ESGF Local Node Deployment Tutorial

This page shows how to deploy an ESGF Node that provides data, index and idp services and belongs to the esgf-test federation. The purpose of this node is to test the process of publication in the ESGF.

This page assumes that command are executed by the root user (not sudo).

Index

0. Prerequisites

1. You have to create a globus account - ?https://www.globusid.org/create

1. Previous installation clean up

Execute `/usr/local/bin/esg-node stop` in order to stop the current ESGF services (in case they are running).

```
[root@spock ~]# /usr/local/bin/esg-node stop


 EEEEEEEEEEEEEEEEEEEE   SSSSSSSSSSSSSSS         GGGGGGGGGGGGGFFFFFFFFFFFFFFFFFFFFF
 E::::::::::::::::::::E SS:::::::::::::::S     GGG::::::::::::GF::::::::::::::::::::F
 E::::::::::::::::::::ES:::::SSSSSS::::::S   GG:::::::::::::::GF::::::::::::::::::::F
 EE::::::EEEEEEEEE:::ES:::::S     SSSSSSS  G:::::GGGGGGGG::::GFF::::::FFFFFFFFF::::F
   E:::::E       EEEEEES:::::S             G:::::G       GGGGGG  F:::::F       FFFFFF
   E:::::E             S:::::S            G:::::G               F:::::F
   E::::::EEEEEEEEEE    S::::SSSS         G:::::G               F::::::FFFFFFFFFF
   E::::::::::::::E     SS::::::SSSSS    G:::::G    GGGGGGGGGG  F::::::::::::::F
   E::::::::::::::E       SSS::::::::SS  G:::::G    G::::::::G  F::::::::::::::F
   E::::::EEEEEEEEEE          SSSSSS::::S G:::::G    GGGGG::::G  F::::::FFFFFFFFFF
   E:::::E                         S:::::SG:::::G        G::::G  F:::::F
   E:::::E       EEEEEE            S:::::S G:::::G       G::::G  F:::::F
 EE::::::EEEEEEEE:::::ESSSSSSS     S:::::S  G:::::GGGGGGGG::::GFF:::::::FF
 E::::::::::::::::::::ES::::::SSSSSS:::::S   GG::::::::::::::GF::::::::FF
 E::::::::::::::::::::ES::::::::::::::SS      GGG::::::GGG:::GF::::::::FF
 EEEEEEEEEEEEEEEEEEEEEE SSSSSSSSSSSSSSS           GGGGGG   GGGGFFFFFFFFFFF.llnl.gov

Checking that you have root privs on spock.meteo.unican.es... [OK]
Checking requisites...

Using IP: 193.144.184.40
Stopping search services...
Using solr_workdir=/usr/local/src/esgf/workbench/esg/solr-5.5.3
Using solr_install_dir=/usr/local/solr-home/slave-8983
Using solr_data_dir=/esg/solr-index/slave-8983
Using solr_server_dir=/usr/local/solr
Using solr_logs_dir=/esg/solr-logs
Using esg_dist_url=http://esg-dn2.nsc.liu.se/esgf/dist
sudo: source: command not found
Sending stop command to Solr running on port 8983 ... waiting 5 seconds to allow Jetty process 16339 to stop gracefully.
```

```
Sending stop command to Solr running on port 8984 ... waiting 5 seconds to allow Jetty process 16554 to stop gracefully.
Stopping Globus Services for Data-Node... (GridFTP) stop_globus_services for datanode
globus-gridftp-server: unrecognized service
Stopping Globus Services for Index-Node... (MyProxy server) stop_globus_services for gateway
Stopping myproxy-server:                                   [  OK  ]
No MyProxy Process Currently Running...
Tomcat (jsvc) process is running...

stop tomcat: /usr/local/tomcat/bin/jsvc -pidfile /var/run/tomcat-jsvc.pid -stop org.apache.catalina.startup.Bootstrap
(please wait)
postmaster (pid  16024) is running...
Stopping postgresql service:                               [  OK  ]
Stopping httpd:                                            [  OK  ]
Running shutdown hooks...

---------------------------
Running Node Services...
node type: [ data index idp compute ] (60)
---------------------------
---------------------------
```

Execute `source /usr/local/bin/esg-purge.sh && esg-purge all`

2. Installation from scratch

Change directory to /usr/local/bin/

```
[root@spock ~]# cd /usr/local/bin/
```

```
[root@spock bin]# wget -O esg-bootstrap http://distrib-coffee.ipsl.jussieu.fr/pub/esgf/dist/devel/esgf-installer/2.4/esg-b
[root@spock bin]# chmod 555 ./esg-bootstrap
[root@spock bin]# ./esg-bootstrap
```

Your directory should look like this:

```
[root@spock bin]# ls
esg-bootstrap  esg-functions  esg-init  esg-node  esg-purge.sh  jar_security_scan  setup-autoinstall
```

Check your node's version:

```
[root@spock bin]# ./esg-node --version
Version: v2.4.24-master-release
Release: Bifrost
Earth Systems Grid Federation (http://esgf.llnl.gov)
ESGF Node Installation Script
```

Set node's type:

```
[root@spock bin]# ./esg-node --set-type data idp index
node type set to: [ index data idp ] (28)
```

Install the node:

```
[root@spock bin]# ./esg-node --install
```

```
Please select the ESGF distribution mirror for this installation (fastest to slowest):
       ----------------------------------------
        [1] http://dist.ceda.ac.uk/esgf
```

```
        [2] http://esg-dn2.nsc.liu.se/esgf
        [3] http://aims1.llnl.gov/esgf
        [4] http://distrib-coffee.ipsl.jussieu.fr/pub/esgf
       ----------------------------------------
select [1] > 1
```

```
Are you ready to begin the installation? [Y/n]
Configured host IP address does not match available IPs...
Detected multiple IP addresses bound to this host...
Please select the IP address to use for this installation
       ----------------------------------------
       [0] : 193.xxx.xxx.xxx
       [1] : 192.xxx.xxx.xxx
       ----------------------------------------
select [] > (select the one that fits your case)
```

```
Welcome to the ESGF Node installation program! :-)

What is the fully qualified domain name of this node? [spock.meteo.unican.es]:
What is the admin password to use for this installation? (alpha-numeric only) []:
Please re-enter password:
What is the name of your organization? [unican]:
Please give this node a "short" name: []: unican
Please give this node a more descriptive "long" name []: unican
What is the namespace to use for this node? (set to your reverse fqdn - Ex: "gov.llnl") [es.unican.meteo]:
What peer group(s) will this node participate in? (esgf-test|esgf-prod) [esgf-test]:
What is the default peer to this node? [spock.meteo.unican.es]:
What is the hostname of the node do you plan to publish to? [spock.meteo.unican.es]:
What email address should notifications be sent as? []:
Is the database external to this node? [y/N]:
Please enter the database connection string...
(form: postgresql://[username]@[host]:[port]/esgcet)
What is the database connection string? [postgresql://dbsuper@localhost:5432/esgcet]: postgresql://
entered: postgresql://dbsuper@localhost:5432/esgcet
What is the (low priv) db account for publisher? [esgcet]:
What is the db password for publisher user (esgcet)? []:
```

```
Enter password for postgres user dbsuper:
Re-enter password for postgres user dbsuper:
Please Enter PostgreSQL port number [5432]:>
```

```
Would you like a "system" or "user" publisher configuration:
       ----------------------------------------
       *[1] : System
        [2] : User
       ----------------------------------------
        [C] : (Custom)
       ----------------------------------------
select [1] >

You have selected: 1
Publisher configuration file -> [/esg/config/esgcet/esg.ini]

Is this correct? [Y/n]
Your publisher configuration file will be: /esg/config/esgcet/esg.ini
What is your organization's id? [unican]:
```

```
Would you like to use the DN: (OU=ESGF.ORG, O=ESGF) ? [Y/n]:
...
```

```
Please enter the password for this keystore   :
```

```
Enter a single ip address which would be cleared to access admin restricted pages.
You will be prompted if you want to enter more ip-addresses

Do you wish to allow further ips? y/n
n
```

```
Create user credentials
Please enter username for tomcat [dnode_user]:
Please enter password for user, "dnode_user" [********]:
Would you like to add another user? [y/N]:
```

```
Please Enter the public (i.e. routable) IP address of this host [193.xxx.xxx.xxx]:>
Do you wish to use an external IDP peer?(N/y):
```

```
Do you want to continue with the Globus installation and setup? [Y/n] :
Do you want to register the MyProxy server with Globus? [Y/n]:
Please provide a Globus username []: YOUR-GLOBUS-USER
Globus password []:
```

When finished, you should see something like this:

```
Running Node Services...
node type: [ data index idp ] (29)
---------------------------
myproxy-s 23071     root    5u  IPv4 1526752      0t0  TCP *:7512 (LISTEN)
java      26088     solr   28u  IPv6 1591850      0t0  TCP 127.0.0.1:7983 (LISTEN)
java      26088     solr   92u  IPv6 1591986      0t0  TCP *:8983 (LISTEN)
java      26257     solr   28u  IPv6 1592730      0t0  TCP 127.0.0.1:7984 (LISTEN)
java      26257     solr   92u  IPv6 1593098      0t0  TCP *:8984 (LISTEN)
postmaste 29509 postgres    3u  IPv6 1449862      0t0  TCP [::1]:5432 (LISTEN)
postmaste 29509 postgres    4u  IPv4 1449863      0t0  TCP 127.0.0.1:5432 (LISTEN)
httpd     12706     root    4u  IPv6 1512235      0t0  TCP *:80 (LISTEN)
---------------------------



Finished!...
In order to see if this node has been installed properly you may direct your browser to:
http://spock.meteo.unican.es/thredds
http://spock.meteo.unican.es/esg-orp
http://spock.meteo.unican.es/

Your peer group membership -- : [esgf-test]
Your specified "default" peer : [spock.meteo.unican.es]
Your specified "index" peer - : [spock.meteo.unican.es] (url = http://spock.meteo.unican.es/)
Your specified "idp" peer --- : [spock.meteo.unican.es] (name = SPOCK.METEO.UNICAN.ES)
Your temporary certificates have been placed in /etc/tempcerts
You can install them by executing this : esg-node --install-keypair /etc/tempcerts/hostcert.pem /etc/tempcerts/hostkey.pem
When promped for the chainfile, specify: /etc/tempcerts/cacert.pem

[Note: Use UNIX group permissions on /esg/content/thredds/esgcet to enable users to be able to publish thredds catalogs fr
%> chgrp -R <appropriate unix group for publishing users> /esg/content/thredds

       ------------------------------------------------------
       Administrators of this node should subscribe to the
       esgf-node-admins@lists.llnl.gov by sending email to: majordomo@lists.llnl.gov
       with the body: subscribe esgf-node-admins
```

```
       ------------------------------------------------------

v2.4.24-master-release

Writing additional settings to db. If these settings already exist, psql will report an error, but ok to disregard.
ERROR:  insert or update on table "permission" violates foreign key constraint "permission_user_id_fkey"
DETAIL:  Key (user_id)=(1) is not present in table "user".
Node installation is complete.
```

Execute the following:

```
[root@spock bin]# ./esg-node --install-keypair /etc/tempcerts/hostcert.pem /etc/tempcerts/hostkey.pem
...
Please set the password for this keystore   :
Please re-enter the password for this keystore:
...
certfile> /etc/tempcerts/cacert.pem
certfile>
...
Is the above information correct? [Y/n]
Is the above information correct? [Y/n]
```

Restart the node:

```
[root@spock bin]# ./esg-node restart
```

Check that everything works (?https://github.com/ESGF/esgf-installer/wiki/ESGF-Post-Installation-Tests).

If the CoG portal does not work follow the instructions on ?https://www.earthsystemcog.org/projects/cog/install_or_upgrade.

Now you should be able to log in the CoG portal using the openid "?https://spock.meteo.unican.es/esgf-idp/openid/rootAdmin" and the password that you chose in the installation process.

Configuration for publishing

The installation process should have created a user in the postgres database, named rootAdmin. You can check it by running `psql -U dbsuper -d esgcet` (to access the postgres cli) and visualizing the table esgf_security.user.

```
esgcet=# select * from esgf_security.user;
id | firstname | middlename |  lastname   |         email          | username  |              password              | dn |
ion_token          | notification_code
----+-----------+------------+-------------+------------------------+-----------+------------------------------------+----
------------------+-------------------
 1 | Admin     |            | User        | emailOfTheAdmin | rootAdmin | hashOfThePassword |    | https://domain/esgf-idp
-b50e-d43692adc5e5 |
```

In order to test the publication, create a new user using the CoG web interface (?https://[index_node_fqdn]). You should click on 'Create Account' and fill the form. Once the user is created using the CoG interface, it should be visible in the esgf_security.user table of the postgres database.

```
esgcet=# select * from esgf_security.user;
id | firstname | middlename |  lastname   |         email          | username  |              password              | dn |
ion_token          | notification_code
----+-----------+------------+-------------+------------------------+-----------+------------------------------------+----
------------------+-------------------
 1 | Admin     |            | User        | emailOfTheAdmin         | rootAdmin | hashOfThePassword                  |
-b50e-d43692adc5e5 |                 0
 2 | zequi     |            | cimadevilla | emailOfZequi            | zequi     | hashOfThePassword                  |
```

Once the user is created, create permissions and roles as follows:

(reference documentation - ?https://acme-climate.atlassian.net/wiki/display/ESGF/Guide+to+ESGF+Publishing+and+Best+Practices)

```
esgcet=# select * from esgf_security.role;
id |   name    |     description
----+-----------+--------------------
 1 | super     | Super User
 2 | none      | None
 3 | default   | Standard
 4 | publisher | Data Publisher
 5 | admin     | Group Administrator
 6 | user      | user role
(6 rows)

esgcet=# select * from esgf_security.group;
id |     name     |     description     | visible | automatic_approval
----+--------------+---------------------+---------+--------------------
 1 | wheel        | Administrator Group | t       | t
 2 | test_group   | test group          | t       | t
 3 | cordex_group | cordex group        | t       | t
(3 rows)

esgcet=# select * from esgf_security.permission;
user_id | group_id | role_id | approved
---------+----------+---------+----------
      2 |        2 |       4 | t
      2 |        2 |       6 | t
      2 |        3 |       6 | t
      2 |        3 |       4 | t
(4 rows)
```

Add the following elements to /esg/config/esgf_policies_local.xml

```
     <policy resource=".*test.*" attribute_type="test_group" attribute_value="user" action="Read"/>
    <policy resource=".*test.*" attribute_type="test_group" attribute_value="publisher" action="Write"/>
    <policy resource=".*cordex.*" attribute_type="cordex_group" attribute_value="user" action="Read"/>
    <policy resource=".*cordex.*" attribute_type="cordex_group" attribute_value="publisher" action="Write"/>
```

Add the following elements to /esg/config/esgf_ats_static.xml

```
    <attribute
       type="test_group"
       attributeService="https://spock.meteo.unican.es/esgf-idp/saml/soap/secure/attributeService.htm"
       description="Test group for test data"
       registrationService="https://spock.meteo.unican.es/esgf-idp/secure/registrationService.htm"/>

   <attribute
       type="cordex_group"
       attributeService="https://spock.meteo.unican.es/esgf-idp/saml/soap/secure/attributeService.htm"
       description="Test group for cordex data"
       registrationService="https://spock.meteo.unican.es/esgf-idp/secure/registrationService.htm"/>
```

Generate your credentials for publication - globus certificate

```
myproxy-logon [ -b ] -s <openid_server> -l <your_esgf_username> -p 7512 -t 72 -o $HOME/.globus/certificate-file
```

The certificate is valid for 72 hours when specified by -t. If you are publishing for the first time, you will need to mkdir $HOME/.globus and use -b to bootstrap its trustroots with the server. The esgf_username is the simply the username portion of your openid rather than the entire openid string, e.g. sashakames, not ?https://pcmdi.llnl.gov/esgf-idp/openid/sashakames

Publish the test dataset

For esgprep and esgpublish to be available, execute `source /etc/esg.env`.

```
[root@spock ~]# esgprep mapfile --project test /esg/data/test/
Collecting files     : 1 files
Mapfile(s) generation: 100% |??????????????????????????????????????????????????????????????| 1/1 files
Mapfile(s) generated : 1 (see /root/mapfiles)
```

```
[root@spock ~]# esgpublish --service fileservice --map mapfiles/test.test.map --project test --thredds --publish --offline
INFO       2017-06-02 14:59:48,405 Replacing files in dataset: test.test, version 1
INFO       2017-06-02 14:59:48,413 File /esg/data/test/sftlf.nc exists, skipping
INFO       2017-06-02 14:59:48,416 New dataset version = 2
INFO       2017-06-02 14:59:48,430 Adding file info to database
INFO       2017-06-02 14:59:48,469 Writing THREDDS catalog /esg/content/thredds/esgcet/1/test.test.v2.xml
INFO       2017-06-02 14:59:48,522 Writing THREDDS ESG master catalog /esg/content/thredds/esgcet/catalog.xml
INFO       2017-06-02 14:59:48,533 Reinitializing THREDDS server
INFO       2017-06-02 14:59:48,830 Publishing: test.test
INFO       2017-06-02 14:59:49,871   Result: SUCCESSFUL
```

Notes:

1. --map must point to the file generated by esgprep mapfile
2. --thredds publish data to the data node
3. --publish publish data to the index node
4. --offline is required for publish the test dataset (Why?)
5. This publication works out of the box because esgf installs by default the required /esg/config/esgcet/esg.test.ini file.

Publish CORDEX datasets

This documentation gives instructions to publish 02_EuroCORDEX_INTERIM_044_v20140616 datasets.

The source code of esgprep must be modified in order to work with symbolic links! See ?https://github.com/IS-ENES-Data/esgf-prepare/pull/3

Change line 308 /usr/local/src/esgf/workbench/esg/esg-publisher/src/python/esgcet/.eggs/esgprep-2.7.1-py2.7.egg/esgprep/drs/handler.py from `globals()[self.mode](self.src, self.dst)` to `symlink(self.src, self.dst)`. See ?https://github.com/IS-ENES-Data/esgf-prepare/blob/master/esgprep/drs/handler.py#L308.

Modify /esg/config/esgcet/esg.cordex.ini to fit your needs. In this case:

```
[project:cordex]

categories =
   project        | enum   | true  | true  | 0
   product        | enum   | true  | true  | 1
   domain         | enum   | true  | true  | 2
   institute      | enum   | true  | true  | 3
   driving_model  | enum   | false | true  | 4
   experiment     | enum   | false | true  | 5
   ensemble       | string | false | true  | 6
   rcm_name       | string | false | true  | 7
   rcm_version    | enum   | false | true  | 8
   time_frequency | enum   | false | true  | 9
   description    | text   | false | false | 99

category_defaults =
   project | CORDEX

filename_format = %(variable)s_%(domain)s_%(driving_model)s_%(experiment)s_%(ensemble)s_%(rcm_model)s_%(rcm_version)s_%(ti

dataset_id = cordex.%(product)s.%(domain)s.%(institute)s.%(driving_model)s.%(experiment)s.%(ensemble)s.%(rcm_name)s.%(rcm_
```

```
directory_format = %(root)s/%(project)s/%(product)s/%(domain)s/%(institute)s/%(driving_model)s/%(experiment)s/%(ensemble)s

product_options = output

domain_options = AFR-44, AFR-44i, ANT-44, ANT-44i, ARC-44, ARC-44i, AUS-44, AUS-44i, CAM-44, CAM-44i, CAS-44, CAS-44i, EAS

institute_options = AUTH-LHTEE, AUTH-Met, AWI, BCCR, CCCma, CHMI, CLMcom, CNRM, CRP-GL, CUNI, DHMZ, DMI, ENEA, GERICS, HMS

driving_model_options = CCCma-CanESM2, CSIRO-QCCCE-CSIRO-Mk3-6-0, CNRM-CERFACS-CNRM-CM5, ECMWF-ERAINT, ICHEC-EC-EARTH, IPS

experiment_options =
  cordex | evaluation | Evaluation
  cordex | historical | Historical
  cordex | rcp26      | RCP2.6
  cordex | rcp45      | RCP4.5
  cordex | rcp60      | RCP6.0
  cordex | rcp85      | RCP8.5

ensemble_pattern = r%(digit)si%(digit)sp%(digit)s

rcm_model_options = AUTH-LHTEE-WRF321B, AUTH-Met-WRF331A, AWI-HIRHAM5, BCCR-WRF331, BCCR-WRF331C, CCCma-CanRCM4, CHMI-ALAD

rcm_version_options = v1, v01, v2, v3, v4, v411, v1a, v5

time_frequency_options = day, fx, mon, sem, 3hr, 6hr

maps = las_time_delta_map, domain_description_map, rcm_name_map

domain_description_map = map(project_id, domain : domain_description)
   cordex | AFR-44  | Africa
   cordex | AFR-44i | Africa
   cordex | ANT-44  | Antarctica
   cordex | ANT-44i | Antarctica
   cordex | ARC-44  | The Arctic
   cordex | ARC-44i | The Arctic
   cordex | AUS-44  | Australasia
   cordex | AUS-44i | Australasia
   cordex | CAM-44  | Central America
   cordex | CAM-44i | Central America
   cordex | CAS-44  | Central Asia
   cordex | CAS-44i | Central Asia
   cordex | EAS-44  | East Asia
   cordex | EAS-44i | East Asia
   cordex | EUR-11  | High-res. Europe
   cordex | EUR-11i | High-res. Europe
   cordex | EUR-44  | Europe
   cordex | EUR-44i | Europe
   cordex | MED-44  | HYMEX Mediterranean
   cordex | MED-44i | HYMEX Mediterranean
   cordex | MNA-22  | Middle East and North Africa
   cordex | MNA-22i | Middle East and North Africa
   cordex | MNA-44  | Middle East and North Africa
   cordex | MNA-44i | Middle East and North Africa
   cordex | NAM-44  | North America
   cordex | NAM-44i | North America
   cordex | SAM-44  | South America
   cordex | SAM-44i | South America
   cordex | WAS-44  | West Asia
   cordex | WAS-44i | West Asia

rcm_name_map = map(project, rcm_model : rcm_name)
```

```
   cordex | AUTH-LHTEE-WRF321B  | WRF321B
   cordex | AUTH-Met-WRF331A    | WRF331A
   cordex | AWI-HIRHAM5         | HIRHAM5
   cordex | BCCR-WRF331         | WRF331
   cordex | BCCR-WRF331C        | WRF331C
   cordex | CCCma-CanRCM4       | CanRCM4
   cordex | CHMI-ALADIN52       | ALADIN52
   cordex | CLMcom-CCLM4-8-17   | CCLM4-8-17
   cordex | CNRM-ALADIN52       | ALADIN52
   cordex | CNRM-ARPEGE51       | ARPEGE51
   cordex | CNRM-ARPEGE52       | ARPEGE52
   cordex | CNRM-ALADIN53       | ALADIN53
   cordex | CRP-GL-WRF331A      | WRF331A
   cordex | CUNI-RegCM4-2       | RegCM4-2
   cordex | DHMZ-RegCM4-2       | RegCM4-2
   cordex | DMI-HIRHAM5         | HIRHAM5
   cordex | ENEA-RegCM4-3       | RegCM4-3
   cordex | GERICS-REMO2009     | REMO2009
   cordex | HMS-ALADIN52        | ALADIN52
   cordex | ICTP-RegCM4-3       | RegCM4-3
   cordex | IDL-WRF331D         | WRF331D
   cordex | IPSL-INERIS-WRF331F | WRF331F
   cordex | IITM-RegCM4-1       | RegCM4-1
   cordex | IITM-RegCM4-4       | RegCM4-4
   cordex | KNMI-RACMO21P       | RACMO21P
   cordex | KNMI-RACMO22E       | RACMO22E
   cordex | KNMI-RACMO22T       | RACMO22T
   cordex | MGO-RRCM            | RRCM
   cordex | MIUB-WRF331A        | WRF331A
   cordex | MOHC-HadGEM3-RA     | HadGEM3-RA
   cordex | MOHC-HadRM3P        | HadRM3P
   cordex | MPI-CSC-REMO2009    | REMO2009
   cordex | NUIM-WRF331F        | WRF331F
   cordex | RMIB-UGent-ALARO-0  | ALARO-0
   cordex | SMHI-RCA4           | RCA4
   cordex | SMHI-RCA4-SN        | RCA4-SN
   cordex | SMHI-RCAO           | RCAO
   cordex | SMHI-RCAO-SN        | RCAO-SN
   cordex | UCAN-WRF331G        | WRF331G
   cordex | UCAN-WRF350I        | WRF350I
   cordex | UCLM-PROMES         | PROMES
   cordex | UHOH-WRF331H        | WRF331H
   cordex | ULg-MAR36           | MAR36
   cordex | UQAM-CRCM5          | CRCM5

las_time_delta_map = map(time_frequency : las_time_delta)
   mon | 1 month
   day | 1 day
   fx  | fixed
   3hr | 3 hours
   sem | seasonal mean
   6hr | 6 hours

project_handler_name = basic_builtin

las_configure = true

thredds_exclude_variables = a, a_bnds, alev1, alevel, alevhalf, alt40, b, b_bnds, basin, bnds, bounds_lat, bounds_lon, dbz

variable_locate = ps,ps_
```

```
variable_per_file = true
```

Generate the DRS structure:

```
esgprep drs list --project cordex /path/to/02_EuroCORDEX_INTERIM_044_v20140616/
esgprep drs tree --project cordex /path/to/02_EuroCORDEX_INTERIM_044_v20140616/
esgprep drs todo --project cordex /path/to/02_EuroCORDEX_INTERIM_044_v20140616/
esgprep drs upgrade --project cordex /path/to/02_EuroCORDEX_INTERIM_044_v20140616/
```

Generate the mapfiles:

```
esgprep mapfile --project cordex ./CORDEX/
```

Publish to data and index nodes:

```
esgpublish --service fileservice --map mapfiles/ --project cordex --thredds --publish
```

Known issues during installation

#error "Psycopg requires PostgreSQL client library (libpq) >= 9.1

This error occurs sometimes during installation but removing the node and installing it from scratch seems to solve it...

```
Traceback (most recent call last):
 File "setup.py", line 110, in <module>
   """,
 File "/usr/local/uvcdat/2.2.0/lib/python2.7/distutils/core.py", line 111, in setup
   _setup_distribution = dist = klass(attrs)
 File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/dist.py", line 239, in __in
 File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/dist.py", line 263, in fetc
 File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/pkg_resources.py", line 568, in resolv
 File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/pkg_resources.py", line 806, in best_m
 File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/pkg_resources.py", line 818, in obtain
 File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/dist.py", line 313, in fetc
 File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/command/easy_install.py", l
 File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/command/easy_install.py", l
 File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/command/easy_install.py", l
 File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/command/easy_install.py", l
 File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/command/easy_install.py", l
distutils.errors.DistutilsError: Setup script exited with error: command 'gcc' failed with exit status 1

Sorry...
This action did not complete successfully
Please re-run this task until successful before continuing further

Also please review the installation FAQ it may assist you
https://github.com/ESGF/esgf.github.io/wiki/ESGFNode%7CFAQ
```

Failed building wheel for Pillow

This error seems unavoidable but it also seems that it doesn't affect the esgf functionality.

References

- ?ESGF Installation From Scratch