

What are Admin Tasks?

- [What are Admin Tasks?](#)
- [Registry format](#)
- [Output format options?](#)
- [How do I get user input for a task?](#)
- [Where do my Admin Tasks scripts go?](#)

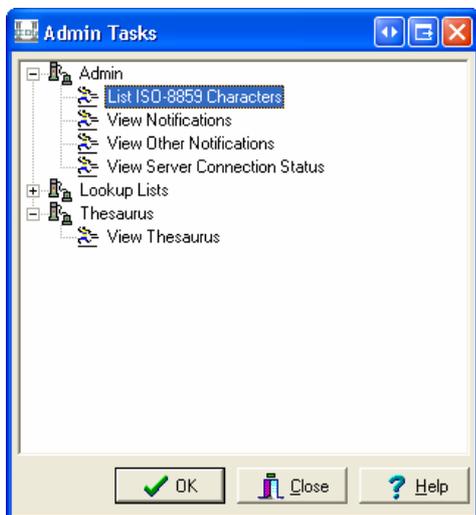
What are Admin Tasks?

The *Admin Tasks* utility is an interface for viewing and invoking tasks that run on the EMu server. Admin Tasks can:

- Execute a command on the server to alter server based files, e.g. a script to change a user password.
- Run a server side report and display the output on a client machine. It may be more efficient and flexible to generate some reports on the EMu server rather than on a client machine. Reports can be output in a numbers of ways, all compatible with the standard EMU reporting facility.
- Create a report that requires user input. An *Admin Task* can ask the user to supply values that can be used as arguments in a server side script.

An *Admin Task* is configured in the EMu Registry. It consists of the name of a script to execute, along with optional instructions for how the output should be displayed. The script command may contain optional prompts asking the user to supply an input value.

Selecting the *Admin*  Admin button on the EMu Command Centre displays the *Admin Tasks* window. This lists all tasks grouped by module:



A user will only see modules in the Admin Tasks list for which they have access rights.

A virtual module of Admin Tasks will display under the Admin heading for all tasks unrelated to any particular module.

To invoke a task, select it in the Admin Tasks list and select the **OK** button. If a task requires user input, the Admin Parameters dialogue box will display. The format of the dialogue will depend on the input parameters requested. Any output generated will display according to the specifications of the Admin Task Registry entry.

The Admin Tasks button will only display for a user in the Command Centre if *eadmin* has been added to their Table Access Registry entry.

Registry format

The format for Admin Tasks Registry entries is:

```
User | username | Table | table | Admin Task | taskname | script  
Group | groupname | Table | table | Admin Task | taskname | script  
Group | Default | Table | table | Admin Task | taskname | script
```

Where:

taskname is the task descriptor shown in the Admin Tasks list under the *table* module heading (e.g. View Notifications in the screenshot above).

table identifies the table heading under which the *taskname* is listed in the Admin Tasks list. The format of *table* is the table's backend name (e.g. *eparties*, *emultimedia*, etc), but it will display in the Admin Tasks list as the module name (e.g. Parties, Multimedia).

A special table name of *eadmin* is used to add entries under the Admin virtual module.

The format of *script* is:

```
{output format} command
```

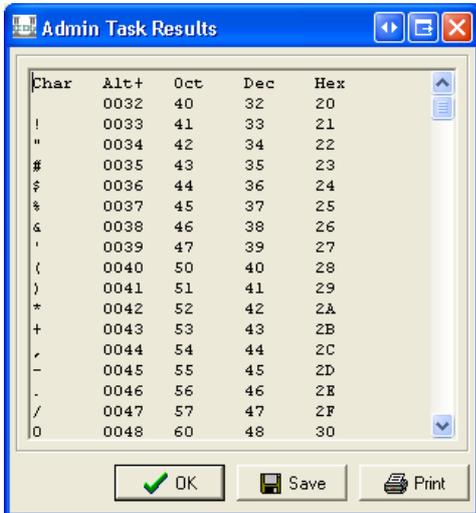
Where:

{output format} is optional. See below for details.

command must be specified. The *command* is the name of the script to be executed on the server. The *command* should reside in the admin directory on the EMu server machine. It is possible to prompt the user for online arguments which are passed through to the *command*. If an *{output format}* is not supplied, any output generated by the *command* is displayed in the Admin Task Results dialogue box.

Output format options

It is possible to specify how output generated by an Admin Task is handled. By default, output will display in the Admin Task Results dialogue box:



Output can be saved and printed.

Specifying output format options makes it possible to display the results of an Admin Task using the standard EMu reporting facility.

The format of output options is:

{action location options}

Where action values are:

- Save** output is saved to file and not displayed on-screen.
In this case the `location` value defines where the file should be saved.
There are no options for this action.
- Launch** output is saved to file as specified by the `location` setting.
When any output is saved, `options` specify which file(s) should be *launched*. The application that launches the file is the application registered for that file type in the Windows Registry.
The files specified by `options` can be hard coded paths or file placeholders of the form `%n` where `n` is the file number in the list of files saved. Thus `%1` is the first file saved, `%2` the second and so on. If a placeholder is specified greater than the number of files generated, it is ignored.
Once launched, a file is removed from the server. Use the `%l` option to stop the removal of files generated on the server (note that files on the client are never removed).
Use the `%f reportfile` option to launch a report. The `reportfile` is the name of the file used to generate the report. The report file is loaded into a cache on the client machine and the option is replaced with the full pathname to the cached report file.
If a generic report is being launched (i.e. a single report that is used for multiple databases) the `%m module` option may be used. This

results in any files that are transferred being sent to the report area for *module* and then the specified report file is taken from *modules* report directory.

Run output is saved to file as specified by the `location` setting.
When any output is saved, `options` define the client side command to be executed. All options available for Launch can be used for Run.

Where `location` values are:

Temp	saves an output file in the Windows temporary file directory (usually C:\Winnt\Temp). The filename used is the same as the filename created on the server. The user is not asked for any input.
ODBC	saves an output file under the ODBC Data Source for the current module. The module used is the one under which the Admin Task is defined. It is expected that the output files generated are in a format suitable for use by the Microsoft ODBC Text Driver.
File	displays the <i>Save File</i> dialogue as each file is copied from the server to the client; the user can specify where the file should be saved.
Directory	when the first file is copied from the client to the server the user is asked to specify a directory into which all files will be saved. The filename used is the same as the filename created on the server.

Note that files generated on the server by the Admin Task and transferred to the client are removed from the server after they are transferred.

How do I get user input for a task?

When a command is run as part of an Admin Task it is possible to prompt the user to provide input (an argument) for the command. The prompt is placed where the command requires an argument: the user's input becomes the argument for the command. The format of the prompt is:

```
[datatype:prompt[:lutname[:separator]]]
```

Where `datatype` is one of:

attachment	It is possible to have a user specify an attachment record (IRN) as an argument for a command. The syntax of the <code>attachment</code> argument is:
------------	---

```
[attachment:prompt:table]
```

where:

`table` is the back-end table to be referenced

`prompt` is explained below

The attachment record's IRN becomes the argument for the command.

date	Displays an edit box in which the user can enter a value.
time	Displays an edit box in which the user can enter a value.
number	Displays an edit box in which the user can enter a value.

<code>text</code>	Displays an edit box in which the user can enter a value.
<code>password</code>	Displays an edit box in which the user can enter a value. Values entered display as an asterisk (*).
<code>combobox</code>	Displays a combo box with a list of values found in the <code>lutname</code> lookup list (see below). The user can make a single selection from the list.
<code>listbox</code>	Displays a list box with a list of values from the <code>lutname</code> lookup list (see below). The user can make multiple selections. The values selected are joined together by an optional separator.

The name of a lookup list in the *eluts* database used to populate the `combobox` and `listbox` with values.

If a level of a (hierarchical) lookup list is required, the `lutname` should be specified as `lutname;level`. As an example, if the second level of the Place Name lookup list was required, then the `lutname` parameter would be `Place Name;2`.

Configuring the datatype

It is possible to specify a default value, width and height for the `datatype`. The format of the command is:

```
[datatype;default value; width; height:...]
```

All parts, except for `datatype` are optional.

Where do my Admin Tasks scripts go?

Admin Tasks scripts should be placed in the `admin` directory on the EMu server. This directory is part of the standard `PATH` setup for EMu on the server, hence absolute path names are not required. Upgrades of EMu do not remove files existing in the `admin` directory.

Also *emusync* will not remove client specific commands found in the `admin` directory (unlike the `bin` and `utils` directories), but will add commands found in the master client. Client specific Admin Tasks (that is tasks specific to a given client, as opposed to general purpose) should be placed in the client's `admin` directory.

Where an *output option* has been specified, the script should output a list of filenames, one per line. These files will be used by the *output option*. If an *output option* is not specified, the script should just print its output, which will be shown in the Admin Task Results form.