Chapter 2: Installation of the EMS

Chapter Contents:

2.1 Introduction -- because you have to start somewhere
2.2 System requirements for installing the EMS
2.3 Installing the EMS
  2.3.1 The basics - The stuff you should know
  2.3.2 What happens during the installation process?
  2.3.3 Installing from DVD
  2.3.4 Installing from the STRC servers
  2.3.5 Local network installation
  2.3.6 Making sure your installation was a success
2.4 Potentially useful installations options and flags
2.5 Updating your EMS with passion
2.6 Potentially useful update options and flags
2.7 Options and flags for every occasion

2.1 Introduction -- Because you have to start somewhere

Installation of the EMS requires the use of ems_install.pl, which was developed to tackle all the challenges you might encounter during the installation of an NWP system. It is highly recommend that you use the most current version of this routine, as your life will become less complicated and more fulfilled if you do. And that's a statement only a few modeling packages can make!

So, “Where might I get this ems_install.pl utility thing,” you ask?

All NOAA-affiliated current and wannabe users of the EMS may simply request the routine from the SOO Science and Training Coordinator (SOO STRC). All non-NOAA users, i.e., everyone else, must register for the EMS on the SOO/STRC site:

  http://strc.comet.ucar.edu/software/uems

Registration allows users to receive notification of updates as they become available, and also allows the developer to advance his world geography skills when a request from Mauritius is submitted.

Note that there may be a delay in receiving the ems_install.pl utility following your registration, as the information is checked both for some semblance of legitimacy and to avoid advertisements for “personal enhancement” products from being sent to others. Once the EMS concierge is able to confirm your registration, which typically takes somewhere between a few minutes and a week should the lone support person be out of the office or comatose, you will receive a second email message with the installation routine attached. Enjoy!
2.2 System requirements for installing the EMS

There are a few system requirements for installing and running the EMS, which are listed below:

- **A relatively current Linux distribution, but not too current** – The EMS has been tested on Red Hat Enterprise, Fedora, CentOS, SuSe, and Ubuntu distributions. Other Linux variants will probably work just fine; however, it’s simply too difficult to keep up with all the releases and updates. Additionally, there is usually a lag before the EMS developer can install a new distribution for testing, so just stick with what works and we’ll both be happier for it.

- **Root permission and/or write permission on a large disc partition.** The ems_install.pl routine was designed to run as root user. You will have the opportunity to assign ownership to an existing or new user during the processes. An existing user could install the EMS provided the user has write permission on the drive where the package will reside.

- **24 Gb of available disk space** – This requirement pertains to the installation of the EMS only. Of course, running an NWP model can use up a significant amount of disk space as you dump simulation data files every minute for 144 hours, so this requirement should be considered as an absolute minimum.

- **The EMS user must be using a Tcsh or Bash shell** – If you are installing the EMS under a known user, then they must be running tcsh or bash; otherwise horrible things may occur, such as the EMS failing to run, and you don’t want that to happen. There are ways around this requirement, but we’re not going there right now.

2.3 Installing the EMS

2.3.1 The basics - The stuff you should know

Whether or not you have an existing EMS release on your system, you can still do a fresh installation. The novice ems_install.pl user should embrace the most basic of options:

```
% ems_install.pl --install [release version]
```

Note that the release version is optional, as indicated by the [square brackets], since the default is the most current official release. *If you are looking to install a release other than the most current, say release 3.2.1.5.22.beta, then you must specify the release number as an argument to “--install”*. For example:

```
% ems_install.pl --install 3.2.1.5.22.beta
```

The above example is for demonstration purposes only, since release 3.2.1.5.22.beta may not actually exist. So don’t try this at home. There is additional information on determining what releases are available in Section 2.5.
2.3.2 What happens during the installation process?

Regardless of what option you choose when installing the EMS, the process will include the following:

- A greeting, just because it’s important and you’re worth the effort
- A prompt for the installation directory (Default: /usr1)
- A check to make sure the directory exists and whether you have write permission
- Determine whether an existing EMS installation resides at that location:
  - Get the version of an existing installation
  - Ask whether you want to rename the existing installation to <uems>.<release>
- A prompt for the name of the user to assign ownership of the package
- An attempt to create a new account and home directory if the user does not exist
- A check that the user’s login shell is either tcsh or bash
- A prompt for a password if the new user was created
- Install the EMS from the specified source
- Do the post-install configuration
- Congratulate you on another wise decision

Note that all sorts of useful information will be printed to the screen while the installation is progressing, so don’t leave the room, even if you have to go.

During the installation process, the ems_install.pl routine will also attempt to:

- Determine the appropriate run-time executables based on your system architecture.
- Attempt to determine the number of physical CPUs (sockets) and cores per CPU on your machine. Should a problem occur, you will be asked to provide this information, and you shall be powerless to resist.
- Install an entry (albeit disabled) in the user’s crontab file to automate the process of updating the EMS. You can configure the ems_install.pl utility to automatically download and install updates from the SOO STRC servers and notify user(s) of any changes. More on the updating capabilities utility later.
- Install an entry (again, disabled) in the user’s crontab file to assist in the automation of real-time forecasting. All you need to do is make the appropriate changes to this entry and you’ll be off and modeling!

See, isn’t that simple? The EMS practically installs itself, which is just the way you like it.

Note that the downloading of EMS package files from the SOO STRC servers may require a considerable amount of time, depending upon the speed of your connection. So be patient, as your effort will be well rewarded.
2.3.3 Installing from DVD

If you are lucky enough to possess one of the fabulous DVDs from the EMS collector series, then consider yourself blessed, because life doesn’t get any better. Besides being a great coaster, the DVD can actually be used to install the EMS. This capability does come with a few caveats so don’t get too excited.

**Ok, go ahead and jump up and down a little.**

In order to install from DVD, you will need to actually mount the DVD, which may require root privilege on your system. For the sake of this guidance, it is assumed that you have mounted the disc under “/media/cdrom”, although the exact location may differ on your machine. Since you’re actually reading these instructions, it’s also assumed you can figure out what to do.

**Step a.** Load EMS DVD

**Step b.** Change directories to DVD drive, e.g. “cd /media/cdrom”

At this point you should see a copy of the install routine on the DVD. Use this version of the utility for installation unless told otherwise by a bowl of fruit or a domesticated farm animal with fur, not feathers. You can’t trust the feathered ones.

**Step c.** Run the `ems_install.pl` routine like you want something good to happen:

```
# ./ems_install.pl --install
```

With any luck something good will happen and the installation will begin. On some systems however, you may see an error such as:

```
# ./ems_install.pl
bash: ./ems_install.pl: /usr/bin/perl: bad interpreter: Permission denied
```

Or

```
% ems_install.pl
ems_install.pl: Permission denied.
```

The above errors likely indicate that your DVD is mounted “noexec”, meaning you can’t run an executable file from the DVD. Silly security restrictions.

All is not lost however (other than the time spent interpreting the error message), because a work-around is available:

**Step c1.** Copy `ems_install.pl` from the DVD to another location such as “/tmp”.

**Step c2.** Run `ems_install.pl` from that location with the following flags:

```
# ./ems_install.pl --dvd /media/cdrom (or wherever you mounted the DVD)
```

That should solve your EMS installation-related problems for now.
2.3.4 Installing from the STRC servers

Unless you are installing the EMS directly from DVD (Section 2.3.3), the default behavior of the utility is to attempt to download the necessary files from data servers at EMS World Headquarters. There is no need to include any special arguments or flags as everything is built into ems_install.pl. Just follow the same basic guidance provided in Section 2.3.1, and most importantly, install it like you mean business!

2.3.5 Local network installation

If you don’t have direct access to the EMS World Headquarters servers, you can still install the system. This method requires that the package files be manually downloaded and placed in a directory where they are accessible. Simply follow these few steps:

Step a. Create a temporary directory where the files are to be downloaded. This can be called anything, provided that you have at least 10Gb of space on that partition. For this example, the directory will be named “/usr1/repository”.

   # mkdir /usr1/repository

Step b. Open up a STRC server in your browser to view the available full releases:

   http://soostrc.comet.ucar.edu/uems/releases

   Index of /<uems>/releases

   Name          Last modified      Size  Description
   3.4.1.14.11/   17-Mar-2014 16:10 -
   3.4.1.14.16/   17-Apr-2014 10:11 -

   Determine which release you want to install, which should be the most current one if you know what’s good for you. The releases are identified by such silly names as “3.4.1.14.66.beta” or “3.5.1.14.12”, because the EMS developer was born without an imagination.

Step c. Create a directory below “/usr1/repository” with the name of the release to be downloaded. For example:

   # mkdir /usr1/repository/3.4.1.14.16

Step d. Again open up the EMS site to the desired release...

   http://soostrc.comet.ucar.edu/uems/releases/3.4.1.14.16

   and download all the package tar files to “/usr1/repository/3.5.1.14.12” directory. There may be quite a few and some are rather large, so while you are waiting you can take up waterfall kayaking or some other worthwhile activity. You have time.
Step e. Once all the files are downloaded and your injuries have healed, run the `ems_install.pl` routine as follows (Replacing the example with the actual location of the downloaded files on your system):

```
# ./ems_install.pl --install --repodir /usr1/repository
```

Step f. Congratulate yourself on another risky yet rewarding skill mastered – the EMS installation, not the waterfall kayaking.

### 2.3.6 Making sure your installation was a success

Following the carefree installation, you should *log out and return* as the EMS user. Make sure your environment is correct by attempting the following commands:

```
% cd $EMS
```

Wherein you should be located at the top level of the EMS

Also try:

```
% ls $EMS_STRC
```

Where you should see the contents of the $EMS/strc directory. If both of the above tests are successful, then try running the "sysinfo" command provided with your installation (and it IS yours now):

```
% sysinfo
```

You should see a summary of your system configuration that includes the Linux distribution along with other information such as your blood pressure (just kidding). Please make sure that the following values are correct:

<table>
<thead>
<tr>
<th>Physical CPUs</th>
<th>#</th>
<th>The number of CPUs or mice that you would see if you opened up the computer case</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cores per CPU</td>
<td>#</td>
<td>The number of cores on each physical CPU, the stuff you can’t see</td>
</tr>
<tr>
<td>Total Processors</td>
<td>#</td>
<td>This value should be just Physical CPUs * Cores per CPU</td>
</tr>
</tbody>
</table>

If either the number of Physical CPUs or the Cores per CPU is incorrect, you will have to change these values in the `$EMS/etc/EMS.cshrc` and `$EMS/etc/EMS.profile` files. If everything appears to be correct, then your installation is complete and you are ready to become a modeler. If not, just give your not-so-local SOO STRC guy a call, or send a message along with some baked goods.
2.4 Potentially useful installations options and flags

There are many command-line flags and options that may be passed to the ems_install.pl routine, not all of which are listed here simply because I’ve long forgotten what most of them do. However, I can recall enough to impart upon you sufficient knowledge to meet most of your installation desires. And that’s really all you want out of life, isn’t it?

Listed below is a description of some of the more useful flags available during an installation. Note that there may be additional flags and options listed in the other sections depending upon whether you are doing an installation or update. So, be sure not to miss them, as the life you save may be your own.

a. Option: --install [install release version|list|listall]

What this option can and can not do for you:

Passing “--install”, without any other options or flags, tells the routine that you want to do a fresh install of the EMS. The default is to install the most current release; however, this behavior may be overridden by including a release number as an argument to “--install”, in which case the specified release will be installed.

Alternatively, you can pass “list” or “listall” as arguments, which replace the previous “--list” and “--listall” flags respectively. Including “list” will provide a listing of all the current releases available on the EMS servers. Using “listall” will include a summary of all the individual package files for each release.

Then, when using “--install”,

% uems_update.pl --install [additional options]
Or
% uems_update.pl --install <release> [additional options]
Or
% uems_update.pl --install list
Or
% uems_update.pl --install listall

If you request a fresh install but have an existing installation on the system, the previous installation will be renamed (“uems.<release number>”) unless the “--scour” flag is passed, in which case previous installation will be wiped off the face of your computer. Any computational domains that reside in the current “uems.<release number>/runs” directory will be transferred to the new “uems>/runs” and updated unless nullified by the “--noruns” flag. If you have domains located elsewhere that you want imported, then you will need to use the “--import=<dir>” option.
b. **Option:** `--dvd <path to dvd rom drive>`

**What this option can and can not do for you:**

You would use the `--dvd` option only if you were installing off a DVD (yes, DVDs are available) and had to copy the `ems_install.pl` routine from the DVD to another location because you could not mount the drive with executable permission. See Section 2.3.3 for guidance on DVD installation.

c. **Option:** `--import=<path to domain directories>[/<domain>]`

**What this option can and can not do for you:**

The `--import=<path to domain directories>` flag allows you to import existing domain directories from a previous EMS installation during a fresh installation. This option is intended for importing either specific domains from a current $EMS/runs directory, or a group of domains within a non-EMS directory (somewhere else on your system).

For Example, the following:

```
% uems_update.pl --install --import=/usr1/mysavedfiles/uems/runs
```

will import any computational domain directories from “/usr1/mysavedfiles/uems/runs” into the new installation. The domains will be localized and updated with any new configuration files, retaining the previous configurations whenever possible. If you have been living right, all will go as planned and your configuration settings will be preserved. *Beginning to sweat a bit now aren’t you?*

If you wish to import only a single domain from a directory, then simply add the domain directory to the end of the string:

```
% uems_update.pl --install --import=/save/uems/runs/bigstorm
```

in which case the “bigstorm” computational domain will be imported to “<uems>/runs” and processed following installation.

You may pass more than one instance of “--import” if you have multiple domains to import:

```
% uems_update.pl --install --import=/saved/runs/bigstorm --import=saved/runs/gostormgo
```

Wherein both “bigstorm” and “gostormgo” domain will be imported to “<uems>/runs” and processed following installation.

If all you wish to do is import computational domains from your current EMS installation (“<uems>/runs”), then it is not necessary to include the “--import” flag as they will be included by default. If you do not want the current domains under “<uems>/runs” to be imported, then include the “--noruns” flag. Including “--noruns” has no impact on the “--import” flag. The “--import” flag is used to import directories moved to a location for safe keeping, which is why you need to include the <path to the domain directories> as an argument; otherwise, the almighty `ems_install.pl` routine would not know where to look.
will not have a clue as to what you want and this sentence will become even longer.

Finally, if you are a bit tentative about all this “import” business, you can always wait until after the installation dust has settled and migrate any existing domain directories to the new location. Then, from each directory run:

```
% ems_domain --localize --update
```

d. **Option:**  --nogeog

**What this option can and can not do for you:**

When including the “--nogeog” flag, you are telling the install routine not to include the large terrestrial data sets. Remember that you actually need these data to run a simulation, but if you already have them locally, you may not need to download them again. You can just copy the files over to the new EMS installation.

However, there are times when these data change and you may need the updated files. Remember, with the EMS, stuff just happens. Just don’t let it happen to you.

e. **Option:**  --noruns

**What this option can and can not do for you:**

When including the “--noruns” flag along with “--install”, you are telling the routine not to import the computational domains from the existing “<ems>/runs” directory to the new installation, which is the default behavior. Passing “--noruns” has no effect on the “--import” option, so go ahead and use them together, or not at all, with complete reckless abandon.

f. **Option:**  --workshop

**What this option can and can not do for you:**

Passing the “--workshop” flag instructs the ems_install.pl routine to include any of the classroom exercises available with a new release or update. If you are updating your system and the files were installed previously then any updates to these files will automatically be included. Conversely, the workshop packages will be excluded during an update if they were not installed previously. You have no control of this fact of life.

These packages are fairly large so if you don’t use them there is no reason to have them installed.
Chapter 02 – Installation of the EMS

---

**g. Option: --relsdire <path/some directory>**

**What this option can and can not do for you:**

Passing the “--relsdire <path/some directory>” flag will override the default location where the downloaded EMS release package files will reside. When installing the system, the tar files are placed in the "<uem>/release/<release number>" directory unless you pass the “--relsdire” option. The default is best unless you have a good reason to change.

**h. Option: --scour**

**What this option can and can not do for you:**

Passing the “--scour” flag will result in any previous EMS installations being removed from your system. The default behavior, i.e., not passing “--scour”, is for the utility to rename any existing installations to “<uem>.<version>”, which is probably what you want, at least until you are happy with the new release.

**i. Option: --tarfile=<tarfile> --tarfile=<tarfile>**

**What this option can and can not do for you:**

Passing “--tarfile” allows you to install or reinstall selected package tar files from an EMS server or local repository. This option will only work with existing EMS installations, and you can only install a tar file for your current release, simply because it would be silly to install a tar file from a different release.

The argument to “--tarfile” is a package tar file that you previously identified by running ems_install.pl --install listall. The name of the file must exactly match a file on the server and the “--tarfile” flag may be passed multiple times.

Here is another fine example of the ems_install.pl power:

```
% ems_install.pl --install --tarfile release.3.2.1.5.2.beta.utils.ncview.tbz
```

The “--install” flag must be included. Just because I said so, that’s why.

The above example will install the ncview package tar file for the EMS release 3.2.1.5.2.beta, assuming that is the same as the currently installed release. The ems_install.pl utility will also do any necessary configuration, because it appreciates your patronage.
Chapter 02 – Installation of the EMS

j. **Option:** --package <package>=[x64|x32]

**What this option can and can not do for you:**

Passing “--package” allows the user to override the default system architecture, either 32- or 64-bit, when installing binary package files. This option is not to be used for installing the primary EMS system binaries (See the “--x64” and “--x32” options), but rather, it is used for downloading and installing the individual packages contained in the “<uems>/utils” directory.

The argument to “--package” is a string (e.g. “nawips”) that will be matched to a tar file package. The string may be followed by a semi-optional “=x64” or “=x32” that will be used to select the 64- or 32-bit version of the binaries respectively. The string is “semi-optional” because not including it will result in the default behavior of installing the correct binaries for your system.

Here is an example:

```
% ems_install.pl --install [other stuff] --package mpich=x32
```

The above example requests the installation of the 32-bit mpich2 binaries on a user's machine. If the machine is 64-bit and the default target binaries have not been overridden, then the 32-bit mpich2 package will be downloaded and installed. If the machine is already 32-bit, then the user is just wasting his or her time.

Note that the matching string, “mpich”, and the requested architecture, “x32”, are separated by an equal sign (=). Yes, the “=” must be included or this will not be your lucky day.

If the user wants to override the default tar file for multiple packages, then the “--package” flag may be passed multiple times:

```
% ems_install.pl --install --package mpich=x32 --package grads=x32
```

2.5 **Updating your EMS with passion**

Eventually, you will want to update your old, crusty, buggy, and yet well-loved EMS with a newer and shinier model containing fresh bugs and that new EMS smell. When that moment arrives, you can call upon your installation tool (yeah, it’s a tool now) to tackle the job. It will someday be known as the Swiss Army knives of installation tools, but for now it’s just the leather awl of installation tools.

Just as with the installation processes, the great EMS unwashed should embrace the most basic of updating commands:

```
% ems_install.pl --update [release version|list|listall] [additional options]
```

Note that the release version is optional, as indicated by the [square brackets], since the default is the most current release. It is recommended that you not include the update release and allow the ems_install.pl to do its magic. If you have been a slacker and find yourself a few dozen updates behind, no problem, as the ems_install.pl will figure out what updates you need to bring your system up to date. With any (a lot of) luck, the utility will download and install each missing update until you are current and just as brilliant as
the day you installed your first EMS. At least that’s the plan.

Alternative arguments to “--update” are “list” and “listall”, which have replaced the “--list” and “--listall” flags respectively. Passing “list” will provide a listing of all the current releases available on the EMS servers for which you are eligible. Including “listall” will include a summary of the individual package files for each release listed by “list”.

If you have any domains located in your “<uems>/runs” directory, then they will also be updated with new configuration files and your existing configuration settings will be retained (no, really!). At the risk of a complete failure, if you do not wish to update your domains, then include “--noconf” flag. The update process does not include a re-localizing the domains under “<ems>/runs” by default but may be accomplished by including the “--localize” flag.

As you may or may not recall, when you installed the system, an entry was placed in your crontab file that allows you to automatically download and install updates. If you fancy this auto-update feature, then you must enable the crontab entry by removing the comment, “#”, from the command line. If automation makes you nervous, you Luddite, then keep it disabled and fly manually while following the guidance provided.

### 2.6 Potentially useful update options and flags

Listed below is a description of some of the more useful flags and options available during an update of the EMS. While the options presented in this section are only used during an update, additional options are available for both installations and updates in the next section, so don’t miss out.

a. **Option: --allyes**

**What this option can and can not do for you:**

Passing the “--allyes” flag tells the ems_install.pl routine, “Yes, yes, I love you and trust you completely.” While I have never heard those words myself, here is your chance to say them without the need to verbalize. The only time you will need the “--allyes” flag is when doing automated updates, since without it the routine will hang while waiting for you to respond to some silly rhetorical question.

b. **Option: --force**

**What this option can and can not do for you:**

Passing the “--force” flag will force the installation of an update even though it’s already installed. Maybe you messed something up and can’t figure out where you went so terribly wrong, or maybe you want to take your frustration out on something. I know, I’ve been there, which is why I included “--force” instead of a “--take_a_deep_breath” option. Just because it feels better.
c. **Option:** --noconf

**What this option can and can not do for you:**

This option is only valid during an EMS update ("--update") and is ignored when doing a clean installation ("--install").

Include the "--noconf" flag if you do not want to update the existing configuration files within each domain directory that exists under "<uems>/runs". This means that all existing domains will be left untouched during an update and will contain the same configuration files and localization as before. Of course, they may no longer work, but you'll take comfort in knowing that you have some control over your life.

Should you have a change of heart or come to find that your EMS (and thus your life) has been ruined by such shortsightedness, you can always update and localize afterwards with the "ems_domain" utility:

```
% ems_domain --update --localize
```

But we are getting ahead of ourselves.

d. **Option:** --updtldir <path/some directory>

**What this option can and can not do for you:**

Passing the "--updtldir <path/some directory>" flag will override the default location where the downloaded EMS update package files will reside. When updating the system, the package tar files are placed in the "<uems>/update/<release number>" directory unless you pass the "--updtldir" option. The default is best unless you have a good reason to change.

### 2.7 Options and flags for every occasion

The options and flags listed below may be used during an update or installation of the EMS, albeit with varying degrees of success.

a. **Option:** --emsdir <path to location of EMS installation>

**What this option can and can not do for you:**

Include the "--emsdir" flag if you want to specify either the location of an existing EMS installation for updates or the future home of a fresh installation when passing the "--install" flag. For the most part, passing this flag is unnecessary, as the ems_install.pl routine will figure most everything out and prompt you for any additional information. So unless you have a good reason for using it, don't.
b. Option:  **--curl** and **--wget**

What this option can and can not do for you:

Passing the “**--curl**” or “**--wget**” flag instructs the installation routine to use a specific utility for http communication. The default is to use whatever tool is available on your system, or randomly select between the two, but you have the option to say otherwise should one method be failing you.

c. Option:  **--debug**

What this option can and can not do for you:

Include the “**--debug**” flag if you want the routine to print out all sorts of information about what is going on inside its brain while it’s attempting to figure out what’s going on inside of yours.

d. Option:  **--emshost <hostname or IP address - emsdata1|emsdata2|soostre>**

What this option can and can not do for you:

Including the “**--emshost**” option instructs the installation utility to contact the specified EMS host for information on update and installation packages. The default behavior is to randomly contact one of the official EMS servers, but you can request a different server should there be any.

e. Option:  **--nolocal**

What this option can and can not do for you:

Passing the “**--nolocal**” option overrides the default behavior of looking for existing EMS package files in the “<uems>/updates” or “<uems>/release” directories. Even if any package tar files are available locally, just look the other way, and download fresh files from a remote location. The old ones smelled funny and were attracting fruit flies.

f. Option:  **--[no]localize**

What this option can and can not do for you:

Including the “**--[no]localize**” flag specifies whether to re-localize any computational domains under the “<uems>/runs” directory. When importing domains during a fresh installation, the default is to re-localize each incoming domain, so passing “**--nolocalize**” would turn this step off. When updating the EMS, the default is to not localize (“**--nolocalize**”) the domains so pass the “**--localize**” flag to override this behavior.

Should you have a change of heart or come to find that your EMS (and thus your life) has been ruined by such shortsightedness, you can always update and localize afterwards with the “ems_domain”
utility:

%  ems_domain --update --localize

But we are getting ahead of ourselves again.

g.  **Option:**  --nounpack

**What this option can and can not do for you:**

Include the “--nounpack” flag if you do not want to install the downloaded EMS package files (the default is to install them), whether for a new installation or an update. Just let them sit in the “<uems>/release|updates/<release number>” directory until they’ve cooled off.

This flag should be named “--noinstall”, but I thought that might be confusing; however, you can try “--noinstall” and see how far it gets you.

h.  **Option:**  --repodir <path to release directories>[/<release>]

**What this option can and can not do for you:**

Use the “--repodir” flag when you want to grab release or update package files from a local source rather than downloading them from the EMS servers. The argument to “--repodir” is the path to a directory containing one or more EMS release or update packages:

%  ems_domain --update|install --repodir=<path to release directories>[/<release>]

The path, <path to release directories>, should lead to a directory where the individual release subdirectories are located (<release>). For example, on your local system there exists a directory containing EMS releases named “/usr/local/uems/releases”. Located beneath “/usr/local/uems/releases” are subdirectories, each containing the package tar files for that release or update. The name of each release subdirectory should represent the release number, such as:

“/usr/local/uems/releases/3.2.1.3.01”
and
“/usr/local/uems/releases/3.6.1.15.56”
and
“/usr/local/uems/releases/3.6.1.15.04”

Then, when using “--repodir”,

%  uems_update.pl --install|update --repodir /usr/local/uems/releases

The install routine will read the contents of any directories below <path to release directories> and select the most current update (“--update”) or release (“--install”).
While you do not need to include a specific release directory as part of the argument to `--repodir`, you may include one should you be targeting a specific release or update, e.g.,

```
% uems_update.pl --install --repodir /usr/local/uems/releases/3.6.1.15.04
```

By including a specific release or update directory you are overriding the default behavior of selecting the most recent package. *This use will also invalidate any arguments passed to `--update` or `--install`.*

i. **Option:** `--x32` and `--x64`

**What these options can do for you:**

Passing either the `--x32` or `--x64` flag will override the machine architecture type, either 32- or 64-bit, and install the specified version of the EMS package binaries. You probably do not need these flags.