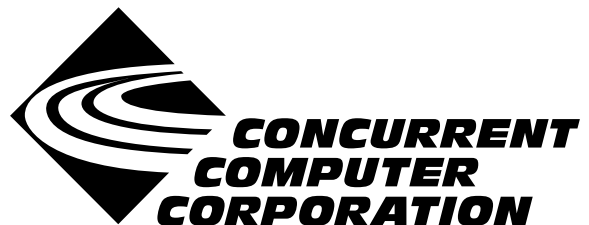


Élan License Manager

Version 5.0.2 Release Notes

August 2000

0891055-5.0.2



Copyright

Copyright 2000 by Concurrent Computer Corporation. All rights reserved. This publication or any part thereof is intended for use with Concurrent Computer Corporation products by Concurrent Computer Corporation personnel, customers, and end-users. It may not be reproduced in any form without the written permission of the publisher.

Disclaimer

The information contained in this document is subject to change without notice. Concurrent Computer Corporation has taken efforts to remove errors from this document, however, Concurrent Computer Corporation's only liability regarding errors that may still exist is to correct said errors upon their being made known to Concurrent Computer Corporation.

License

Duplication of this manual without the written consent of Concurrent Computer Corporation is prohibited. Any copy of this manual reproduced with permission must include the Concurrent Computer Corporation copyright notice.

Trademark Acknowledgments

NightStar, PowerMAXION, PowerMAX OS, TurboHawk, and Power Hawk are trademarks of Concurrent Computer Corporation.

Night Hawk is a registered trademark of Concurrent Computer Corporation.

Élan License Manager is a trademark of Élan Computer Group, Inc.

PowerStack is a trademark of Motorola, Inc.

X Window System is a trademark of The Open Group.

Contents

1.0 Introduction	1
2.0 Documentation	2
3.0 Prerequisites	3
3.1 Software	3
3.2 Hardware	3
4.0 System Installation	4
4.1 Installation Options	4
4.2 Obtaining Licenses	4
5.0 Overview of Élan License Manager 5.0.2	6
5.1 Changes in this Release	6
5.1.1 License Acquisition Delay	6
5.2 License Allocation	6
5.2.1 Floating Licenses	6
5.2.2 Node-Locked Licenses	6
5.2.3 Reserved Licenses	7
5.2.4 Excluded Users	7
5.2.5 Domain Licensing	7
5.2.6 Held Licenses	8
5.3 Redundant License Servers	9
5.4 Running License Managed Applications	10
5.4.1 Error Messages	11
5.5 Administrative Commands	12
5.6 Compatibility with Previous Releases	13
6.0 Cautions	14
7.0 Direct Software Support	15

1.0. Introduction

Élan License Manager™ is a software package that controls single-or multi-user access to software applications in a single-user or networked multi-user environment.¹ The principal functions of the Élan License Manager include:

- **User Based Licensing** - To allow *per-user* floating licensing that controls the number of concurrent users for an application. Licenses are provided to users, not machines.
- **Component Licensing** - The ability to license features or components of an application individually.
- **Timed Software** - The ability to establish an expiration date on software for creating evaluation or trial copies.
- **Copy Protection** - A form of copy protection ensures that licensed software applications operate only on an authorized CPU or network. Unlike MS-DOS copy protection, this copy protection scheme still permits backup copies, changing disks, reinstallation, etc.

Certain Concurrent applications are now released as licensed products using the Élan License Manager. **e1an51m** is required to manage the licenses required by those products.

The Élan License Manager is based on the client/server model with a *license manager* running on one or more *servers* and the *application* running on one or more *clients*. The *server* and *client* may be the same machine or they may be different machines.

The *license manager* maintains a central database of licenses and activity. The *license manager* verifies valid clients, grants licenses to valid client applications, and collects the licenses when the application terminates. The *license manager* runs as a *daemon* invoked at system start-up.

The *application* may run on the server or on one or more clients on a network. Applications communicate with the license manager via sockets using datagram connections to request and return licenses.

Licenses are available to anyone on the network or CPU - the applications need not be tied to any particular nodes (although they can be). This scheme of flexible licensing has been coined *dynamic licensing* or *floating licensing*. It is far more convenient than the older "one copy on one CPU" license scheme, and it increases the availability of the application within the site at no increase in cost.

Élan License Manager licenses objects called *features*. *Features* are considered components or major functions of an application. Applications can have one *feature* (i.e. the ability to invoke the application) or many. Each *feature* may have its own license count and expiration date.

Obtaining and administering licenses for application features is described later in this document.

1. Much of the text of this paper comes from the *Élan License Manager Technical Overview* and the *Élan License Manager Programmer's Guide* by the Élan Computer Group, Inc., and is used with their permission.

2.0. Documentation

Table 2-1 lists the *Élan License Manager 5.0.2* documentation available from Concurrent.

Table 2-1. *Élan License Manager Version 5.0.2* Documentation

Manual Name	Pub. Number
<i>Élan License Manager Version 5.0.2 Release Notes</i>	0891055-5.0.2

Copies of the Concurrent documentation can be ordered by contacting the Concurrent Software Support Center. The toll-free number for calls within the continental United States is 1-800-245-6453. For calls outside the continental United States, the number is 1-954-973-5354.

Online manual pages are provided for **elmadmin(1)**, **elmalert(1)**, **elmd(1)**, **elmrpt(1)**, **elmusage(1)**, **elmver(1)** and **elm_resource(5)**.

Additionally, the *Élan License Manager 5.0.2 Release Notes* are available online by using the X Window System™ utility, **nhelp**.

Further, the *Élan License Manager 5.0.2 Release Notes* are also available on Concurrent Computer Corporation's web site at www.ccur.com.

3.0. Prerequisites

Prerequisites for Élan License Manager Version 5.0.2 are as follows:

3.1. Software

- PowerMAX OS™ Version 4.3 or later

3.2. Hardware

- Computer Systems:
 - Power Hawk™ 620 and 640
 - Power Hawk 710, 720 and 740
 - PowerStack™ II and III
 - Night Hawk® Series 6000
 - TurboHawk™
 - PowerMAXION™
- Board-Level Products:
 - Motorola MVME2604
 - Motorola MVME4604

4.0. System Installation

The Élan License Manager product is installed as standard PowerMAX OS software packages and utilizes the standard PowerMAX OS product installation mechanism, **pkgadd** (see **pkgadd(1)**).

The package names are **elan51m** and **elan41m**. These names are case-sensitive.

elan51m	the “main” Élan License Manager package (and the only one needed if all tools are upgraded at the same time)
elan41m	a license server daemon used to provide support for older tools which have not yet been upgraded

NOTE

The **elan1m** package should be removed before installing **elan51m**.

Please refer to the “Installing Add-on Software” chapter in the *System Administration Volume I* (0890429) manual and the *PowerMAX OS Release Notes* (0890454) for instructions on software installation.

The **elan51m** product needs to be installed on a particular system only if:

- Licensed applications (e.g., any of the NightStar™ tools) are used at this site, *and*
- This system is used as one of the systems on the local network that runs the license manager, or users on this system should be allowed to run the license manager reporting commands.

The **elan41m** product is provided to allow both new and old license managers to run on the same server machine until all the tools using the Élan License Manager have been upgraded. See “Compatibility with Previous Releases” on page 13 for more information.

4.1. Installation Options

During installation, the installer is asked if the license manager daemon, **elmd**, should be started at boot time on this system. If the answer is "yes", then the script

```
/etc/init.d/Selmd
```

is linked into the directory of commands that are executed at each system boot. You may use this command to start the daemon at times other than system boot.

See **elmd(1)** and **elm_resource(5)** for more information about starting the license manager daemon.

4.2. Obtaining Licenses

Licenses are granted by obtaining a *key* from Concurrent. This key is generated specifically for your CPU or network and encodes the number of licenses and their expiration date. You must obtain the appropriate keys before you can run licensed products. Each Concurrent product that requires a license describes in its documentation which components of the product are licensed.

The steps for obtaining and installing licenses are:

1. Read the release notes for the product that you wish to obtain licenses for and find the *feature alias* (usually a derivative of the name of the product).
2. Determine the number of license managers you wish to run at your site and the machines where they will run (see below).
3. On one of the machines where you run a license manager, run the program **elmadmin**:

```
elmadmin -c -n name
```

where *name* is the *feature alias* provided in the documentation for the product that is to get a license. You will be asked for the number of managers (servers) you wish to run and, if more than one, the host names for each. When you have answered all questions, **elmadmin** displays a *server code* for each server where you run a license manager. The program then prompts you for a *key* that you must now obtain from Concurrent.²

4. Contact Concurrent Software Distribution at 1-800-666-5405 (continental United States only) or 1-954-283-1836 to obtain your *key*. You need to provide the *feature alias* and each *server code* generated by **elmadmin**.
5. Enter your key for **elmadmin**. The licenses are now initialized and you should be able to run the protected application.

Here is a sample dialogue with **elmadmin**.

```
# elmadmin -c -n MAXAda
Elan License Manager - Copyright 1989-1997 Elan Computer Group, Inc.

On how many hosts will you run the license server? (default=1):
Code for "amber2" is: V5-194/8720 1204 2535 695
Please enter your key: 1763 0272 5157 0245 5103 0315 7265 7200 2980
5312 5

License Replacement Pack #01 ID=919265500 (Tue Sep 23 14:12:09 2025
GMT)
Feature name: MAXAda [31000]
Number of licenses: 5

Successfully installed key for feature MAXAda
(/usr/lib/powerworks_licenses/31000.101).
Notifying server of new key ... OK
```

The system administrator may need to interact with the license manager using the *feature name*. The *feature name* is a five-digit number that the *feature alias* is associated with. The *feature name* is provided in the application's documentation. When placing entries in the *resource* file (see below), be sure to use the numeric *feature name*, not the *feature alias*.

2. If you cannot contact Concurrent immediately, it is safe to interrupt the **elmadmin** program at any time. Rerunning it later after contacting Concurrent and entering the same responses generates the same server codes.

5.0. Overview of Élan License Manager 5.0.2

Élan License Manager 5.0.2 is released primarily as a patch to Élan License Manager 5.0.1.

5.1. Changes in this Release

5.1.1. License Acquisition Delay

In previous releases, the Élan License Manager could take up to 15 seconds to grant a license for a particular feature when the count of licenses in use for that feature reached the maximum allowed.

NOTE

In such circumstances, each MAXAda compilation could take, at a minimum, 15 seconds to complete.

A workaround in the past was to set the `POWERWORKS_ELMHOST` environment variable:

```
POWERWORKS_ELMHOST=@host
```

where `host` was the name of the system where the Élan License Manager was running.

This workaround is no longer needed with Élan License Manager 5.0.2.

5.2. License Allocation

Élan License Manager implements two principal schemes of licensing: *floating* and *non-floating* licenses. As a refinement, the features of *reserving* and *holding* licenses are also available. These concepts are discussed in the following sections.

See `elm_resource(5)` for additional information.

5.2.1. Floating Licenses

These licenses are generally available to *anyone* on a network who can reach the server running the license manager daemon. As such, these licenses are known as *floating* licenses - the licenses float on the network and anyone on the network can use them. When 100 licenses, for example, are encoded in the license key for a product, then *any* 100 users on the network can use the product concurrently.

When the concurrent usage reaches the licensed limit, 100 in this example, a subsequent request to use the product is denied because all licenses are currently in use.

5.2.2. Node-Locked Licenses

It is occasionally desired that licenses not float on a network. Usually the facilities of the *resource* file (see below) can be used to restrict a license to a particular system. Alternatively, the site can request that their licenses be *node-locked* on a per-CPU basis. The licenses are only available on the CPU that the license manager daemon is running on. Node-locked licenses must be specified when contacting Concurrent for the license key.

If, at a later time, floating licenses are desired, the non-floating key can be converted to a floating license key simply by requesting a new key from Concurrent.

5.2.3. Reserved Licenses

Whether licenses float on a network or not, sometimes you would like to *reserve* licenses for particular people, groups, or even machines.

For example, let's say a company of 400 employees purchases 100 floating licenses for a product. However, the company would like their simulation department of 20 always to have access to the application. These 20 people can be named in the *resource* file, and 20 licenses are reserved for them.

Similarly, a company may buy many floating licenses for a diagnostic package, but they would like the lab machines always to have access to the package. The *resource* file may list these machines, and reserve some licenses for these particular machines. The remainder of the licenses float.

The *resource* file looks similar to the `/etc/group` file. Here's an example portion of a *resource* file:

```
11000:development:jim,tom,bill,cindy,brad:10:0
11000:simulation:barbara,vince,todd:2:0
11000:proj-lead:eddy:1:0
11000:demo-room:@demosys.mycorp.com:1:0
```

In the example above, 10 licenses for *NightView* (the feature name for *NightView* is "11000") have been reserved for the development department, 2 licenses for a group of 3 in the `simulation` department, 1 license for the boss, `eddy`, and 1 license has been reserved for the demonstration room machine named `demosys.mycorp.com`. The remaining licenses, if any, float as usual.

Notice that one may reserve *fewer* licenses than there are group members. We have done this in our example above with the `simulation` group. In such a case, any member of this group acquires a license from the reserved pool first, thus affording a high probability to this group of obtaining a license. If the reserved pool is already fully allocated, however, licenses are taken from the remaining non-reserved pool.

The *resource* file is used for floating or non-floating licenses alike. The distinction is simply that the non-floating licenses stay with a particular machine. Licenses may still be reserved for individuals. It makes little sense, of course, to reserve to a machine name in the *resource* file if the licenses are already non-floating.

Please refer to `elm_resource(5)` and `elmd(1)` for more details on the *resource* file.

5.2.4. Excluded Users

Users and/or hosts may be *excluded* from using specified features by listing them in the *resource* file with the word `EXCLUDE` in the license count field. For example:

```
11000:user-group5:harriet,@wanda:EXCLUDE:0
```

would disallow `harriet` or anyone logged onto machine `wanda` from checking out licenses for *NightView*. `EXCLUDE`-ed users must be listed before any reserved licenses.

5.2.5. Domain Licensing

It is desirable to limit clients that may check out licenses by network *domain*. For example, a system administrator may wish to limit certain licenses only to the network domain servicing the department that purchased the license.

For example, if a company owned a Class-A network but was also on the Internet, the administrator might wish to limit license server requests from only those on the Class-A network.

License requests are limited to a domain of clients via the *resource* file. The token %IPACCEPT followed by a *mask*, specified in the *resource* file, limits client accesses to *mask*.

The *mask* appears in the *resource* file as a line of the form:

```
%IPACCEPT mask
```

where *mask* is a quadruple of expressions "*e*" separated by periods: "*e.e.e.e*". Each *e* may be any one of the following, or a comma separated list of any of the following:

<i>N</i>	A decimal integer between 0 and 255.
<i>N-M</i>	An integer range. <i>N</i> and <i>M</i> are decimal integers between 0 and 255, and <i>N</i> should be less than <i>M</i> .
*	An asterisk matches all integers.
THIS	The word "THIS" (or "this") means the server's IP octet value in the designated position. For example, if the server address were 192.100.42.16, then "THIS.*.*.*" is the same as "192.*.*.*" and "THIS.THIS.THIS.*" would be the same as "192.100.42.*".

For example, a *mask* of:

```
THIS.*.*.*
```

would limit requests from only those on the same Class-A network as the license server. Also,

```
THIS.100,200-224,226.*.*
```

would limit requests from only those on the same Class-A network and from Class-B subnetworks 100, 200 through 224, and 226.

5.2.6. Held Licenses

Held licenses are those licenses that are *held* or retained as checked out by the license server *after* an application exits. These licenses are each marked with the original owner's user and host name, and are held for a specific duration, the *hold period*, for this person's reuse. If the hold period has not yet expired and the same person (the same *user@host*) requests a license that is currently *held*, the held license is transferred to him or her. When the hold period passes, however, such licenses are freed to the free license pool for anyone's use.

Held licenses are useful for several different purposes.

1. A minimum license usage period may be established for short-lived applications. For example, a compiler often runs for a brief period. As such, a standard floating license per se may not be an effective form of licensing since, by probabilities, it is usually checked in and, therefore, usually granted to anyone who asks for it. However, if the application has a minimum usage period of 10 minutes or so, a more equitable per-license usage is realized. The *held* license feature may be used to implement such a scheme.
2. When a series of applications are used together, perhaps in a series from a shell script, *holding* a common feature license for as little as a minute guarantees that each application in series is still able to obtain a license and the user is not stalled during a job.

3. A user or, more often, system administrator, may wish to allow a user to exit and shortly thereafter re-enter an application without losing his or her license to another. That is, a license may be held for 5 minutes or so after the application exits, allowing a user to re-enter with his or her license still available.

A minimum hold period may be defined by the application. A hold period greater than the minimum hold period (if one) may be designated by the user or system administrator via the *resource* file.

The final entry in each *resource* file line is the hold period in seconds for this group. For example,

```
11000:proj-lead:eddy:1:900
11000:demo-room:@demosys.mycorp.com:0:600
11000:others::0:60
```

In this example, when eddy uses *NightView*, the license is held for him for 15 minutes (900 seconds) after exiting the application.

In the second example, if anyone uses *NightView* on the machine `demosys.mycorp.com`, his or her license is held for 10 minutes.

In the third example, everyone else (indicated by an empty member list) has his or her license for *NightView* held for 1 minute after exiting the application.

If the value in the held field is less than the minimum hold period defined by the application, the minimum is used.

5.3. Redundant License Servers

Many customers install applications on a single system, usually a *file server*, and run a single license server on the same system. Of course, if this file server goes down they no longer have access to the application - or any other files on the server, for that matter.

In some cases, however, it is desirable to maintain one or more *backup* systems: if the main file server crashes work can be continued on a backup file system. The application and all support files must be installed on the backup file server, and the license manager must also be installed on these systems to afford license control from these backup servers.

Élan License Manager supports redundant license servers with a technique similar to other fault-tolerant systems such as the space shuttle. The license server, **elmd**, may be run on multiple servers with a special key called a redundant license key. The *redundant* key is distinct from a standard key in that it encodes the system IDs for *all* systems that are authorized to run the license server. This key is shared among all redundant servers. A redundant key is generated by supplying server codes for each of the systems where the license manager is run.

The set of systems that the license manager runs on can be changed at any time by requesting a new key from Concurrent with server codes for the new machines (or possibly the one single machine) that the license manager is to be run on.

When the redundant model is used, all license servers on a network communicate with each other and maintain status information on each other. When a cluster of **elmds** are started on a network, one automatically receives the status of *master*, and the remaining **elmds** receive the status of *shadow*. Client communication is automatically routed to the current *master* for license acquisition and authentication.

For security, the license manager requires that a *majority* of license servers remain running. The reason for this requirement is that if a network were divided in two, then each section would have to decide who can continue to issue licenses. If both continued, each would be authorizing licenses - twice as many

licenses as are valid. However, because only one side can have a mathematical majority of the original number of servers, the remaining majority only wins and takes over license distribution. The minority side (or sides) become inactive.

For a majority side to exist, the original number of servers must be an odd number.

When using **elmadmin** to generate server codes and install a license key, a key file is created on the machine where **elmadmin** is to run. The key file must be made accessible to the other servers by one of the following methods:

1. placing the key file in a file system shared by all machines, or
2. by copying it to a location on each machine.

It is preferable to place the key in the same directory path on each system. If this is not practical, use the **-e** option of **elmd** on each system to specify the key location.

The host name of each of the redundant servers should be mentioned in the *resource* file. This may be done by placing lines of the form:

```
%SERVER hostname
```

into the *resource* file, where *hostname* is the name of one of the hosts. There should be one line for each of the redundant servers.

5.4. Running License Managed Applications

When running an application whose usage is managed by the Élan License Manager, be prepared for the application to exit immediately upon invocation if there are not enough licenses or if the license manager is down (or if the application cannot connect to the *master* server). The documentation for each application states what happens when a license is unavailable when you try to run it, or if the license is lost during execution of the application.

Usually an application must send a broadcast request on the network to locate the master license server. If you know the hostname that the license server is running on, you may set the environment variable "POWERWORKS_ELMHOST" to the hostname and avoid the network broadcast. The variable may also contain a list of hostnames, separated by colons. In this case, parallel connections are attempted to all the listed hosts and the first to reply handles the license management. For example, to restrict the license manager handling your applications to a manager on only one of the systems *hoss*, *ben*, or *littlejoe*, use

```
POWERWORKS_ELMHOST=hoss:ben:littlejoe
export POWERWORKS_ELMHOST
```

Usually, the application broadcasts to all the specified hosts in parallel. To serialize the attempts, preface the host list by an "@" sign. This tries each host until it times out and then tries the next, stopping with the first responsive host. When you have only one host in your list, it is best to use the @ prefix.

By default, the Concurrent implementation of the Élan License Manager 5.0.2 (**elan51m**) uses socket port 6462. The **elan41m** product uses socket port 6461. If either of these port numbers conflict with an application at your site, you may change which port is used by the Élan License Manager by modifying the */etc/services* file by adding a line of the form:

```
service-name      xxx/udp  concurrent-license-server
```

where *service-name* is either *ccur_elmd* for **elan51m** or *powerworks_elmd* for **elan41m** and *xxx* is an otherwise unused port number of your own choosing.

5.4.1. Error Messages

The following messages occur during failed attempts to start an application that must connect to the license manager.

License server initializing; please wait

or...

License server coming up in *n* seconds; please wait

or...

License server coming up in *n* minutes; please wait

The license server has just started and the application must wait for the server to complete its initialization, this includes waiting for a synchronization period specified with the **-s** option to **elmd**.

License server is down

No operational license server could be found running on the network. Check the value of the POWERWORKS_ELMHOST environment variable, if set.

Insufficient servers running

The client is running with redundant servers but there are not enough servers active. For license validation purposes, a majority of servers must remain active in the redundant mode.

License server host unknown

The hostname is not valid. Check that the POWERWORKS_ELMHOST environment variable, if set, is set to the name of the proper host for the license manager daemon and that that hostname appears in the **/etc/hosts** file or is defined by the DNS server.

No licenses are currently available

There are no licenses available for the designated feature.

License authorization failure

Authorization failure. There was a client or daemon fraud or an attempt to acquire a non-floating license from another host.

License has expired

The feature license has expired or there was a date validation error.

Communications socket failure

A **socket (2)** connection failed, probable network failure.

Network error: unable to send message to license server

A **sendto(2)** system call failed when trying to send a message to the license daemon, probable network failure.

License lost

A reconnect after a server crash failed to reestablish all the client's prior licenses. That is, one or more licenses was denied for some reason during a reconnect.

No license file found for feature

or...

No such feature

There is no license key file installed for the specified feature.

5.5. Administrative Commands

The Élan License Manager product includes these administrative commands:

elmd	The license server controlling all licensing checkouts and reclaims. This daemon program runs on the server and issues and collects license requests. elmd is initiated at boot time by the selmd boot script.
elmadmin	Creates license keys, performs daemon control functions, and lists current license activity.
elmalert	Provides an end-user or system administrator with information regarding when licenses are near expiration or when a license for a particular feature becomes available. Notification may be done via screen message or by email.
elmrpt	Provides a summary report on daily license activity, indicating the number of license requests, the number issued, and the number denied. It optionally gives a day to day breakdown or total breakdown. When using this command, it is necessary to specify the log file name as a command line parameter, usually "/usr/adm/elm.log" .
elmusage	Produces a snapshot report of current license activity, updating every 10 seconds.
elmver	A diagnostic tool used to extract version information from applications managed with the Élan License Manager.
elm4d	The licence server used to provide support for older tools while getting them all upgraded.

elmd and **elm4d** are installed in the `/usr/lib/powerworks_licenses/bin` directory. Other commands are installed in the `/usr/bin` directory. Online reference manual pages for each of these commands are available as part of this product. In addition, **elm_resource(5)** describes the format of the resource file.

5.6. Compatibility with Previous Releases

The **elan51m** license manager is not backward-compatible with the previous **elan1m** license manager. Therefore, all tools that currently use **elan1m** will need to be upgraded in order to use the new license manager. However, the old keys you already have on your system are valid with the new license manager, so there is no need to obtain new keys.

It may not be convenient to simultaneously upgrade all the tools on all the machines in a network, so the **elan41m** package is provided to allow both the old and new license managers to run on the same server machine.

This is intended as a temporary measure while all the tools are being upgraded. The **elan41m** package provides a license server which will only use the license keys you already have. Any new license keys you add after upgrading to **elan51m** will be available solely to newer tools which are getting licenses from the **elan51m** version of the license manager.

Also, none of the reporting or administrative tools will work with **elan41m**. For example, the **elmadmin** tool will only be able to show you information about licenses in use by tools talking to the **elan51m** server. The same goes for **elmrpt**, etc.

Finally, any application that is expecting to use the 5.0.2 license manager will not be able to connect to an older version of the Élan License Manager.

6.0. Cautions

All Concurrent applications use numeric license keys (provided by Concurrent) which look similar to credit card numbers, for example:

```
1763 0272 5157 0245 5103 0315 7265 7200 2980 5312 5
```

In addition, the server codes (generated by the **elmadmin** program) are preceded by a “V5-”. For example:

```
V5-194/8720 1204 2535 695
```

If you use **elmadmin** to generate a server code and it does not look vaguely like the number above, then you probably did not use the **-n** option flag. Re-run the command with the **-n** option.

7.0. Direct Software Support

Software support is available from a central source. If you need assistance or information about your system, please contact the Concurrent Software Support Center at our toll free number (1-800-245-6453). Our customers outside the continental United States can contact us directly at 1-954-973-5354. The Software Support Center operates Monday through Friday from 8 a.m. to 7 p.m., Eastern Standard Time.

Calling the Software Support Center gives you immediate access to a broad range of skilled personnel and guarantees you a prompt response from the person most qualified to assist you. If you have a question requiring on-site assistance or consultation, the Software Support Center staff will arrange for a field analyst to return your call and schedule a visit.

