Locating '.ecf' file

The script file can be any file.(i.e. .ecf, .py). If you want to use pure python tasks, then add the following variables.

```
edit ECF_EXTN .py.  # search for files matching task name and extension .py
```

This is required since the default extension is '.ecf'. For more examples see the cookbook

ecFlow looks for files using the following search process when trying to locate the '.ecf' associated with a task.

- First, it uses the variable **ECF_SCRIPT** and tries to open that file. **ECF_SCRIPT** is generated from **ECF_HOME/SUITE/FAMILY/TASK**.

- **DEFAULT**:
  Otherwise, if variable **ECF_FILES** exists, it must point to a directory that is searched in reverse order, e.g. let's assume that the node name is **/o /12/fc/model** and that **ECF_FILES** is defined as **/home/ecmwf/emos_ECF/def/o/ECFfiles**. The order of files tried is as follows:

  ```
  /home/ecmwf/emos_ECF/def/o/ECFfiles/o/12/fc/model.ecf
  /home/ecmwf/emos_ECF/def/o/ECFfiles/12/fc/model.ecf
  /home/ecmwf/emos_ECF/def/o/ECFfiles/fc/model.ecf
  /home/ecmwf/emos_ECF/def/o/ECFfiles/model.ecf
  ```

  This may at first be seen as overkill, but you can put all the files for a number of suites in one distinct file system/directory.

  if the original **ECF_SCRIPT** did not exist, ecFlow will check the directories for the job file in **ECF_HOME** (**ECF_SCRIPT** is derived from **ECF_HOME** .). If a directory does not exist, ecFlow will create it. This helps to clean up old job-files and output and makes the maintenance of the scripts easier. It also guarantees that the output can be redirected into the file without the job creating the directory. (e.g. NQS option **QSUB -ro**, or when using redirection.)

  Using **ECF_FILES** means that you do not have to create and maintain a link-jungle, e.g. the **model.ecf** above exists in a number of different families in ECMWF operational suites. The file is placed in a directory **.../ECFfiles/fc/** and used by nodes **/o/00/fc/model**, **/o/12/fc/model** etc. This *trick* works nicely as long as there are no other tasks named **model** in the same family.

  ALTERNATIVE:

  The search can be reversed, by **adding** a variable ECF_FILES_LOOKUP, with a value of "prune_leaf". (from ecflow 4.12.0) .

  ```
  edit ECF_FILES_LOOKUP prune_leaf
  ```

  Then ecFlow will use the following search pattern

  ```
  /home/ecmwf/emos_ECF/def/o/ECFfiles/o/12/fc/model.ecf
  /home/ecmwf/emos_ECF/def/o/ECFfiles/o/12/model.ecf
  /home/ecmwf/emos_ECF/def/o/ECFfiles/o/model.ecf
  /home/ecmwf/emos_ECF/def/o/ECFfiles/model.ecf
  ```

  > ⚠ However please be aware this will also affect the search in ECF_HOME

  If the ECF_FILES directory does not exist, the server will try variable substitution. This allows additional configuration.

  ```
   edit ECF_FILES /home/ecmwf/emos/def/o/%FILE_DIR:ECFfiles%
  ```

- Finally, it searches ECF_HOME directory. (using the DEFAULT or ALTERNATIVE lookup methods)