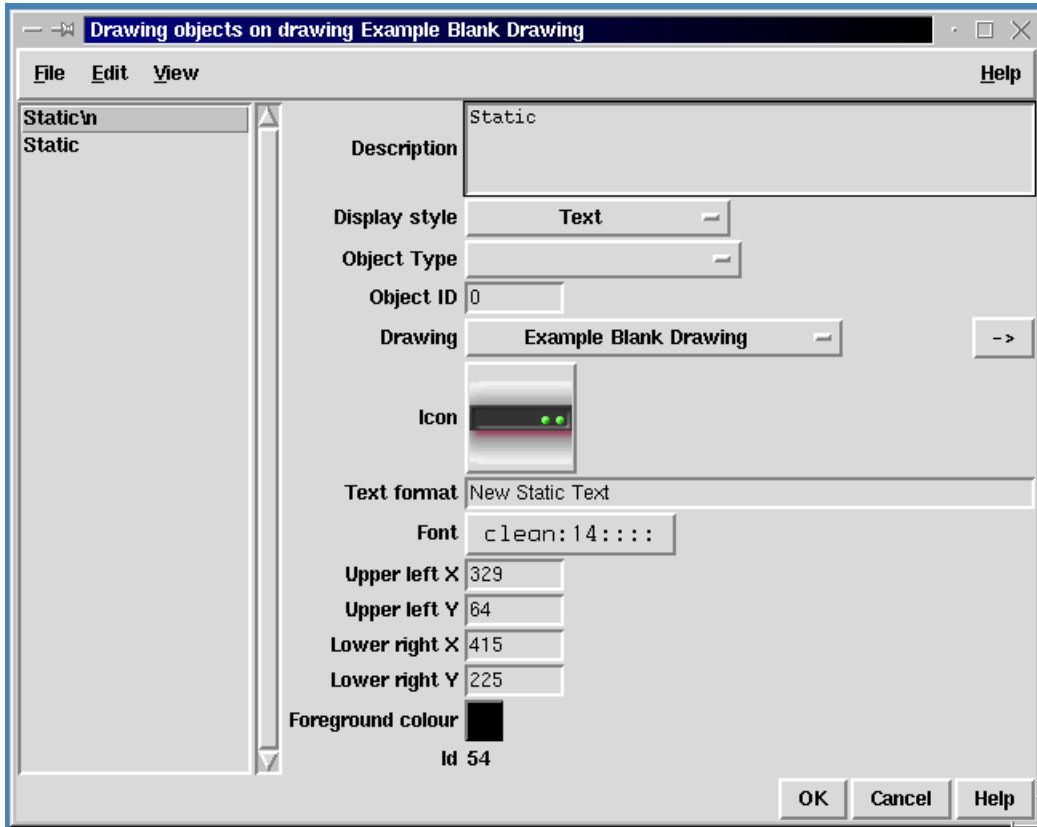
---

**FIGURE 95.** New Static Text added to the Drawing



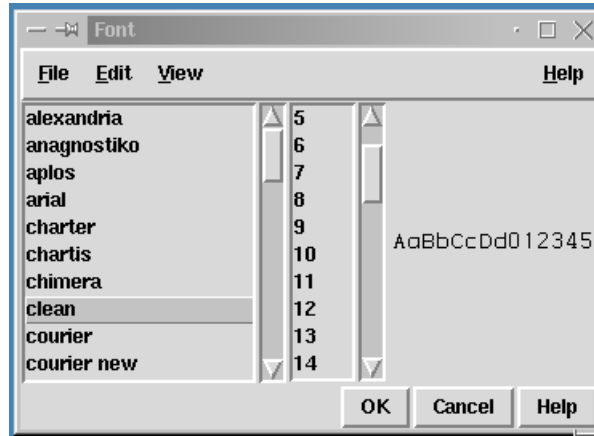
The text can be formatted by editing the drawing object and changing the text, format, font and colour. Refer to Figure 96 on page 106 for details.

FIGURE 96. Drawing Object for New Static Text



The Text format window contains the new text string and the format of the new text can be modified by clicking the Font button. Refer to Figure 97 on page 107 for details.

**FIGURE 97.** Font Selector



This screen shows the font selector used to modify text strings for drawings.

### 9.3.4 New Object

Database objects can be added directly to drawings with Edit->New Object.

A default icon for each type of object will be selected automatically. The icon can be changed later, refer to Figure 90 on page 100 for details.

---

**FIGURE 98.** New Object added to the Drawing



This screen shows a new connector object added to the drawing.

The drawing object associated with the new object can be edited in the same way as an icon (see Section 9.3.1 on page 98).

## 10.0 Help Menu

---

FIGURE 99. Nets Main Screen Help Menu



This screen shows the Nets Help Menu.

### 10.1 Help Screen

FIGURE 100. Nets Help Screen



This screen shows the About Nets information.

### 10.2 Getting Help

If you need help with Nets user tasks, you should consider using the Nets mailing list, or (if you have a support contract) the Nets support email address.

However, before posting requesting information or help, try the following:

- Read and reread the manuals and guides, especially this Users Guide.
- Try some experiments in a test or training database to become familiar with the capabilities of the system.
- Look for similar questions and/or discussion in the Nets mailing list archive:

<http://www.starport.net/~nets>

If you solve your problem, consider posting a summary and the solution, so we can add it to the Nets FAQ.

If you find a bug, let us know so we can fix it. The best way to let us know is to post to the Nets mailing list.

