**Hour Glass Scheduler Configuration Details**

**Scheduler date/time format**

The following documentation attempts to explain this mysterious string/pattern that is used to schedule the hour\_glass jobs:

<cron-expression>59 \* \* \* \* ? \*</cron-expression>

As a background, the hour\_glass scheduler is a java wrapper for the crontab functionality available on the linux operating system.

Commands are executed by hour\_glass when the minute, hour, and month of year fields match the current time, and when at least one of the two day fields (day of month, or day of week) match the current time.

A field may be an asterisk (\*), which always stands for "first-last".

Ranges of numbers are allowed. Ranges are two numbers separated with a hyphen. The specified range is inclusive. For example, 8-11 for an "hours" entry specifies execution at hours 8, 9, 10 and 11.

Lists are allowed. A list is a set of numbers (or ranges) separated by commas. Examples: "1,2,5,9", "0-4,8-12".

Step values can be used in conjunction with ranges. Following a range with "/" specifies skips of the number's value through the range. For example, "0-23/2" can be used in the hours field to specify command execution every other hour (the alternative in the V7 standard is "0,2,4,6,8,10,12,14,16,18,20,22"). Steps are also permitted after an asterisk, so if you want to say "every two hours", just use "\*/2".

Names can also be used for the "month" and "day of week" fields. Use the first three letters of the particular day or month (case doesn't matter). Ranges or lists of names are not allowed.

Note: The day of a command's execution can be specified by two fields - day of month, and day of week. If both fields are restricted (i.e., aren't \*), the command will be run when either field matches the current time. For example, "30 4 1, 15 \* 5" would cause a command to be run at 4:30 am on the 1st and 15th of each month, plus every Friday.

Instead of the first five fields, one of eight special strings may appear:

|  |  |  |
| --- | --- | --- |
| String | Meaning | Comments |
| @reboot | Run once, at startup. |  |
| @yearly | Run once a year, "0 0 1 1 \*". |  |
| @annually | (same as @yearly) |  |
| @monthly | Run once a month, "0 0 1 \* \*". |  |
| @weekly | Run once a week, "0 0 \* \* 0". |  |
| @daily | Run once a day, "0 0 \* \* \*". |  |
| @midnight | (same as @daily) |  |
| @hourly | Run once an hour, "0 \* \* \* \*". |  |

An example of hour\_glass format with commented fields is as follows:

# Minute Hour Day of Month Month Day of Week

# (0-59) (0-23) (1-31) (1-12 or Jan-Dec) (0-6 or Sun-Sat)

0 2 12 \* \*

This line executes the "find" command at 2AM on the 12th of every month.

**Examples**

Here are some more examples of hour\_glass settings.

This line executes the job command every minute of every hour of every day of every month.

\* \* \* \* \*

This line executes the job command every 12am and 12pm on the 1st day of every 2nd month.

0 0,12 1 \*/2 \*

This line executes the job command every 2am on the 1st through the 10th of each month.

0 2 1-10 \* \*

This line is and example of running a hour\_glass job every month at 4am on Mondays, and on the days between 15-21. This is because using the day of month and day of week fields with restrictions (no \*) makes this an "or" condition not an "and" condition. Both will be executed.

0 4 15-21 \* 1

Run on every second Sunday of every month.

0 4 8-14 \* \*

**Troubleshooting**

1) When hour\_glass job is run from the current users it is executed as that user. It does not however source any files in the users home directory like their .cshrc or .bashrc or any other file. If you need hour\_glass to source (read) any file that your script will need you should do it from the script hour\_glass is calling. Setting paths, sourcing files, setting environment variables, etc.

2) If the users account has permission to access hour\_glass but no usable shell in /etc/passwd then the hour\_glass will not run. You will have to give the account a shell for the hour\_glass to run.

3) If your hour\_glass jobs are not running check if the hour\_glass daemon is running.

4) You also might want to check if the /etc/security/access.conf file exists. You might need to add your user in there.

5) hour\_glass parameters and arguments are not parsed for environmental substitutions. You cannot use things like $PATH, $HOME, or ~/sbin. You can set things like MAILTO= or PATH= and other environment variables the /bin/sh shell uses.

6) Watch out using day of month and day of week together. Day of month and day of week fields with restrictions (no \*) makes this an "or" condition not an "and" condition. When either field is true it will be executed.